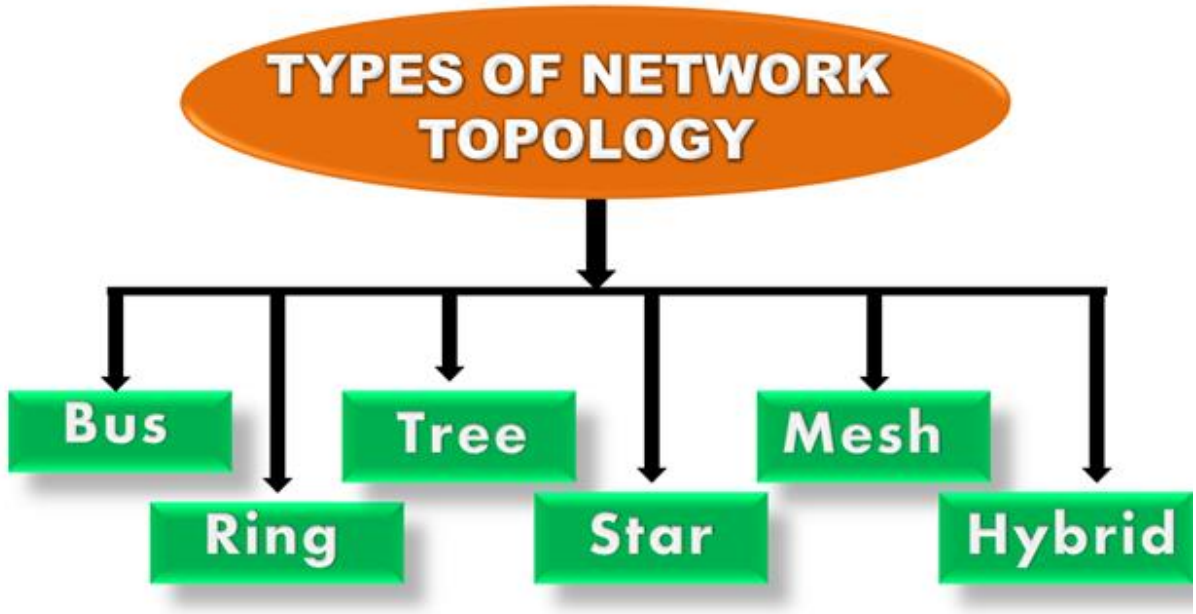
Unit 3

Topology

# What is Topology?

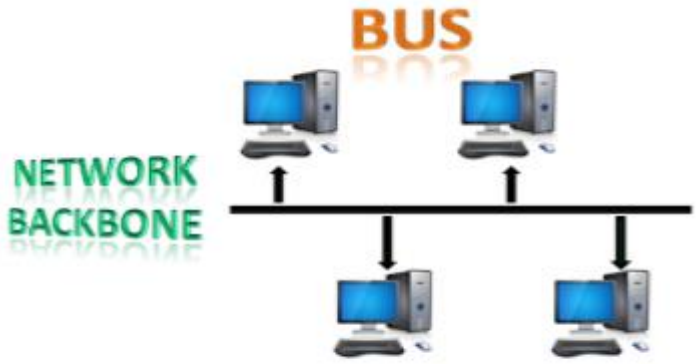
Topology defines the structure of the network of how all the components are interconnected to each other. There are two types of topology: physical and logical topology.

Physical topology is the geometric representation of all the nodes in a network.




---

## Bus Topology



- The bus topology is designed in such a way that all the stations are connected through a single cable known as a backbone cable.
- Each node is either connected to the backbone cable by drop cable or directly connected to the backbone cable.
- When a node wants to send a message over the network, it puts a message over the network. All the stations available in the network will receive the message whether it has been addressed or not.
- The bus topology is mainly used in 802.3 (ethernet) and 802.4 standard networks.
- The configuration of a bus topology is quite simpler as compared to other topologies.
- The backbone cable is considered as a **"single lane"** through which the message is broadcast to all the stations.
- The most common access method of the bus topologies is **CSMA** (Carrier Sense Multiple Access).

**CSMA:** It is a media access control used to control the data flow so that data integrity is maintained, i.e., the packets do not get lost. There are two alternative ways of handling the problems that occur when two nodes send the messages simultaneously.

- **CSMA CD:** CSMA CD (**Collision detection**) is an access method used to detect the collision. Once the collision is detected, the sender will stop transmitting the data. Therefore, it works on "**recovery after the collision**".
- **CSMA CA: CSMA CA (Collision Avoidance)** is an access method used to avoid the collision by checking whether the transmission media is busy or not. If busy, then the sender waits until the media becomes idle. This technique effectively reduces the possibility of the collision. It does not work on "recovery after the collision".

## Advantages of Bus topology:

- **Low-cost cable:** In bus topology, nodes are directly connected to the cable without passing through a hub. Therefore, the initial cost of installation is low.
- **Moderate data speeds:** Coaxial or twisted pair cables are mainly used in bus-based networks that support upto 10 Mbps.

- **Familiar technology:** Bus topology is a familiar technology as the installation and troubleshooting techniques are well known, and hardware components are easily available.
- **Limited failure:** A failure in one node will not have any effect on other nodes.

### Disadvantages of Bus topology:

- **Extensive cabling:** A bus topology is quite simpler, but still it requires a lot of cabling.
- **Difficult troubleshooting:** It requires specialized test equipment to determine the cable faults. If any fault occurs in the cable, then it would disrupt the communication for all the nodes.
- **Signal interference:** If two nodes send the messages simultaneously, then the signals of both the nodes collide with each other.
- **Reconfiguration difficult:** Adding new devices to the network would slow down the network.
- **Attenuation:** Attenuation is a loss of signal leads to communication issues. Repeaters are used to regenerate the signal.

---

## Ring Topology


- Ring topology is like a bus topology, but with connected ends.
- The node that receives the message from the previous computer will retransmit to the next node.
- The data flows in one direction, i.e., it is unidirectional.

- The data flows in a single loop continuously known as an endless loop.
- It has no terminated ends, i.e., each node is connected to other node and having no termination point.
- The data in a ring topology flow in a clockwise direction.
- The most common access method of the ring topology is **token passing**.
    - **Token passing:** It is a network access method in which token is passed from one node to another node.
    - **Token:** It is a frame that circulates around the network.

## Working of Token passing

- A token moves around the network, and it is passed from computer to computer until it reaches the destination.
- The sender modifies the token by putting the address along with the data.
- The data is passed from one device to another device until the destination address matches. Once the token received by the destination device, then it sends the acknowledgment to the sender.
- In a ring topology, a token is used as a carrier.

