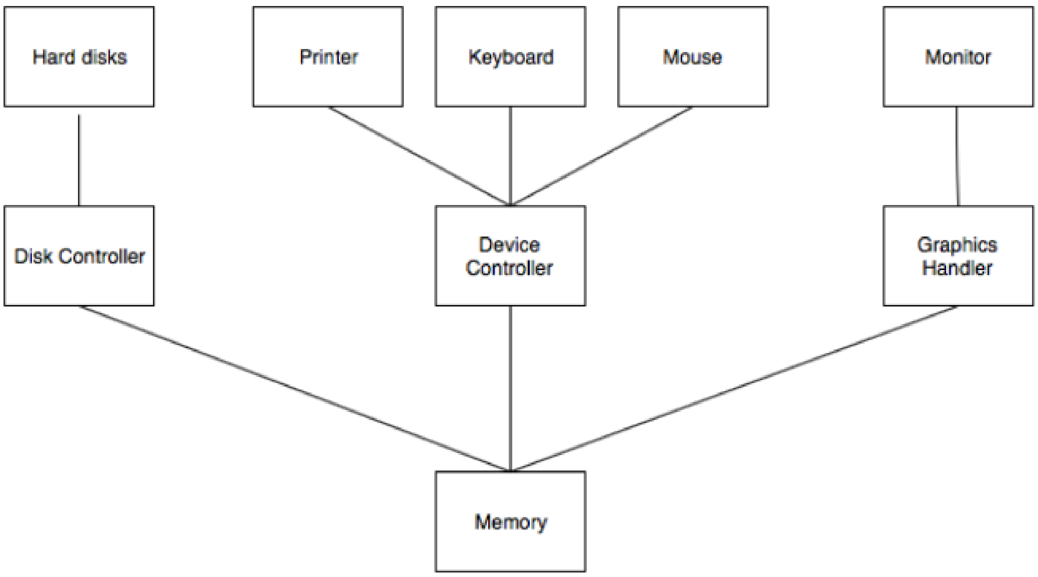
Unit:1 Introduction to Operating System

Computer System Organisation

The computer system is a combination of many parts such as peripheral devices, **secondary memory**, **CPU**, etc. This can be explained more clearly using a diagram.



The salient points about the above figure displaying Computer System Organisation is –

- The I/O devices and the CPU both execute concurrently. Some of the processes are scheduled for the CPU and at the same time, some are undergoing input/output operations.
- There are multiple device controllers, each in charge of a particular device such as keyboard, mouse, printer etc.
- There is buffer available for each of the devices. The input and output data can be stored in these buffers.
- The data is moved from memory to the respective device buffers by the CPU for I/O operations and then this data is moved back from the buffers to memory.
- The device controllers use an interrupt to inform the CPU that I/O operation is completed.

Computer System and its Organisation

The organisation of the computer system consists of different units. All the units have specific components designed to perform some particular task.

The units of the computer system are:

- Input Unit
- Output Unit
- Central Processing Unit (CPU)
- Storage Unit

1. Input unit

All the information, data, or commands are provided to the computer systems as input through input devices. The keyboard, microphone, scanner, or mouse are the input devices that help the user enter the data in distinct forms.

The keyboard allows the user to enter input as characters, numbers, or symbols. Through the mouse, you can select any file or folder. No matter in what form the input is entered, it is changed into binary codes (machine language) of 1s and 0s. This is done as the computer can only understand the machine language. Once the input has been received, then comes the work of the Central Processing Unit (CPU).

2. Output Unit

The output devices provide the result of the command provided to the computers. Monitor, keyboard, printer are some of the output devices connected to the computer systems.

For example, if the user is writing something on the word processor through the keyboard, it gets displayed on the monitor screen. Since the monitor screen provides the result, it is the output unit.

3. Central Processing Unit

After receiving the input through the input unit, the data goes to the CPU, where the complete processing of the data occurs. The entered information and data are stored in the primary memory, RAM (Random Access Memory), by the CPU. For processing, any data CPU takes it from the RAM. Now CPU has its own three main components:

- *Arithmetic Logic Unit*

The function of the Arithmetic Logic Unit, as the name suggests, performs arithmetic and logical operations. Once the input is entered, it is stored in RAM and sent to ALU for processing.

