

MODULE – 3

BACKUP, ARCHIVE, AND REPLICATION

3.1 INTRODUCTION TO BUSINESS CONTINUITY

Business Continuity (BC):

Business continuity (BC) is an integrated and enterprise wide process that includes all activities (internal and external to IT) that a business must perform to mitigate the impact of planned and unplanned downtime.

BC entails preparing for, responding to, and recovering from a system outage that adversely affects business operations. It involves proactive measures, such as business impact analysis, risk assessments, deployment of BC technology solutions (backup and replication), and reactive measures, such as disaster recovery and restart, to be invoked in the event of a failure.

The goal of a BC solution is to ensure the “**information availability**” required to conduct vital business operations.

3.1.1 Information Availability:

Information availability (IA) refers to the ability of the infrastructure to function according to business expectations during its specified time of operation. Information availability ensures that people (employees, customers, suppliers, and partners) can access information whenever they need it. Information availability can be defined in terms of:

1. Reliability,
2. Accessibility
3. Timeliness.

1. **Reliability:** This reflects a component’s ability to function without failure, under stated conditions, for a specified amount of time.
2. **Accessibility:** This is the state within which the required information is accessible at the right place, to the right user. The period of time during which the system is in an accessible state is termed **system uptime**; when it is not accessible it is termed **system**

downtime.

3. **Timeliness:** Defines the exact moment or the time window (a particular time of the day, week, month, and/or year as specified) during which information must be accessible. For example, if online access to an application is required between 8:00 am and 10:00 pm each day, any disruptions to data availability outside of this time slot are not considered to affect timeliness.

3.1.1.1 Causes of Information Unavailability

Various planned and unplanned incidents result in data unavailability.

- **Planned outages** include installation/integration/maintenance of new hardware, software upgrades or patches, taking backups, application and data restores, facility operations (renovation and construction), and refresh/migration of the testing to the production environment.
- **Unplanned outages** include failure caused by database corruption, component failure, and human errors.
- **Disasters (natural or man-made)** such as flood, fire, earthquake, and contamination are another type of incident that may cause data unavailability.

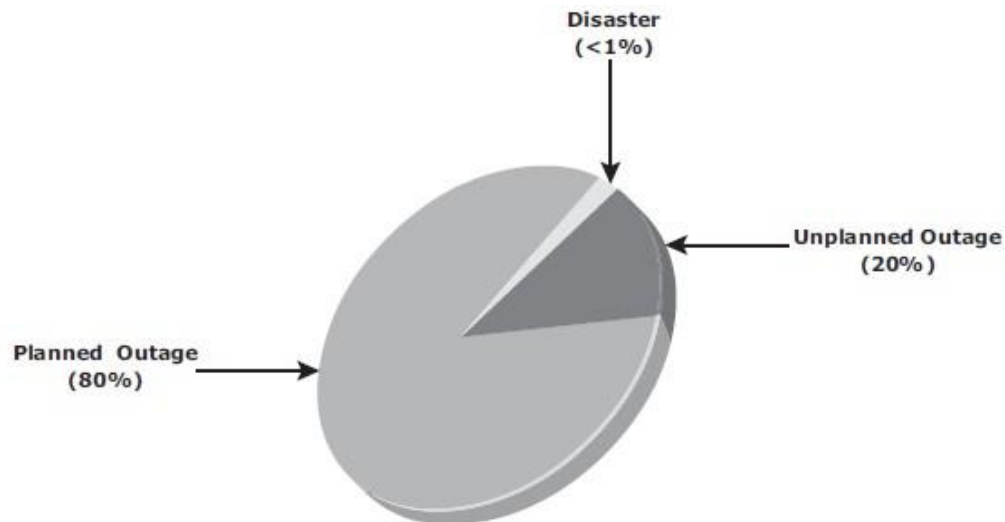


Fig 3.1: Disruptors of Information Availability

As illustrated in Fig 3.1 above, the majority of outages are planned. Planned outages are expected and scheduled, but still cause data to be unavailable.

3.1.1.2 Consequences of Downtime

- Information unavailability or downtime results in loss of productivity, loss of revenue, poor financial performance, and damage to reputation.
- Loss of productivity includes reduced output per unit of labor, equipment, and capital.
- Loss of revenue includes direct loss, compensatory payments, future revenue loss, billing loss, and investment loss.
- Poor financial performance affects revenue recognition, cash flow, discounts, payment guarantees, credit rating, and stock price.
- Damages to reputations may result in a loss of confidence or credibility with customers, suppliers, financial markets, banks, and business partners.
- An important metric, *average cost of downtime per hour*, provides a key estimate in determining the appropriate BC solutions. It is calculated as follows:

$$\text{Average cost of downtime per hour} = \text{average productivity loss per hour} + \text{average revenue loss per hour}$$

Where:

$$\text{Productivity loss per hour} = (\text{total salaries and benefits of all employees per week}) / (\text{average number of working hours per week})$$

$$\text{Average revenue loss per hour} = (\text{total revenue of an organization per week}) / (\text{average number of hours per week that an organization is open for business})$$

3.1.1.3 Measuring Information Availability

- Information availability (IA) relies on the availability of physical and virtual components of a data center. Failure of these components might disrupt IA. A failure is the termination of a component's capability to perform a required function. The component's capability can be restored by performing an external corrective action, such as a manual reboot, a repair, or replacement of the failed component(s).
- Proactive risk analysis performed as part of the BC planning process considers the component failure rate and average repair time, which are measured by MTBF and MTTR:

- **Mean Time Between Failure (MTBF):** It is the average time available for a system or component to perform its normal operations between failures.
- **Mean Time To Repair (MTTR):** It is the average time required to repair a failed component. MTTR includes the total time required to do the following activities: Detect the fault, mobilize the maintenance team, diagnose the fault, obtain the spare parts, repair, test, and restore the data.

Fig 3.2 illustrates the various information availability metrics that represent system uptime and downtime.

