INFORMATION STORAGE AND MANAGEMENT

CS52216CC

CREDIT : 3

MODULE-I

Introduction to Information Storage: Information Storage,Evolution of Storage Architecture, Data Center Infrastructure, Virtualization and Cloud Computing.

Data Center Environment: Application, DatabaseManagement System, Host, Connectivity, Storage, Disk Drive Component, Disk Drive Performance.

Why Information management?

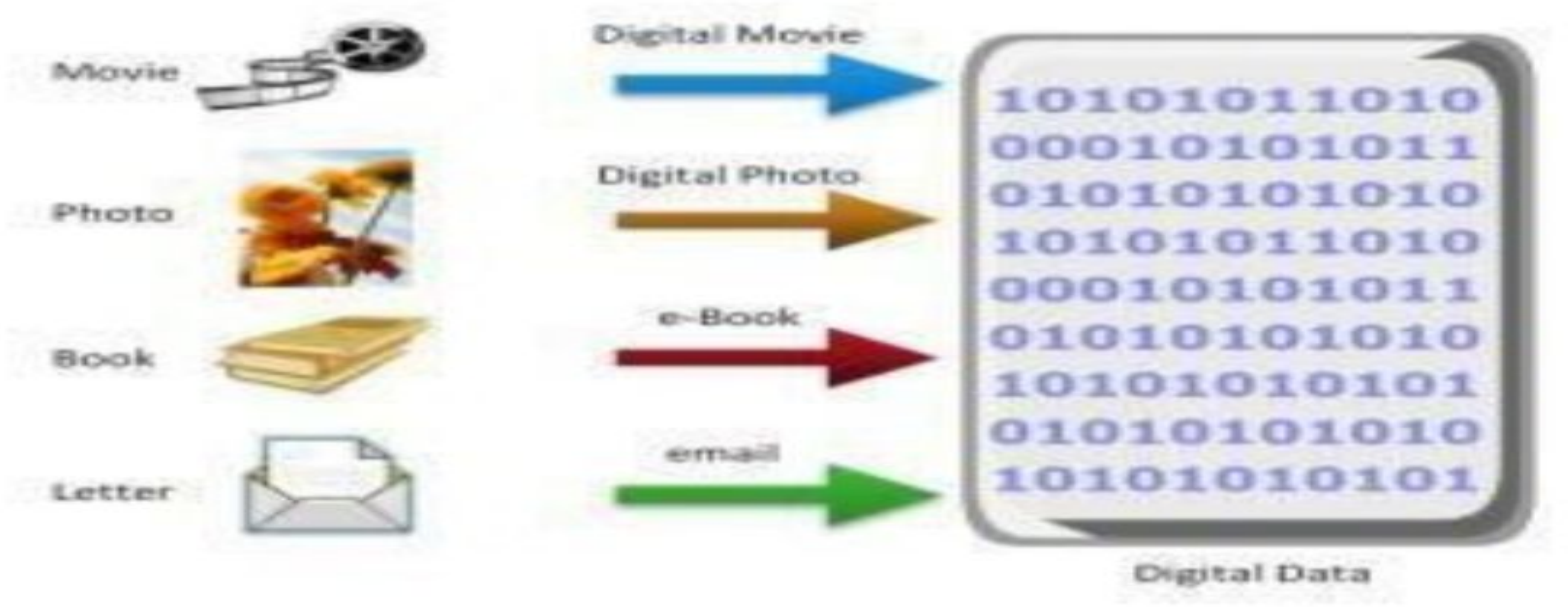
- Information management is the practice of **collecting, storing, organizing, protecting, and providing access to information** so that it is available when and where it’s needed whether for personal use or for critical business operations like online transactions, reservations, patient care, or inventory control.
- It ensures that the growing volume of data we generate is **handled efficiently, securely, and in a way that supports decision-making and daily activities**.

- **Information storage** is a place usually a digital system where data is kept so it can be **saved and accessed later**.
- For businesses, this storage is essential because it holds the data they need to create important information for their daily work, such as sales records, customer details, or inventory data. In short, it’s a **digital warehouse for business information**.
- Businesses use data to derive information that is critical to their day-to-day operations.
- Storage is a repository that enables users to store and retrieve this digital data.

- Data
- Data is a collection of raw facts from which conclusions may be drawn.
- Eg: a printed book, a family photograph, a movie on videotape, e-mail message, an e-book, a bitmapped image, or a digital movie are all examples of data.

Real-world content → converted to binary (0s & 1s) → stored as digital data → processed back for user access.

Fig 1.1: Digital data



Several reasons have caused **digital data** to grow rapidly

1. Increase in data processing capabilities:
2. Lower cost of digital storage:
3. Affordable and faster communication technology:
4. Proliferation of applications and smart devices:

- Increase in data processing capabilities:
- Increase in data processing capabilities: Modern-day computers provide a significant increase in processing and storage capabilities. This enables the conversion of various types of content and media from conventional forms to digital formats.

- Lower cost of digital storage:
- Lower cost of digital storage: Technological advances and decrease in the cost of storage devices have provided low-cost solutions and encouraged the development of less expensive data storage devices. This cost benefit has increased the rate at which data is being generated and stored.

- Affordable and faster communication technology:
- The rate of sharing digital data is now much faster than traditional approaches. A handwritten letter may take a week to reach its destination, whereas it only takes a few seconds for an e-mail message to reach

Types of data

- **Structured data**
- Structured data is organized in rows and columns in a rigidly defined format so that applications can retrieve and process it efficiently.
- Structured data is typically stored using a database management system(DBMS).

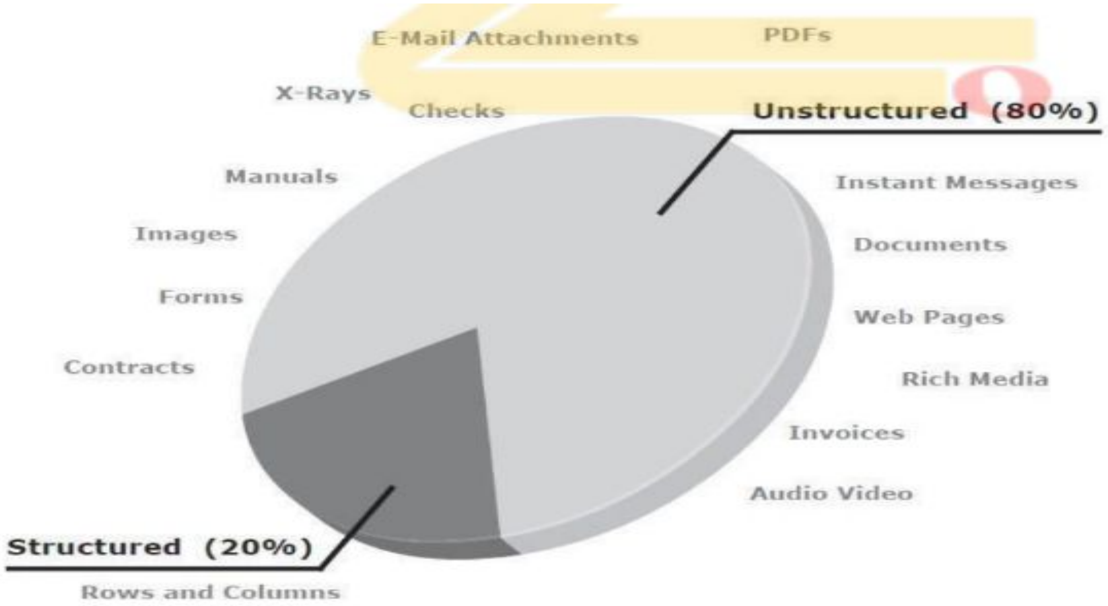


Fig 1.2: Types of data

Big Data

Big data refers to data sets whose sizes are beyond the capability of commonly used software tools to capture, store, manage, and process within acceptable time limits.

It includes both structured and unstructured data generated by a variety of sources, including business application transactions, web pages, videos, images, e-mails, social media, and so on.

- The big data ecosystem consists of the following:

1. Devices that collect data from multiple locations and also generate new data about this data (metadata).
2. Data collectors who gather data from devices and users.
3. Data aggregators that compile the collected data to extract meaningful information.
4. Data users and buyers who benefit from the information collected and aggregated by others in the data value chain .

Bigdata Ecosystem

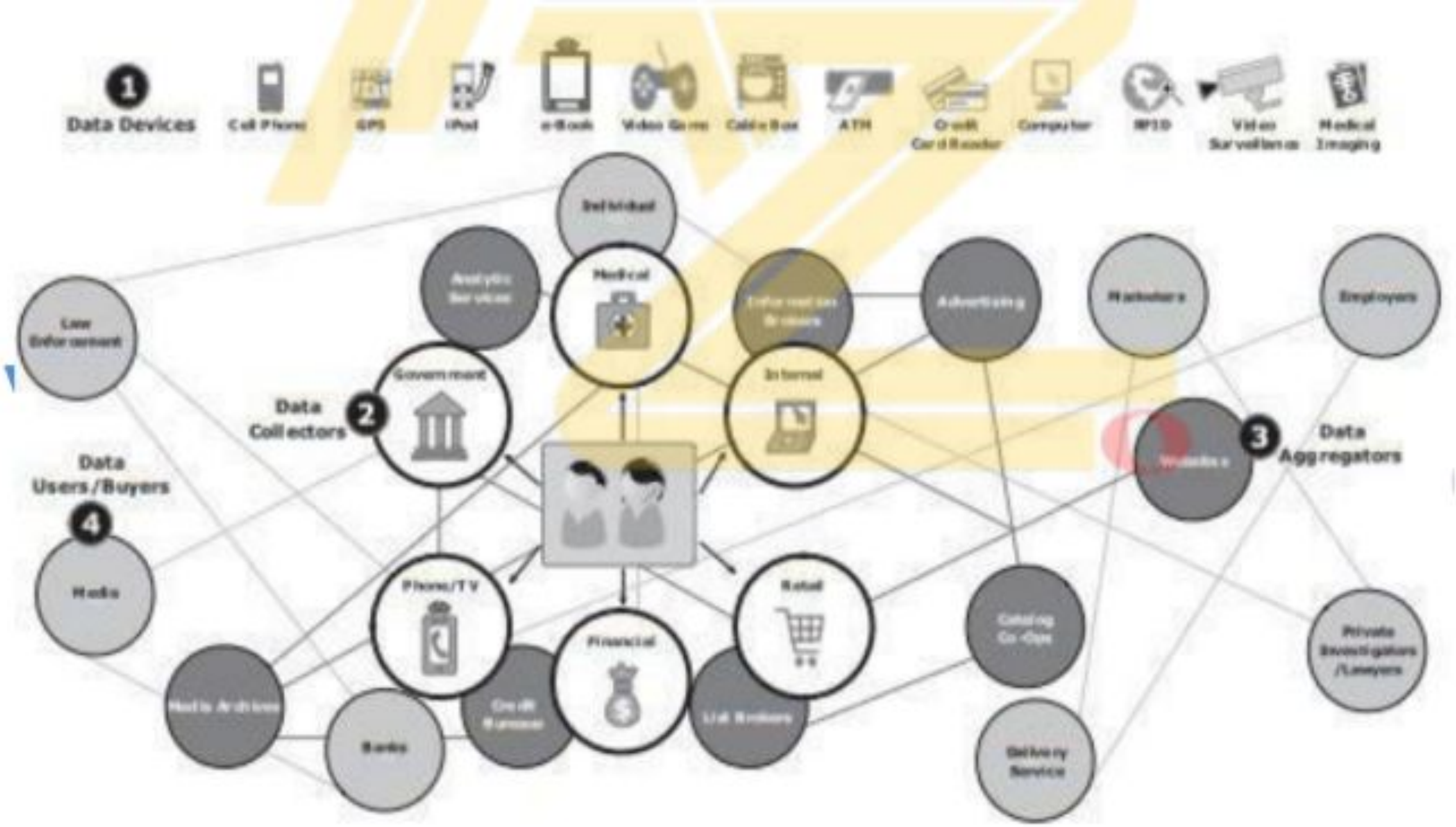


Fig 1.3: Big data Ecosystem

- Big data Analysis in real time requires new techniques, architectures, and tools that provide :

1. high performance,
2. massively parallel processing (MPP) data platforms,
3. advanced analytics on the data sets.

