**S.K.D.A.V. GOVERNMENT POLYTECHNIC, ROURKELA**

****

**SUBJECT-OPERATING SYSTEM**

**SEMESTER-4TH**

**BRANCH-INFORMATION TECHNOLOGY**

## MODULE-1

## INTRODUCTION

## DEFINITION OF OS:-

## In the Computer System (comprises of Hardware and software), Hardware can only understand machine code (in the form of 0 and 1) which doesn't make any sense to a naive user.

We need a system which can act as an intermediary and manage all the processes and resources present in the system.

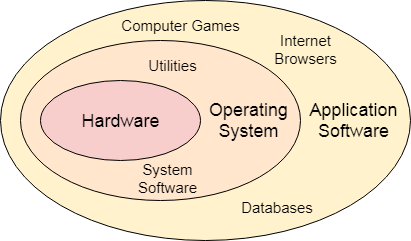
An **Operating System** can be defined as an **interface between user and hardware**. It is responsible for the execution of all the processes, Resource Allocation, CPU management, File Management and many other tasks.

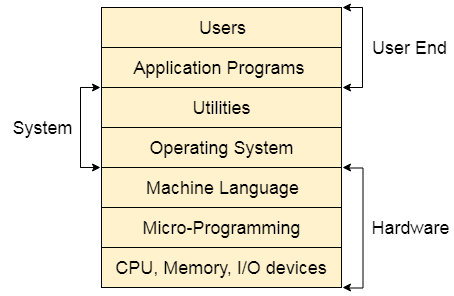
The purpose of an operating system is to provide an environment in which a user can execute programs in convenient and efficient manner.

## STRUCTURE OF A COMPUTER SYSTEM

## A Computer System consists of:

* Users (people who are using the computer)
* Application Programs (Compilers, Databases, Games, Video player, Browsers, etc.)
* System Programs (Shells, Editors, Compilers, etc.)
* Operating System ( A special program which acts as an interface between user and hardware )
* Hardware ( CPU, Disks, Memory, etc)





## Objectives of Operating System

An operating system consists of a special program that controls the execution of the application program. The OS acts as an intermediary between applications and hardware components. OS can be thought of as having three objectives. These are:

* **Convenience**: It makes a computer more suitable to use.
* **Efficiency**: It provides the computer system resources with efficiency and in easy to use format.
* **Ability to develop**: It should be built in such a way that it permits the efficient development, testing, and installation of new system functions without interfering with service.

## Functions of Operating System

**Security –**  
The operating system uses password protection to protect user data and similar other techniques. it also prevents unauthorized access to programs and user data.

**Control over system performance –**  
Monitors overall system health to help improve performance. records the response time between service requests and system response to have a complete view of the system health. This can help improve performance by providing important information needed to troubleshoot problems.

**Job accounting –**  
Operating system Keeps track of time and resources used by various tasks and users, this information can be used to track resource usage for a particular user or group of user.

**Error detecting aids –**  
Operating system constantly monitors the system to detect errors and avoid the malfunctioning of computer system.

**Coordination between other software and users –**  
Operating systems also coordinate and assign interpreters, compilers, assemblers and other software to the various users of the computer systems.

**Memory Management –**  
The operating system manages the Primary Memory or Main Memory. Main memory is made up of a large array of bytes or words where each byte or word is assigned a certain address. Main memory is a fast storage and it can be accessed directly by the CPU. For a program to be executed, it should be first loaded in the main memory. An Operating System performs the following activities for memory management:

It keeps tracks of primary memory, i.e., which bytes of memory are used by which user program. The memory addresses that have already been allocated and the memory addresses of the memory that has not yet been used. In multi programming, the OS decides the order in which process are granted access to memory, and for how long. It Allocates the memory to a process when the process requests it and deallocates the memory when the process has terminated or is performing an I/O operation.

**Processor Management –**  
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.

Keeps tracks of the status of processes. The program which perform this task is known as traffic controller. Allocates the CPU that is processor to a process. De-allocates processor when a process is no more required.

**Device Management –**  
An OS manages device communication via their respective drivers. It performs the following activities for device management. Keeps tracks of all devices connected to system. designates a program responsible for every device known as the Input/Output controller. Decides which process gets access to a certain device and for how long. Allocates devices in an effective and efficient way. Deallocates devices when they are no longer required.

**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 and status of every file and more… These facilities are collectively known as the file system.

Moreover, Operating System also provides certain services to the computer system in one form or the other.  
**The Operating System provides certain services to the users which can be listed in the following manner:**

**Program Execution**: The Operating System is responsible for execution of all types of programs whether it be user programs or system programs. The Operating System utilises various resources available for the efficient running of all types of functionalities.