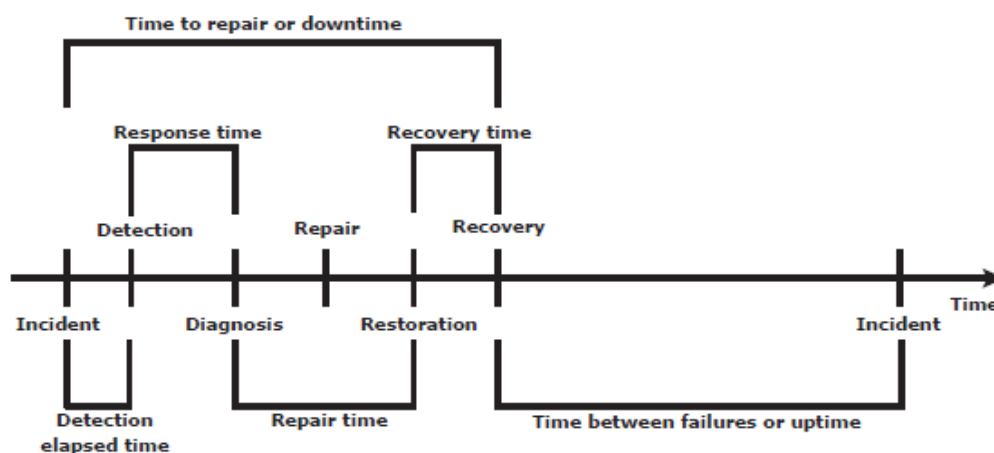


Fig 3-2: Information availability metrics

IA is the time period that a system is in a condition to perform its intended function upon demand. It can be expressed in terms of system uptime and downtime and measured as the amount or percentage of system uptime:

$$\text{IA} = \text{system uptime} / (\text{system uptime} + \text{system downtime})$$

In terms of MTBF and MTTR, IA could also be expressed as

$$\text{IA} = \text{MTBF} / (\text{MTBF} + \text{MTTR})$$

Uptime per year is based on the exact timeliness requirements of the service, this calculation leads to the number of “9s” representation for availability metrics.

Table 3-1 lists the approximate amount of downtime allowed for a service to achieve certain levels of 9s availability. For example, a service that is said to be “five 9s available” is available for 99.999 percent of the scheduled time in a year (24×365).

UPTIME (%)	DOWNTIME (%)	DOWNTIME PER YEAR	DOWNTIME PER WEEK
98	2	7.3 days	3 hr, 22 minutes
99	1	3.65 days	1 hr, 41 minutes
99.8	0.2	17 hr, 31 minutes	20 minutes, 10 secs
99.9	0.1	8 hr, 45 minutes	10 minutes, 5 secs
99.99	0.01	52.5 minutes	1 minute
99.999	0.001	5.25 minutes	6 secs
99.9999	0.0001	31.5 secs	0.6 secs

Table 3-1: Availability percentage and Allowable downtime

3.1.2 BC Terminology

This section defines common terms related to BC operations which are used in this module to explain advanced concepts:

- **Disaster recovery:** This is the coordinated process of restoring systems, data, and the infrastructure required to support key ongoing business operations in the event of a disaster. It is the process of restoring a previous copy of the data and applying logs or other necessary processes to that copy to bring it to a known point of consistency. Once all recoveries are completed, the data is validated to ensure that it is correct.
- **Disaster restart:** This is the process of restarting business operations with mirrored consistent copies of data and applications.
- **Recovery-Point Objective (RPO):** This is the point in time to which systems and data must be recovered after an outage. It defines the amount of data loss that a business can endure. A large RPO signifies high tolerance to information loss in a business. Based on the RPO, organizations plan for the minimum frequency with which a backup or replica must be made. For example, if the RPO is six hours, backups or replicas must be made at least once in 6 hours. Fig 3.3 (a) shows various RPOs and their corresponding ideal recovery strategies. An organization can plan for an appropriate BC technology solution on the basis of the RPO it sets. For example:
 - **RPO of 24 hours:** This ensures that backups are created on an offsite tape drive every midnight. The corresponding recovery strategy is to restore data from the set of last

backup tapes.

- **RPO of 1 hour:** Shipping database logs to the remote site every hour. The corresponding recovery strategy is to recover the database at the point of the last log shipment.
- **RPO in the order of minutes:** Mirroring data asynchronously to a remote site
- **Near zero RPO:** This mirrors mission-critical data synchronously to a remote site.

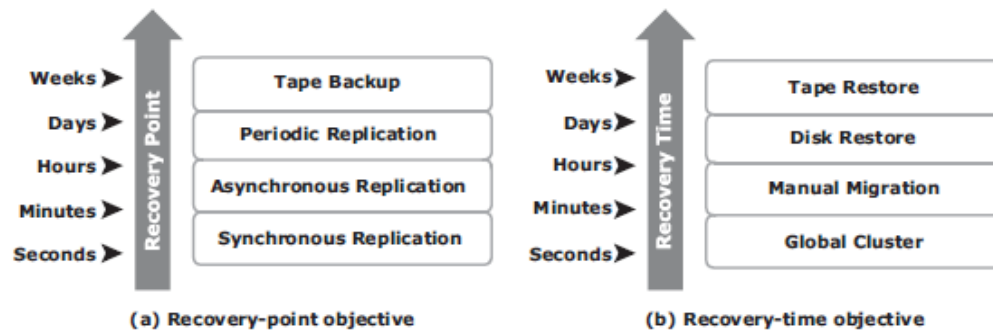


Fig 3.3: Strategies to meet RPO and RTO targets

- **Recovery-Time Objective (RTO):** The time within which systems and applications must be recovered after an outage. It defines the amount of downtime that a business can endure and survive. Businesses can optimize disaster recovery plans after defining the RTO for a given system. For example, if the RTO is two hours, then use a disk backup because it enables a faster restore than a tape backup. However, for an RTO of one week, tape backup will likely meet requirements. Some examples of RTOs and the recovery strategies to ensure data availability are listed below (refer to Fig 3.3 (b)):

- **RTO of 72 hours:** Restore from backup tapes at a cold site.
- **RTO of 12 hours:** Restore from tapes at a hot site.
- **RTO of few hours:** Use a data vault to a hot site.
- **RTO of a few seconds:** Cluster production servers with bidirectional mirroring, enabling the applications to run at both sites simultaneously.