After the arithmetic and logical operations are done, the output/result gets stored in primary memory for a while. Then it is transferred to secondary memory, that is, a storage unit. From there, it is sent to the output unit, which provides the output to the user.

- *Control Unit*

The function of the control unit is to get the information from the RAM (primary memory), decode the data, and then implement them. Through the control unit, the CPU interacts with the input/output device and the system's memory.

The Control unit is one of the main components of the CPU. It is responsible for directing the system's operation by executing the program's instructions.

- *Register Set*

The Register Set usually varies from one system to another. It includes a general-purpose register that saves the output temporarily in the primary memory. The other registers are special-purpose registers. Their work is to execute the programs.

4. Storage Unit

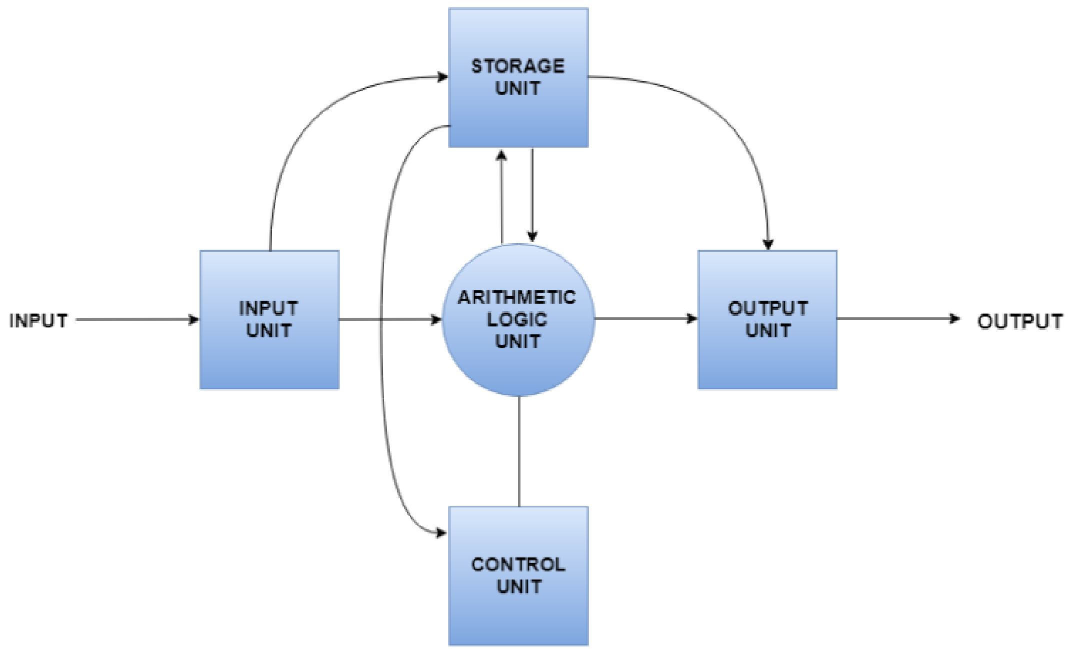
As read above, the data or input that is entered is stored in the primary memory, i.e., RAM, before the processing occurs. Even after the execution, the output is stored temporarily in the primary memory and then in the second memory; it is sent to the output unit to provide the results.

The storage unit is used to store all kinds of data at every processing step. The devices such as hard disks allow the user to store data in different forms. A hard disk is a storage unit that provides the computer with sufficient space to store data and programs.

Computer System Architecture

A computer system is basically a machine that simplifies complicated tasks. It should maximize performance and reduce costs as well as power consumption. The different components in the Computer System Architecture are Input Unit, Output Unit, Storage Unit, **Arithmetic Logic Unit**, **Control Unit** etc.

A diagram that shows the flow of data between these units is as follows −



The input data travels from input unit to ALU. Similarly, the computed data travels from ALU to output unit. The data constantly moves from storage unit to ALU and back again. This is because stored data is computed on before being stored again. The control unit controls all the other units as well as their data.

Details about all the computer units are −

- **Input Unit**
  The input unit provides data to the computer system from the outside. So, basically it links the external environment with the computer. It takes data from the input devices, converts it into machine language and then loads it into the computer system. Keyboard, mouse etc. are the most commonly used input devices.

- **Output Unit**

  The output unit provides the results of computer process to the users i.e it links the computer with the external environment. Most of the output data is the form of audio or video. The different output devices are monitors, printers, speakers, headphones etc.

- **Storage Unit**

  Storage unit contains many computer components that are used to store data. It is traditionally divided into **primary storage and secondary storage**. Primary storage is also known as the main memory and is the memory directly accessible by the **CPU**. Secondary or external storage is not directly accessible by the CPU. The data from secondary storage needs to be brought into the primary storage before the CPU can use it. Secondary storage contains a large amount of data permanently.

- **Arithmetic Logic Unit**

  All the calculations related to the computer system are performed by the arithmetic logic unit. It can perform operations like addition, subtraction, multiplication, division etc. The control unit transfers data from storage unit to arithmetic logic unit when calculations need to be performed. The arithmetic logic unit and the control unit together form the central processing unit.

- **Control Unit**

  This unit controls all the other units of the computer system and so is known as its central nervous system. It transfers data throughout the computer as required including from storage unit to central processing unit and vice versa. The control unit also dictates how the memory, input output devices, arithmetic logic unit etc. should behave.

Operating System Structure

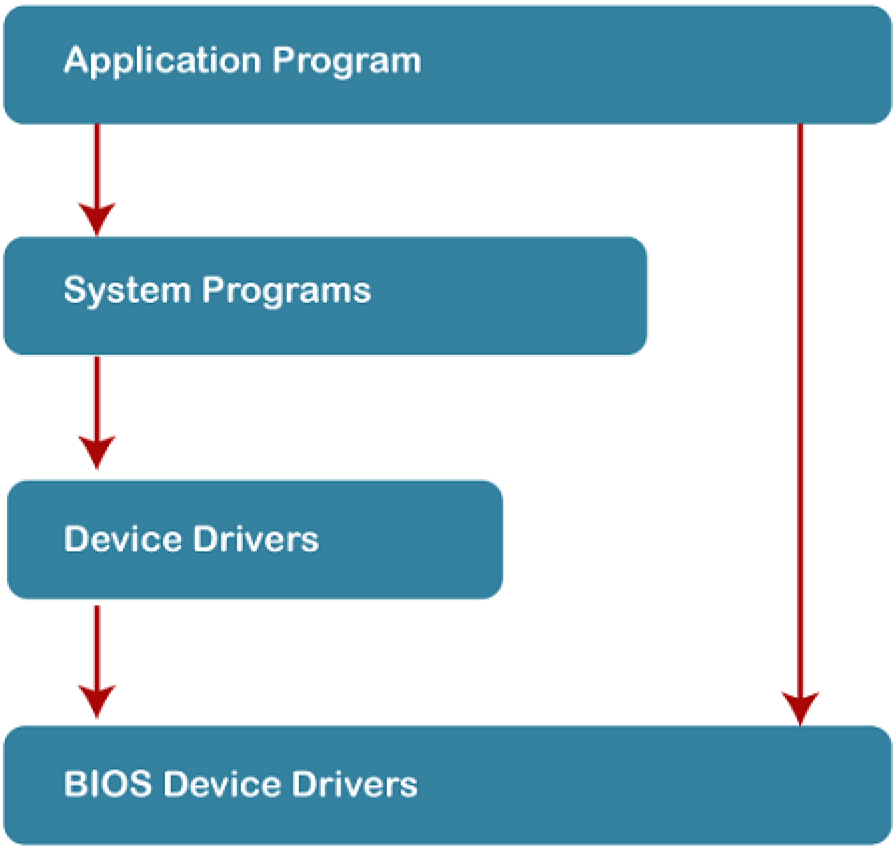
An operating system is a design that enables user application programs to communicate with the hardware of the machine. The operating system should be built with the utmost care because it is such a complicated structure and should be simple to use and modify. Partially developing the operating system is a simple approach to accomplish this. Each of these components needs to have distinct inputs, outputs, and functionalities.

It is the most straightforward operating system structure, but it lacks definition and is only appropriate for usage with tiny and restricted systems. Since the interfaces and degrees of functionality in this structure are clearly defined, programs are able to access I/O routines, which may result in unauthorized access to I/O procedures.