Information

- Data, whether structured or unstructured, does not fulfil any purpose for individuals or businesses unless it is presented in a meaningful form.
- Information is the intelligence and knowledge derived from data.
- Businesses analyze raw data in order to identify meaningful trends. On the basis of these trends, a company can plan or modify its strategy.
- For example, a retailer identifies customers' preferred products and brand names by analyzing their purchase patterns and maintaining an inventory of those products.
- Because information is critical to the success of a business, there is an ever present concern about its availability and protection.

Storage

- Data created by individuals or businesses must be stored so that it is easily accessible for further processing.
- In a computing environment, devices designed for storing data are termed storage devices or simply storage.
- The type of storage used varies based on the type of data and the rate at which it is created and used.
- Devices such as memory in a cell phone or digital camera, DVDs, CD-ROMs, and hard disks in personal computers are examples of storage devices.
- Businesses have several options available for storing data including internal hard disks, external disk arrays and tapes.

Evolution of Storage Architecture

- Historically, organizations had centralized computers (mainframe) and information storage devices (tape reels and disk packs) in their data center.
- The evolution of open systems and the affordability and ease of deployment that they offer made it possible for business units/departments to have their own servers and storage.

Evolution of Storage Architecture

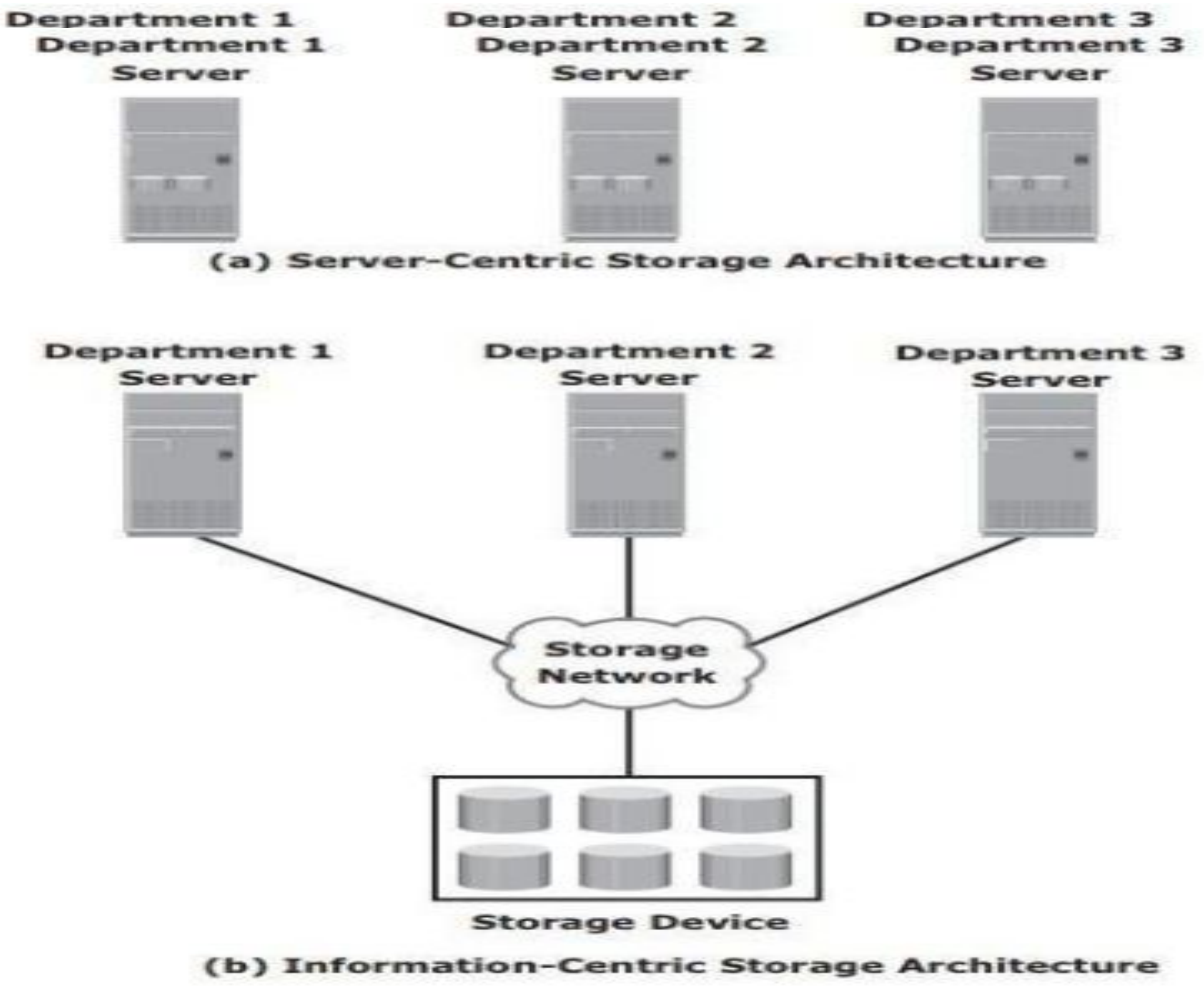
- In earlier implementations of open systems, the storage was typically internal to the server. This approach is referred to as server-centric storage architecture.
- In this server-centric storage architecture, each server has a limited number of storage devices, and any administrative tasks, such as maintenance of the server or increasing storage capacity, might result in unavailability of information.

Evolution of Storage Architecture

- The rapid increase in the number of departmental servers in an enterprise resulted in unprotected, unmanaged, fragmented islands of information and increased capital and operating expenses.
- To overcome these challenges, storage evolved from

server-centric to information-centric

Information centric storage architecture



- In information-centric architecture, storage devices are managed centrally and independent of servers.
- These centrally-managed storage devices are shared with multiple servers.
- When a new server is deployed in the environment, storage is assigned from the same shared storage devices to that server.

- The capacity of shared storage can be increased dynamically by adding more storage devices without impacting information availability.
- In this architecture, information management is easier and

cost-effective.

- Storage technology and architecture continues to evolve, which enables organizations to consolidate, protect, optimize, and leverage their data to achieve the highest return on information assets.

Data Center Infrastructure

- Organizations maintain data centers to provide centralized data processing capabilities across the enterprise.
- The data center infrastructure includes computers, storage systems, network devices, dedicated power backups, and environmental controls (such as air conditioning and fire suppression).

Key Data Center Elements

- Five core elements are essential for the basic functionality of a data center:

1) Application: An application is a computer program that provides the logic for computing operations. Eg: order processing system.

2) Database: More commonly, a database management system (DBMS) provides a structured way to store data in logically organized tables that are interrelated. A DBMS optimizes the storage and retrieval of data.

3) Host or compute: A computing platform (hardware, firmware, and software) that runs applications and databases.

4) Network: A data path that facilitates communication among various networked devices.

5) Storage array: A device that stores data persistently for subsequent use.

Fig 1.5 shows an example of an order processing system that involves the five core elements of a data center and illustrates their functionality in a business process.

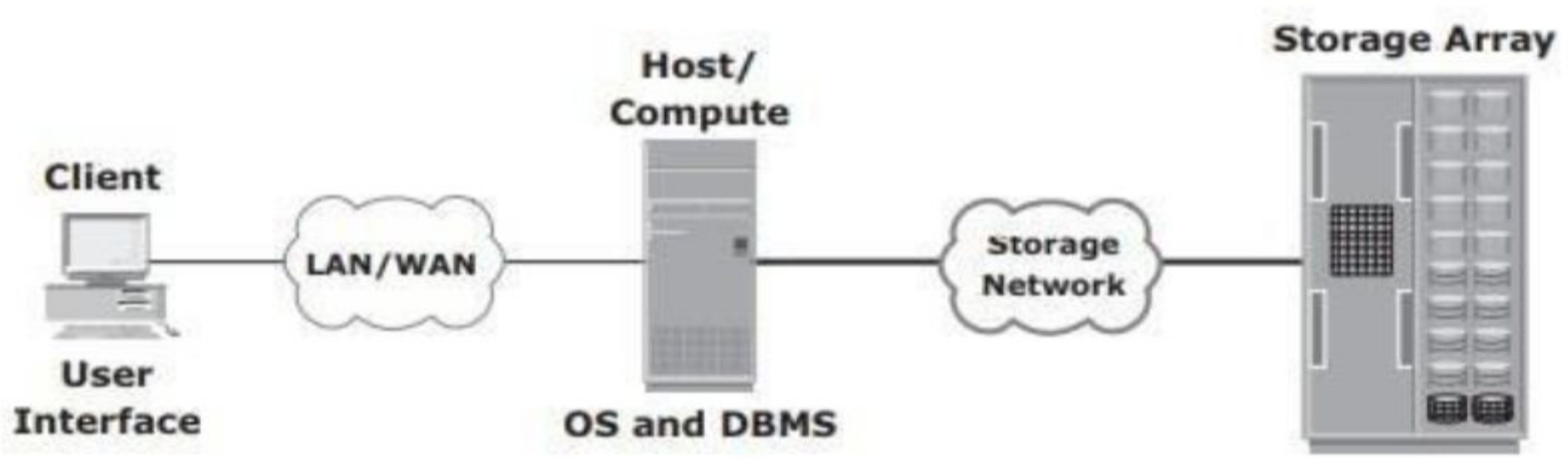


Fig 1.5: Example of an online order transaction system

1) A customer places an order through a client machine connected over a LAN/ WAN to a host running an order-processing application.

2) The client accesses the DBMS on the host through the application to provide order- related information, such as the customer name, address, payment method, products ordered, and quantity

ordered.

4) The Storage Network provides the communication link between the host and the storage array and transports the request to read or write commands between them.

5) The storage array, after receiving the read or write request from the host, performs the necessary operations to store the data on physical disks.