**Handling Input/Output Operations**: The Operating System is responsible for handling all sort of inputs, i.e, from keyboard, mouse, desktop, etc. The Operating System does all interfacing in the most appropriate manner regrading all kind of Inputs and Outputs.  
For example, there is difference in nature of all types of peripheral devices such as mouse or keyboard, then Operating System is responsible for handling data between them.

**Manipulation of File System**: The Operating System is responsible for making of decisions regarding the storage of all types of data or files, i.e, floppy disk/hard disk/pen drive, etc. The Operating System decides as how should the data should be manipulated and stored.

**Error Detection and Handling**: The Operating System is responsible for detection of any types of error or bugs that can occur while any task. The well secured OS sometimes also acts as countermeasure for preventing any sort of breach to the Computer System from any external source and probably handling them.

**Resource Allocation:** The Operating System ensures the proper use of all the resources available by deciding which resource to be used by whom for how much time. All the decisions are taken by the Operating System.

**Accounting:** The Operating System tracks an account of all the functionalities taking place in the computer system at a time. All the details such as the types of errors occurred are recorded by the Operating System.

**Information and Resource Protection:** The Operating System is responsible for using all the information and resources available on the machine in the most protected way. The Operating System must foil an attempt from any external resource to hamper any sort of data or information.

All these services are ensured by the Operating System for the convenience of the users to make the programming task easier. All different kinds of Operating System more or less provide the same services.

**Following are the major applications of an operating system −**

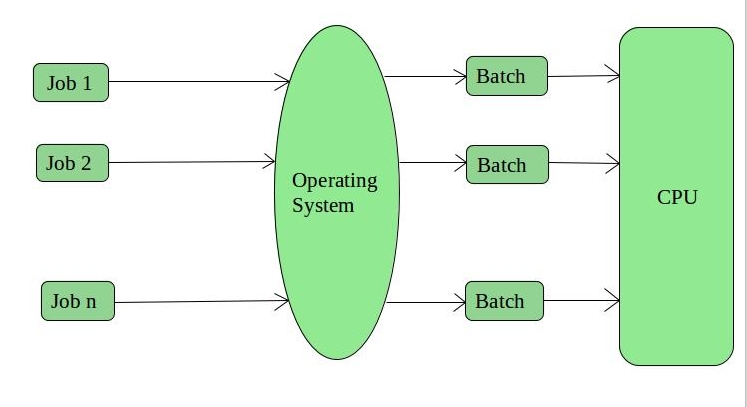
* An operating system is accountable for the formation and deletion of files and directories.
* An operating system manages the process of deletion, suspension, resumption, and synchronization.
* An operating system manages memory space by allocation and de-allocation.
* An operating system stores, organizes, and names and protects the existing files.
* Further, an operating system manages all the components and devices of the computers system including modems, printers, plotters, etc.
* In case, if any device fails, the operating system detects and notify.
* An operating system protects from destruction as well as from unauthorized use.
* An operating system facilitates the interface to user and hardware.

**TYPES OF OPERATING SYSTEMS**

An [Operating System](https://www.geeksforgeeks.org/operating-system-introduction-operating-system-set-1/) performs all the basic tasks like managing file,process, and memory. Thus operating system acts as manager of all the resources, i.e. **resource manager**. Thus operating system becomes an interface between user and machine.

**1. Batch Operating System –**

This type of operating system does not interact with the computer directly. There is an operator which takes similar jobs having same requirement and group them into batches. It is the responsibility of operator to sort the jobs with similar needs.



**Advantages of Batch Operating System:**

* It is very difficult to guess or know the time required by any job to complete. Processors of the batch systems know how long the job would be when it is in queue
* Multiple users can share the batch systems
* The idle time for batch system is very less
* It is easy to manage large work repeatedly in batch systems

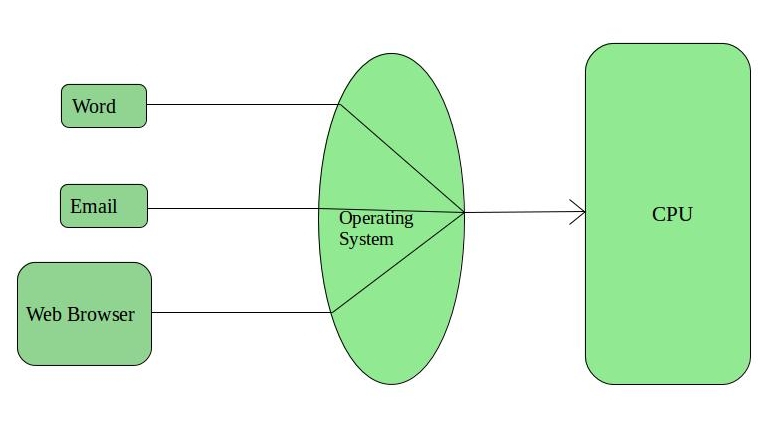
**Disadvantages of Batch Operating System:**

* The computer operators should be well known with batch systems
* Batch systems are hard to debug
* It is sometime costly
* The other jobs will have to wait for an unknown time if any job fails

**Examples of Batch based Operating System:** Payroll System, Bank Statements etc.

**2. Time-Sharing Operating Systems –**

Each task is given some time to execute, so that all the tasks work smoothly. Each user gets time of CPU as they use single system. These systems are also known as Multitasking Systems. The task can be from single user or from different users also. The time that each task gets to execute is called quantum. After this time interval is over OS switches over to next task.