3.1.3 BC Planning Life Cycle

BC planning must follow a disciplined approach like any other planning process. Organizations today dedicate specialized resources to develop and maintain BC plans. From the conceptualization to the realization of the BC plan, a life cycle of activities can be defined for the BC process.

The BC planning lifecycle includes five stages shown below (Fig 3.4):

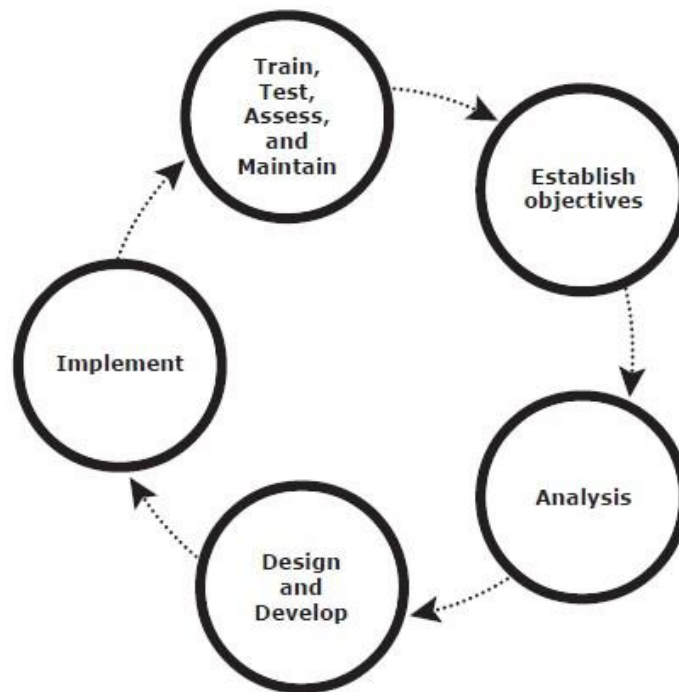


Fig 3.4: BC Planning Lifecycle

Several activities are performed at each stage of the BC planning lifecycle, including the following key activities:

1. Establishing objectives

- Determine BC requirements.
- Estimate the scope and budget to achieve requirements.
- Select a BC team by considering subject matter experts from all areas of the business, whether internal or external.
- Create BC policies.

2. Analyzing

- Collect information on data profiles, business processes, infrastructure support, dependencies, and frequency of using business infrastructure.
- Identify critical business needs and assign recovery priorities.
- Create a risk analysis for critical areas and mitigation strategies.
- Conduct a Business Impact Analysis (BIA).
- Create a cost and benefit analysis based on the consequences of data unavailability.

3. Designing and developing

- Define the team structure and assign individual roles and responsibilities. For example, different teams are formed for activities such as emergency response, damage assessment, and infrastructure and application recovery.
- Design data protection strategies and develop infrastructure.
- Develop contingency scenarios.
- Develop emergency response procedures.
- Detail recovery and restart procedures.

4. Implementing

- Implement risk management and mitigation procedures that include backup, replication, and management of resources.
- Prepare the disaster recovery sites that can be utilized if a disaster affects the primary data center.
- Implement redundancy for every resource in a data center to avoid single points of failure.

5. Training, testing, assessing, and maintaining

- Train the employees who are responsible for backup and replication of business-critical data on a regular basis or whenever there is a modification in the BC plan
- Train employees on emergency response procedures when disasters are declared.
- Train the recovery team on recovery procedures based on contingency scenarios.
- Perform damage assessment processes and review recovery plans.
- Test the BC plan regularly to evaluate its performance and identify its limitations.
- Assess the performance reports and identify limitations.

→ Update the BC plans and recovery/restart procedures to reflect regular changes within the data center.

3.1.4 Failure Analysis

3.1.4.1 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.
- Fig 3.5 depicts a system setup in which an application, running on a VM, 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
- In a setup where **each component must function as required to ensure data availability**, the failure of a single physical or virtual component causes the failure of the entire data center or an application, resulting in disruption of business operations.
- In this example, failure of a hypervisor can affect all the running VMs and the virtual network, which are hosted on it.
- There can be several similar single points of failure identified in this example. A VM, a hypervisor, an HBA/NIC on the server, the physical server, the IP network, the FC switch, the storage array ports, or even the storage array could be a potential single point of failure. To avoid single points of failure, it is essential to implement a fault-tolerant mechanism.

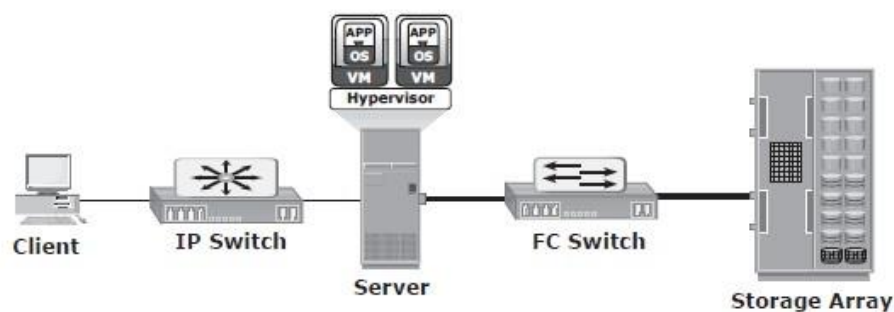


Fig 3.5: Single Point of Failure

3.1.4.2 Resolving Single Points of Failure

- 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.
- Data centers follow stringent guidelines to implement fault tolerance for uninterrupted information availability. Careful analysis is performed to eliminate every single point of failure.
- The example shown in Fig 3.6 represents all enhancements of the system shown in Fig 3.5 in the infrastructure to mitigate single points of failure:
 - Configuration of redundant HBAs at a server to mitigate single HBA failure
 - Configuration of NIC (network interface card) teaming at a server allows protection against single physical NIC failure. It allows grouping of two or more physical NICs and treating them as a single logical device. NIC teaming eliminates the single point of failure associated with a single physical NIC.
 - Configuration of redundant switches to account for a switch failure
 - Configuration of multiple storage array ports to mitigate a port failure
 - RAID and hot spare configuration to ensure continuous operation in the event of disk failure
 - Implementation of a redundant storage array at a remote site to mitigate local site failure
 - Implementing server (or compute) clustering, a fault-tolerance mechanism whereby two or more servers in a cluster access the same set of data volumes. Clustered servers exchange a heartbeat to inform each other about their health. If one of the servers or hypervisors fails, the other server or hypervisor can take up the workload.
 - Implementing a VM Fault Tolerance mechanism ensures BC in the event of a server failure. This technique creates duplicate copies of each VM on another server so that when a VM failure is detected, the duplicate VM can be used for failover. The two VMs are kept in synchronization with each other in order to perform successful failover.