Key characteristics of data center elements are:

- Key characteristics of data center elements are:

1) Availability: All data center elements should be designed to ensure accessibility. The inability of users to access data can have a significant negative impact on a business.

2) Security: Polices, procedures, and proper integration of the data center core elements that will prevent unauthorized access to information must be established. Specific mechanisms must enable servers to access only their allocated resources on storage arrays.

Key characteristics of data center elements are:

- Scalability: Data center operations should be able to allocate additional processing capabilities (eg: servers, new applications, and additional databases) or storage on demand, without interrupting business operations.
- Performance: All the core elements of the data center should be able to provide optimal performance and service all processing requests at high speed. The infrastructure should be able to support performance

requirements.

Key characteristics of data center elements are:

- Data integrity: Data integrity refers to mechanisms such as error correction codes or parity bits which ensure that data is written to disk exactly as it was received. Any variation in data during its retrieval implies corruption, which may affect the operations of the

organization.

Key characteristics of data center elements are:

- Capacity: Data center operations require adequate resources to store and process large amounts of data efficiently. When capacity requirements increase, the data center must be able to provide additional capacity without interrupting availability, or, at the very least, with minimal disruption. Capacity may be managed byreallocation of existing resources, rather than by adding new resources.

Key characteristics of data center elements are:

- Manageability: A data center should perform all operations and activities in the most efficient manner. Manageability can be achieved through automation and the reduction of human (manual) intervention in common tasks

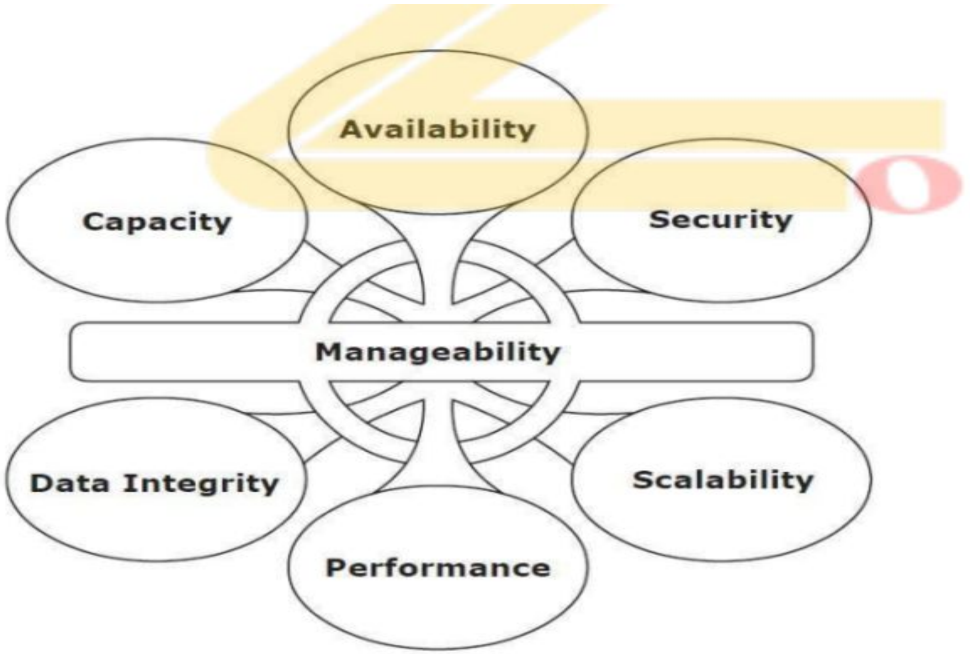


Fig 1.6: Key characteristics of data center elements

Virtualization

- Virtualization is a technique of **abstracting physical resources** such as compute, storage, and network, and making them appear as **logical resources.**
- It has been used in IT for many years in different forms, such as virtual memory and disk partitioning.
- Virtualization enables pooling of multiple physical resources to provide a single aggregated view—for example, storage devices appearing as one large storage system or CPU power of multiple servers combined together.

Virtualization

- It allows centralized management and easy creation of virtual resources like virtual disks or virtual servers with specific CPU and memory.
- These virtual resources share physical resources, leading to better utilization.
- Capacity can be added or removed based on business needs without affecting users or applications.
- Overall, virtualization reduces cost, saves space and energy, and supports green computing.

Cloud computing

- Cloud computing allows people or businesses to use IT resources (like servers, storage, databases, software, and networks) as a service over the internet instead of owning physical hardware.
- It gives scalable and flexible computing, meaning resources can be increased or decreased easily, whenever needed.
- Users can scale up or down storage, processing power, or other resources with very little effort and without needing to contact the service provider.

Cloud computing

- It supports self-service, meaning users can request and use resources automatically through online systems, without manual setup.
- It follows a pay-as-you-go model, so users only pay for what they actually use (e.g., CPU hours, data storage, or data transfer).
- Cloud systems are usually built on virtualized data centers, which pool resources and make them quickly available when requested.

Data Center Environment

Key Data center Elements

- Application
- An application is a computer program that provides the logic for computing operations.
- The application sends requests to the underlying operating system to perform read/write (R/W) operations on the storage devices.
- Applications deployed in a data center environment are commonly categorized as business applications, infrastructure management applications, data protection applications, and security applications.
- Some examples of these applications are e-mail, enterprise resource planning (ERP),decision support system (DSS), resource management, backup, authentication and antivirus applications, and so on

- DBMS
- A database is a structured way to store data in logically organized tables that are interrelated.
- A DBMS controls the creation, maintenance, and use of a database.

- Host(or) Compute
- The computers on which applications run are referred to as hosts. Hosts can range from simple laptops to complex clusters of servers.
- Hosts can be physical or virtual machines.
- A compute virtualization software enables creating virtual machines on top of a physical compute infrastructure.

- A host consists of

- CPU: The CPU consists of four components-Arithmetic Logic Unit (ALU), control unit, registers, and L1 cache
- Memory: There are two types of memory on a host, Random AccessMemory (RAM) and Read-Only Memory (ROM)
- I/O devices : keyboard, mouse, monitor
- a collection of software to perform computing operations- This software includes the operating system, file system, logical volume manager, device drivers, and so on.

- Operating System
- In a traditional computing environment, an operating system controls all aspects of computing.
- It works between the application and the physical components of a compute system.
- In a virtualized compute environment, the virtualization layer works between the operating system and the hardware resources.

- Functions of OS
- data access
- monitors and responds to user actions and the environment
- organizes and controls hardware components
- manages the allocation of hardware resources
- It provides basic security for the access and usage of all managed resources
- performs basic storage management tasks
- manages the file system, volume manager, and device drivers.

Memory virtualization

- **Memory virtualization** is a technique that allows a computer to use more memory than the physical RAM available by creating *virtual memory*. I
- t is managed by the Virtual Memory Manager (VMM), which uses a portion of the disk called **swap space** (or page file) as if it were physical memory.
- The system divides memory into fixed-size pages, and a process called **paging** moves inactive pages from RAM to the swap space and brings them back when needed.
- This helps run larger or more applications on a host, even with limited physical memory.

- Device Drivers
- A device driver is special software that permits the operating system to interact with a specific device, such as a printer, a mouse, or a disk drive.

- A **Volume Manager** (specifically a Logical Volume Manager or LVM) is software that sits between the file system and the physical disk to provide flexible and efficient storage management. Earlier, entire disks had to be allocated to a file system, which caused problems like lack of flexibility, difficulty in extending storage, and underutilization of space. LVM solved this by allowing disks to be divided into smaller **logical volumes (LVs)** through partitioning or by combining multiple smaller disks into one large **logical volume** through concatenation.

- The main components of LVM are **physical volumes (PVs)**, which are the actual disks, **volume groups (VGs)**, which are collections of PVs, and **logical volumes (LVs)**, which are partitions created within VGs. This setup makes it possible to dynamically extend file system capacity and better utilize available storage.

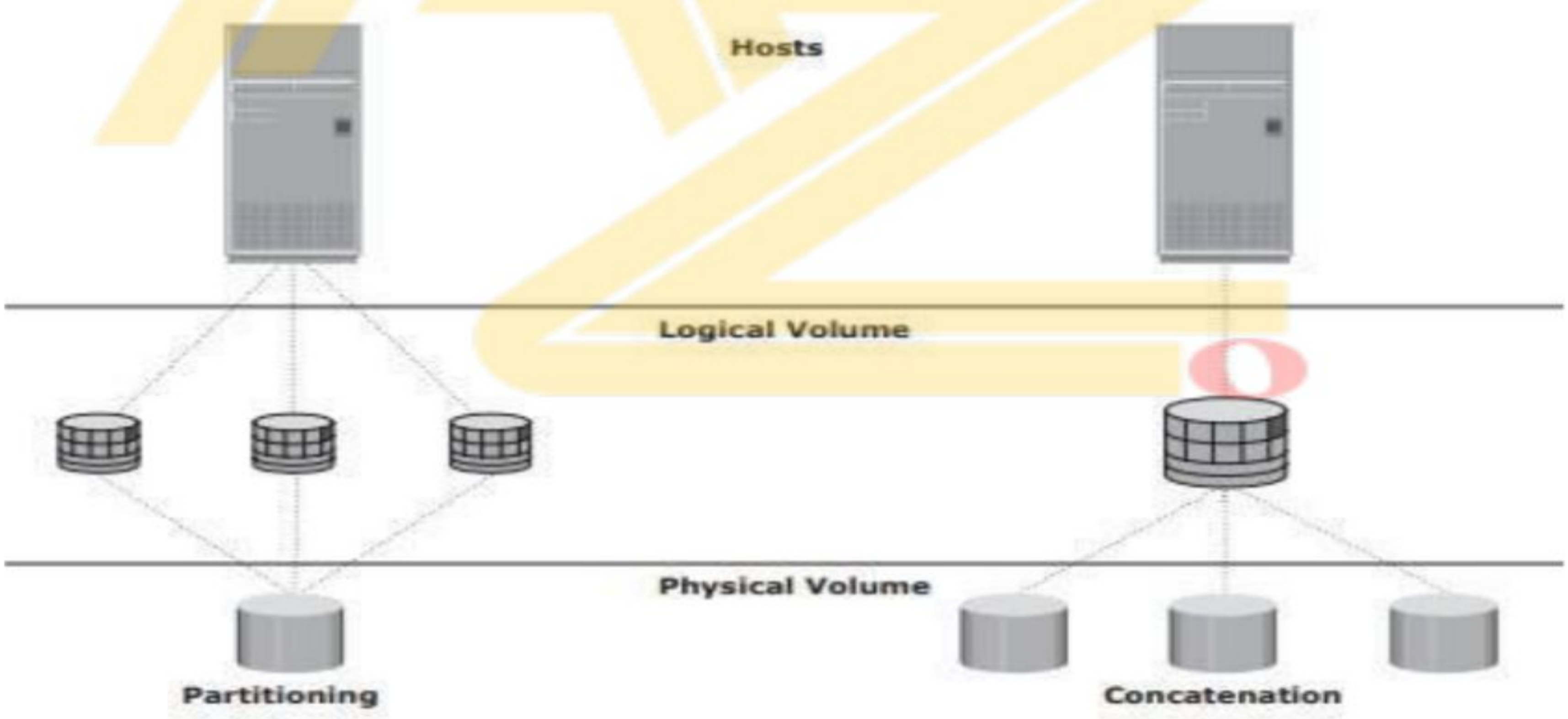


Fig 1.7: Disk Partitioning and concatenation

File System

- File Systemof related records or data stored as a unit with a name.
- A file system is a hierarchical structure of files.
- A file system enables easy access to data files residing within a disk drive, a disk partition, or a logical volume.
- It provides users with the functionality to create, modify, delete, and access files.
- Access to files on the disks is controlled by the permissions assigned to the file by the owner, which are also maintained by the file system.
- A file system organizes data in a structured hierarchical manner via the use of directories, which are containers for storing pointers to multiple files.