**Advantages of Time-Sharing OS:**

* Each task gets an equal opportunity
* Less chances of duplication of software
* CPU idle time can be reduced

**Disadvantages of Time-Sharing OS:**

* Reliability problem
* One must have to take care of security and integrity of user programs and data
* Data communication problem

**Examples of Time-Sharing OSs are:** Multics, Unix etc.

## 3. Multiprocessor Systems

A Multiprocessor system consists of several processors that share a common physical memory. Multiprocessor system provides higher computing power and speed. In multiprocessor system all processors operate under single operating system. Multiplicity of the processors and how they do act together are transparent to the others.

**Advantages of Multiprocessor Systems**

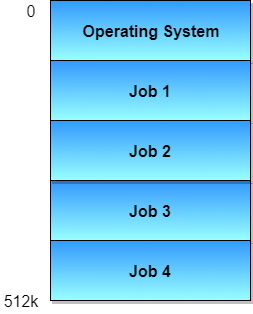
1. Enhanced performance
2. Execution of several tasks by different processors concurrently, increases the system's throughput without speeding up the execution of a single task.
3. If possible, system divides task into many subtasks and then these subtasks can be executed in parallel in different processors. Thereby speeding up the execution of single tasks.

## 4. Multiprogramming Batch Systems

* In this the operating system picks up and begins to execute one of the jobs from memory.
* Once this job needs an I/O operation operating system switches to another job (CPU and OS always busy).
* Jobs in the memory are always less than the number of jobs on disk(Job Pool).
* If several jobs are ready to run at the same time, then the system chooses which one to run through the process of **CPU Scheduling**.
* In Non-multiprogrammed system, there are moments when CPU sits idle and does not do any work.
* In Multiprogramming system, CPU will never be idle and keeps on processing.

**Time Sharing Systems** are very similar to Multiprogramming batch systems. In fact time sharing systems are an extension of multiprogramming systems.

In Time sharing systems the prime focus is on **minimizing the response time**, while in multiprogramming the prime focus is to maximize the CPU usage.



**5. Real-Time Operating System –**

These types of OSs serve the real-time systems. The time interval required to process and respond to inputs is very small. This time interval is called **response time**.

**Real-time systems** are used when there are times requirements are very strict like missile systems, air traffic control systems, robots etc.

**Two types of Real-Time Operating System which are as follows:**

* **Hard Real-Time Systems:**  
  These OSs are meant for the applications where time constraints are very strict and even the shortest possible delay is not acceptable. These systems are built for saving life like automatic parachutes or air bags which are required to be readily available in case of any accident. Virtual memory is almost never found in these systems.
* **Soft Real-Time Systems:**  
  These OSs are for applications where for time-constraint is less strict.

**Advantages of RTOS:**

* **Maximum Consumption:** Maximum utilization of devices and system,thus more output from all the resources
* **Task Shifting:** Time assigned for shifting tasks in these systems are very less. For example in older systems it takes about 10 micro seconds in shifting one task to another and in latest systems it takes 3 micro seconds.
* **Focus on Application:** Focus on running applications and less importance to applications which are in queue.
* **Real time operating system in embedded system:** Since sizes of programs are small, RTOS can also be used in embedded systems like in transport and others.
* **Error Free:** These types of systems are error free.
* **Memory Allocation:** Memory allocation is best managed in these types of systems.

**Disadvantages of RTOS:**

* **Limited Tasks:** Very few tasks run at the same time and their concentration is very less on few applications to avoid errors.
* **Use heavy system resources:** Sometimes the system resources are not so good and they are expensive as well.
* **Complex Algorithms:** The algorithms are very complex and difficult for the designer to write on.
* **Device driver and interrupt signals:** It needs specific device drivers and interrupt signals to response earliest to interrupts.
* **Thread Priority:** It is not good to set thread priority as these systems are very less prone to switching tasks.

**Examples of Real-Time Operating Systems are:** Scientific experiments, medical imaging systems, industrial control systems, weapon systems, robots, air traffic control systems, etc.

Let’s see the difference between DOS and Windows: