Great Place To Work. Certified FEB 2023 – FEB 2024 INDIA

UPES

UNIVERSITY OF TOMORROW

nirf RANKING 2023

#RANKED 52 IN INDIA

#University Category

QS WORLD UNIVERSITY RANKINGS

NO.1 PVT. UNIVERSITY IN ACADEMIC REPUTATION IN INDIA

ACCREDITED WITH GRADE A NAAC

ACCREDITED **GRADE 'A'** BY NAAC

E-LEAD

PERFECT SCORE OF **150/150** AS A TESTAMENT TO EXECEPTIONAL E-LEARNING METHODS

Unit 1 : Introduction to Operating System

Lecture 2

Dr. Hemant Petwal

School of Computer Science

UPES, Dehradun

India

UPES
UNIVERSITY OF TOMORROW

Table of Contents

UPES
UNIVERSITY OF TOMORROW

Learning & Course Outcomes

Learning Outcomes

LO1: Understand the history and evolution of operating system

LO2: Define an operating system and demonstrate its functionality

Course Outcomes

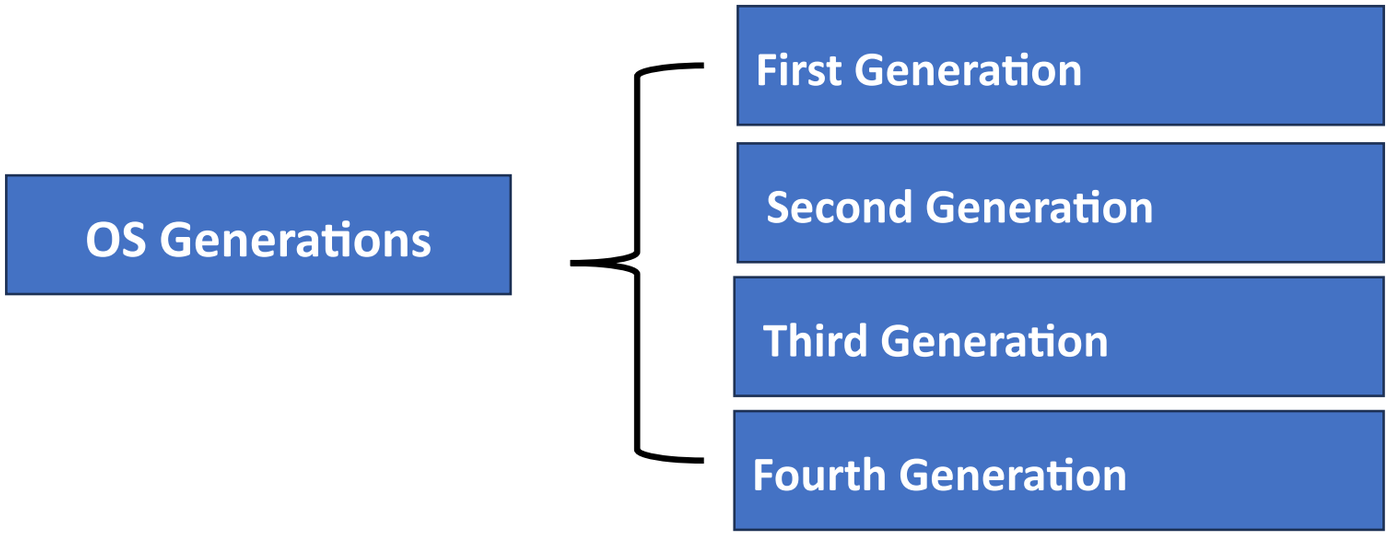
CO1: Demonstrate a comprehensive understanding of operating systems

UPES
UNIVERSITY OF TOMORROW

Operating System: History

- An operating system is a type of software that acts as an interface between the user and the hardware.
- It is responsible for handling various critical functions of the computer or any other machine. Various tasks that are handled by OS.

Evolution (Generation) of Operating System



UPES
UNIVERSITY OF TOMORROW

Operating System: Generations

The First Generation (1940 – 1950)

- An OS was not included in the creation of the first electrical computer.
- Early computer users had complete control over the device and wrote programs in pure machine language for every task.
- During the computer generation, a programmer can merely execute and solve basic mathematical calculations. an operating system is not needed for these computations.

The Second Generation (1955 – 1965)

- GMOSIS, the first operating system (OS) was developed in the early 1950s.
- General Motors has created the operating system for the IBM Computer.
- The second-generation operating system was built on a single-stream batch processing system
- It gathers all related jobs into groups or batches and then submits them to the operating system using a punch card to finish all of them.

The Third Generation (1965 – 1980)

- In 2nd Generation, the operating system cleans up after each work is finished before reading and starting the subsequent job on a punch card
- Operating system designers were able to create a new operating system in the late 1960s that was capable of multiprogramming—the simultaneous execution of several tasks in a single computer program.

UPES
UNIVERSITY OF TOMORROW

Operating System: Generations

The Fourth Generation (1980 – Present Day)

- The evolution of the personal computer is linked to the fourth generation of operating systems. Nonetheless, the third-generation minicomputers and the personal computer have many similarities
- At that time, minicomputers were only slightly more expensive than personal computers, which were highly expensive.
- The development of Microsoft and the Windows operating system was a significant influence in the creation of personal computers.

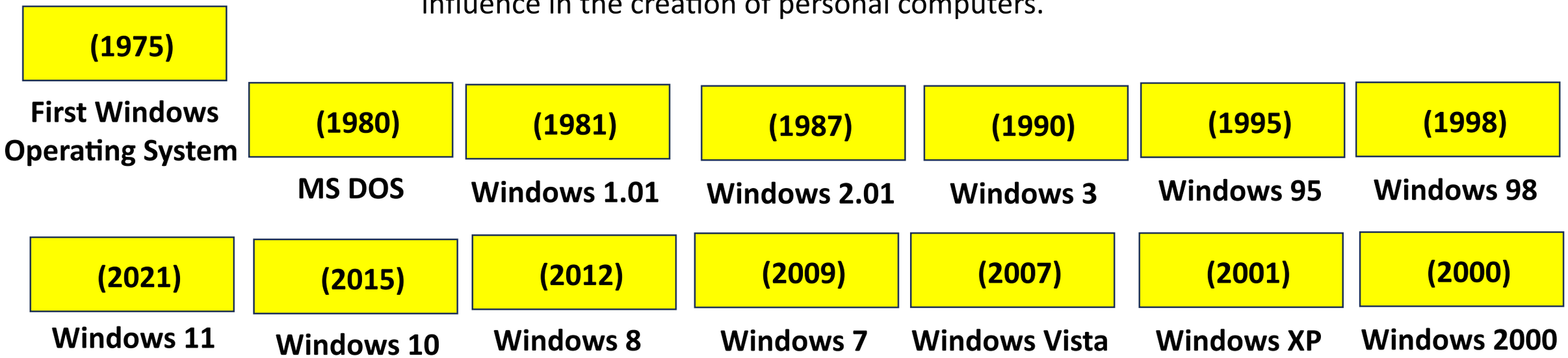


Fig 1. OS Generations

UPES
UNIVERSITY OF TOMORROW

Operating System: Types

An OS performs all the basic tasks like managing files, processes, and memory. Thus, OS acts as the manager of all the resources, i.e. resource manager and becomes an interface between the user and the machine. It is one of the most required software that is present in the device.

There are several types of Operating Systems:

- Batch Operating System
- Multi-Programming System
- Multi-Processing System
- Multi-Tasking Operating System
- Time-Sharing Operating System
- Distributed Operating System
- Network Operating System
- Real-Time Operating System

Operating System: Batch Processing OS

The batch-processing operating system was very popular in the **1970s**. In batch operating system the jobs were performed in batches. This means Jobs having similar requirements are grouped and executed as a group to speed up processing. Users using batch operating systems do not interact with the computer directly. It is the responsibility of the operator to sort jobs with similar needs

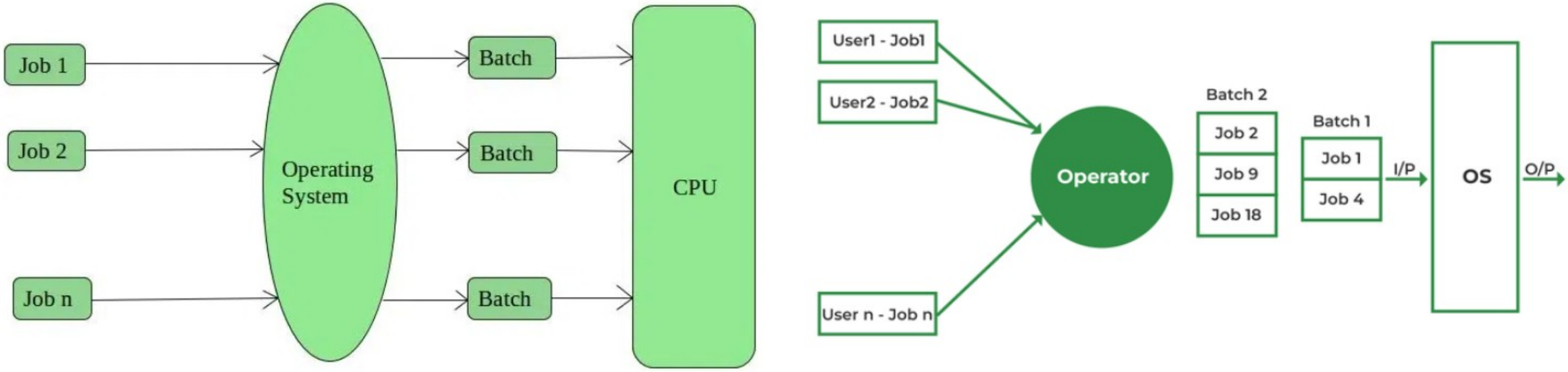


Fig 2. Batch Processing OS

Operating System: Batch Processing OS

Advantages of Batch processing OS

- Multiple users can share the batch systems.
- The idle time for the batch system is very less.
- It is easy to manage large work repeatedly in batch systems.
- Some examples are Payroll Systems, Bank Statements, etc.

