

Unit-1 Introduction of Operating System

❖ Fundamental Goals of Operating system:

The operating system controls and coordinates the user of the hardware among the various application programs for the various users.

It hides complexity of its hardware from user and provide easy interface.

An operating system is a control program. It controls the execution of user programs to prevent errors and improper use of the computer.

The **Primary goal** of operating system is **convenience for the user**. The operating system makes the use of system easier.

The secondary goal of operating system is **efficient operation of the computer system**.

A Computer system can be divided into four components:

1. Hardware
2. Operating System
3. Application Programs
4. Users

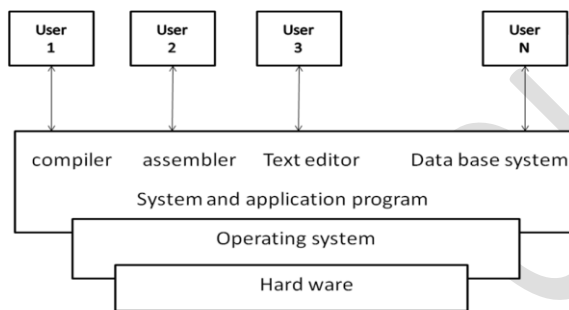


Fig :Component of Computer System

1. **Hardware** :The **hardware** of computer system includes the **Central Processing Unit (CPU)**, the **memory**, and the **input/output (I/O) devices**. They are the basic resources.
2. **Application program** includes compilers, database system, games, and business programs. They define the way in which the hardware resources are to be used to solve the computing problems of the users.
3. **Users** :There are many different users trying to solve different problems using different application programs.
4. **Operating system** :The operating system controls and coordinates the user of the hardware among the various application programs for the various users.

❖ Overview of Operating Systems

❖ Multi programming Operating System

Multiprogramming **increases CPU utilization** by organizing jobs so that the CPU always has one job to execute.

Operating system
job1
job2
job3
job4

The operating system keeps several jobs in memory simultaneously. This set of jobs is a subset of the jobs kept in the **job pool** the number of jobs that can be kept simultaneously in memory is usually much smaller than the number of jobs that can be in the job pool.

Operating system picks and begins to execute one of the jobs in memory and the other job may have to wait for some task, such as I/O operation.

In multiprogramming system, the CPU simply switches to, and executes another job. When that job needs to wait, the CPU is switched to another job.

Eventually the first job finishes waiting and gets CPU back .CPU has always at least one job to execute.

In multiprogramming operating system all the jobs that enter the system are kept in the pool. This pool consists of all processes residing on disk awaiting allocation of main memory.

When the operating system selects a job pool, it loads that job into main memory is called **job scheduling**.

If several jobs are ready to run at the same time, the system must choose among them, this decision is called **CPU scheduling**.

Required feature to do multi programming

To store several programs in memory simultaneously, better memory management & protection are required.

As main memory is limited, virtual memory and swapping are used to accommodate more than one program in memory.

CPU scheduling should be sophisticated, to provide fairness to all programs.

Disk management and file system is complex now.

Synchronization and communication among running programs and problems like dead lock should be overcome.

❖ Time Sharing Operating System

It is a logical extension of multiprogramming operating system.

The ability to execute more than one task at the same time, a task being a program is called **multitasking**.

The terms multitasking and multiprocessing are often used interchangeably although multiprocessing implies that more than one CPU is involved.

In multitasking, only one CPU is involved, but it switches from one program to another so quickly that each user may interact with each program while it is running. So it gives the appearance of executing all of the programs at the same time.

A **time – shared operating system** uses CPU scheduling and multiprogramming to provides each user with a small portion of a time – shared computer. Each user has at least one separate program in memory.

A program in execution is called process.

Time slice is amount of that is given to process to execute on processor (CPU).

A time – Sharing operating system must have memory management, synchronization, communication, protection file system and disk management.

There are two basic types of multitasking:

- 1) Pre emptive multitasking (forcibly)
- 2) Non-preemptive cooperative. (co operative or voluntarily)

In **preemptive** - a program is allowed to execute only for some maximum time duration, after this time duration CPU is forcibly taken away from the program.

- Minimize the response time for user.

Ex: OS/2, Windows 95, Windows NT, the Amiga operating system and UNIX use

In **Non-preemptive** multitasking, each program can control the CPU for as long as it needs it.

- If a program is not using the CPU .However, it can allow another program to use it temporarily.

Ex: Microsoft Windows 3.x and the MultiFinder (for Macintosh computers)

❖ Real Time Operating System

A real-time operating system is used where there are rigid time requirement on the operation of a processor or the flow of data, and thus is often used as a control device in a dedicated application.

RTOS has very little user-interface ,capability and no end user utilities. RTOS always controls inputs and respond within defined time constraint.

Systems that control scientific experiments, medical imaging systems, industrial control systems, and some displays, automobile-engine fuel-injection system are real-time system.

A real-time operating system has well-defined, fixed time constraints processing must be done with the defined constraints, or the system will fail.

- **There are two flavors of real-time systems:**

1. **Hard real-time:** A hard real-time system guarantees that critical tasks complete on time. This goal requires that all delays in the system be bounded , from the retrieval of stored data to the time that takes the operating system to finish any request made of it.

- Secondary storage of any sort is usually limited or missing , with data being stored in short-term memory or in read-only memory(ROM).

- Most advanced operating system features such as virtual memory is not found in real-time system.

Ex: oil refinery (plant),ATM

2. **Soft real-time system:** A less restrictive type of real-time operating system is a soft real-time system, where a critical real-time task gets priority over other tasks , and retains that priority until it completes.

- Soft real-time system have more limited utility that hard real-time system.

- They are risky if used for industrial control and robotics because of their lack of deadline support.

- They are useful in multimedia , virtual reality, and advanced scientific projects such as undersea exploration and planetary rovers.

Ex: digital audio ,multimedia

❖ Multithreading Operating System

- A thread also called a **lightweight process (L W P)** is a basic unit of CPU utilization; it comprises a thread ID, a program counter, a register set, and a stack.

- Thread is a flow of control within a process.

- It shares with other threads belonging to the same process its code section, data section and other operating system resources.

- Process is a **heavy process**, has a single thread of control.

- If the process has multiple threads of control, it can do more than one task at a time.

For eg .A web browser might have one thread display images or text while another thread retrieves data from the network.

A word process may have a thread for displaying graphics, another thread for reading keystroke from users, and a third thread for performing spelling and grammar checking.

Advantage of multithread programming

1. Responsiveness:

Multithreading an interactive application may allow a program to continue running even if part of it is blocked or is performing lengthy operation, thereby increasing responsiveness to the user.

2. Resource sharing

Threads share the memory and resources of the process to which they belong. The Benefit of cost sharing is that it allows an application to have several different threads of activity all within the same address space.

3. Economy

Allocating memory and resources for process creation is costly. Threads share resources of the process to which they belong, it is more economical to create and context switch threads.

4. Utilization of multiprocessor architecture

Multithreading can be greatly increased in multiprocessor architectures, where each thread may be running in parallel on a multiprocessor.

A single-threaded process can only run on one CPU, no matter how many are available. Multithreading on a multi-CPU machine increases concurrency.

❖ Distributed Operating System

A distributed system is a collection of loosely coupled processors interconnected by a communication network.

In distributed operating system **processors do not share memory or a clock**. Instead, each processor has its own local memory and the processors communicate with one another through various communication lines, such as high-speed buses or telephone-lines.

- These systems are usually referred to as **loosely coupled system(or distributed system)**.
- Distributed operating system evolved from computer networks in which a number of largely independent hosts are connected by communication links and protocols.
- A distributed operating system governs the operation of a distributed computer system and provides a virtual machine abstraction to its users. The key objective of distributed operating system is **transparency**.
- A distributed operating system has **less autonomous environment**. Use of such system types the commands and it runs but it is up to OS-level to decide which processor is allocated to him/her for performing task.

- **User of such systems is not aware of the multiplicity of machine**, means he/she are not aware of the fact that multiple computers are being used. It seems like a virtual uniprocessor. Components and resource distribution should be hidden from users and application program unless they explicitly demand.
- Distributed operating system usually **provides the means for system-wide sharing of resources, such as computational capacity, files, and I/O devices.**
- Distributed operating system also may facilitate access to remote resources, Communication with remote processes and distribution of computations along with services necessary for pooling of shared system resources include global naming, distributed file system, and facilitate for distribution of computations, such as internodes process communication and remote procedure calls.

Advantages of distributed systems:

1. Resource sharing:

If a number of different sites are connected to one another, than a user at one site may be able to use the resources available at another.

2. Reliability:

If one site fails in a Distributed system, the remaining sites can continue operating, giving the system better reliability.

3. Communication:

Several sites are connected to one another by a communication network. The user at different sites has the opportunity to exchange information.

4. Computation speeds up.

In distributed system computation (work load) distributes on different machine (,thus it provides load balancing.)Overloaded systems work gets distributed on other system. so computation can be done speedily.

❖ Operating System services

The operating system provides an environment for the execution of programs. The operating system provides services to programs to the users of those programs .The operating system services are provided for the convenience of the programmer.

Operating system provides services to the user program and system.

For user point of view services:

1. User Interface: it provides interface to user so user can perform some action. It provides three types of interfaces,

a) Command line Interface (CLI), in which user can type command and methods for executing them.

- b) **Batch interface**, in which commands and directives to control those commands are entered into files ,and those files are executed.
- c) **Graphical User Interface (GUI)**,in which interface is a window system with pointing device to direct I/O ,chooses from menus, keyboard to enter text.

2. Program execution: The system must be able to load a program into memory and run it. The program must be able to end its execution either normally or abnormally (indicating error).

3. I/O operations: A running program may require I/O involves a file or an I/O device. For specific devices, special functions may be desired. For efficiency protection , users usually cannot control I/O devices directly. The operating system provides a means to do I/O.

4. File-system manipulation: The programs need to read and write files and programs must also create and delete files. The operating system maintains the file system.

5. Communications: In many conditions the process needs to exchange information with another process. Such communications can occur in two ways:

- a) The first takes place between processes that are executing on the same computer.
- b) The second takes place between processes that are executing on different computer systems that are tied together by a network.

Communications may be implemented via **shared memory or via message passing** , in which the information is moved between processes by operating system.

6. Error Detection: The operating system constantly needs to be aware of possible errors. Errors may occur in the CPU and memory hardware, in I/O devices and in user programs.

For each type of errors, the operating system should take the proper action to ensure correct and consistent computing.

For System point of view services

Operating a system function that exists not for helping the user, but rather ensuring the efficient operation of the system.

1. Resource allocation: Operating system manages many resources like CPU cycles, main memory, and file storage. Resources are allocated to multiple users.

Some resources have special allocation code, whereas others may have general request and release code.

For eg. In determining how best to use the CPU, the operating system have CPU- scheduling routines that takes account the speed of the CPU,the jobs that must be executed, the number of registers.

2. Accounting: The tracking of which user use how many and which kind of computer resources can be used for billing or simply accumulating usage statistics.

Usage statistics may be a valuable tool for researchers who wish to reconfigure the system to improve computing services.

3.Protection: Protection involves ensuring that all access to system resources is controlled.

❖ Generations of Operating System

The First Generation (1940 to early 1950s)

When the first electronic computer was developed in 1940, it was created without any operating system. In early times, users have full access to the computer machine and write a program for each task in absolute machine language. The programmer can perform and solve only simple mathematical calculations during the computer generation, and this calculation does not require an operating system.

The Second Generation (1955 - 1965)

The first operating system (OS) was created in the early 1950s and was known as **GMOS. General Motors** has developed OS for the **IBM** computer. The second-generation operating system was based on a single stream batch processing system because it collects all similar jobs in groups or batches and then submits the jobs to the operating system using a punch card to complete all jobs in a machine.

The Third Generation (1965 - 1980)

During the late 1960s, operating system designers were very capable of developing a new operating system that could simultaneously perform multiple tasks in a single computer program called multiprogramming. The introduction of **multiprogramming** plays a very important role in developing operating systems that allow a CPU to be busy every time by performing different tasks on a computer at the same time.

The Fourth Generation (1980 - Present Day)

The fourth generation of operating systems is related to the development of the personal computer. However, the personal computer is very similar to the minicomputers that were developed in the third generation. The cost of a personal computer was very high at that time; there were small fractions of minicomputers costs. Currently, most Windows users use the Windows 10 operating system. Besides the Windows operating system, Apple is another popular operating system built in the 1980s, and this operating system was developed by Steve Jobs, a co-founder of Apple. They named the operating system Macintosh OS or Mac OS.

❖ Case Study

1. Linux

Linux is a family of free and open-source operating systems based on the Linux kernel. Operating systems based on Linux are known as *Linux distributions* or *distros*. Examples include Debian, Ubuntu, Fedora, CentOS, Gentoo, Arch Linux, and many others.

The Linux kernel has been under active development since 1991, and has proven to be extremely versatile and adaptable. You can find computers that run Linux in a wide variety of contexts all over the world, from web servers to cell phones. Today, 90% of all cloud infrastructure and 74% of the world's smartphones are powered by Linux.

However, newcomers to Linux may find it somewhat difficult to approach, as Linux filesystems have a different structure than those found on Windows or MacOS. Additionally, Linux-based operating systems depend heavily on working with the command line interface, while most personal computers rely on graphical interfaces.

This guide serves as an introduction to important

The terms “terminal,” “shell,” and “command line interface” are often used interchangeably, but there are subtle differences between them:

- A *terminal* is an input and output environment that presents a text-only window running a shell.

A *shell* is a program that exposes the computer's operating system to a user or program. In Linux systems, the shell presented in a terminal is a command line interpreter.

A *command line interface* is a user interface (managed by a command line interpreter program) which processes commands to a computer program and outputs the results.

2. Latest Windows Operating System

Windows 11 is the latest major release of Microsoft's Windows NT operating system, released in October 2021. It is a free upgrade to its predecessor, Windows 10 (2015), and is available for any Windows 10 devices that meet the new Windows 11 system requirements.