This organizational structure is used by the MS-DOS operating system:

- There are four layers that make up the MS-DOS operating system, and each has its own set of features.
- These layers include ROM BIOS device drivers, MS-DOS device drivers, application programs, and system programs.
- The MS-DOS operating system benefits from layering because each level can be defined independently and, when necessary, can interact with one another.
- If the system is built in layers, it will be simpler to design, manage, and update. Because of this, simple structures can be used to build constrained systems that are less complex.
- When a user program fails, the operating system as whole crashes.
- Because MS-DOS systems have a low level of abstraction, programs and I/O procedures are visible to end users, giving them the potential for unwanted access.

The following figure illustrates layering in simple structure:

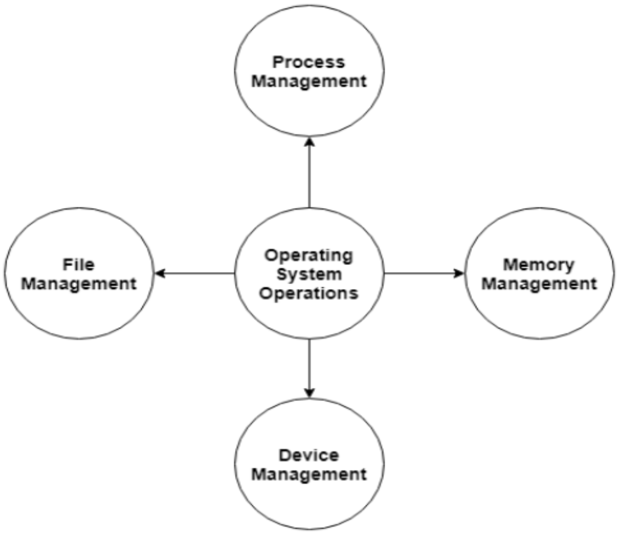


Application Program
System Programs
Device Drivers
BIOS Device Drivers

Operating System Operations

An **operating system** is a construct that allows the user application programs to interact with the system hardware. Operating system by itself does not provide any function but it provides an atmosphere in which different applications and programs can do useful work.

The major operations of the operating system are process management, memory management, device management and file management. These are given in detail as follows:



Process Management

The operating system is responsible for managing the processes i.e assigning the processor to a process at a time. This is known as process scheduling. The different algorithms used for process scheduling are FCFS (first come first served), SJF (shortest job first), priority scheduling, round robin scheduling etc.

There are many scheduling queues that are used to handle processes in process management. When the processes enter the system, they are put into the job queue. The processes that are ready to execute in the main memory are kept in the ready queue. The processes that are waiting for the I/O device are kept in the device queue.

Memory Management

Memory management plays an important part in operating system. It deals with memory and the moving of processes from disk to primary memory for execution and back again.

The activities performed by the operating system for memory management are –

- The operating system assigns memory to the processes as required. This can be done using best fit, first fit and worst fit algorithms.
- All the memory is tracked by the operating system i.e. it nodes what memory parts are in use by the processes and which are empty.
- The operating system deallocated memory from processes as required. This may happen when a process has been terminated or if it no longer needs the memory.

Device Management

There are many I/O devices handled by the operating system such as mouse, keyboard, disk drive etc. There are different device drivers that can be connected to the operating system to handle a specific device. The device controller is an interface between the device and the device driver. The user applications can access all the I/O devices using the device drivers, which are device specific codes.

File Management

Files are used to provide a uniform view of data storage by the operating system. All the files are mapped onto physical devices that are usually non-volatile so data is safe in the case of system failure.

The files can be accessed by the system in two ways i.e. sequential access and direct access –

- **Sequential Access**
  The information in a file is processed in order using sequential access. The files records are accessed on after another. Most of the file systems such as editors, compilers etc. use sequential access.
- **Direct Access**
  In direct access or relative access, the files can be accessed in random for read and write operations. The direct access model is based on the disk model of a file, since it allows random accesses.

Process Management in OS

A Program does nothing unless its instructions are executed by a CPU. A program in execution is called a process. In order to accomplish its task, process needs the computer resources.

There may exist more than one process in the system which may require the same resource at the same time. Therefore, the operating system has to manage all the processes and the resources in a convenient and efficient way.

Some resources may need to be executed by one process at one time to maintain the consistency otherwise the system can become inconsistent and deadlock may occur.