Disadvantages of Batch processing OS

- The computer operators should be well known with batch systems.
- Batch systems are hard to debug and sometimes costly.
- The other jobs will have to wait for an unknown time if any job fails.
- In batch operating system the processing time for jobs is commonly difficult to accurately predict while they are in the queue.
- It is difficult to accurately predict the exact time required for a job to complete while it is in the queue.

UPES
UNIVERSITY OF TOMORROW

Operating System: Multi-Programming OS

Before Multiprogramming, there were single tasking operating systems like MS DOS that used to allow only one program to be loaded at a time and run. These systems were not efficient as CPU was not used efficiently. E.g., in a single tasking system if the current program waits for some input/output to finish, the CPU is not used. The idea of multiprogramming is to assign CPUs to other processes while the current process might not be finished.

- Multiprogramming OS is an ability of an operating system that executes more than one program using a single processor machine.
- In Multi-programming, More than one task or program or jobs are present inside the main memory at one point of time.

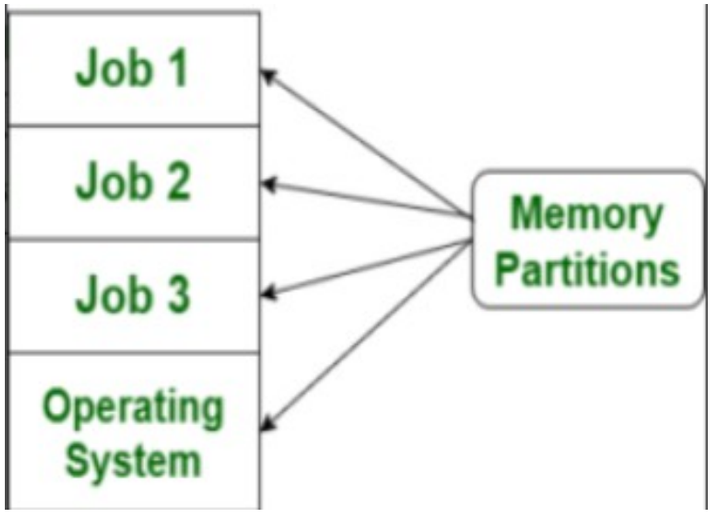


Fig 3. Multi-Programming OS

Operating System: Multi-Programming OS

Job

In systems using multiprogramming a program loaded to memory and ready to execute is called a job.

States: In a multiprogramming system, a job can be in one of three states.

- **Running**: The job is currently executing on the CPU. At any time, at most one job can be in this state.
- **Ready**: The job is ready to run but currently not selected to do so.
- **Waiting**: The job is blocked from running on the CPU while waiting for an I/O request to be completed.

UPES
UNIVERSITY OF TOMORROW

Operating System: Multi-Programming OS

State Transitions

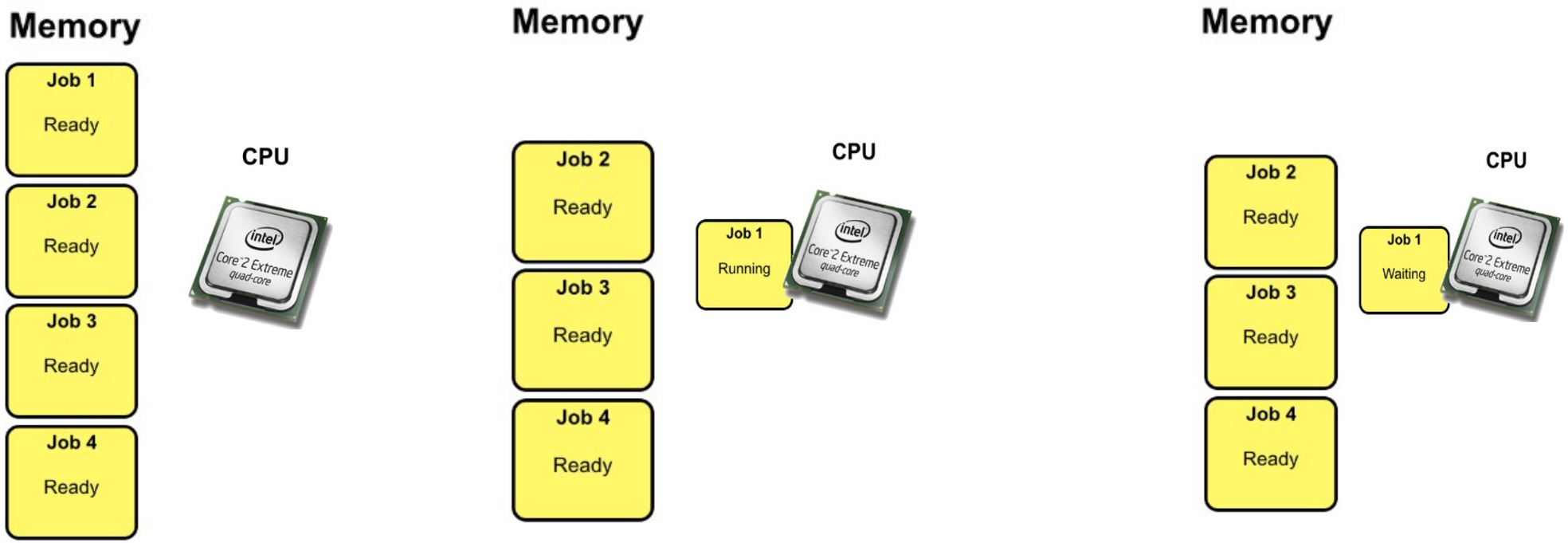
In a multiprogramming system, the following state transitions are possible

From	To	Description
Running	Waiting	When a running job requests I/O, the job changes state from running to waiting.
Running	Ready	When an I/O requests completes, the running job changes state from running to ready.
Waiting	Ready	When an I/O requests completes, the job waiting for the request to complete changes state from waiting to ready.
Ready	Running	When an I/O requests completes, one of the ready jobs are selected to run on the CPU and changes state from ready to running.

Fig 4. State Transition in Multi-Programming System

UPES
UNIVERSITY OF TOMORROW

Operating System: Multi-Programming OS



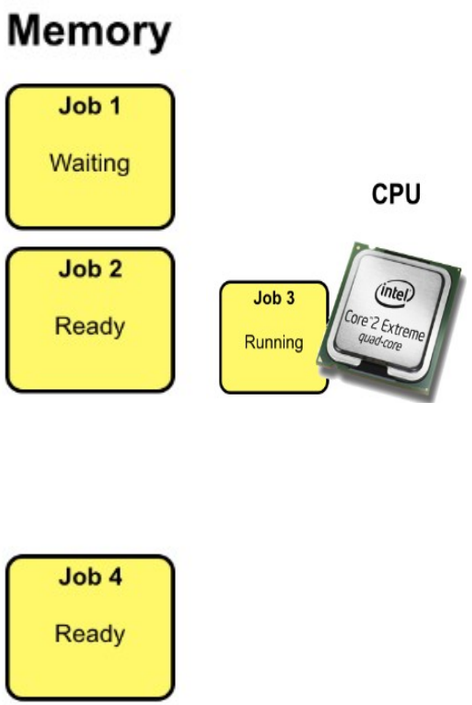
Initially, all jobs are int the ready state.

One of the ready jobs (Job 1) is selected to execute on the CPU and changes state from ready to running.