- All file systems maintain a pointer map to the directories, subdirectories, and files that are part of the file system.
- Examples of common file systems are:
- FAT 32 (File Allocation Table) for Microsoft Windows
- NT File System (NTFS) for Microsoft Windows
- UNIX File System (UFS) for UNIX
- Extended File System (EXT2/3) for Linux

Volume Manager

- In the beginning, computers saw a disk as **one big block of storage**.
- The **entire disk** had to be given to **one file system** or **one application**.
- This caused problems:
    - **No flexibility** – you couldn’t easily share or resize space.
    - If the disk became **full**, there was no way to increase its size.
    - If the disk was **too large**, giving it all to one file system wasted space (underutilization).
- Solution → Logical Volume Manager (LVM)

Logical Volume Manager (LVM)

- Acts like a **middle layer** between the file system and the physical disk.
- **Features**:
- **Partitioning** : Split a big disk into smaller, virtual parts called **Logical Volumes (LVs)**.
  (Like dividing a notebook into separate sections for different subjects).
- **Concatenation** : Combine many small disks to act as one big virtual disk.
  (Like joining many notebooks to look like a single large notebook).
- **Dynamic extension** :If one logical volume gets full, you can easily extend its size

- If you have many small disks, you can join them together so the computer sees them as one big disk.
- This big disk is called a logical volume.

Basic LVM Components

- **Physical Volume (PV)**
- Each real hard disk (or part of it) that is connected to the system.
- Example: Think of each PV as a **brick**.
- **Volume Group (VG)**
- A collection of one or more physical volumes.
- When you create a VG, each PV inside it is divided into equal pieces called **Physical Extents (PEs)**.
- Example: A VG is like a **wall** built by combining many bricks.
- **Logical Volume (LV)**
- Created inside a Volume Group.
- Works like a **virtual disk partition** that applications or the operating system can use.
- Example: If the VG is the whole **wall**, the LV is like a **window or door** made in that wall, customized for use.

Disk Partitioning and Concatention

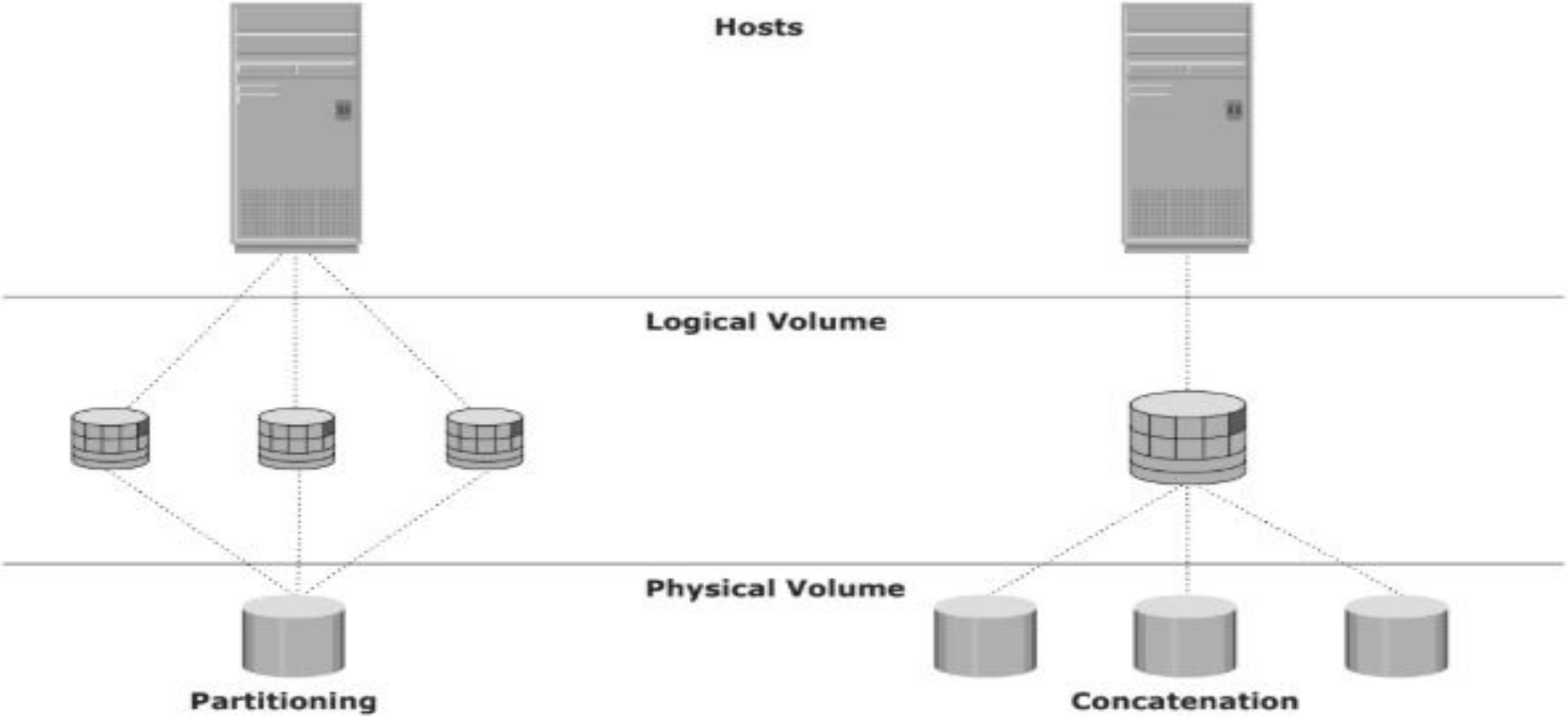


Figure 2-1: Disk partitioning and concatenation

- **1. Partitioning (left side)**
- A **single physical disk** is divided into **multiple smaller parts (partitions)**.
- Each partition acts like a separate disk for the computer.
- Example: A 500 GB disk split into 3 partitions of 100 GB, 200 GB, and 200 GB.
- **2. Concatenation (right side)**
- **Multiple physical disks** are combined to look like **one big logical disk**.
- The computer sees them as a single volume, even though they are multiple disks.
- Example: Three 200 GB disks combined to form one large 600 GB logical disk.
- **Partitioning:** Split one disk into many smaller parts.
- **Concatenation:** Combine many disks into one big volume.

File System

- **File and File System**
- A **file** is just a collection of related data stored together under a name.
- A **file system** is like a **hierarchical structure (tree)** that organizes these files.

What a File System Does

- Stores files inside a **disk, disk partition, or logical volume**.
- Lets users **create, modify, delete, and open files**.
- Controls **permissions** (who can read, write, or execute a file).
- Organizes files into **directories and subdirectories** (like folders in Windows).
- Keeps a **map (pointers)** that tells where each directory, subdirectory, and file is located.

- **Examples of File Systems**
- **FAT32** – old Windows file system.
- **NTFS** – modern Windows file system.
- **UFS** – used in UNIX.
- **EXT2/EXT3** – used in Linux.

- **Metadata (extra info about files)**
- The file system also stores **metadata** (data about data).
- In UNIX, metadata includes:
  - **Superblock** → overall info about the file system (type, size, creation date, layout).
  - **Inodes** → info about each file (size, owner, permissions, last modified date, links, and where the file's data is stored).
  - **Free block list** → shows which blocks are free or already used.

Process of mapping user files to disk Storage

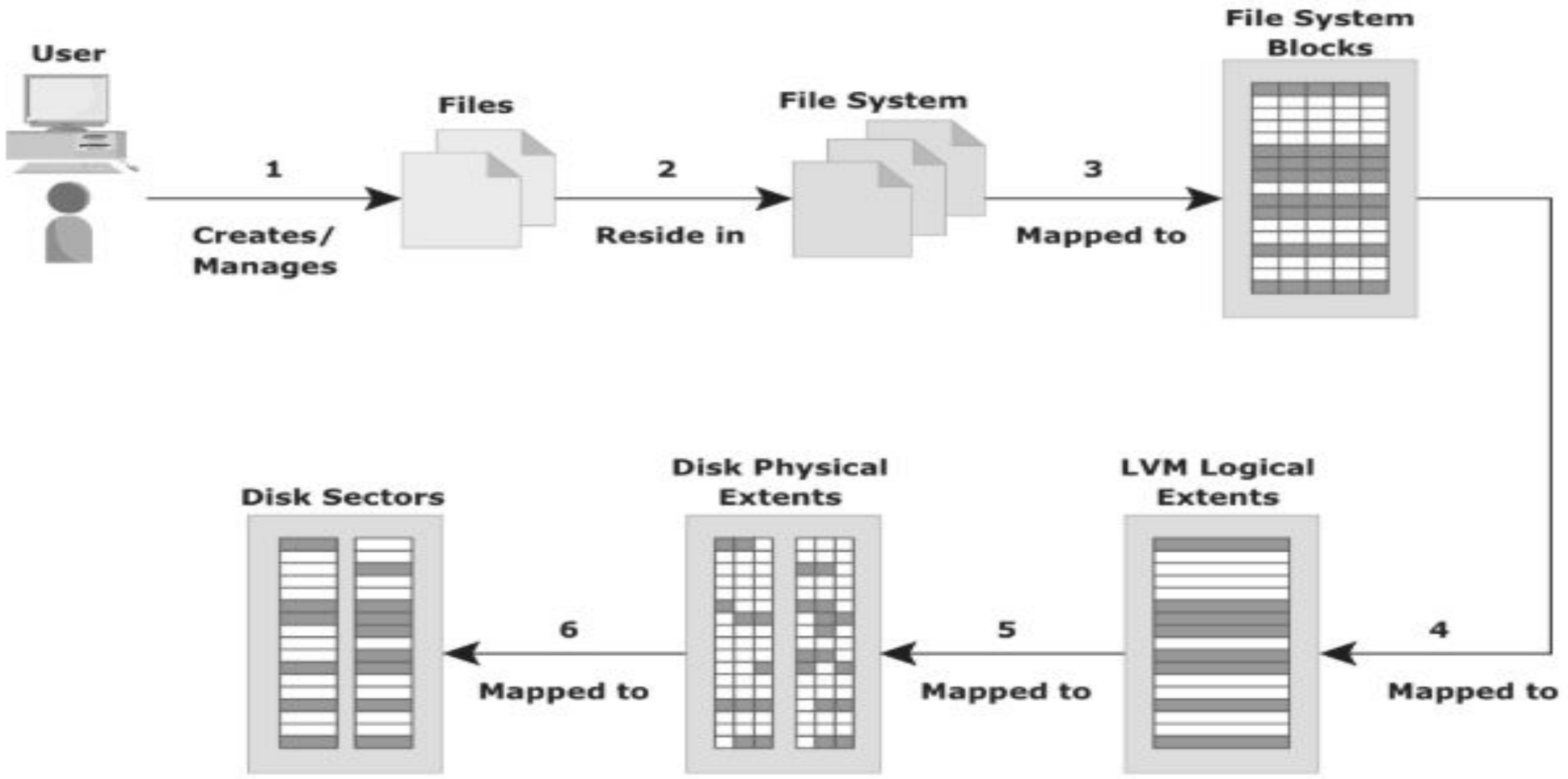


Figure 2-2: Process of mapping user files to disk storage

1. User creates files → Applications and users make files (like documents, images, etc.).
2. Files go into a file system → These files are placed inside a file system (like NTFS, EXT3, etc.).
3. File system uses blocks → The file system breaks the files into small fixed-size units called file system blocks.
4. Blocks mapped to logical extents → These blocks are linked to logical extents inside a Logical Volume (LV).
5. Logical extents mapped to physical extents → The Logical Volume Manager (LVM) or OS connects those logical extents to real physical extents on disks.
6. Physical extents stored in sectors → Finally, the physical extents are written into the actual disk sectors (the smallest physical storage unit in a hard drive).