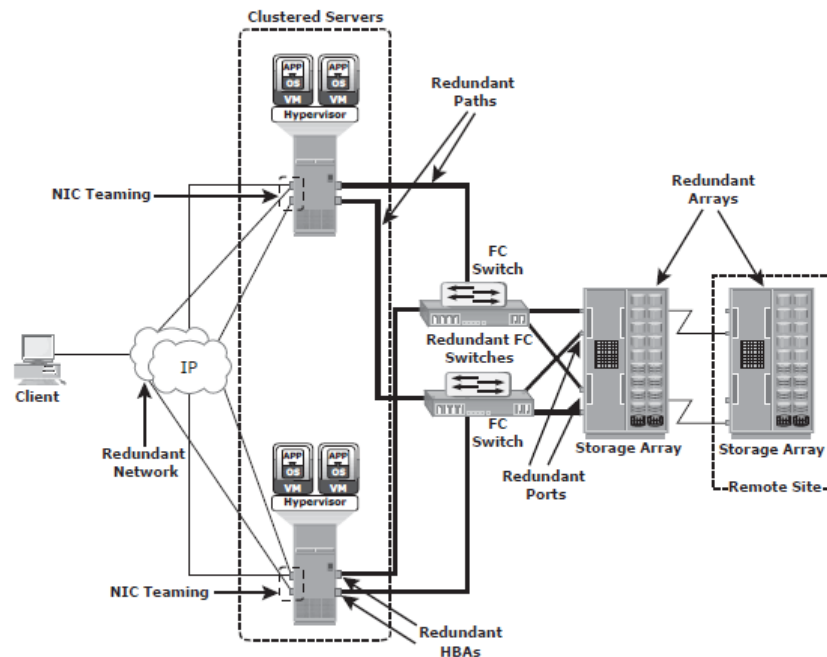


Fig 3.6: Resolving single points of failure

3.1.4.3 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.
- In a virtual environment, multipathing is enabled either by using the hypervisor's built-in capability or by running a third-party software module, added to the hypervisor.

3.1.5 BC Technology Solutions

After analyzing the business impact of an outage, designing appropriate solutions to recover from a failure is the next important activity. One or more copies of the original data are maintained using any of the following strategies, so that data can be recovered and business operations can be restarted using an alternate copy:

1. **Backup:** Data backup is a predominant method of ensuring data availability. The frequency of backup is determined based on RPO, RTO, and the frequency of data changes.
2. **Storage array-based replication (local):** Data can be replicated to a separate location within the same storage array. The replica is used independently for other business operations. Replicas can also be used for restoring operations if data corruption occurs.
3. **Storage array-based replication (remote):** Data in a storage array can be replicated to another storage array located at a remote site. If the storage array is lost due to a disaster, business operations can be started from the remote storage array.

3.2 **Backup and Recovery**

- **Data Backup** is a copy of production data, created and retained for the sole purpose of recovering lost or corrupted data.
- Evaluating the various backup methods along with their recovery considerations and retention requirements is an essential step to implement a successful backup and recovery solution.
- Organizations generate and maintain large volumes of data, and most of the data is fixed content. This fixed content is rarely accessed after a period of time. Still, this data needs to be retained for several years to meet regulatory compliance.
- **Data archiving** is the process of moving data that is no longer actively used, from primary storage to a low-cost secondary storage. This data is retained in the secondary storage for a long term to meet regulatory requirements. This reduces the amount of data to be backed up and the time required to back up the data.

3.2.1 **Backup Purpose**

Backups are performed to serve three purposes: *disaster recovery*, *operational recovery*, and *archival*. These are discussed in the following sections.

3.2.1.1 **Disaster Recovery**

- Backups are 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. Organizations can bring production systems online in a relatively short period of time if a disaster occurs.

3.2.1.2 **Operational Recovery**

- Data in the production environment changes with every business transaction and operation.
- Operational recovery is the use of backups to restore data if data loss or logical

corruption occurs during routine processing.

- For example, it is common for a user to accidentally delete an important email or for a file to become corrupted, which can be restored from operational backup.

3.2.13 Archival

- Backups are also performed to address archival requirements.
- 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.

3.2.2 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. This method of backup is also referred to as an *online backup*.
- In a **cold backup**, the application is not active or shutdown during the backup process and is also called as *offline backup*.
- The hot backup of online production data becomes more challenging because data is actively used and changed.
- An open file is locked by the operating system and is not backed up during the backup process. In such situations, an *open file agent* is required to back up the open file.
- In database environments, the use of open file agents is not enough, because the agent should also support a consistent backup of all the database components.
- 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.
- The disadvantage associated with a hot backup is that the agents usually affect the overall application performance.

- 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. The PIT copy is created from the production volume and used as the source for the backup. This reduces the impact on the production volume.
- 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.

3.2.3 Backup Topologies

- Three basic topologies are used in a backup environment:
 1. Direct attached backup
 2. LAN based backup, and
 3. 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 Fig 3.7 device is directly attached and dedicated to the backup client. 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. Network-based topologies (LAN-based and SAN-based) provide the solution to optimize the utilization of backup devices.

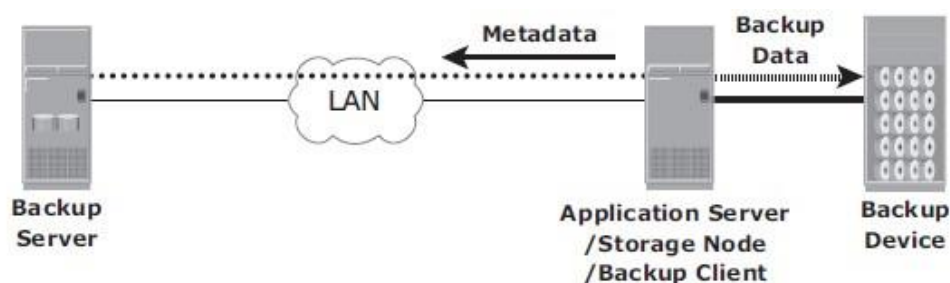


Fig 3.7: Direct-attached backup topology

- In **LAN-based backup**, the clients, backup server, storage node, and backup device are connected to the LAN (see Fig 3.8). 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.
- 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.

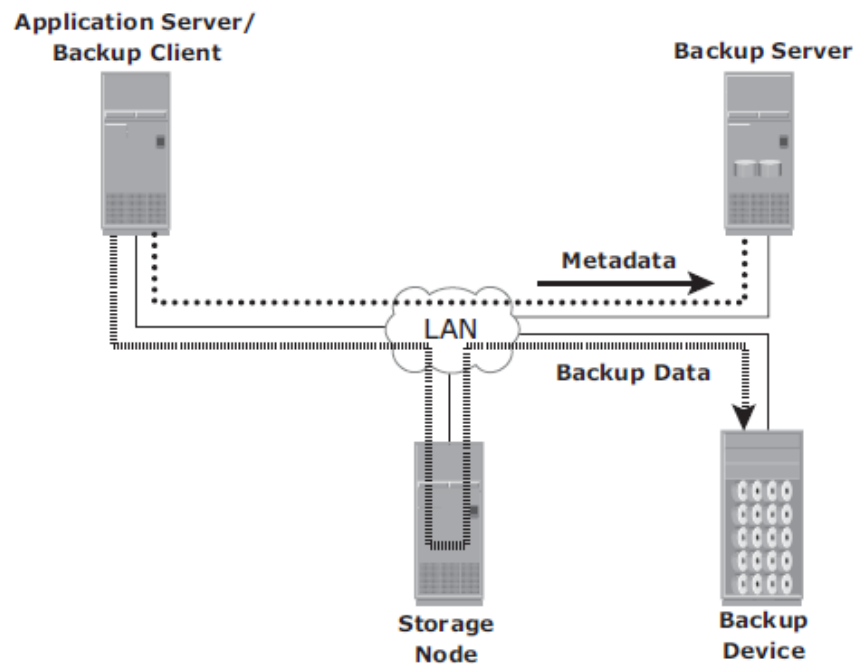


Fig 3.8: LAN-based backup topology

- The **SAN-based backup** is also known as the *LAN-free backup*. Fig 3.9 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.
- In the example from Fig 3.9, a client sends the data to be backed up to the backup device over the SAN. Therefore, the backup data traffic is restricted to the SAN, and only the backup metadata is transported over the LAN. The volume of metadata is insignificant when compared to the production data; the LAN performance is not degraded in this configuration.

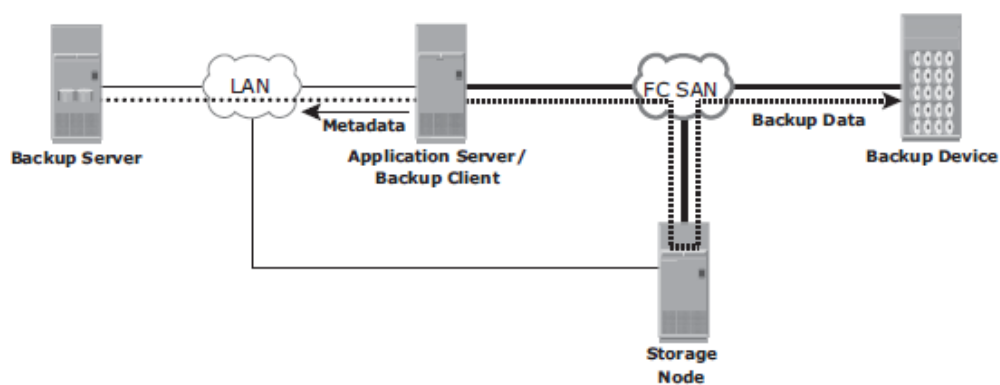


Fig 3.9: SAN-based backup topology

- 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 Fig 3.10. This topology might be implemented for several reasons, including cost, server location, reduction in administrative overhead, and performance considerations.

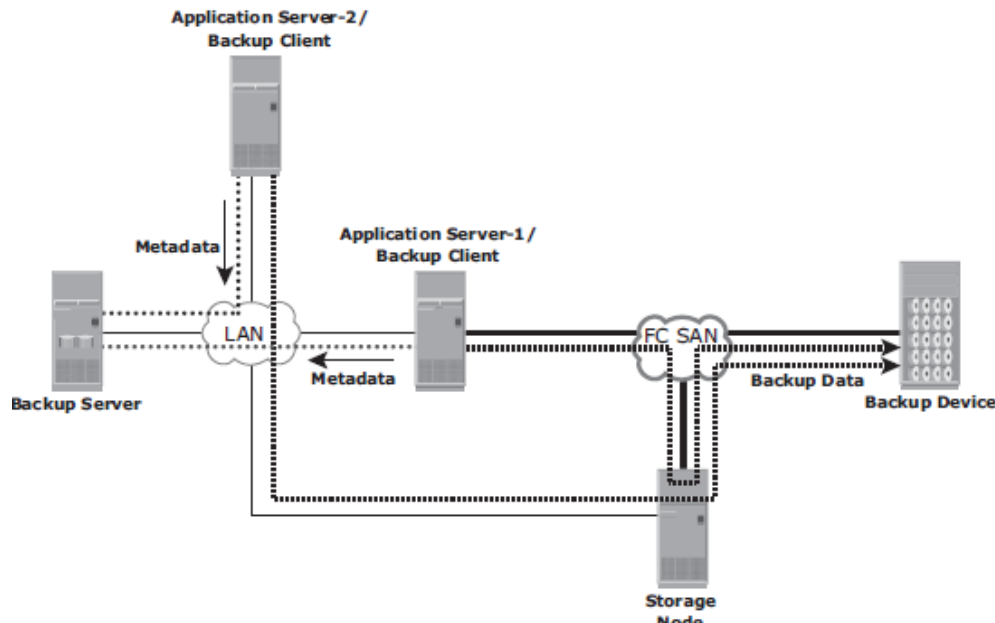


Fig 3.10: Mixed backup topology

3.2.4 Backup Technologies

- A wide range of technology solutions are currently available for backup targets.
- Tapes and disks are the two most commonly used backup media. Virtual tape libraries use disks as backup medium emulating tapes, providing enhanced backup and recovery capabilities.