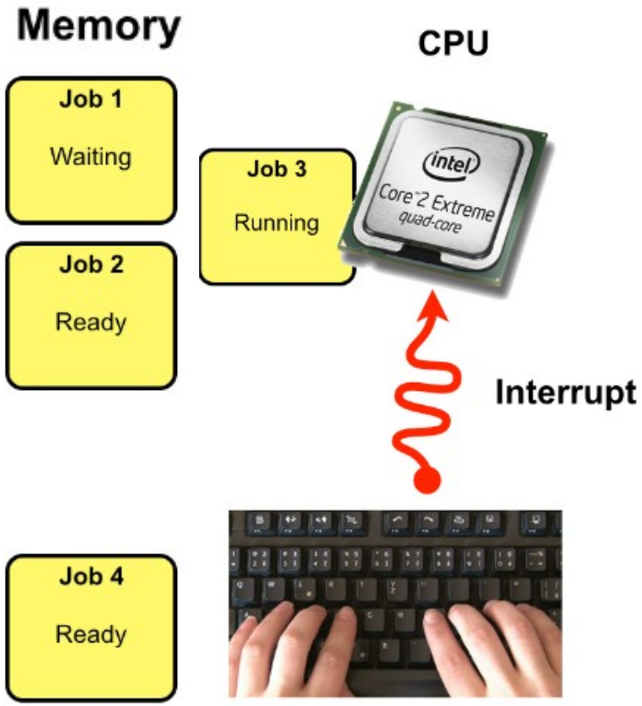
The running job 1 makes a request for I/O and the state changes from running to waiting.

Fig 5. Working of Multi-Programming OS

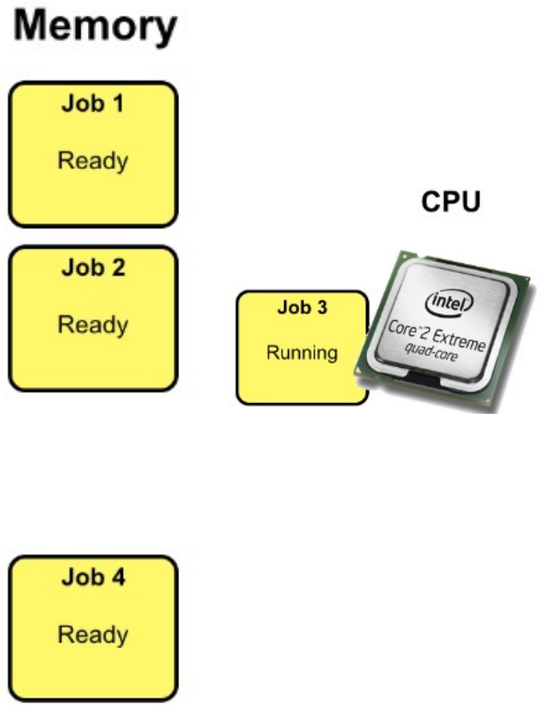
Operating System: Multi-Programming OS



Instead of idle waiting for the I/O request to complete, one of the ready jobs (Job 3) is selected to execute on the CPU and have its state change from ready to running.



Eventually the the I/O request job 1 is waiting for will complete and the CPU will be notified by an interrupt. In this example, job 1 was waiting for a keypress on the keyboard.



The state of the waiting job (job 1) will change from waiting to ready.

UPES
UNIVERSITY OF TOMORROW

Operating System: Multi-Programming OS

Advantages of Multiprogramming

- Need Single CPU for implementation.
- Context switch between process.
- Switching happens when current process undergoes waiting state.
- CPU idle time is reduced.
- High resource utilization.
- High Performance.

Disadvantages of Multiprogramming

- Prior knowledge of scheduling algorithms (An algorithm that decides which next process will get hold of the CPU) is required.
- If it has a large number of jobs, then long-term jobs will have to require a long wait.
- Memory management is needed in the operating system because all types of tasks are stored in the main memory.
- Using multiprogramming up to a larger extent can cause a heat-up issue.

UPES
UNIVERSITY OF TOMORROW

Operating System: Multi-Processing System

- A multiprocessing operating system is defined as a type of operating system that makes use multiple central processing units within a single system to improve performance.
- It enables a system to support more than one processor and divide the tasks among them.
- Every process requires a CPU for its execution. So, this allows multiple processes to execute parallelly on different processing units.
- All available processors are connected to peripheral devices, computer buses, physical memory, and clocks.
- The main aim of the multi-processing operating system is to increase the speed of execution of the system.
- For example, UNIX, LINUX, and Solaris are the most widely used multi-processing operating system.

Operating System: Multi-Processing System

Working of Multi-Processing OS

- In a multiprocessing operating system, the workload is divided among the multiple processors or cores.
- Each processor handles a specific task, which allows for improved performance and faster execution.
- After the completion of the task, the results from each processor are compiled to produce a single output.
- The operating system manages the allocation of resources and ensures that each processor is assigned a task it can handle efficiently.
- This results in better resource utilization and optimized system performance.

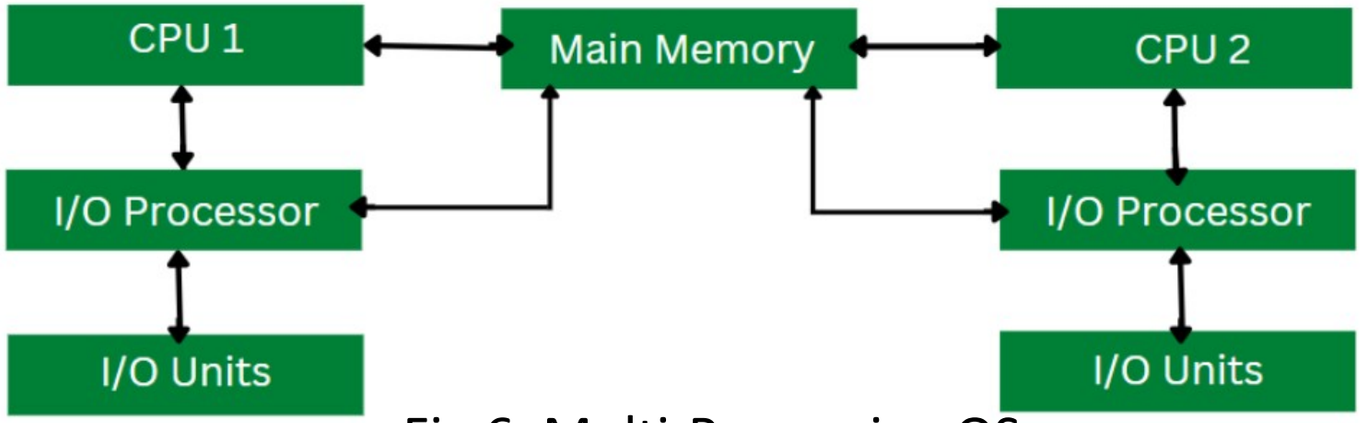


Fig 6. Multi-Processing OS

Operating System: Multi-Processing System

Advantages of Multiprocessing OS

- **Increased reliability:** Due to the multiprocessing system, processing tasks can be distributed among several processors. This increases reliability as if one processor fails; the task can be given to another processor for completion.
- **Increased throughout:** As several processors increase, more work can be done in less
- **The economy of Scale:** As multiprocessors systems share peripherals, secondary storage devices, and power supplies, they are relatively cheaper than single-processor systems.

Disadvantages of Multiprocessing operating System

- Multiprocessing Operating Systems are complex and require specialized knowledge.
- The cost of a Multiprocessing Operating system can be high because of the need for specialized hardware resources.
- They may face compatibility issues with software that is not designed to work with multiprocessing operating systems.
- Achieve Synchronization between multiple processors in a multiprocessing operating system is a challenging task.

UPES
UNIVERSITY OF TOMORROW

Operating System: Multi-Tasking OS

- The term "multitasking" is commonly used in modern computer systems.
- It is an advancement of multiprogramming systems, which enables the execution of several programs concurrently.
- A multitasking operating system enables a user to perform multiple computer tasks simultaneously. These tasks are referred to as processes, and they share processing resources such as the CPU.
- The operating system maintains a record of the progress of each process and enables users to switch between them without losing any data.