Compute virtualization

- Compute virtualization is a technique that hides the physical hardware from the operating system, allowing multiple virtual machines (VMs) to run on a single physical server.
- Each VM behaves like a separate computer with its own CPU, memory, storage, and network, even though they all share the same underlying hardware.
- This is managed by a **hypervisor**, which allocates resources to each VM and keeps them isolated from one another. By enabling multiple operating systems and applications to run on one machine, compute
- virtualization improves server utilization, reduces hardware costs, and allows for easier server management and consolidation.

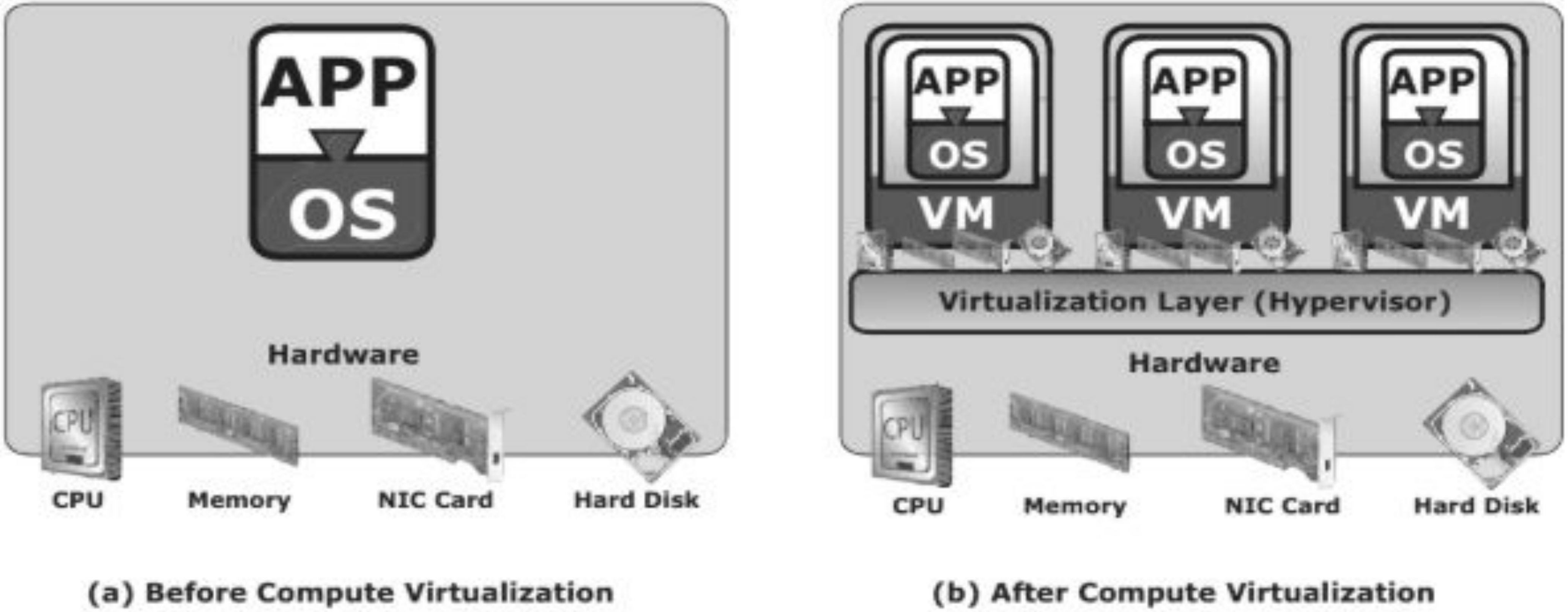


Figure 2-3: Server virtualization

- **Before virtualization (a):** One physical server could run only one operating system and one application, so most hardware resources stayed unused.
- **After virtualization (b):** A **hypervisor** is added, which lets one physical server run **many virtual machines (VMs)**. Each VM has its own operating system and application, and they all share the server’s hardware efficiently. This saves cost, uses resources better, and reduces the need for many physical servers.

Connectivity

- Connectivity means how a host (like a computer) connects to storage devices (like hard drives or storage systems).
- This connection is made possible through physical parts (like cables or ports) and specific communication rules called interface protocols.

Physical Components of Connectivity

- The physical components of connectivity are the hardware parts that link a host (like a computer) to storage devices. There are three main components: the host interface device, port, and cable.
- A **host interface device**, also known as a **host adapter**, connects the host to other hosts or storage devices. Common examples include the **Host Bus Adapter (HBA)** and **Network Interface Card (NIC)**. The HBA is a special type of hardware (an ASIC board) that handles data transfer between the host and storage, reducing the load on the CPU. Most hosts have multiple HBAs.

- A **port** is a connection point on the host or adapter that allows communication with storage devices. One HBA can have one or more ports.
- **Cables** physically connect the host to storage devices. These can use **copper** or **fiber optic** materials to transmit data.

Physical Component of Connectivity

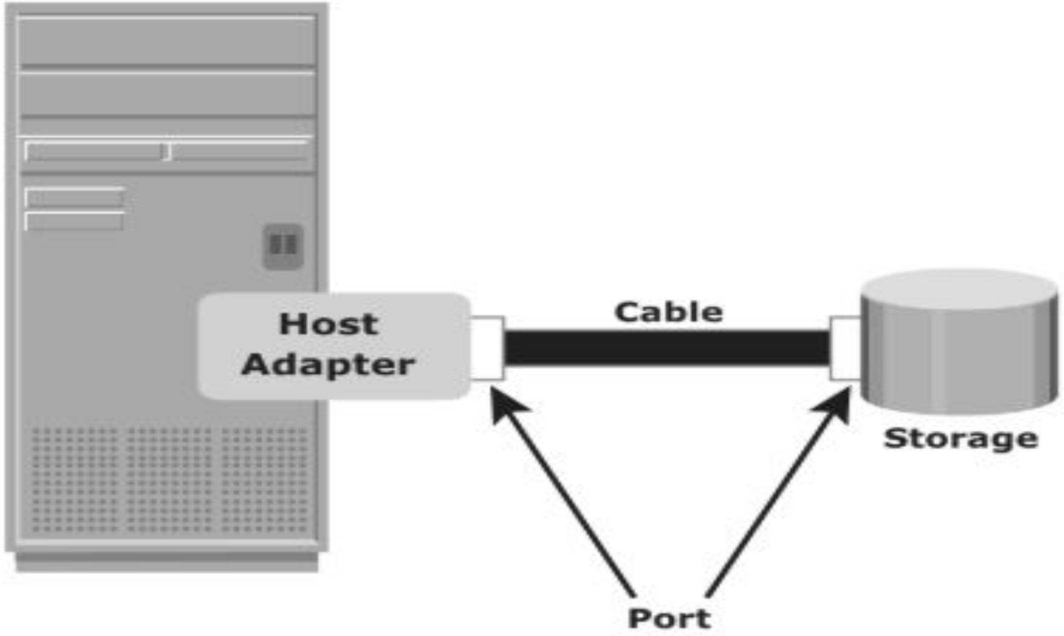


Figure 2-4: Physical components of connectivity

- **Host Adapter**: This is a hardware component inside the host (often an HBA - Host Bus Adapter). It allows the host to communicate with storage devices.
- **Cable**: This is the physical connection (usually copper or fiber optic) that carries the data between the host and the storage.
- **Port**: These are the connection points on both the host adapter and the storage device where the cable is plugged in.
- **Storage**: This is the external storage device (such as a disk array or SAN) that stores the data.

Interface Protocols

- A **protocol** is a set of rules that allows the **host** (like a computer) and the **storage device** to talk to each other. These protocols work through **interface devices** (also called controllers) on both ends — at the host and at the storage.
- Some common **protocols** used to connect hosts to storage devices include:
- **IDE/ATA (Integrated Device Electronics/Advanced Technology Attachment)**
- **SCSI (Small Computer System Interface)**
- **Fibre Channel (FC)**
- **IP (Internet Protocol)**
- These protocols make sure data is sent and received properly between the host and storage.

IDE/ATA and Serial ATA

- <u>**IDE/ATA and Serial ATA**</u>
- **IDE/ATA** is an older way to connect storage devices like hard drives and CD-ROMs to a computer. It sends many bits of data at once, so it’s also called **Parallel ATA (PATA)**.
- One ATA connection can link **two devices**, but this can slow things down. So, for better performance, it's best to use one device per connection.
- **SATA (Serial ATA)** is a newer version that sends **one bit at a time**, but much faster. It's also cheaper and better for performance. **SATA 3.0** can move data very quickly up to **6 gigabits per second**.
- Today, **SATA has mostly replaced IDE/PATA** in modern computers.

SCSI

- **SCSI** is a protocol often used in high-end computers to connect storage devices. It sends data in **parallel** and gives **better speed, flexibility, and compatibility** than ATA. However, it's more **expensive**, so it's not common in regular home computers.
- SCSI has improved over time and now includes many versions. It can connect **up to 16 devices** on one connection and can transfer data at speeds up to **640 MB per second** (with Ultra-640 SCSI).
- **Serial Attached SCSI (SAS)** is a newer version that uses **serial (one bit at a time)** communication instead of parallel. It connects devices **one-to-one (point-to-point)**. The updated version, **SAS 2.0**, can transfer data up to **6 gigabits per second**.

Fibre Channel

- **Fibre Channel** is a popular protocol used for **fast communication** between computers and storage devices.
- It supports **very high speeds**, similar to a **gigabit network**.
- Fibre Channel sends data **serially** (one bit at a time) and can work over both **copper cables** and **optical fiber**.
- The latest version, called **16FC**, can transfer data at speeds up to **16 gigabits per second (Gb/s)**.