The operating system is responsible for the following activities in connection with Process Management

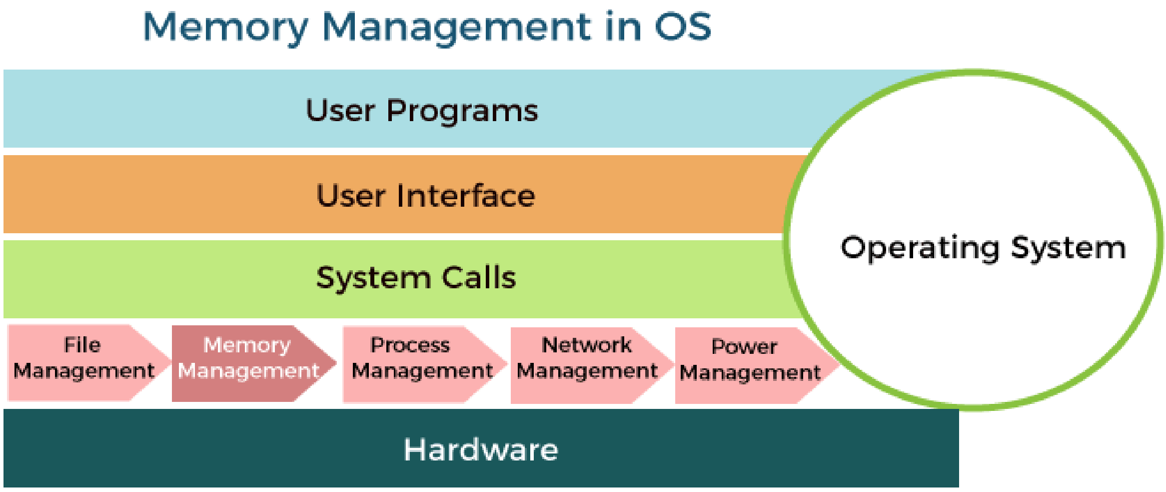
1. Scheduling processes and threads on the CPUs.
2. Creating and deleting both user and system processes.
3. Suspending and resuming processes.
4. Providing mechanisms for process synchronization.
5. Providing mechanisms for process communication.

Memory Management in Operating System (OS)

In this article, we will understand memory management in detail.

What do you mean by memory management?

Memory is the important part of the computer that is used to store the data. Its management is critical to the computer system because the amount of main memory available in a computer system is very limited. At any time, many processes are competing for it. Moreover, to increase performance, several processes are executed simultaneously. For this, we must keep several processes in the main memory, so it is even more important to manage them effectively.



Role of Memory management

Following are the important roles of memory management in a computer system:

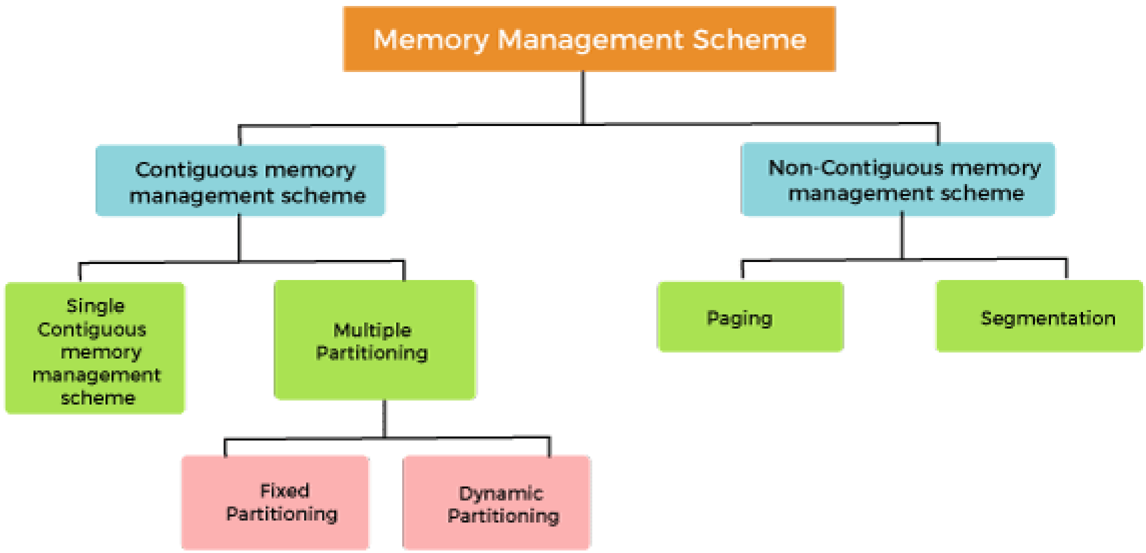
- Memory manager is used to keep track of the status of memory locations, whether it is free or allocated. It addresses primary memory by providing abstractions so that software perceives a large memory is allocated to it.
- Memory manager permits computers with a small amount of main memory to execute programs larger than the size or amount of available memory. It does this by moving information back and forth between primary memory and secondary memory by using the concept of swapping.
- The memory manager is responsible for protecting the memory allocated to each process from being corrupted by another process. If this is not ensured, then the system may exhibit unpredictable behavior.