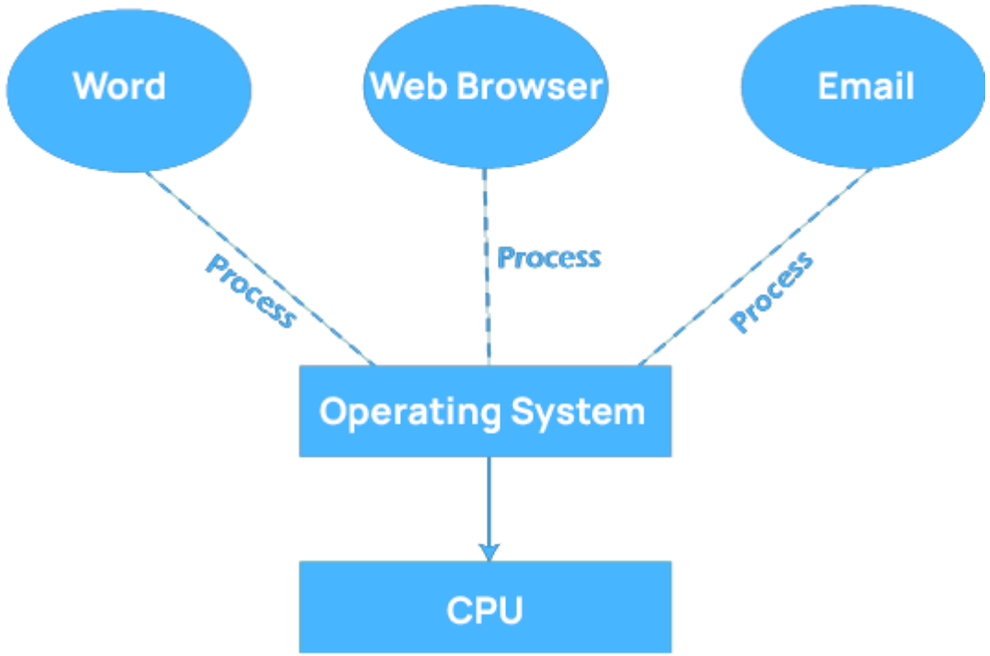


Fig 7. Multi-Tasking OS

Operating System: Multi-Tasking OS

- Early operating system could execute various programs at the same time, although multitasking was not fully supported.
- As a result, a single software could consume the entire CPU of the computer while completing a certain activity.
- Basic operating system functions, such as file copying, prevented the user from completing other tasks, such as opening and closing windows.
- Fortunately, because modern operating systems have complete multitasking capability, numerous programs can run concurrently without interfering with one other.
- In addition, many operating system processes can run at the same time.

Operating System: Multi-Tasking OS

Features of Multi-Tasking Operating System

Time Sharing: Many processes are allocated with resources of computer in respective time slots, processors time is shared with multiple processes.

Context Switching: Context switching is a process of saving the context of one process and loading the context of another process. In simpler terms it is loading another process when the prior process has finished its execution.

Multi-Threading: Multithreading is the ability of a program or an operating system to enable more than one user at a time without requiring multiple copies of the program running on the computer.

Hardware Interrupt: When a process or an event requires urgent attention, hardware or software will signal with an interrupt. It informs the processor that a high-priority task has arisen that necessitates interrupting the running process.

UPES
UNIVERSITY OF TOMORROW

Operating System: Multi-Tasking OS

Types of Multi-Tasking OS

Pre-emptive Multi-Tasking Operating System:

- In pre-emptive multitasking, the operating system can initiate a context switching from the running process to another process.
- In other words, the operating system allows stopping the execution of the currently running process and allocating the CPU to some other process.
- The OS uses some criteria to decide for how long a process should execute before allowing another process to use the operating system.
- The mechanism of taking control of the operating system from one process and giving it to another process is called pre-emption. Here are some Examples UNIX, Windows 95, Windows NT operating system.

Operating System: Multi-Tasking OS

Types of Multi-Tasking OS

Non-pre-emptive Multi-Tasking Operating System:

- Non-pre-emptive Multi-Tasking Operating System is also known as cooperative multitasking, this operating system never initiates context switching from the running process to another process.
- A context switch occurs only when the processes voluntarily yield control periodically or when idle or logically blocked to allow multiple applications to execute simultaneously.
- Also, in this multitasking, all the processes cooperate for the scheduling scheme to work. Example – Macintosh OS version 8.0-9.2.2 and Windows 3.x operating system.

Operating System: Multi-Tasking OS

Advantages of Multitasking Operating System

- Multitasking operating systems allow multiple applications to run concurrently without affecting CPU performance, making them suitable for multiple users working simultaneously.
- Multitasking operating systems possess a robust virtual memory system that eliminates long wait times for program execution by shifting applications to virtual memory if necessary.
- Additionally, to prevent CPU wait times, all jobs in Multitasking Operating System are given a predetermined duration.
- Multitasking operating systems can efficiently manage computer resources such as I/O devices, RAM, hard disks, and CPUs.
- Users can concurrently run several programs, such as internet browsers, games, Microsoft Excel, PowerPoint, and other utilities in a Multitasking Operating System.

Operating System: Multi-Tasking OS

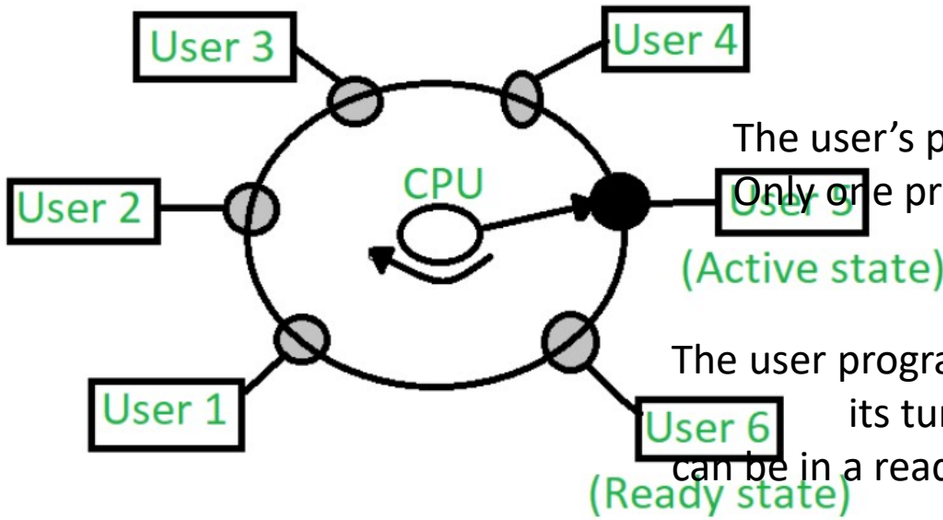
Disadvantages of Multitasking Operating System

- Due to the slower pace of the processors in a Multitasking Operating System, the computer system may perform slowly and experience longer response times when running multiple programs. To fix this issue, additional processing power may be required.
- Running numerous programs concurrently in a Multitasking Operating System can overload the main memory, leading to slow system performance and increased reaction times since the CPU cannot allocate sufficient time for each program.
- In a multitasking operating system, multiple processors work simultaneously to complete tasks, resulting in increased CPU heat generation.

Operating System: Time Sharing OS

Multi-programmed, batched systems provide an environment where various system resources were used effectively, but it did not provide for user interaction with computer systems. Time-sharing is a logical extension of multi-programming. A time-sharing operating system i design allows multiple users (processes) to concurrently share the same system resources, such as the CPU, memory, and peripherals. It enables each user or process to have the illusion of having dedicated access to the system while effectively sharing resources in a time-sliced manner.

Waiting State:
The user's program is waiting for some input/output operation. More than one user can be in a waiting state at a time.



The user's program is under the control of the CPU. Only one program is available in this state.

The user program is ready to execute but it is waiting for its turn to get the CPU. More than one user can be in a ready state at a time.

Fig 8. Time Shared OS

UPES
UNIVERSITY OF TOMORROW

Operating System: Time Sharing OS

- Time-shared operating system uses CPU scheduling and multi-programming to provide each user with a small portion of a shared computer at once.
- Each user has at least one separate program in memory.
- A program is loaded into memory and executes, it performs a short period of time either before completion or to complete I/O.
- This short period of time during which the user gets the attention of the CPU is known as time slice, time slot, or quantum. It is typically of the order of 10 to 100 milliseconds.
- An alarm clock mechanism to send an interrupt signal to the CPU after every time slice.

Operating System: Time Sharing OS

Advantages of Time-Sharing OS

- Each task gets an equal opportunity.
- Fewer chances of duplication of software.
- CPU idle time can be reduced.

Disadvantages of Time-Sharing OS

- Reliability problem.
- One must have to take of the security and integrity of user programs and data.
- Data communication problem.

Operating System: Distributed OS

A distributed operating system is one in which several computer systems connected through a single communication channel. Moreover, these systems have their individual processors and memory. Furthermore, these processors communicate through high-speed buses or telephone lines. These individual systems that connect through a single channel are considered as a single unit. We can also call them loosely coupled systems. The individual components or systems of the network are nodes.