### Advantages of Ring topology:

- **Network Management:** Faulty devices can be removed from the network without bringing the network down.
- **Product availability:** Many hardware and software tools for network operation and monitoring are available.
- **Cost:** Twisted pair cabling is inexpensive and easily available. Therefore, the installation cost is very low.
- **Reliable:** It is a more reliable network because the communication system is not dependent on the single host computer.

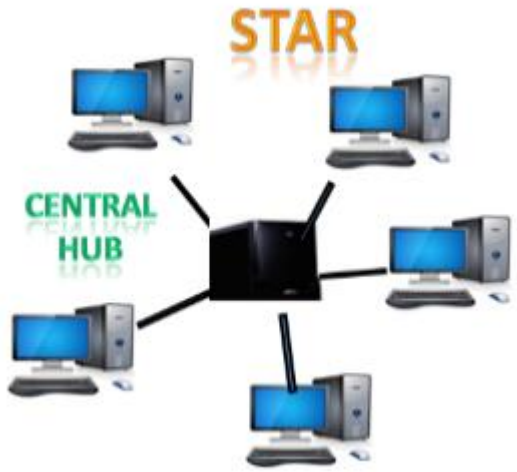
### Disadvantages of Ring topology:

- **Difficult troubleshooting:** It requires specialized test equipment to determine the cable faults. If any fault occurs in the cable, then it would disrupt the communication for all the nodes.
- **Failure:** The breakdown in one station leads to the failure of the overall network.

- **Reconfiguration difficult:** Adding new devices to the network would slow down the network.
- **Delay:** Communication delay is directly proportional to the number of nodes. Adding new devices increases the communication delay.

---

## Star Topology




- Star topology is an arrangement of the network in which every node is connected to the central hub, switch or a central computer.
- The central computer is known as a **server**, and the peripheral devices attached to the server are known as **clients**.
- Coaxial cable or RJ-45 cables are used to connect the computers.
- Hubs or Switches are mainly used as connection devices in a **physical star topology**.
- Star topology is the most popular topology in network implementation.

### Advantages of Star topology

- **Efficient troubleshooting:** Troubleshooting is quite efficient in a star topology as compared to bus topology. In a bus topology, the manager has to inspect the kilometers of cable. In a star topology, all the stations are connected to the centralized network. Therefore, the network administrator has to go to the single station to troubleshoot the problem.

- **Network control:** Complex network control features can be easily implemented in the star topology. Any changes made in the star topology are automatically accommodated.
- **Limited failure:** As each station is connected to the central hub with its own cable, therefore failure in one cable will not affect the entire network.
- **Familiar technology:** Star topology is a familiar technology as its tools are cost-effective.
- **Easily expandable:** It is easily expandable as new stations can be added to the open ports on the hub.
- **Cost effective:** Star topology networks are cost-effective as it uses inexpensive coaxial cable.
- **High data speeds:** It supports a bandwidth of approx 100Mbps. Ethernet 100BaseT is one of the most popular Star topology networks.

### Disadvantages of Star topology

- **A Central point of failure:** If the central hub or switch goes down, then all the connected nodes will not be able to communicate with each other.
- **Cable:** Sometimes cable routing becomes difficult when a significant amount of routing is required.

---

## Tree topology


- Tree topology combines the characteristics of bus topology and star topology.
- A tree topology is a type of structure in which all the computers are connected with each other in hierarchical fashion.

- The top-most node in tree topology is known as a root node, and all other nodes are the descendants of the root node.
- There is only one path exists between two nodes for the data transmission. Thus, it forms a parent-child hierarchy.

### Advantages of Tree topology

- **Support for broadband transmission:** Tree topology is mainly used to provide broadband transmission, i.e., signals are sent over long distances without being attenuated.
- **Easily expandable:** We can add the new device to the existing network. Therefore, we can say that tree topology is easily expandable.
- **Easily manageable:** In tree topology, the whole network is divided into segments known as star networks which can be easily managed and maintained.
- **Error detection:** Error detection and error correction are very easy in a tree topology.
- **Limited failure:** The breakdown in one station does not affect the entire network.
- **Point-to-point wiring:** It has point-to-point wiring for individual segments.

### Disadvantages of Tree topology

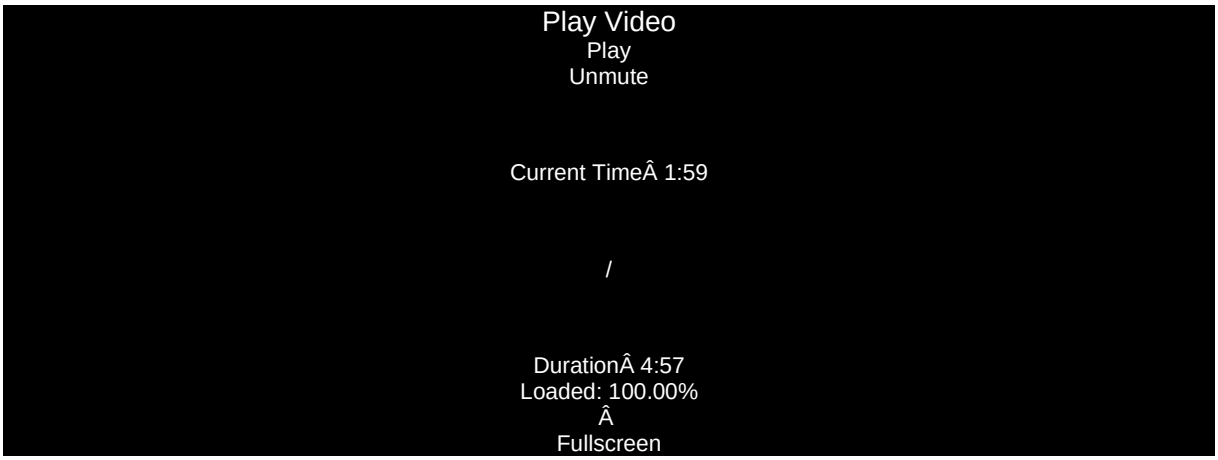
- **Difficult troubleshooting:** If any fault occurs in the node, then it becomes difficult to troubleshoot the problem.
- **High cost:** Devices required for broadband transmission are very costly.
- **Failure:** A tree topology mainly relies on main bus cable and failure in main bus cable will damage the overall network.
- **Reconfiguration difficult:** If new devices are added, then it becomes difficult to reconfigure.