Internet Protocol (IP)

- IP is a network system usually used to connect one computer to another. Recently, new technologies have made it possible to use IP to connect computers to storage devices as well. Using IP has benefits like being cheaper and well-developed, and it lets companies use their current IP networks. Two common examples of this are iSCSI and FCIP, which use IP to connect computers to storage.

Storage

- Storage is very important in a data center.
- Storage devices save information using three main types: magnetic, optical, or solid-state.
- Magnetic storage includes things like disks and tapes.
- Optical storage means CDs and DVDs.
- Solid-state storage includes flash drives.

- Tapes used to be popular for backups because they were cheap. But tapes have problems:
- You have to go through the tape step by step to find data, which takes time.
- Only one program can use the tape at a time.
- Tapes get worn out because the machine touches the tape.
- It’s hard to manage and use tapes.

- Because of these problems and cheaper disks, tapes are not used much anymore in big data centers.
- Optical discs like CDs and DVDs are used by people to save pictures, back up files, or share small programs. But they don’t hold a lot of data and are slow, so businesses don’t use them much.
- One good thing about optical discs is you can write data once and read it many times without changing it. For example, CD-ROMs do this. This is good for storing data that doesn’t need to change. Sometimes many discs are stored together in a machine called a jukebox.
- Disk drives are now the most common storage. They are fast and can handle many users at the same time. They also hold a lot of data. Sometimes many disks are put together to store even more data and work faster.

Disk Drive Components

- **Disk Drive Components**
  A hard disk drive has several main parts: the platter, spindle, read-write head, actuator arm, and controller board.
- The drive works by moving the arm quickly over spinning platters that are covered with magnetic material.
- The read-write head on the arm reads and writes data on the platters.
- Data can be written and erased on the platters many times.
  The next sections explain the parts of the disk drive, how data is stored on disks, and what affects the disk's speed.

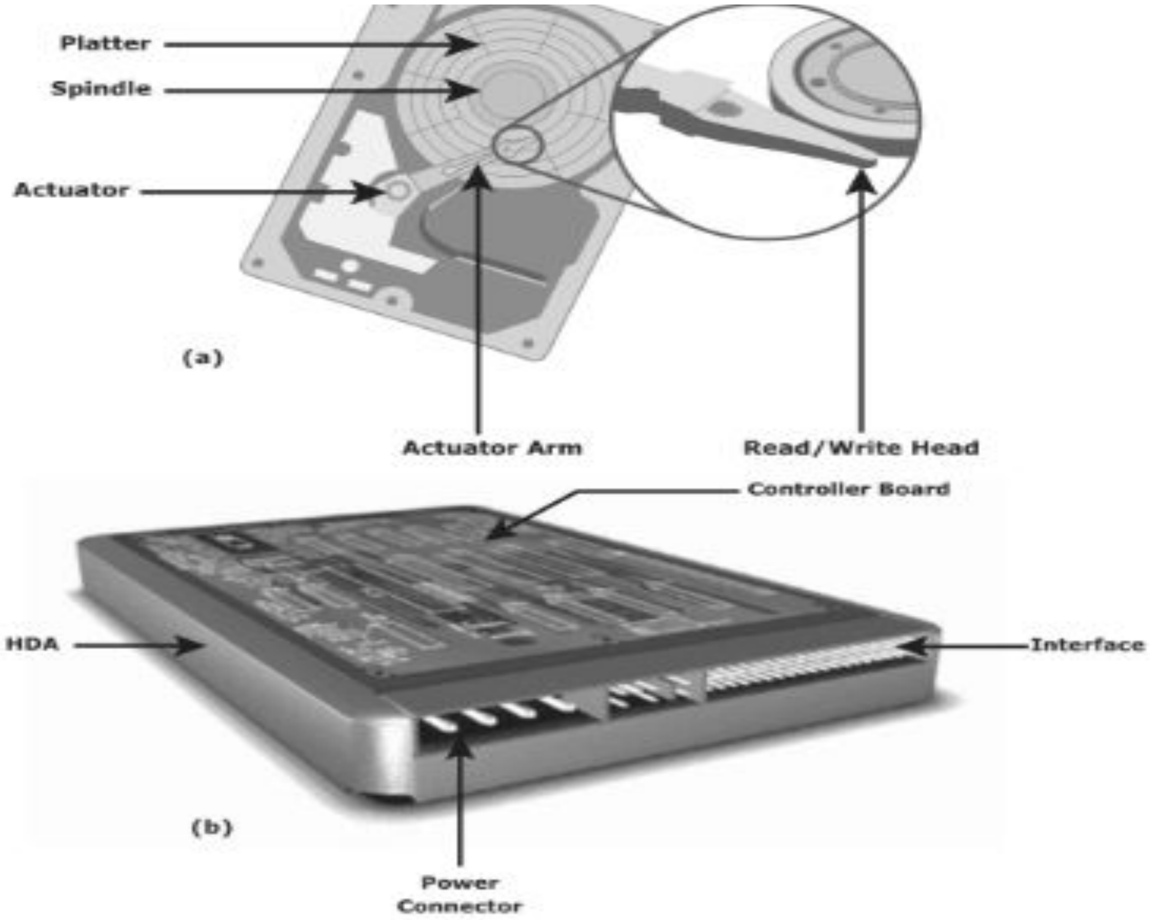


Figure 2-5: Disk drive components

Disk Drive Components

- **Platter**
- A hard disk drive has one or more flat, round disks called platters
- Data is stored on these platters using 0s and 1s.

- The platters spin inside a sealed case called the Head Disk Assembly (HDA).

- Each platter is a hard disk coated with magnetic material on both sides.
- Data is saved by changing the magnetic areas on the surface.

- Data can be read or written on both sides of the platter.

- The total storage of the drive depends on how many platters it has and how much each one can hold

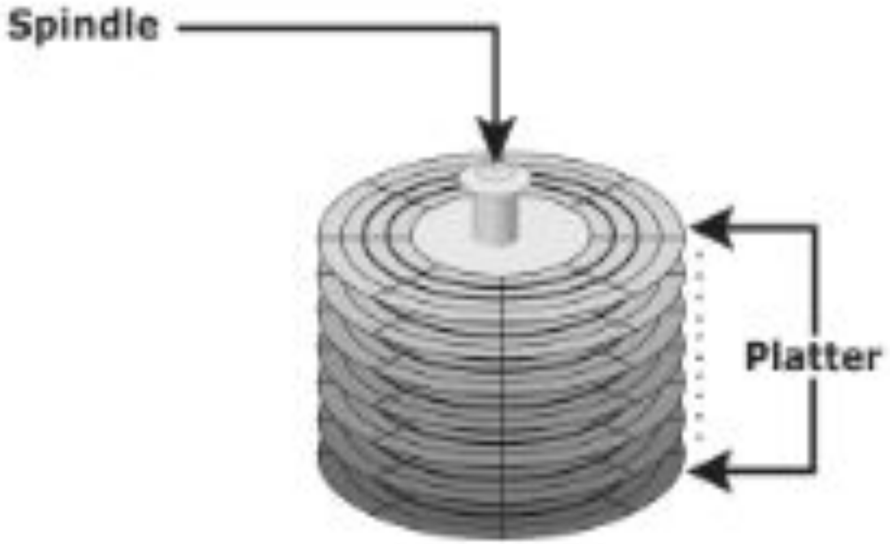


Figure 2-6: Spindle and platter

Spindle

- The spindle holds all the platters together (see Figure 2-6) and is attached to a motor.
- The motor spins the platters at a steady speed.Platters usually spin thousands of times per minute, such as 5,400, 7,200, 10,000, or 15,000 revolutions per minute (rpm).
- With new technology, platter speeds are getting faster, but there is a limit to how fast they can go.

The read/write (R/W) heads

- The read/write (R/W) heads, shown in Figure 2-7, read data from and write data to the platters.
- Each platter has two R/W heads, one for each side.
  When writing, the head changes the magnetic pattern on the platter.
- When reading, it senses the magnetic pattern.
  The R/W head doesn’t touch the platter surface while reading or writing.
  When the platters spin, a tiny air gap called the “head flying height” keeps the head just above the platter.
  When the platters stop spinning, the head rests on a special part of the platter near the spindle called the landing zone.
  The landing zone has a lubricant to reduce rubbing between the head and platter.

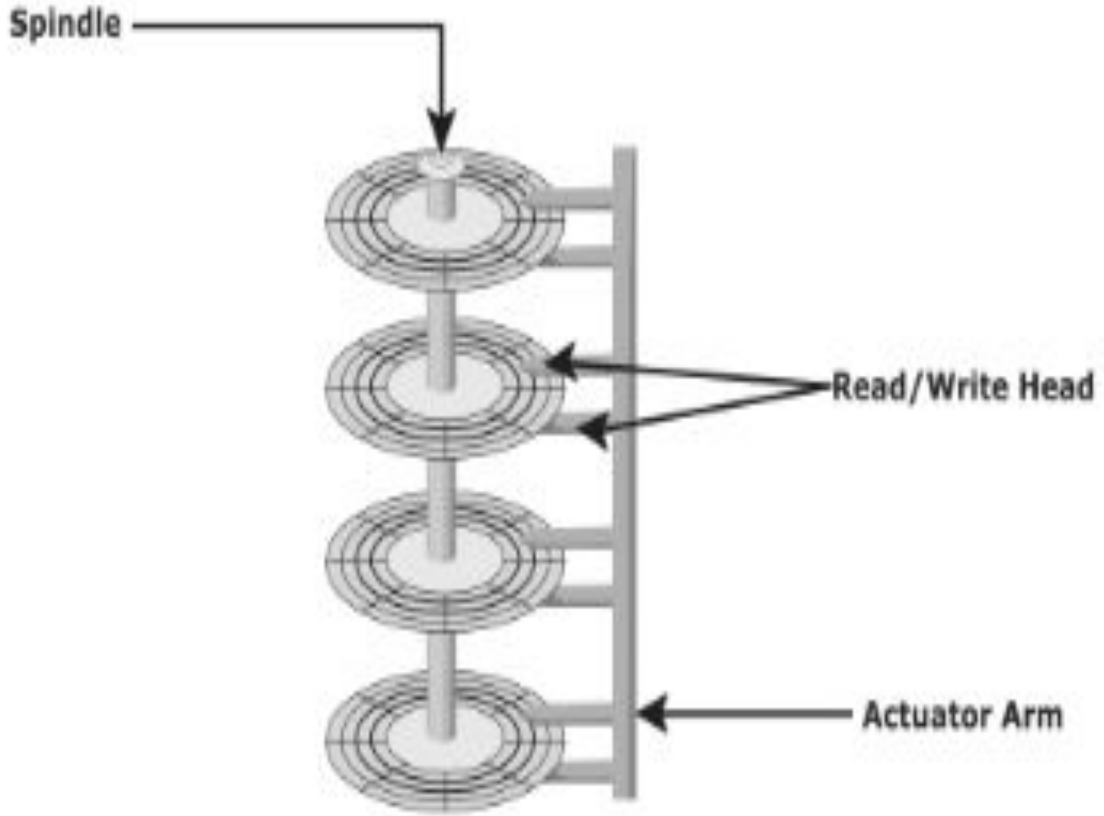


Figure 2-7: Actuator arm assembly

- **Actuator Arm Assembly**
  The read/write heads are fixed to the actuator arm assembly, which moves the heads to the right spot on the platters to read or write data (see Figure 2-7).
  All the heads for the platters move together on this one arm.