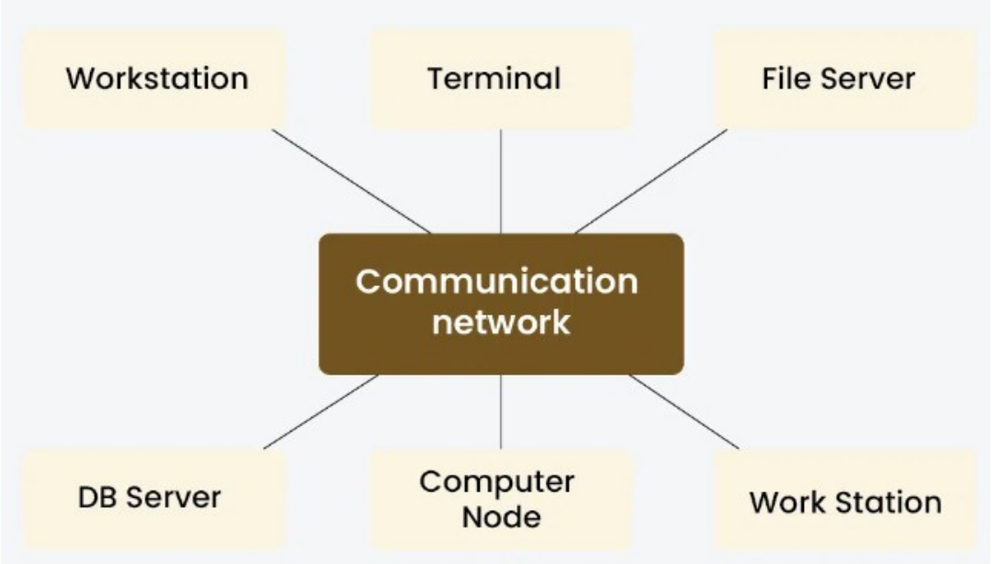


Fig 9. Distributed OS

Operating System: Distributed OS

Types of Distributed OS

- **Client-Server Systems**

In a client-server system within a distributed operating system, clients request services or resources from servers over a network. Clients initiate communication, send requests, and handle user interfaces, while servers listen for requests, perform tasks, and manage resources.

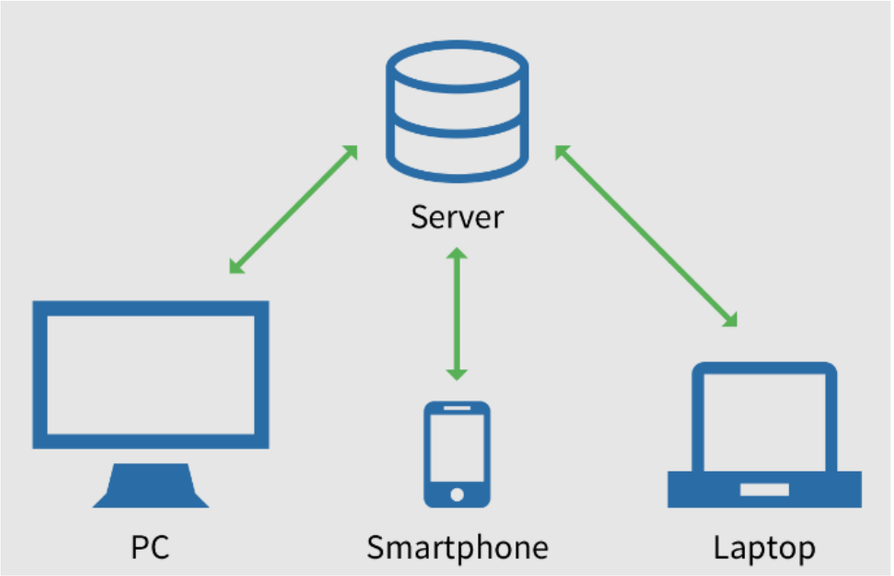


Fig 10. Client-Server OS

Operating System: Distributed OS

Types of Distributed OS

Peer-to-Peer(P2P) Systems

In peer-to-peer (P2P) systems, interconnected nodes directly communicate and collaborate without centralized control. Each node can act as both a client and a server, sharing resources and services with other nodes. P2P systems enable decentralized resource sharing, self-organization, and fault tolerance.

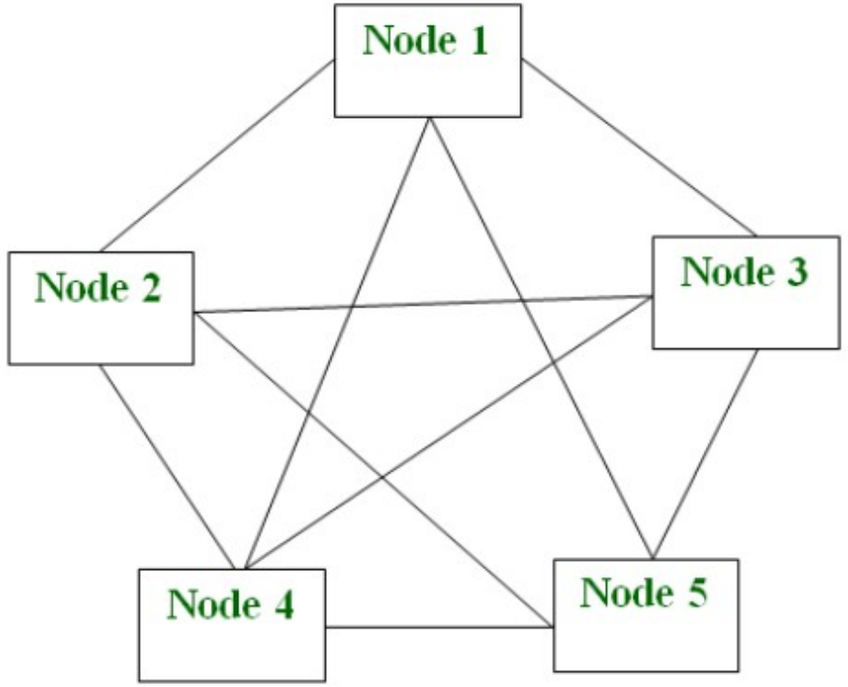


Fig 11. Peer-to-Peer OS

Operating System: Distributed OS

Types of Distributed OS

Middleware

Middleware acts as a bridge between different software applications or components, enabling communication and interaction across distributed systems. It abstracts complexities of network communication, providing services like message passing, remote procedure calls (RPC), and object management.

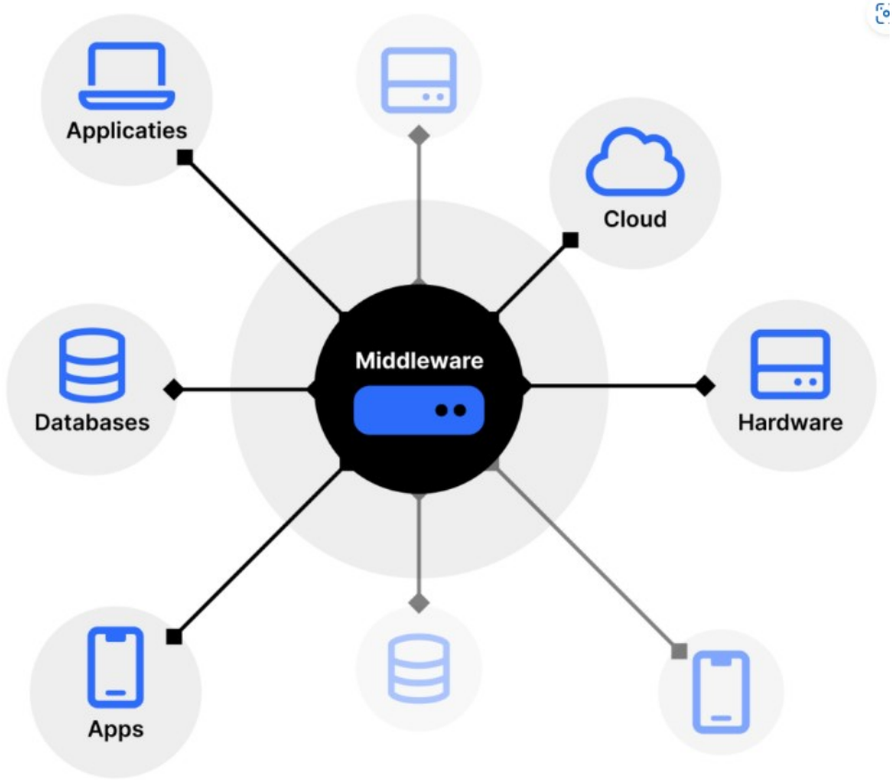


Fig 12. Middleware OS

Operating System: Distributed OS

Types of Distributed OS

Three-Tier OS

In a distributed operating system, “Tier” refer to the physical separation of components. the three-tier architecture divides tasks into presentation, logic, and data layers. The presentation tier, comprising client machines or devices, handles user interaction. The logic tier, distributed across multiple nodes or servers, executes processing logic and coordinates system functions.

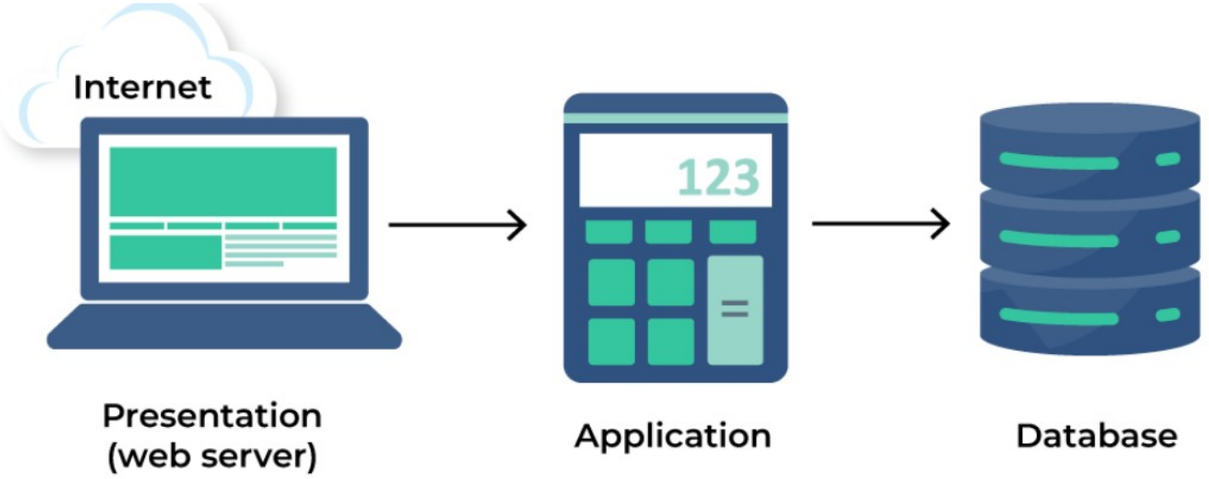


Fig 13. Three-Tier OS

Operating System: Distributed OS

Types of Distributed OS

N-Tier OS

In an N-tier architecture, applications are structured into multiple tiers or layers beyond the traditional three-tier model. Each tier performs specific functions, such as presentation, logic, data processing, and data storage, with the flexibility to add more tiers as needed. In a distributed operating system, this architecture enables complex applications to be divided into modular components distributed across multiple nodes or servers.

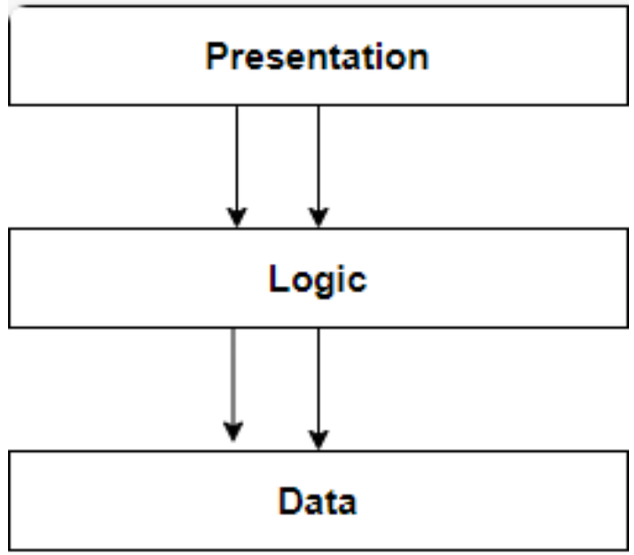


Fig 14. N-Tier OS

Operating System: Distributed OS

Examples of Distributed OS

- **Solaris:** The SUN multiprocessor workstations are the intended use for it.
- **OSF/1:** The Open Foundation Software Company designed it, and it works with Unix.
- **Micros:** All nodes in the system are assigned work by the MICROS operating system, which also guarantees a balanced data load.
- **DYNIX:** It is created for computers with many processors, known as Symmetry.
- **Locus:** It can be viewed simultaneously from both local and distant files without any locatic restrictions.
- **Mach:** It permits the features of multitasking and multithreading.

UPES
UNIVERSITY OF TOMORROW

Operating System: Distributed OS

Advantages of Distributed OS

- It can increase data availability throughout the system by sharing all resources (CPU, disk, network interface, nodes, computers, and so on) between sites.
- Because all data is replicated across all sites, it reduces the probability of data corruption because users can access data from another operating site if one site fails.
- Data transfer from one site to another is accelerated by it.
- Since it may be accessible from both local and remote sites, it is an open system.
- It facilitates a reduction in the time needed to process data.
- Most distributed systems are composed of multiple nodes that work together to provide fault tolerance. Even if one machine malfunctions, the system still functions.

Operating System: Distributed OS

Disadvantages of Distributed OS

- Which tasks need to be completed, when they need to be completed, and where they need to be completed must be determined by the system. The restrictions of a scheduler might result in unpredictable runtimes and unused hardware.
- Since the nodes and connections in DOS need to be secured, it is challenging to establish sufficient security.
- Comparing a DOS-connected database to a single-user system, the latter is easier to maintain and less complex.
- Compared to other systems, the underlying software is incredibly sophisticated and poorly understood.
- Compiling, analyzing, displaying, and keeping track of hardware utilization metrics for large clusters may be quite challenging.

UPES
UNIVERSITY OF TOMORROW

Operating System: Network OS

- Since an operating system is that the operating system is the interface between the computer hardware and the user. In daily life, we use the operating system on our devices which provides a good GUI, and many more features.
- 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.

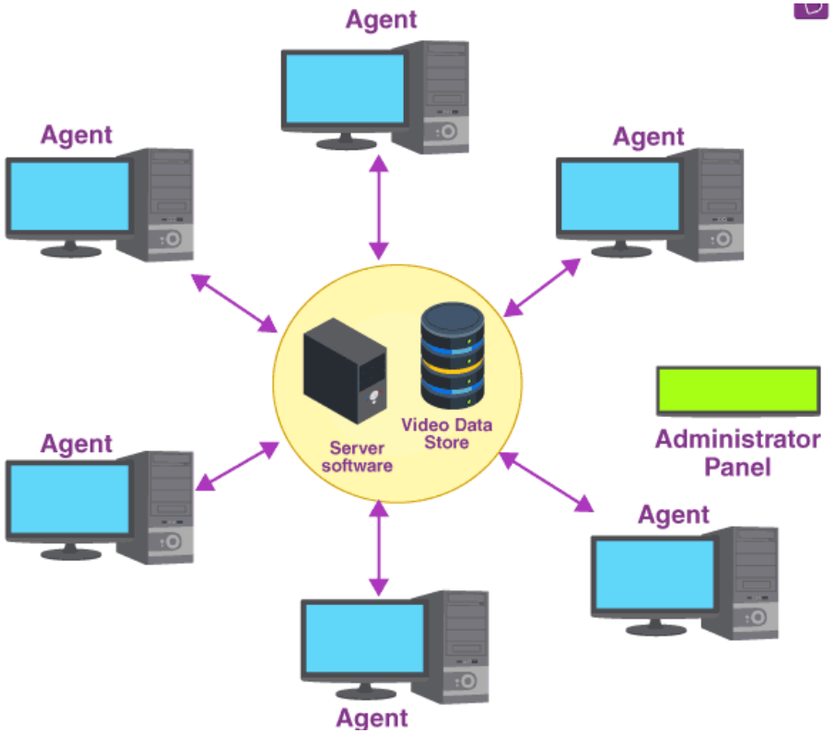


Fig 15. Network OS

Operating System: Network OS

Functions of the Network OS

- 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.

Types of Network OS

- **Peer to Peer**

Peer-to-peer network operating systems allow the sharing of 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.

Operating System: Network OS

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.

Examples of Network OS

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

UPES
UNIVERSITY OF TOMORROW

Operating System: Real Time OS

Real-time OS are used in environments where a large number of events, mostly external to the computer system, must be accepted and processed in a short time or within certain deadlines. such applications are industrial control, telephone switching equipment, flight control, and real-time simulations. With a Real Time OS, the processing time is measured in tenths of seconds. This system is time-bound and has a fixed deadline. The processing in this type of system must occur within the specified constraints. Otherwise, This will lead to system failure.