---

## Mesh topology

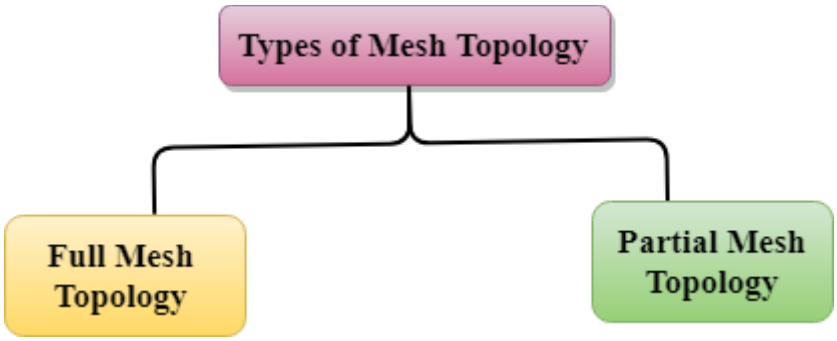

- Mesh technology is an arrangement of the network in which computers are interconnected with each other through various redundant connections.
- There are multiple paths from one computer to another computer.
- It does not contain the switch, hub or any central computer which acts as a central point of communication.
- The Internet is an example of the mesh topology.
- Mesh topology is mainly used for WAN implementations where communication failures are a critical concern.
- Mesh topology is mainly used for wireless networks.
- Mesh topology can be formed by using the formula: **Number of cables = (n*(n-1))/2;**

Where n is the number of nodes that represents the network.



**Mesh topology is divided into two categories:**

- Fully connected mesh topology
- Partially connected mesh topology




- **Full Mesh Topology:** In a full mesh topology, each computer is connected to all the computers available in the network.
- **Partial Mesh Topology:** In a partial mesh topology, not all but certain computers are connected to those computers with which they communicate frequently.

## Advantages of Mesh topology:

**Reliable:** The mesh topology networks are very reliable as if any link breakdown will not affect the communication between connected computers.

**Fast Communication:** Communication is very fast between the nodes.

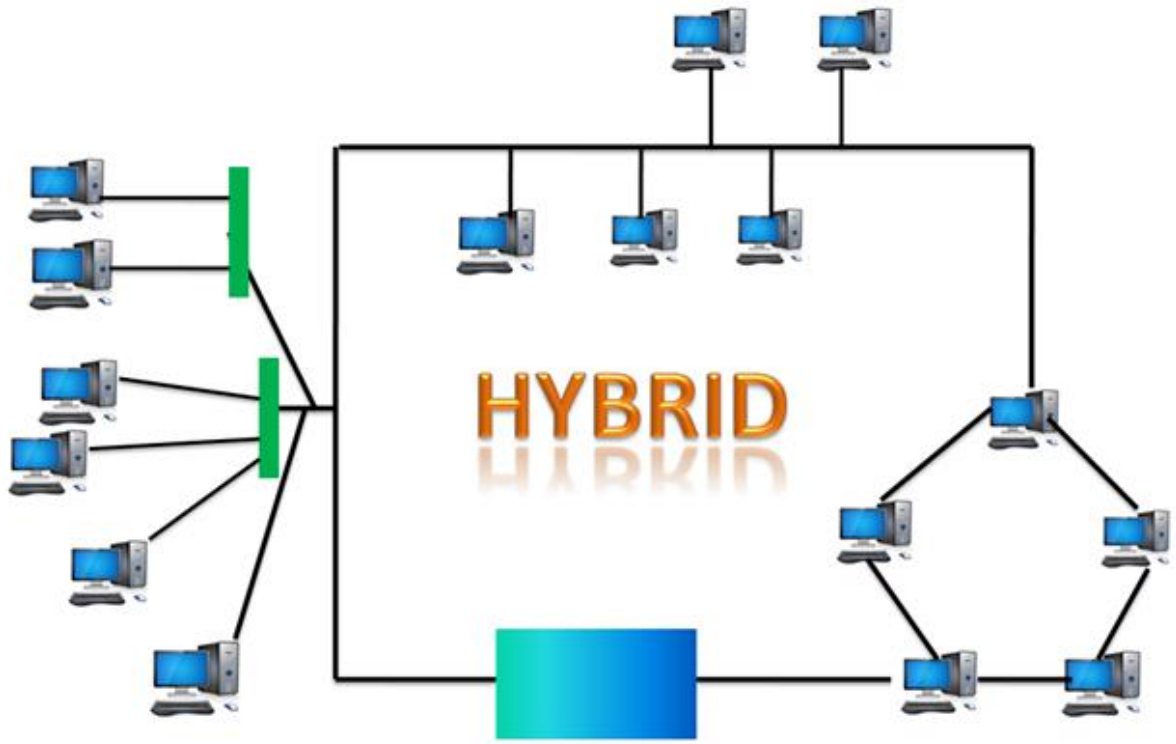
**Easier Reconfiguration:** Adding new devices would not disrupt the communication between other devices.

### Disadvantages of Mesh topology

- **Cost:** A mesh topology contains a large number of connected devices such as a router and more transmission media than other topologies.
- **Management:** Mesh topology networks are very large and very difficult to maintain and manage. If the network is not monitored carefully, then the communication link failure goes undetected.
- **Efficiency:** In this topology, redundant connections are high that reduces the efficiency of the network.

---

## Hybrid Topology




- The combination of various different topologies is known as **Hybrid topology**.
- A Hybrid topology is a connection between different links and nodes to transfer the data.
- When two or more different topologies are combined together is termed as Hybrid topology and if similar topologies are connected with each other will

not result in Hybrid topology. For example, if there exist a ring topology in one branch of ICICI bank and bus topology in another branch of ICICI bank, connecting these two topologies will result in Hybrid topology.

### Advantages of Hybrid Topology

- **Reliable:** If a fault occurs in any part of the network will not affect the functioning of the rest of the network.
- **Scalable:** Size of the network can be easily expanded by adding new devices without affecting the functionality of the existing network.
- **Flexible:** This topology is very flexible as it can be designed according to the requirements of the organization.
- **Effective:** Hybrid topology is very effective as it can be designed in such a way that the strength of the network is maximized and weakness of the network is minimized.

### Disadvantages of Hybrid topology

- **Complex design:** The major drawback of the Hybrid topology is the design of the Hybrid network. It is very difficult to design the architecture of the Hybrid network.
- **Costly Hub:** The Hubs used in the Hybrid topology are very expensive as these hubs are different from usual Hubs used in other topologies.
- **Costly infrastructure:** The infrastructure cost is very high as a hybrid network requires a lot of cabling, network devices, etc.