3.2.4.1 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.
- A tape cartridge is composed of magnetic tapes in a plastic enclosure.
- 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.

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. Fig 3-14 shows a 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.

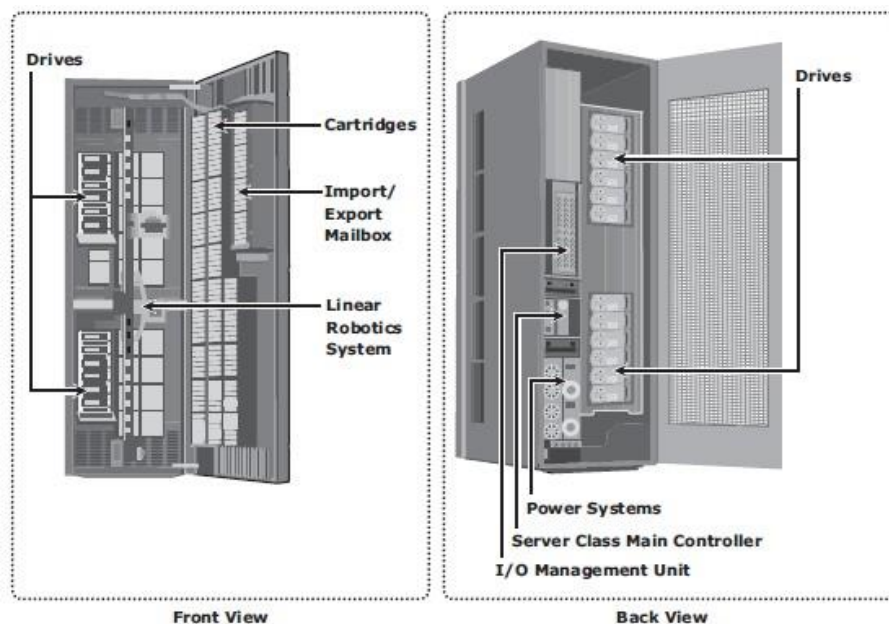


Fig 3.11: Physical tape library

- 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 (Fig 3.11 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 Fig 3.12, 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.

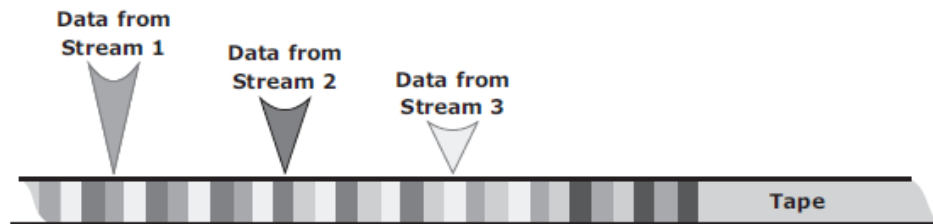


Fig 3.12: Physical tape library

- Many times, even the buffering and speed adjustment features of a tape drive fail to prevent the gaps, causing the “*shoe shining effect*” or “*backhitching*.” This is the repeated back and forth motion a tape drive makes when there is an interruption in the backup data stream. This repeated back-and-forth motion not only causes a degradation of service, but also excessive wear and tear to tapes.
- When the tape operation finishes, 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.

Limitations of Tape

- 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.

3.2.4.2 Backup to Disk

- Because of *increased availability*, low cost **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 TCO* (Total cost of ownership), and *improved quality of service*. 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.
- Fig 3.13 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, a restore from disk took 24 minutes compared to the restore from a tape, which took 108 minutes for the same environment.
- Recovering from 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 and RTO.
- Backup to disk does not offer any inherent offsite capability, and is dependent on other technologies such as local and remote replication.
- Some backup products also 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.

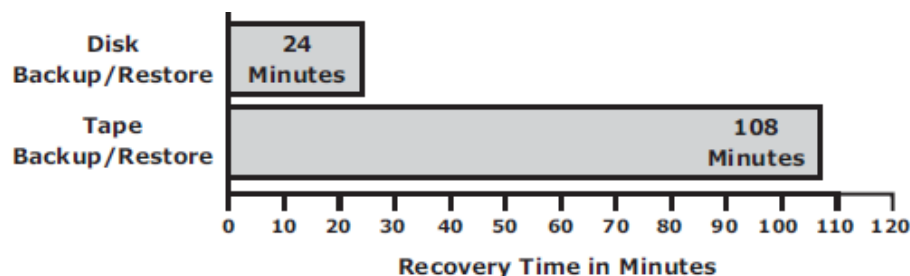


Fig 3.13: Tape versus Disk restore

3.2.4.3 Backup to Virtual Tape

- Virtual tapes are disk drives emulated and presented as tapes to the backup software.
- The key benefit of using a virtual tape is that it does not require any additional modules, configuration, or changes in the legacy backup software. This preserves the investment made in the backup software.

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 the backup software, there is no difference between a physical tape library and a virtual tape library.
- Fig 3.14 shows a virtual tape library that uses 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 backing up 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.
- 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.
- 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 data 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.
- 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.

- Connecting the engines of a virtual tape library appliance to a physical tape library enables 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 tapes offers several advantages over both physical tapes and disks.
- Compared to physical tapes, virtual tapes offer better single stream performance, better reliability, and random disk access characteristics.
- Backup and restore operations benefit from the disk's random access characteristics because they are always online and provide faster backup and recovery.
- A virtual tape drive 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, a virtual tape library offers easy installation and administration because it is preconfigured by the manufacturer.
- However, a virtual tape library is generally used only for backup purposes. In a backup-to-disk environment, the disk systems are used for both production and backup data.