- **Drive Controller Board**
- The controller is a circuit board attached to the bottom of the disk drive.
- It has a small computer (microprocessor), memory, and special software (firmware).
- The firmware controls the power and speed of the motor that spins the platters.
- It also manages how the drive talks to the computer.
- The controller moves the actuator arm, switches between read/write heads, and helps make data access faster.

- **Physical Disk Structure**
- Data is stored in circles called tracks on the disk. Tracks are counted starting from the outside edge. Tracks are packed close together.
- Each track is split into small parts called sectors. Sectors are the smallest parts of storage you can use. The disk is made with these tracks and sectors already set up. Older disks had fewer sectors per track, new ones have more. There can be thousands of tracks on one disk.

Physical Disk Structure

- Data is stored in circles called tracks on the disk. Tracks are counted starting from the outside edge. Tracks are packed close together.
- Each track is split into small parts called sectors. Sectors are the smallest parts of storage you can use. The disk is made with these tracks and sectors already set up. Older disks had fewer sectors per track, new ones have more. There can be thousands of tracks on one disk.

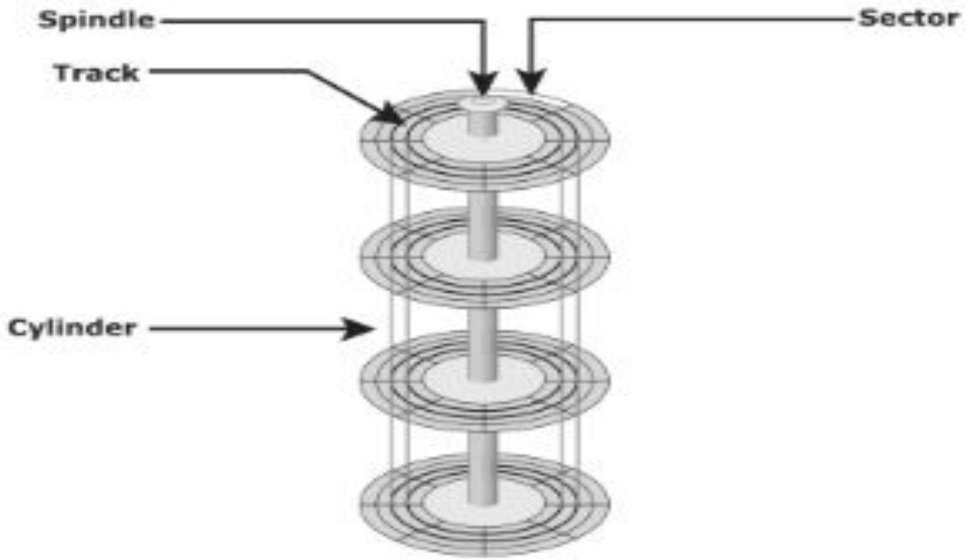


Figure 2-8: Disk structure: sectors, tracks, and cylinders

Zoned Bit Recording

Disk platters have circular tracks. Outer tracks are longer than inner tracks, so they can hold more data.
Older disks used the same number of sectors on all tracks, which wasted space on the outer tracks.
Zoned bit recording fixes this by dividing tracks into zones based on how far they are from the center.

Zoned Bit Recording

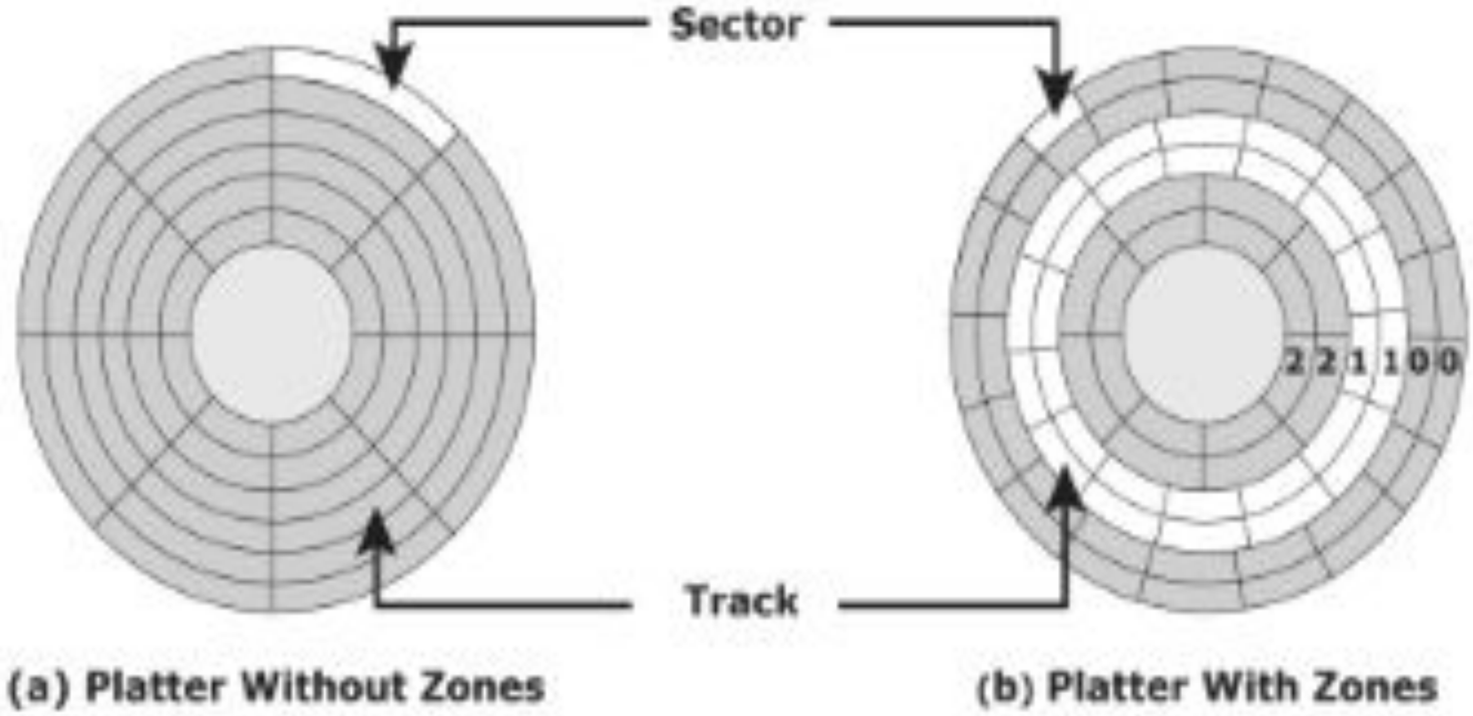


Figure 2-9: Zoned bit recording

- **Logical Block Addressing**
  Older drives used physical addresses called cylinder, head, and sector (CHS) to find data on the disk .
- The computer needed to know the disk’s details.
  Logical Block Addressing (LBA) makes this easier by using a single number to find data
- The disk controller changes the LBA number into the physical CHS address.
- The computer only needs to know how many blocks the disk has.
  Each logical block matches one physical sector on the disk.

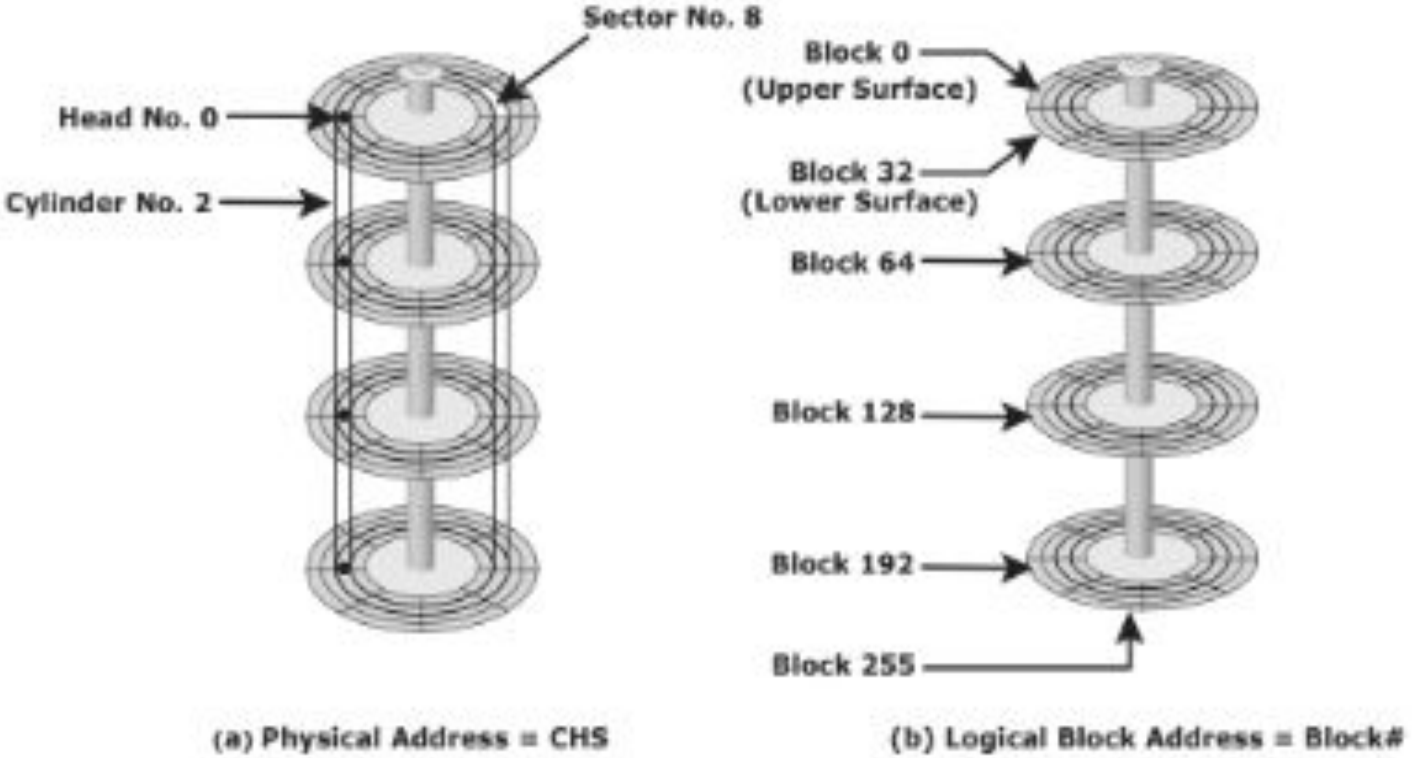


Figure 2-10: Physical address and logical block address

Logical Block Addressing

- (a) Physical Address (CHS):This way of addressing data on a hard drive uses three parts:
- Cylinder: Think of it as a stack of circles on the disk.
- Head: The part that reads the data on one circle.
- Sector: A small section on the circle where data is stored.So, the location of data is given by which cylinder, which head, and which sector it is on.
- (b) Logical Block Address (Block#):Instead of using three parts, this method just numbers each block of data one by one from 0 upwards.This makes it easier to find data because you just use the block number.