Connection Strategies:

Connection strategies: Once messege are able to reach there destinations, processes can institute communcations sessions information.

pairs of process that want to communicate over the network can be connected in a number if ways.

three most common sehems are circuit switching,message switching, packet switiching.

# What is a Network Operating System?

The basic definition of an **operating system** is that the operating system is the interface between the computer hardware and the user. And in daily life, we use the operating system on our devices which provides a good GUI, and many more features with it. Similarly, a network operating system(NOS) is software that

connects multiple devices and computers on the network and allows them to share resources on the network. Let's see what are the functions of the network operating system.

**Functions of the NOS :**

Following are the main functions of NOS :

- Creating and managing user accounts on the network.
- Controlling access to resources on the network.
- Provide communication services between the devices on the network.
- Monitor and troubleshoot the network.
- Configuring and Managing the resources on the network.

Now let's see the type of Network Operating systems.

**Types of Network operating systems :**

There are mainly two types of networks, one is peer to peer and another is client/server. Now let's see each type one by one.

- **Peer to Peer –**
  Peer-to-peer network operating systems allow sharing resources and files with small-sized networks and having fewer resources. In general, peer-to-peer network operating systems are used on LAN.
- **Client/server –**
  Client-server network operating systems provide users access to resources through the central server. This NOS is too expensive to implement and maintain. This operating system is good for the big networks which provide many services.

**Features of network operating systems :**

Let's see what are the functions of the network operating system.

- Printers and application sharing on the network.
- File systems and database sharing.
- Provide good security by using functionality like user authentication and access control.
- Create backups of data.
- Inter-networking.

Now let's see what are the advantages of NOS.

**Advantages of Network operating systems :**

- Highly stable due to central server.
- Provide good security.
- Upgradation of new technology and hardware can be easily implemented in the network.
- Provide remote access to servers from different locations.

**Disadvantages of Network operating systems :**

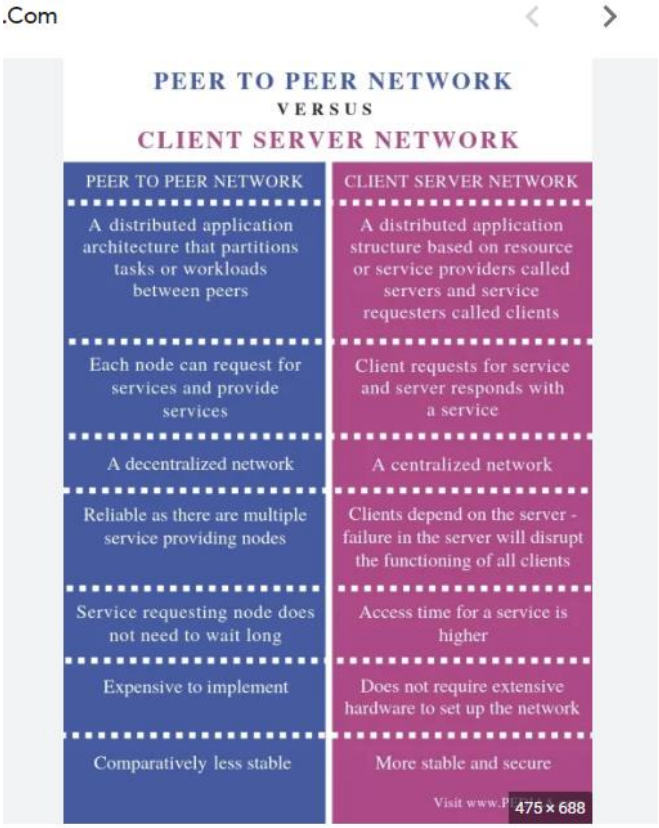
- Depend on the central location to perform the operations.
- High cost to buying server.
- Regular updating and maintenance are required.

Now let's see what are the examples of network operating systems.

**Examples of Network Operating systems :**

Following are the examples of network operating systems.

- Microsoft Windows Server
- UNIX/Linux
- Artisoft's LANtastic
- Banyan's VINES

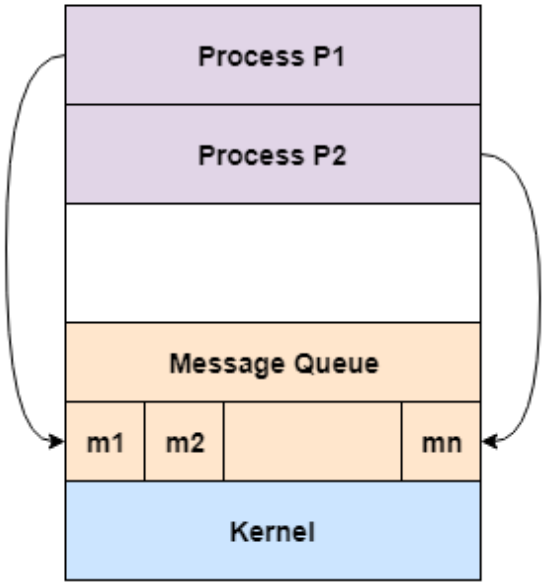



Message passing

Process communication is the mechanism provided by the operating system that allows processes to communicate with each other. This communication could involve a process letting another process know that some event has occurred or transferring of data from one process to another. One of the models of process communication is the message passing model.

Message passing model allows multiple processes to read and write data to the message queue without being connected to each other. Messages are stored on the queue until their recipient retrieves them. Message queues are quite useful for interprocess communication and are used by most operating systems.

A diagram that demonstrates message passing model of process communication is given as follows –




Message Passing Model

In the above diagram, both the processes P1 and P2 can access the message queue and store and retrieve data.

## Advantages of Message Passing Model

Some of the advantages of message passing model are given as follows –

- The message passing model is much easier to implement than the shared memory model.
- It is easier to build parallel hardware using message passing model as it is quite tolerant of higher communication latencies.

## Disadvantage of Message Passing Model

The message passing model has slower communication than the shared memory model because the connection setup takes time.

**Ans:Access Matrix** is a security model of protection state in computer system. It is represented as a matrix. Access matrix is used to define the rights of each process executing in the domain with respect to each object. The rows of matrix represent domains and columns represent objects. Each cell of matrix represents set of access rights which are given to the processes of domain means each entry(i, j) defines the set of operations that a process executing in domain Di can invoke on object Oj.

the security problem

the process of ensuring OS availability, confidentiality, integrity is known as operating system security. OS security refers to the processes or measures taken to protect the operating system from dangers, including viruses, worms, malware, and remote hacker intrusions. Operating system security comprises all preventive-control procedures that protect any system assets that could be stolen, modified, or deleted if OS security is breached.

program threats

# Program Threats

Operating system's processes and kernel do the designated task as instructed. If a user program made these process do malicious tasks, then it is known as **Program Threats**. One of the common example of program threat is a program installed in a computer which can store and send user credentials via network to some hacker. Following is the list of some well-known program threats.