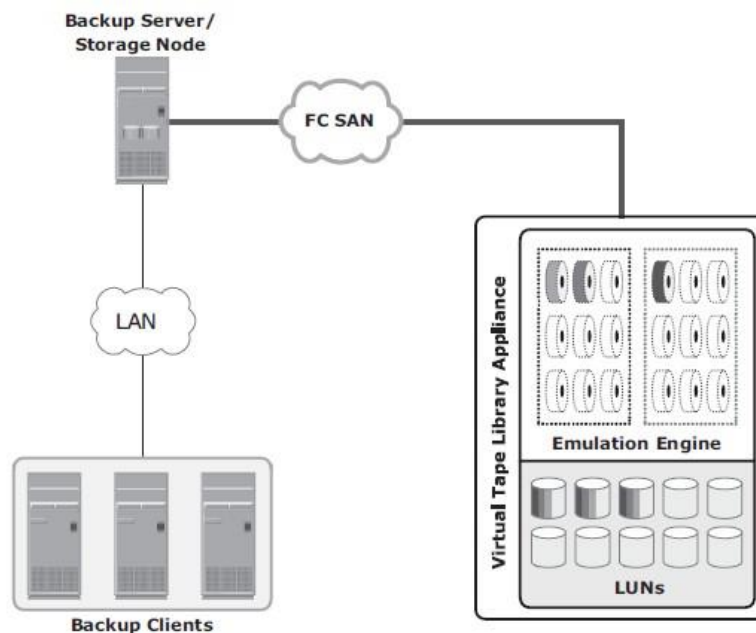


Fig 3.14: Virtual Tape Library

3.2.4.4 Backup Targets Comparison

Table 3.2 shows a comparison between various backup targets.

FEATURES	TAPE	DISK	VIRTUAL TAPE
Offsite Replication Capabilities	No	Yes	Yes
Reliability	No inherent protection methods	Yes	Yes
Performance	Subject to mechanical operations, loading time	Faster single stream	Faster single stream
Use	Backup only	Multiple (backup, production)	Backup only

Table 3.2: Backup targets comparison

3.2.5 Data Deduplication for Backup

- **Data deduplication** is the process of identifying and eliminating redundant data. When duplicate data is detected during backup, the data is discarded and only the pointer is created to refer the copy of the data that is already backed up.
- Data deduplication helps to reduce the storage requirement for backup, shorten the backup window, and remove the network burden. It also helps to store more backups on the disk and retain the data on the disk for a longer time.

3.2.5.1 Data Deduplication Methods

- There are two methods of deduplication: *file level* and *subfile level*.
- The differences exist in the amount of data reduction each method produces and the time each approach takes to determine the unique content.
- **File-level deduplication** (also called single-instance storage) detects and removes redundant copies of identical files. It enables storing only one copy of the file; the subsequent copies are replaced with a pointer that points to the original file.
- File-level deduplication is simple and fast but does not address the problem of duplicate content inside the files. For example, two 10-MB PowerPoint presentations with a difference in just the title page are not considered as duplicate files, and each file will be stored separately.

- *Subfile deduplication* breaks the file into smaller chunks and then uses a specialized algorithm to detect redundant data within and across the file. As a result, subfile deduplication eliminates duplicate data across files.
- There are two forms of subfile deduplication: fixed-length block and variable-length segment.
- *The fixed-length block deduplication* divides the files into fixed length blocks and uses a hash algorithm to find the duplicate data.
- Although simple in design, fixed-length blocks might miss many opportunities to discover redundant data because the block boundary of similar data might be different. Consider the addition of a person's name to a document's title page. This shifts the whole document, and all the blocks appear to have changed, causing the failure of the deduplication method to detect equivalencies.
- In *variable-length segment deduplication*, if there is a change in the segment, the boundary for only that segment is adjusted, leaving the remaining segments unchanged. This method vastly improves the ability to find duplicate data segments compared to fixed-block.

3.2.5.2 Data Deduplication Implementation

Deduplication for backup can happen at the data source or the backup target.

Source-Based Data Deduplication

- *Source-based data deduplication* eliminates redundant data at the source before it transmits to the backup device.
- Source-based data deduplication can dramatically reduce the amount of backup data sent over the network during backup processes. It provides the benefits of a shorter backup window and requires less network bandwidth. There is also a substantial reduction in the capacity required to store the backup images.
- Fig 3.15 shows source-based data deduplication.
- Source-based deduplication increases the overhead on the backup client, which impacts the performance of the backup and application running on the client.
- Source-based deduplication might also require a change of backup software if it is not supported by backup software.

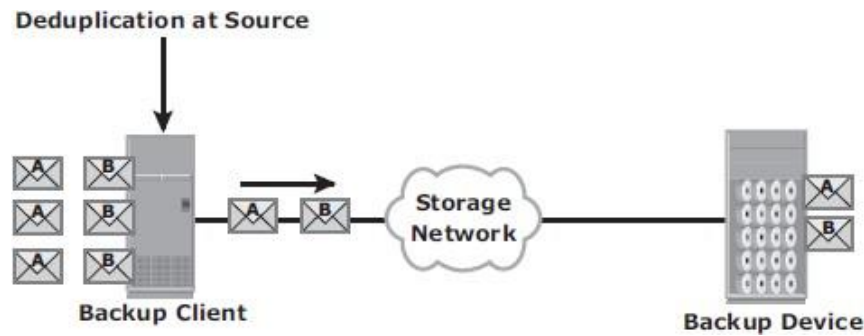


Fig 3.15: Source-based data deduplication

Target-Based Data Deduplication

- Target-based data deduplication is an alternative to source-based data deduplication.
- Target-based data deduplication occurs at the backup device, which offloads the backup client from the deduplication process.
- Fig 3.16 shows target-based data deduplication.
- In this case, the backup client sends the data to the backup device and the data is deduplicated at the backup device, either *immediately (inline)* or at a *scheduled time (post-process)*.
- Because deduplication occurs at the target, all the backup data needs to be transferred over the network, which increases network bandwidth requirements. Target-based data deduplication does not require any changes in the existing backup software.
- *Inline deduplication* performs deduplication on the backup data before it is stored on the backup device. Hence, this method reduces the storage capacity needed for the backup.
- Inline deduplication introduces overhead in the form of the time required to identify and remove duplication in the data. So, this method is best suited for an environment with a large backup window.
- *Post-process deduplication* enables the backup data to be stored or written on the backup device first and then deduplicated later.
- This method is suitable for situations with tighter backup windows. However, post-process deduplication requires more storage capacity to store the backup images before they are deduplicated.