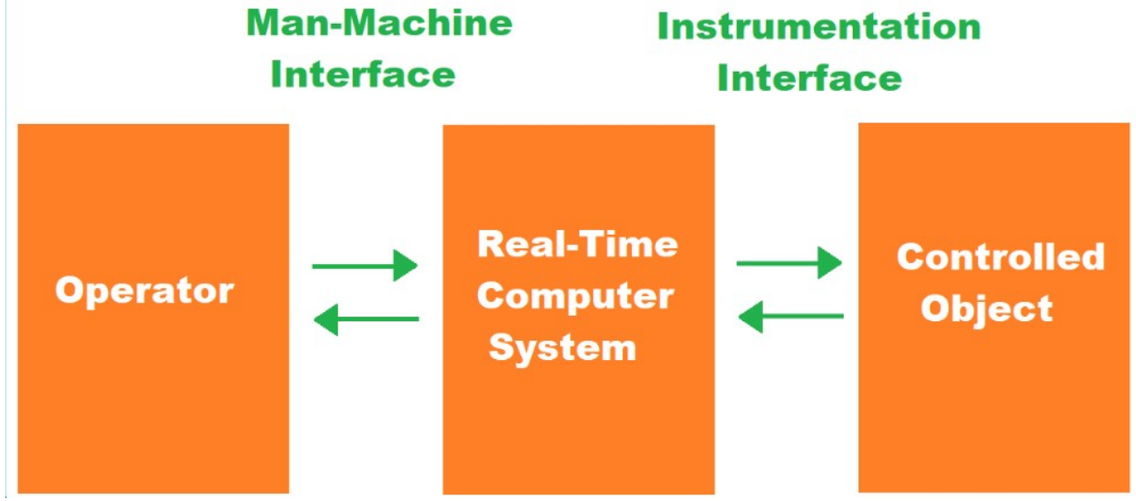


Fig 16. Real Time OS

Operating System: Real Time OS

Types of Real Time OS

Hard Real-Time OS

These operating systems guarantee that critical tasks are completed within a range of time.
For example, a robot is hired to weld a car body. If the robot welds too early or too late, the car cannot be sold, so it is a hard real-time system that requires complete car welding by the robot hardly on time., scientific experiments, medical imaging systems, robots, air traffic control systems, etc.

Soft real-time OS

This operating system provides some relaxation in the time limit.
For example – Multimedia systems, digital audio systems, etc. Explicit, programmer-defined, and controlled processes are encountered in real-time systems. A separate process is changed by handling a single external event. The process is activated upon the occurrence of the related event signaled by an interrupt.

Operating System: Real Time OS

Types of Real Time OS

Firm Real-time Operating System

Real Time OS of this type have to follow deadlines as well. In spite of its small impact, missing a deadline can have unintended consequences, including a reduction in the quality of the product.
Example: Multimedia applications.

Deterministic Real-Time OS

Consistency is the main key in this type of real-time operating system. It ensures that all the task and processes execute with predictable timing all the time, which make it more suitable for applications in which timing accuracy is very important. Examples: INTEGRITY, PikeOS.

Operating System: Real Time OS

Advantages of Real-Time OS

- Maximum utilization of devices and systems. Thus, more output from all the resources.
- Time assigned for shifting tasks in these systems is very less. For example, in older systems, it takes about 10 microseconds. Shifting one task to another and in the latest systems, it takes 3 microseconds.
- Focus on running applications and less importance to applications that are in the queue.
- Since the size of programs is small, RTOS can also be embedded systems like in transport and others.
- These types of systems are error-free.
- Memory allocation is best managed in these types of systems.

Operating System: Real Time OS

Dis-advantages of Real-Time OS

- Very few tasks run simultaneously, and their concentration is very less on few applications to avoid errors.
- Sometimes the system resources are not so good, and they are expensive as well.
- The algorithms are very complex and difficult for the designer to write on.
- It needs specific device drivers and interrupts signals to respond earliest to interrupts.
- It is not good to set thread priority as these systems are very less prone to switching tasks.
- Real-Time OS performs minimal task switching.

Operating System: Functions

Memory Management

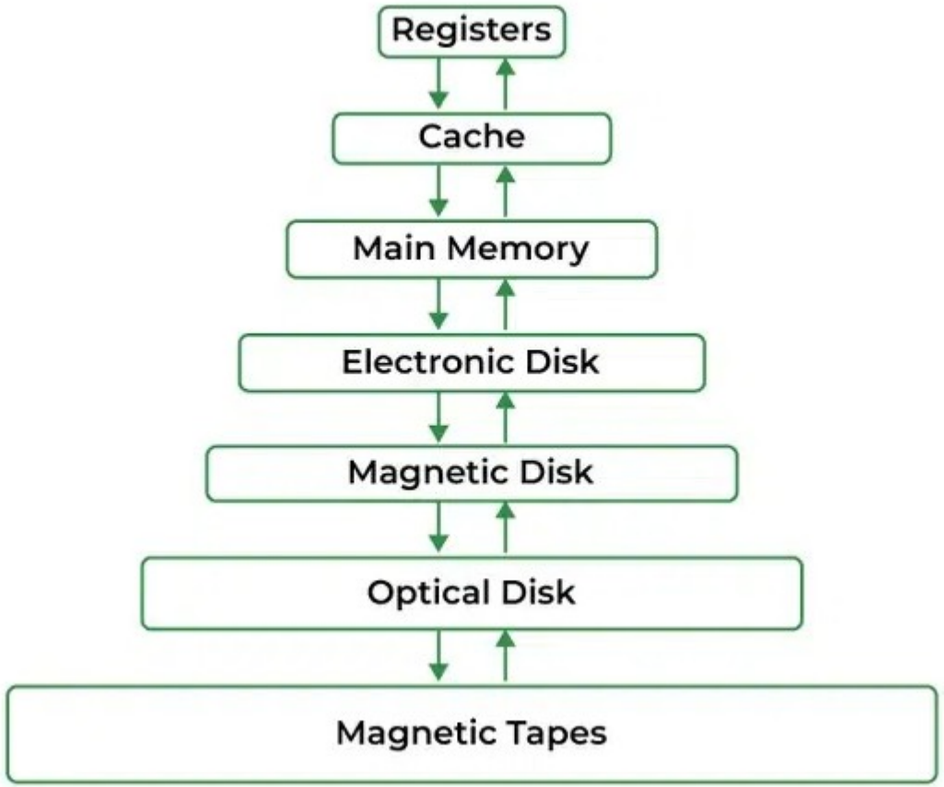


Fig 17 . Memory Management in OS

Operating System handles the following responsibilities

- It is the management of the main or primary memory. Whatever program is executed, it must be present in the main memory.
- It is used for achieving better concurrency, system performance, and memory utilization.
- Main memory is a quick storage area that may be accessed directly by the CPU.
- When the program is completed, the memory region is released and can be used by other programs.
- Therefore, there can be more than one program present at a time. Hence, it is required to manage the memory.

Operating System: Functions

- OS Allocates and deallocates the memory.
- OS Keeps a record of which part of primary memory is used by whom and how much.
- OS Distributes the memory while multiprocessing.
- In multiprogramming, the operating system selects which processes acquire memory when and how much memory they get.
- Memory management moves processes from primary memory to secondary memory and vice versa. It also keeps track of available memory, memory allocation, and unallocated.

Operating System: Functions

- It keeps track of the status of each memory location, whether it is allocated or free.
- It enables computer systems to run programs that require more main memory than the amount of free main memory available on the system. This is achieved by moving data between primary and secondary memory.
- It addresses the system's primary memory by providing abstractions such that the programs running on the system perceive a large memory is allocated to them.
- It is the job of memory management to protect the memory allocated to all the processes from being corrupted by other processes. If this is not done, the computer may exhibit unexpected/faulty behavior.
- Memory management enables sharing of memory spaces among processes, with the help of which, multiple programs can reside at the same memory location (although only one at a time).

Operating System: Functions

Operating System handles the following responsibilities

Process Management

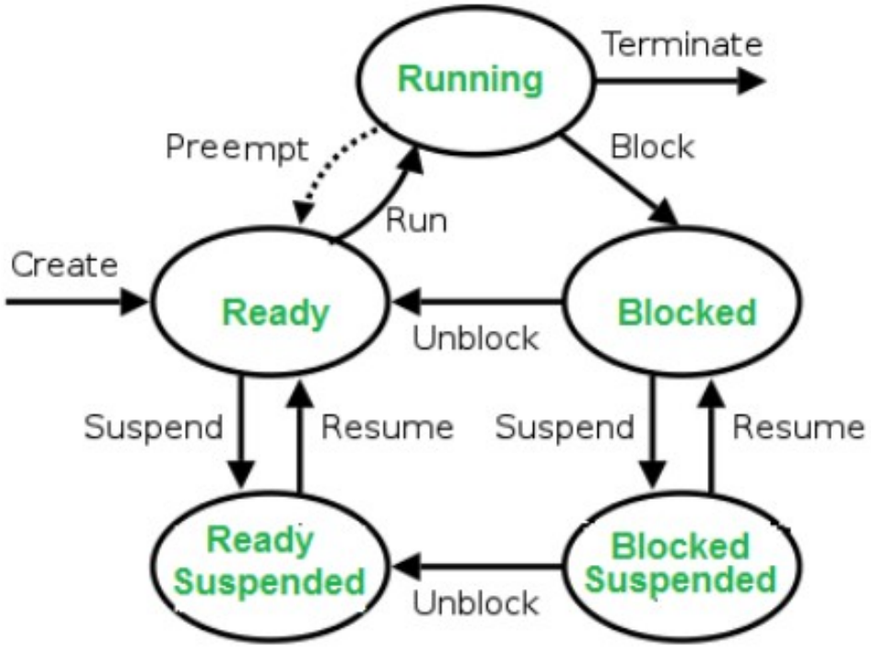


Fig 18. Process Management in OS

- In a multi-programming environment, the OS decides the order in which processes have access to the processor, and how much processing time each process has.
- This function of OS is called Process Scheduling.
- An Operating System performs the following activities for Processor Management.
- An operating system manages the processor's work by allocating various jobs to it and ensuring that each process receives enough time from the processor to function properly.
- Keeps track of the status of processes. The program which performs this task is known as a traffic controller. Allocates the CPU that is a processor to a process. De-allocates processor when a process is no longer required.