- **Trojan Horse** – Such program traps user login credentials and stores them to send to malicious user who can later on login to computer and can access system resources.
- **Trap Door** – If a program which is designed to work as required, have a security hole in its code and perform illegal action without knowledge of user then it is called to have a trap door.
- **Logic Bomb** – Logic bomb is a situation when a program misbehaves only when certain conditions met otherwise it works as a genuine program. It is harder to detect.
- **Virus** – Virus as name suggest can replicate themselves on computer system. They are highly dangerous and can modify/delete user files, crash systems. A virus is generatlly a small code embedded in a program. As user accesses the program, the virus starts getting embedded in other files/ programs and can make system unusable for user

# System Threats

System threats refers to misuse of system services and network connections to put user in trouble. System threats can be used to launch program threats on a complete network called as program attack. System threats creates such an environment that operating system resources/ user files are misused. Following is the list of some well-known system threats.

- **Worm** – Worm is a process which can choked down a system performance by using system resources to extreme levels. A Worm process generates its multiple copies where each copy uses system resources, prevents all other processes to get required resources. Worms processes can even shut down an entire network.
- **Port Scanning** – Port scanning is a mechanism or means by which a hacker can detects system vulnerabilities to make an attack on the system.
- **Denial of Service** – Denial of service attacks normally prevents user to make legitimate use of the system. For example, a user may not be able to use internet if denial of service attacks browser's content settings.

- Network threats: **Passive Network Threats**: Activities such as wiretapping and idle scans that are designed to intercept traffic travelling through the network.
- **Active Network Threats**: Activities such as Denial of Service (DoS) attacks and SQL injection attacks where the attacker is attempting to execute commands to disrupt the network's normal operation.

# Unit:2

File Concepts:

## File Structure

A File Structure should be according to a required format that the operating system can understand.

- A file has a certain defined structure according to its type.
- A text file is a sequence of characters organized into lines.
- A source file is a sequence of procedures and functions.
- An object file is a sequence of bytes organized into blocks that are understandable by the machine.
- When operating system defines different file structures, it also contains the code to support these file structure. Unix, MS-DOS support minimum number of file structure.

## File Attributes

A file has a name and data. Moreover, it also stores meta information like file creation date and time, current size, last modified date, etc. All this information is called the attributes of a file system.

Here, are some important File attributes used in OS:

- **Name:** It is the only information stored in a human-readable form.
- **Identifier**: Every file is identified by a unique tag number within a file system known as an identifier.
- **Location:** Points to file location on device.
- **Type:** This attribute is required for systems that support various types of files.
- **Size**. Attribute used to display the current file size.
- **Protection**. This attribute assigns and controls the access rights of reading, writing, and executing the file.
- **Time, date and security:** It is used for protection, security, and also used for monitoring

# File Operations

- **File is an abstract data type**
- **Create**
- **Write**
- **Read**
- **Reposition within file**
- **Delete**
- **Truncate**
- **Open file: Several pieces of data are needed to manage open files:**
  - **File pointer: pointer to last read/write location, per process that has the file open**
  - **File-open count: counter of number of times a file is open – to allow removal of data from open-file table when last processes closes it**

- **Disk location of the file: cache of data access information**
- **Access rights: per-process access mode information**

## Access method:

When a file is used, information is read and accessed into computer memory and there are several ways to access this information of the file. Some systems provide only one access method for files. Other systems, such as those of IBM, support many access methods, and choosing the right one for a particular application is a major design problem.

There are three ways to access a file into a computer system: Sequential-Access, Direct Access, Index sequential Method.

1. **Sequential Access –**
   It is the simplest access method. Information in the file is processed in order, one record after the other. This mode of access is by far the most common; for example, editor and compiler usually access the file in this fashion.
   Read and write make up the bulk of the operation on a file. A read operation *-read next-* read the next position of the file and automatically advance a file pointer, which keeps track I/O location. Similarly, for the -write *next-* append to the end of the file and advance to the newly written material.
   **Key points:**
   - Data is accessed one record right after another record in an order.
   - When we use read command, it move ahead pointer by one
   - When we use write command, it will allocate memory and move the pointer to the end of the file
   - Such a method is reasonable for tape.

2. **Direct Access –**
   Another method is *direct access method* also known as *relative access method*. A filed-length logical record that allows the program to read and write record rapidly. in no particular order. The direct access is based on the disk model of a file since disk allows random access to any file block. For direct access, the file is viewed as a numbered sequence of block or record. Thus, we may read block 14 then block 59, and then we can write block 17. There is no restriction on the order of reading and writing for a direct access file.
   A block number provided by the user to the operating system is normally a *relative block number*, the first relative block of the file is 0 and then 1 and so on.

## Directory Structure:

A **directory** is a container that is used to contain folders and files. It organizes files and folders in a hierarchical manner.

## Disk Structure

- **Disk can be subdivided into partitions**
- **Disks or partitions can be RAID(Redundant Array of Independent Disks) protected against failure**
- **Disk or partition can be used raw – without a file system, or formatted with a file system**
- **Partitions also known as minidisks, slices**
- **Entity containing file system known as a volume**
- **Each volume containing file system also tracks that file system's info in device directory or volume table of contents**
- **As well as general-purpose file systems there are many special-purpose file systems, frequently all within the same operating system or computer**

## Free space management:

The system keeps tracks of the free disk blocks for allocating space to files when they are created. Also, to reuse the space released from deleting the files, free space management becomes crucial. The system maintains a free space list which keeps track of the disk blocks that are not allocated to some file or directory. The free space list can be implemented mainly as:

1. **Bitmap or Bit vector –**
   A Bitmap or Bit Vector is series or collection of bits where each bit corresponds to a disk block. The bit can take two values: 0 and 1: *0 indicates that the block is allocated* and 1 indicates a free block. The given instance of disk blocks on the disk in *Figure 1* (where green blocks are allocated) can be represented by a bitmap of 16 bits as: **0000111000000110**.

**Disk:**

# Types of Hard Drives

Currently, hard drives are divided into 4 major types:

- Parallel Advanced Technology Attachment (PATA)
- Serial Advanced Technology Attachment (SATA)
- Small Computer System Interface (SCSI)
- Solid State Drive (SSD)

These names come from the way they connect to the computer. In this article, I'm now going to elaborate on each of these types of hard drives as concisely as possible.