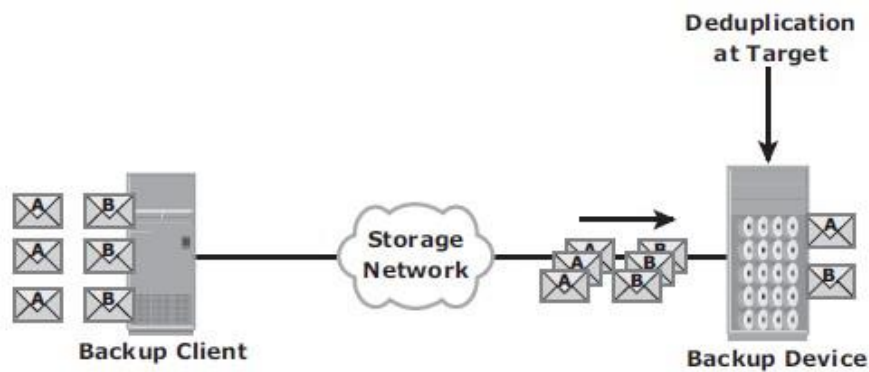


Fig 3.16: Target-based data deduplication

3.2.6 Backup in Virtualized Environments

- There are two approaches for performing a backup in a virtualized environment: the *traditional backup* approach and the *image-based* backup approach.
- In the *traditional backup* approach, a backup agent is installed either on the virtual machine (VM) or on the hypervisor.
- Fig 3.17 shows the traditional VM backup approach.
- If the backup agent is installed on a VM, the VM appears as a physical server to the agent. The backup agent installed on the VM backs up the VM data to the backup device. The agent does not capture VM files, such as the virtual BIOS file, VM swap file, logs, and configuration files. Therefore, for a VM restore, a user needs to manually re-create the VM and then restore data onto it.
- If the backup agent is installed on the hypervisor, the VMs appear as a set of files to the agent. So, VM files can be backed up by performing a file system backup from a hypervisor. This approach is relatively simple because it requires having the agent just on the hypervisor instead of all the VMs.
- The traditional backup method can cause high CPU utilization on the server being backed up.
- So the backup should be performed when the server resources are idle or during a low activity period on the network.
- And also allocate enough resources to manage the backup on each server when a large number of VMs are in the environment.

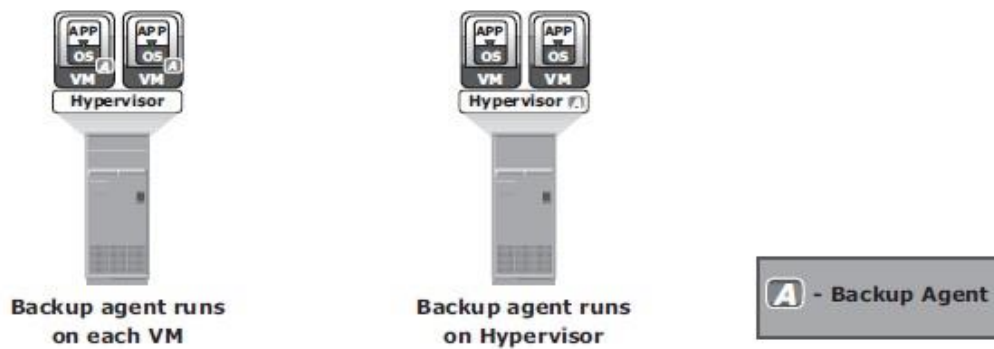


Fig 3.17: Traditional VM backup

- *Image-based backup* operates at the hypervisor level and essentially takes a snapshot of the VM.
- It creates a copy of the guest OS and all the data associated with it (snapshot of VM disk files), including the VM state and application configurations. The backup is saved as a single file called an “*image*,” and this image is mounted on the separate physical machine—proxy server, which acts as a backup client.
- The backup software then backs up these image files normally. (see Fig 3.18).
- This effectively offloads the backup processing from the hypervisor and transfers the load on the proxy server, thereby reducing the impact to VMs running on the hypervisor.
- Image-based backup enables quick restoration of a VM.

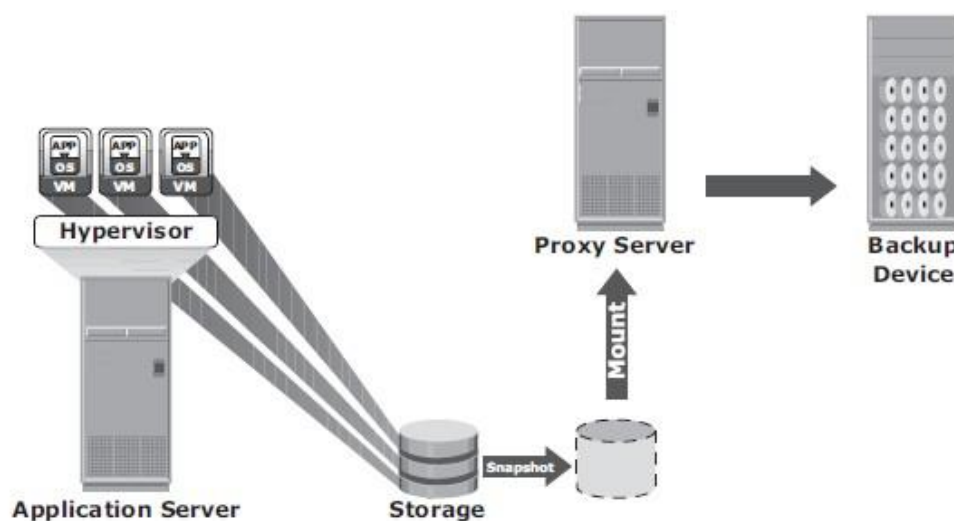


Fig 3.18: Image-based backup

SVIT