Disk service time

- Disk service time is the total time a disk takes to finish an I/O request. It depends on three parts:
- Seek time,
- Rotational delay,
- Data transfer speed.

Seek time

- Seek time (also called access time) is the time the disk takes to move its read/write head to the right place.

Seek Time Types:

- **Full Stroke:** Time for the read/write head to move from the innermost track to the outermost track.
- **Average:** Average time to move from one random track to another (usually about one-third of a full stroke).
- **Track-to-Track:** Time to move between two nearby tracks.
- These times are measured in milliseconds. Modern disks usually have an average seek time of 3 to 15 ms. Seek time affects random track reading more than nearby tracks.
- To reduce seek time, data can be stored only on part of the disk (for example, using only 40% of a 500 GB disk, making it work like a 200 GB disk). This method is called **short-stroking**.

Rotational Latency

When the disk spins, the read/write head waits for the correct sector to come under it.

The time taken for this spin is called **rotational latency**, measured in milliseconds.

- The faster the disk spins, the lower the latency. On average, it is half the time of one full rotation.
- A 5,400 rpm disk has about **5.5 ms latency**.
- A 15,000 rpm disk has about **2 ms latency** (since $0.5 \div 250 = 2$ ms).

- **Data Transfer Rate**
  The data transfer rate is the average speed at which data moves between the disk and the computer (HBA).
- **Read operation:** Data goes from disk platters → read/write heads → disk buffer → interface → HBA.
- **Write operation:** Data goes from HBA → interface → buffer → read/write heads → disk platters.

- There are two types of transfer rates:

- Internal transfer rate: Speed of data moving from the platter to the buffer (affected by seek time and rotational delay).

- External transfer rate: Speed of data moving from the buffer to the HBA through the interface (example: ATA 133 MB/s).

Data Transfer Rate

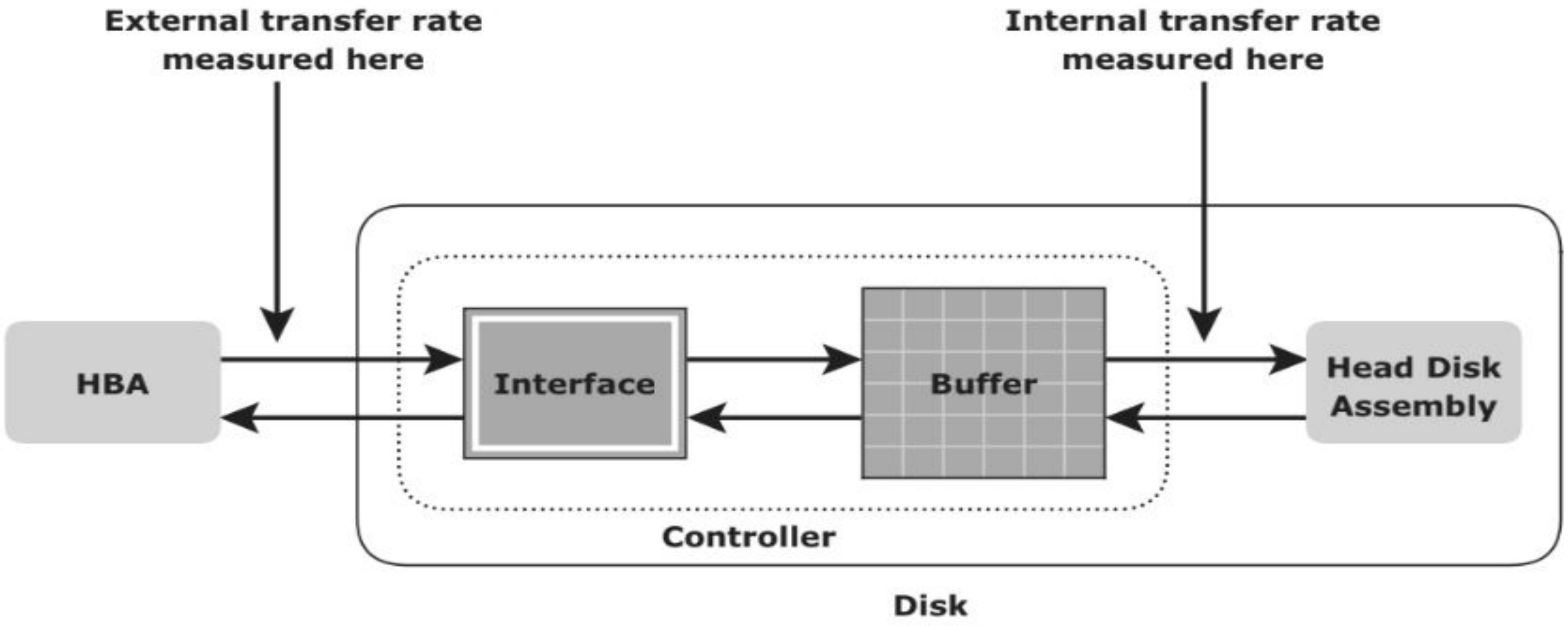


Figure 2-11: Data transfer rate

Disk I/O Controller Utilization

- The use of a disk I/O controller affects how fast I/O requests are handled.
- A disk can be seen as two parts:
- **Queue:** Where I/O requests wait before being handled.
- **Disk I/O Controller:** Handles the requests one by one.
- The requests arrive at a certain speed (arrival rate) and wait in the queue until the controller processes them.
- The response time depends on how many requests come in, how long the queue is, and how fast the controller processes each request.

I/O Processing

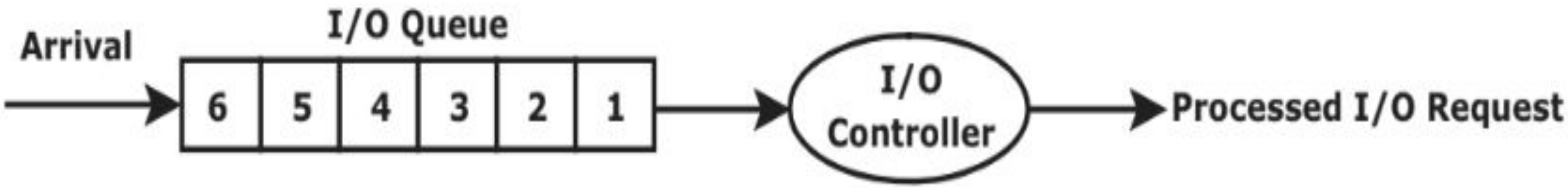


Figure 2-12: I/O processing

I/O Processing

1. Arrival: New I/O requests (like 6, 5, 4 …) enter the system.
2. I/O Queue: The requests wait in line (queue) before being handled.
3. I/O Controller: Takes the requests one by one from the queue and processes them.
4. Processed I/O Request: After processing, the requests are sent out as completed.

$$\text{Average response time } (T_R) = \text{Service time } (T_S) \,/\, (1 - \text{Utilization})$$

- **TS** = time the controller takes to finish one I/O.
- **Utilization** = how busy the controller is.

As utilization gets closer to **100%** (controller fully busy), the response time becomes extremely large (almost infinite).

Utilization versus response time

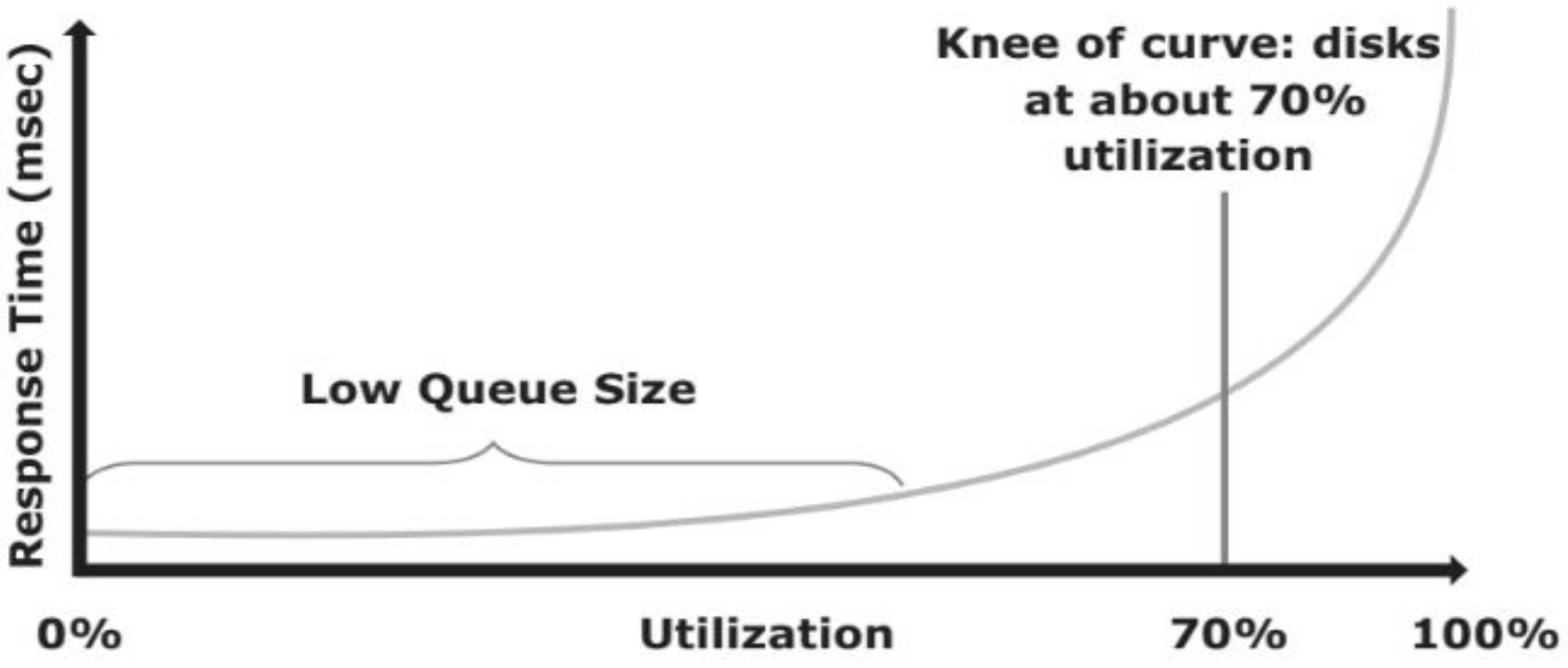


Figure 2-13: Utilization versus response time

- The graph shows the relation between **disk utilization** and **response time**:
- When the disk is used less (low utilization), the response time is small because the queue is short.
- As utilization increases, response time also increases.
- Around **70% utilization** (knee of the curve), the response time starts rising sharply.
- When utilization is close to **100%**, the response time becomes very high because many requests are waiting.