**Parallel Advanced Technology Attachment (PATA)**

The PATA hard drives were first introduced to the market by Compaq and Western Digital in 1986. They can have up to 80GB capacity and transfer data as fast as 133 MB/S.

They were named Parallel Advanced Technology Attachment because they use a parallel ATA interface to connect to the computer. Apart from PATA, they are also called Integrated Drive Electronics (IDE) and Enhanced Integrated Drive Electronics (EIDE).

PATA hard drives are made of mechanical moving parts and are based on parallel signaling technology – meaning they transmit multiple bits of data simultaneously.

**Serial Advanced Technology Attachment (SATA)**

In recent times, a lot of desktop and laptop computers have gotten SATA hard drives because they have superseded PATA hard drives in size, power consumption, and even better pricing.

The mode of connection to a computer remains the same as PATA, but instead of parallel signaling technology for data transmission, they use serial signaling technology. This means that they transfer data one bit at a time.

A notable advantage SATA hard drives have over PATA hard drives is the transmission of data at a rate of 150 – 300 MB/S. In addition, they have thinner cables and a cable limit of 1 meter.

**Small Computer System Interface (SCSI)**

SCSI hard drives are upgrades over SATA and PATA drives for many reasons such as round-the-clock operations, speed, storage, and several others.

For connection, SCSI hard drives use a small computer system interface – which is a standard for connecting peripheral devices such as printers, scanners, and others.

Best of all, they allow the connection of peripheral devices such as printers, scanners, and other hard drives. In addition, they transmit data at 320 MB/S and you can connect them internally or externally.

Connections through SCSI on personal computers have now been replaced by the Universal Serial BUS (USB). This means that SCSI is no longer used as consumer hardware.

**Solid State Drive (SSD)**

SSD hard drives are one of the latest hard drive technologoes at the time of writing this article.

Unlike the hard drive technologies before SSD drives, they don't consist of moving parts and they don't use magnetism for storing data.

Instead, they use integrated circuits (ICs) just like third-generation computers. This makes them more durable, faster, and less prone to damage and corruption.

SSD hard drives have a notable advantage of transferring data at speed of 550 MB/S and allow faster boot times than the types of hard drives before them.

**Disk management**

**Here are some common things that you can do in Disk Management:**

- **Partition a Drive**
- **Format a Drive**

- **Change a Drive's Letter**
- **Shrink a Partition**
- **Delete a Partition**
- **Change a Drive's File System**

## What is Disk Access Time in Disk Scheduling?

Disk Access Time is defined as the total time required by the computer to process a read/write request and then retrieve the required data from the disk storage.

There are two components in disk access time. The first component is the ***seek time*** which occurs when the read and write arm seeks the desired track. The second component is latency or wait *time* which occurs when the head write arm waits for the desired sector on the track to spin around.

# Disk Scheduling

As we know, a process needs two type of time, CPU time and IO time. For I/O, it requests the Operating system to access the disk.

However, the operating system must be fare enough to satisfy each request and at the same time, operating system must maintain the efficiency and speed of process execution.

The technique that operating system uses to determine the request which is to be satisfied next is called disk scheduling.

**FCFS**

- **Service requests in the order they come**
- **Fair to all requests**
- **Can cause very large total seek time over all requests if the load is moderate to high**

**SSTF**

- **Selects the request with the minimum seek time from the current head position**
- **SSTF scheduling is a form of SJF scheduling**
  - **May cause starvation of some requests like SJF**
  - **But not optimal, unlike SJF**
- **Minimizes seek time, but not fair**
- **May work well if the load is not high**

**SCAN**

- **The disk arm starts at one end of the disk, and**
- **moves toward the other end, servicing requests**
- **until it gets to the other end of the disk, where**
- **the head movement is reversed and servicing**
- **continues**
- **Sometimes called the *elevator algorithm***

**C-SCAN**

- **Provides a more uniform wait time than SCAN**
- **The head moves from one end of the disk to the other. servicing requests as it goes. When it reaches the other end, however, it immediately returns to the beginning of the disk, without servicing any requests on the return trip**
- **Treats the cylinders as a circular list that wraps around from the last cylinder to the first one**

**C-LOOK**

- **Version of C-SCAN**

**Arm only goes as far as the last request in each direction, then reverses direction immediately, without first going all the way to the end of the disk**

**Selecting a Disk-Scheduling Algorithm**

- **SSTF is common and has a natural appeal**
- **SCAN and C-SCAN perform better for systems that place a heavy load on the disk**
- **Performance depends on the number and types of requests**
- **Requests for disk service can be influenced by the file allocation method**
- **The disk-scheduling algorithm should be written as a separate module of the operating system, allowing it to be replaced with a different algorithm if necessary**

**Either SSTF or C-LOOK is a reasonable choice for the default algorithm (depending on load)**

**What is RAID?**

- **The basic idea of RAID was to combine multiple small, inexpensive disk drives into an array of disk drives which yields performance exceeding that of a Single Large Expensive Drive (SLED). Additionally, this array of drives appears to the computer as a single logical storage unit or drive.**
- **This concept is an example of storage virtualization**
- **It is a way of storing the same data in different places (thus, redundantly) on multiple hard disks.**
- **By placing data on multiple disks, I/O (input/output) operations can overlap in a balanced way, improving performance.**

- **Since multiple disks increases the mean time between failures (MTBF), storing data redundantly also increases fault tolerance .**

**Why RAID?**

- **RAID is now used as an umbrella term for computer data storage schemes that can divide and replicate data among multiple physical disk drives.**
- **The physical disks are said to be in a RAID array , which is accessed by the operating system as one single disk.**
- **The different schemes or architectures are named by the word RAID followed by a number (e.g., RAID 0, RAID 1).**
- **Each scheme provides a different balance between two key goals:**
    1. **increase data reliability & capacity**
    2. **increase input/output performance.**

**The Different RAID Levels**

- **RAID 0**
- **RAID 1**
- **RAID 2**
- **RAID 3**
- **RAID 4**
- **RAID 5**
- **RAID 6**

**What is Address Binding?**

**Address binding is the process of mapping the program's logical or virtual addresses to corresponding physical or main memory addresses**

**Compile time. The compiler translates symbolic addresses to absolute addresses. If you know at compile time where the process will reside in memory, then absolute code can be generated (Static).**