- Memory managers should enable sharing of memory space between processes. Thus, two programs can reside at the same memory location although at different times.

Memory Management Techniques:

The memory management techniques can be classified into following main categories:

- Contiguous memory management schemes
- Non-Contiguous memory management schemes



Classification of memory management schemes

Contiguous memory management schemes:

In a Contiguous memory management scheme, each program occupies a single contiguous block of storage locations, i.e., a set of memory locations with consecutive addresses.

Single contiguous memory management schemes:

The Single contiguous memory management scheme is the simplest memory management scheme used in the earliest generation of computer systems. In this scheme, the main memory is divided into two contiguous areas or partitions. The operating systems reside permanently in one partition, generally at the lower memory, and the user process is loaded into the other partition.

Advantages of Single contiguous memory management schemes:

- Simple to implement.
- Easy to manage and design.
- In a Single contiguous memory management scheme, once a process is loaded, it is given full processor's time, and no other processor will interrupt it.

Disadvantages of Single contiguous memory management schemes:

- Wastage of memory space due to unused memory as the process is unlikely to use all the available memory space.
- The CPU remains idle, waiting for the disk to load the binary image into the main memory.
- It can not be executed if the program is too large to fit the entire available main memory space.
- It does not support multiprogramming, i.e., it cannot handle multiple programs simultaneously.

Protection and Security in Operating System

Protection and security requires that computer resources such as **CPU**, **softwares**, **memory** etc. are protected. This extends to the operating system as well as the data in the system. This can be done by ensuring integrity, confidentiality and availability in the operating system. The system must be protect against unauthorized access, viruses, worms etc.

Protection and Security Methods

The different methods that may provide protect and security for different computer systems are −

Authentication: This deals with identifying each user in the system and making sure they are who they claim to be. The operating system makes sure that all the users are authenticated before they access the system. The different ways to make sure that the users are authentic are:

- **Username/ Password**
  Each user has a distinct username and password combination and they need to enter it correctly before they can access the system.
- **User Key/ User Card**
  The users need to punch a card into the card slot or use they individual key on a keypad to access the system.
- **User Attribute Identification**
  Different user attribute identifications that can be used are fingerprint, eye retina etc. These are unique for each user and are compared with the existing samples in the database. The user can only access the system if there is a match.

One Time Password: These passwords provide a lot of security for authentication purposes. A onetime password can be generated exclusively for a login every time a user wants to enter the system. It cannot be used more than once. The various ways a onetime password can be implemented are −

- **Random Numbers**
  The system can ask for numbers that correspond to alphabets that are pre arranged. This combination can be changed each time a login is required.
- **Secret Key**
  A hardware device can create a secret key related to the user id for login. This key can change each time

Special Purpose Computer Systems

There are various classes of computer systems based on their computational speed, usage, and hardware. The following are some special-purpose systems according to specific applications. They use:

Real-time embedded systems Multimedia systems

Multimedia systems

Handheld and portable systems

These are explained as following below.

Real-time embedded systems Multimedia systems:

Embedded systems are small computers having a limited set of hardware like a small processor capable of processing a limited set of instructions (often called as an Application Specific Integrated Circuit (ASIC) a small memory (RAM or EPROM) and I/O devices. These systems usually do specific tasks.

Examples are microwave ovens, robots in a manufacturing unit, latest automotive engines, etc.

A variety of embedded systems exists of which some are computers with standard operating systems, some have dedicated programs embedded in their limited memories and often some don't even have any software, hardware (ASIC) to do processing. Nearly all embedded systems use real-time operating system because they are used as control devices and have rigid time requirements. Sensors are used to input data such as temperature, air pressure, etc, from the environment to embedded system where that data is analyzed and several other controls are adjusted by embedded system itself to control the situation of system.