UPES
UNIVERSITY OF TOMORROW

Operating System: Functions

Operating System handles the following responsibilities

Device Management

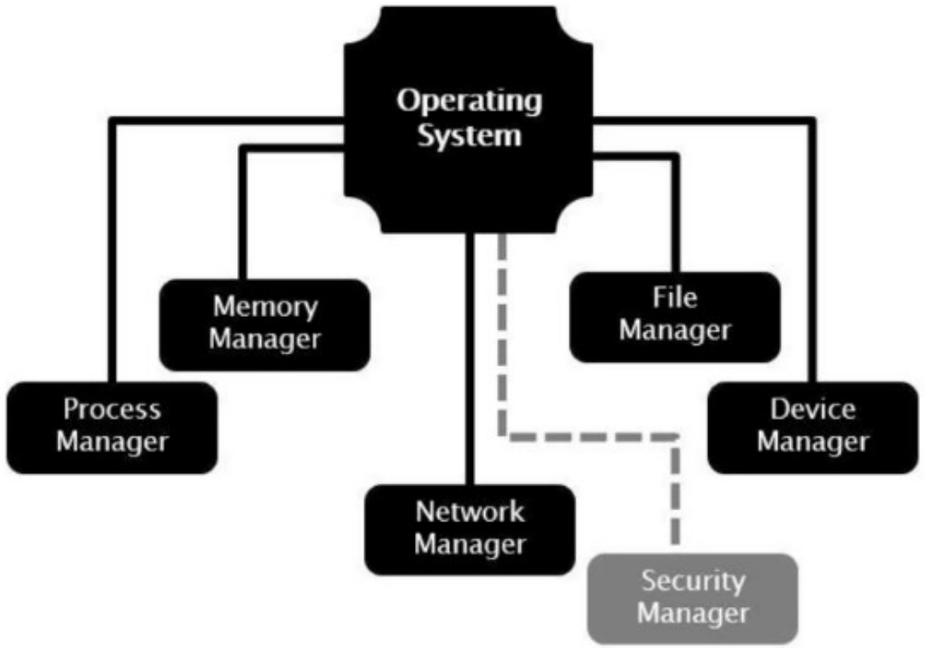


Fig 19. Device Management in OS

- OS Keeps track of all devices connected to the system.
- It designates a program responsible for every device known as the Input/Output controller.
- It decides which process gets access to a certain device and for how long.
- It allocates devices effectively and efficiently. Deallocates devices when they are no longer required.
- There are various input and output devices. An OS controls the working of these input-output devices.
- It receives the requests from these devices, performs a specific task, and communicates back to the requesting process.

Operating System: Functions

Operating System handles the following responsibilities

File Management

- A file system is organized into directories for efficient or easy navigation and usage.
- These directories may contain other directories and other files.
- An Operating System carries out the following file management activities.
- It keeps track of where information is stored, user access settings, the status of every file, and more.
- These facilities are collectively known as the file system.
- An OS keeps track of information regarding the creation, deletion, transfer, copy, and storage of files in an organized way.
- It also maintains the integrity of the data stored in these files, including the file directory structure, by protecting against unauthorized access.

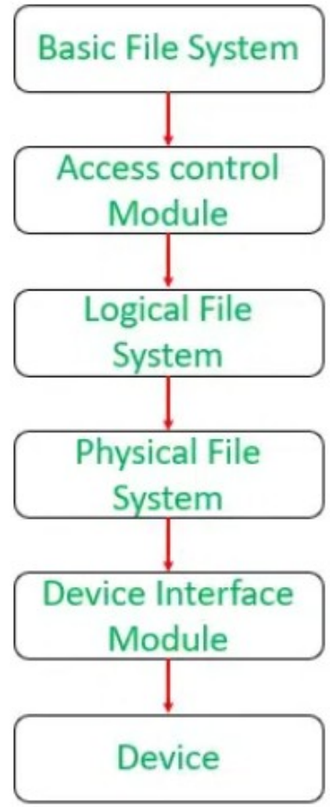


Fig 20 . File Management in OS

Operating System: Functions

Operating System handles the following responsibilities

User Interface or Command Interpreter

- The user interacts with the computer system through the operating system.
- Hence OS acts as an interface between the user and the computer hardware.
- This user interface is offered through a set of commands or a graphical user interface (GUI). Through this interface, the user makes interacts with the applications and the machine hardware.

Fig 21 .Command Interpreter in OS

Booting the Computer

- The process of starting or restarting the computer is known as booting.
- If the computer is switched off completely and if turned on, then it is called cold booting.
- Warm booting is a process of using the operating system to restart the computer.

Operating System: Functions

Operating System handles the following responsibilities

Security

- Protection against unauthorized access through login.
- Protection against intrusion by keeping the firewall active.
- Protecting the system memory against malicious access.
- Displaying messages related to system vulnerabilities.

Error-Detecting Aids

- The operating system constantly monitors the system to detect errors and avoid malfunctioning computer systems.
- From time to time, the operating system checks the system for any external threat or malicious software activity.

Network Management

- OS manage how data is packaged and sent over the network, making sure it arrives safely and in the right order.
- OS also enable to set up your network connections, like Wi-Fi or Ethernet, and keep an eye on how your network is doing.
- OS keep make sure your computer is using the network efficiently and securely.

UPES
UNIVERSITY OF TOMORROW

MCQ

Q1. Programs in the first generation of computers were written in:

A. Assembly language
B. Machine language
C. High-level language
D. Structured language

Q2. Second-generation operating systems were built on:

A. Time-sharing
B. Batch processing
C. Multiprogramming
D. Real-time processing

Q3. The fourth generation of operating systems is linked with the evolution of:

A. Supercomputers
B. Mainframes
C. Personal computers
D. Embedded systems

Q4. In a multiprogramming system, a program ready to execute is called a:

A. Job
B. Task
C. Thread
D. Process

MCQ

Q5. Which operating system can execute more than one program using a single processor?

A. Single-tasking OS B. Batch OS

C. Multiprogramming OS D. Real-time OS

Q6. Which OS design allows multiple users to share system resources concurrently?

A. Batch OS B. Time-sharing OS

C. Single-user OS D. Network OS

Q7. In a multiprogramming system, the state where a job is blocked from running while waiting for an I/O request is:

A. Running B. Ready

C. Waiting D. Executing

Q8. The main challenge in a multiprogramming system is:

A. Resource management B. User interaction

C. Speed of execution D. File management

UPES
UNIVERSITY OF TOMORROW

MCQ

Question No- Answer	Option	Description
Q1- Answer	B	Machine language
Q2- Answer	B	Batch processing
Q3- Answer	C	Personal computers
Q4- Answer	A	Job
Q5- Answer	C	Multiprogramming OS
Q6- Answer	B	Time-sharing OS
Q7- Answer	C	Waiting
Q8- Answer	A	Resource management

Summary/Key Points

- An operating system is a type of software that acts as an interface between the user and the hardware
- Operating system is evolved in Four generations: First, Second, Third and Fourth Generation
- There are several types of Operating Systems:
  - Batch Operating System
  - Multi-Programming System
  - Multi-Processing System
  - Multi-Tasking Operating System
  - Time-Sharing Operating System
  - Distributed Operating System
  - Network Operating System
  - Real-Time Operating System

UPES
UNIVERSITY OF TOMORROW

Summary/Key Points

- Operating system handles some major responsibility which includes:
  - Memory Management
  - Process Management
  - Device management
  - File management
  - User interface & Command Interpreter
  - Security
  - Error detection
  - Network Management

Reference Material

- Operating Systems Concepts (10th Ed.) Silberschatz A, Peterson J and Galvin P, John Wiley & Sons, Inc. 2018.

  Topics:

  - Operating System: History, Page No: 3-10
  - Operating System: Types, Page No: 15-21
  - Operating System: Functions, Page No: 21-27

- Modern Operating Systems (4th Ed.) by Andrew S. Tanenbaum and Herbert Bos, 2007
- Operating Systems: Principles and Practice by Thomas Anderson and Michael Dahlin, 2014

UPES
UNIVERSITY OF TOMORROW

Coming Up-Next Lecture

- Service of Operating System
- Computing Environments

UPES
UNIVERSITY OF TOMORROW

Thank You

UPES
UNIVERSITY OF TOMORROW