**Load time. The compiler translates symbolic addresses to relative (relocatable) addresses. The loader translates these to absolute addresses. If it is not known at compile time where the process will reside in memory, then the compiler must generate relocatable code (Static).**

**Relocatable means that the program image can reside anywhere in physical memory.**

**Execution time. If the process can be moved during its execution from one memory segment to another, then binding must be delayed until run time. The absolute addresses are generated by hardware. Most general-purpose OSs use this method (Dynamic).**

**what is Memory Management?**

**Main Memory refers to a physical memory that is the internal memory to the computer. The word main is used to distinguish it from external mass storage devices such as disk drives. Main memory is also known as RAM.**

- **At times one program is dependent on some other program. In such a case, rather than loading all the dependent programs, CPU links the dependent programs to the main executing program when its required. This mechanism is known as Dynamic Linking.**

- **Linking postponed until execution time.**
- **Small piece of code, stub, used to locate the appropriate memory-resident library routine.**
- **Stub replaces itself with the address of the routine, and executes the routine.**
- **Operating system needed to check if routine is in processes' memory address.**
- **Dynamic linking is particularly useful for libraries.**
- **All the programs are loaded in the main memory for execution. Sometimes complete program is loaded into the memory, but some times a certain part or routine of the program is loaded into the main memory only when it is called by the program, this mechanism is called Dynamic Loading, this enhance the performance.**
    - **Routine is not loaded until it is called**
    - **Better memory-space utilization; unused routine is never loaded.**
    - **Useful when large amounts of code are needed to handle infrequently occurring cases.**
    - **No special support from the OS is required - implemented through program design.**

**What is Process Address Space**

- **The process address space is the set of logical addresses that a process references in its code. For example, when 32-bit addressing is in use, addresses can range from 0 to 0x7fffffff; that is, $2^{31}$ possible numbers, for a total theoretical size of 2 gigabytes.**

- **The operating system takes care of mapping the logical addresses to physical addresses at the time of memory allocation to the program.**
- **There are three types of addresses used in a program before and after memory is allocated −**
- **Symbolic addresses**
- **The addresses used in a source code. The variable names, constants, and instruction labels are the basic elements of the symbolic address space.**
- **Relative addresses**
- **At the time of compilation, a compiler converts symbolic addresses into relative addresses.**
- **Physical addresses**
- **The loader generates these addresses at the time when a program is loaded into main memory.**
- **Note: Virtual and physical addresses are the same in compile-time and load-time address-binding schemes.**

**What is Memory Allocation**

**Memory allocation is a process by which computer programs are assigned memory or space.**

**Main memory usually has two partitions −**

- **Low Memory − Operating system resides in this memory.**
- **High Memory − User processes are held in high memory.**

**High Memory can be partitioned in following ways:**

**Contiguous Allocation : a) Single/Fixed-Size partition**

**b) Multiple/Variable-Size Partition**

**Dynamic Storage Allocation: a) First Fit**

**b) Best Fit**

**c) Worst Fit**

**Operating system maintains information about:**

**a) allocated partitions**

**b) free partitions (hole)**

**what is SINGLE PARTITION ALLOCATION**

- **Division of physical memory into fixed sized regions. (Allows addresses spaces to be distinct = one user can't muck with another user, or the system.)**
- **The number of partitions determines the level of multiprogramming. Partition is given to a process when it's scheduled.**
- **Protection around each partition determined by**
  - **bounds ( upper, lower )**
  - **base / limit.**
- **These limits are set in the hardware.**

  **Q:What is Multiple-partition Allocation**
- **Degree of multiprogramming is limited by number of partitions**
- **Variable-partition sizes are kept for efficiency (sized to a given process' needs)**
- **Hole – block of available memory; holes of various size are scattered throughout memory**
- **When a process arrives, it is allocated memory from a hole large enough to accommodate it**

- **Process that exits frees its partition, adjacent free partitions are merged.**

**Q: What is Dynamic Storage-Allocation Problem?**

**Memory allocation is a process by which computer programs are assigned memory or space. It is of three types :**

- **First Fit: The first hole that is big enough is allocated to program.**
- **Best Fit: The smallest hole that is big enough is allocated to program.**
  - **must search entire list, unless ordered by size, Consumes CPU time**
- **Worst Fit: The largest hole that is big enough is allocated to program.**
  - **must search entire list**
  - **Avoid small holes (external fragmentation). This occurs when there are many small pieces of free memory.**
- **What should be the minimum size allocated, allocated in what chunk size?**
- **Also avoid internal fragmentation. This is when memory is handed out in some fixed way (power of 2 for instance) and requesting program doesn't use all of it.**

## what is Swapping in Operating Systems (OS)?

Let's suppose there are several processes like P1, P2, P3, and P4 that are ready to be executed inside the ready queue, and processes P1 and P2 are very memory consuming so when the processes start executing there may be a scenario where the memory will not be available for the execution of the process P3 and P4 as there is a limited amount of memory available for process execution.

**Swapping in the operating system** is a memory management scheme that temporarily swaps out an idle or blocked process from the main memory to secondary memory which

ensures proper memory utilization and memory availability for those processes which are ready to be executed.

When that memory-consuming process goes into a termination state means its execution is over due to which the memory dedicated to their execution becomes free then the swapped-out processes are brought back into the main memory and their execution starts.

The area of the secondary memory where swapped-out processes are stored is called **swap space**. The swapping method forms a temporary queue of swapped processes in the secondary memory.

In the case of high-priority processes, the process with low priority is swapped out of the main memory and stored in swap space then the process with high priority is swapped into the main memory to be executed first.

The main goals of an operating system include **Maximum utilization of the CPU**. This means that there should be a process execution every time, the **CPU** should never stay idle and there should not be any **Process starvation** or **blocking**.

Different process management and memory management schemes are designed to fulfill such goals of an operating system.

Swapping in **OS** is done to get access to data present in secondary memory and transfer it to the main memory so that it can be used by the application programs.

It can affect the performance of the system but it helps in running more than one process by managing the memory. Therefore swapping in os is also known as the **memory compaction technique**.

## What is Fragmentation?

Fragmentation is an unwanted problem in the operating system in which the processes are loaded and unloaded from memory, and free memory space is fragmented. Processes can't be assigned to memory blocks due to their small size, and the memory blocks stay unused. It is also necessary to understand that as programs are loaded and deleted from memory, they generate free space or a hole in the memory. These small blocks cannot be allotted to new arriving processes, resulting in inefficient memory use.


The conditions of fragmentation depend on the memory allocation system. As the process is loaded and unloaded from memory, these areas are fragmented into small pieces of memory that cannot be allocated to incoming processes. It is called **fragmentation**.

## Causes of Fragmentation

User processes are loaded and unloaded from the main memory, and processes are kept in memory blocks in the main memory. Many spaces remain after process loading and swapping that another process cannot load due to their size. Main memory is available, but its space is insufficient to load another process because of the dynamical allocation of main memory processes.

## Types of Fragmentation

There are mainly two types of fragmentation in the operating system. These are as follows:

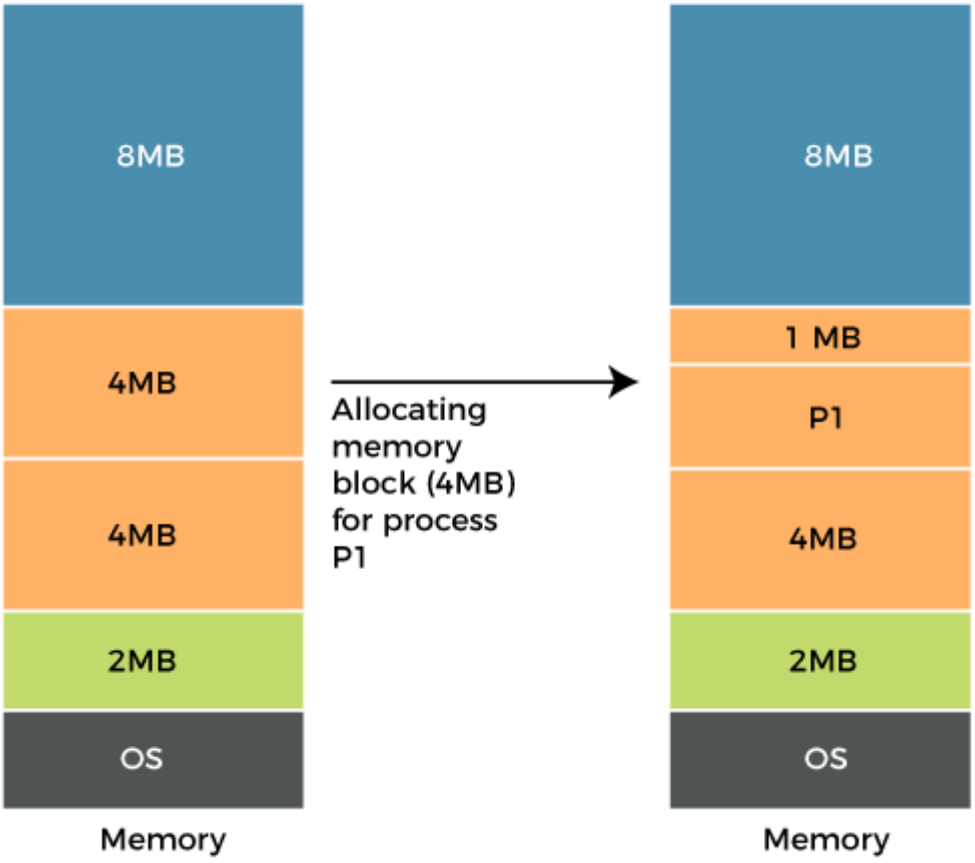
1. **Internal Fragmentation**
2. **External Fragmentation**

### Internal Fragmentation

When a process is allocated to a memory block, and if the process is smaller than the amount of memory requested, a free space is created in the given memory block. Due to this, the free space of the memory block is unused, which causes **internal** fragmentation.

**For Example:**

Assume that memory allocation in RAM is done using fixed partitioning (i.e., memory blocks of fixed sizes). **2MB, 4MB, 4MB**, and **8MB** are the available sizes. The Operating System uses a part of this RAM.




Let's suppose a process **P1** with a size of **3MB** arrives and is given a memory block of **4MB**. As a result, the **1MB** of free space in this block is unused and cannot be used to allocate memory to another process. It is known as **internal fragmentation**.

**How to avoid internal fragmentation?**

The problem of internal fragmentation may arise due to the fixed sizes of the memory blocks. It may be solved by assigning space to the process via dynamic partitioning. Dynamic partitioning allocates only the amount of space requested by the process. As a result, there is no internal fragmentation.

## External Fragmentation

External fragmentation happens when a dynamic memory allocation method allocates some memory but leaves a small amount of memory unusable. The quantity of available memory is substantially reduced if there is too much external fragmentation. There is enough memory space to complete a request, but it is not contiguous. It's known as **external** fragmentation.

# Compaction in Operating System

Compaction is a technique to collect all the free memory present in form of fragments into one large chunk of free memory, which can be used to run other processes.

### What is Pagging?

Paging is a memory management scheme that eliminates the need for contiguous allocation of physical memory. The process of retrieving processes in the form of pages from the secondary storage into the main memory is known as paging. The basic purpose of paging is to separate each procedure into pages. Additionally, frames will be used to split the main memory.This scheme permits the physical address space of a process to be non – contiguous.

Let us look some important terminologies:

- Logical Address or Virtual Address (represented in bits): An address generated by the CPU
- Logical Address Space or Virtual Address Space( represented in words or bytes): The set of all logical addresses generated by a program
- Physical Address (represented in bits): An address actually available on memory unit
- Physical Address Space (represented in words or bytes): The set of all physical addresses corresponding to the logical addresses

# Segmentation

In Operating Systems, Segmentation is a memory management technique in which the memory is divided into the variable size parts. Each part is known as a segment which can be allocated to a process.

The details about each segment are stored in a table called a segment table. Segment table is stored in one (or many) of the segments.

Segment table contains mainly two information about segment:

1. Base: It is the base address of the segment
2. Limit: It is the length of the segment.

# Virtual Memory in Operating System

Virtual Memory is a storage allocation scheme in which secondary memory can be addressed as though it were part of the main memory. The addresses a program may use to reference memory are distinguished from the addresses the memory system uses to identify physical storage sites, and program-generated addresses are translated automatically to the corresponding machine addresses.

The size of virtual storage is limited by the addressing scheme of the computer system and the amount of secondary memory is available not by the actual number of the main storage locations.

It is a technique that is implemented using both hardware and software. It maps memory addresses used by a program, called virtual addresses, into physical addresses in computer memory.

1. All memory references within a process are logical addresses that are dynamically translated into physical addresses at run time. This means that a process can be swapped in and out of the main memory such that it occupies different places in the main memory at different times during the course of execution.
2. A process may be broken into a number of pieces and these pieces need not be continuously located in the main memory during execution. The combination of dynamic run-time address translation and use of page or segment table permits this.