A few examples are home appliance controllers, weapon controllers, boiler temperature controllers, fuel injection systems in automobile engines, etc. A real-time system needs that processing to be done in fixed time constraints.

Multimedia systems:

Multimedia refers to data of multiple types that includes audio and video including conventional data like text files, word-processing documents, spreadsheets, etc. It requires that audio and video data must be processed based upon certain time restrictions. This is called streaming. It is usually 30 frames per second for a video file.

Applications such as video conferencing, movies and clips downloaded over the internet, mp3, DVD, VCD playing, and recording are examples of various multimedia applications. A multimedia application is usually a combination of both audio and video. Multimedia is not limited to desktop operating systems or computers but it is also becoming popular in handheld.

Handheld and portable systems:

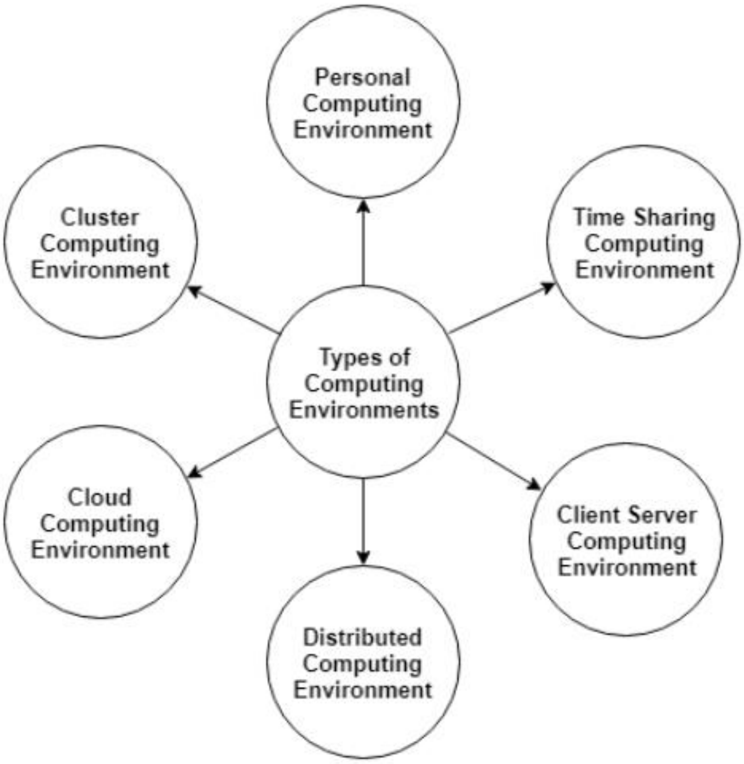
Hand-held systems refers to small portable devices that can be carried along and are capable of performing normal operations. They are usually battery-powered. Examples include Personal Digital Assistants (PDAs), mobile phones, palm-top computers, pocket PCs etc. As they are handheld devices, their weights and sizes have certain limitations as a result they are equipped with small memories, slow processors and small display screens, etc.

The physical memory capacity is very less (512 KB to 128 MB) hence the operating systems of these devices must manage the memory efficiently. As the processors are slower due to battery problems, the operating system should not burden.

Types of Computing Environments

A computer system uses many devices, arranged in different ways to solve many problems. This constitutes a computing environment where many computers are used to process and exchange information to handle multiple issues.

The different types of Computing Environments are −



Let us begin with Personal Computing Environment −

Personal Computing Environment

In the personal computing environment, there is a single computer system. All the system processes are available on the computer and executed there. The different devices that constitute a personal computing environment are laptops, mobiles, printers, computer systems, scanners etc.

Time Sharing Computing Environment

The time sharing computing environment allows multiple users to share the system simultaneously. Each user is provided a time slice and the processor switches rapidly among the users according to it. Because of this, each user believes that they are the only ones using the system.

Client Server Computing Environment

In client server computing, the client requests a resource and the server provides that resource. A server may serve multiple clients at the same time while a client is in contact with only one server. Both the client and server usually communicate via a computer network but sometimes they may reside in the same system.

Distributed Computing Environment

A distributed computing environment contains multiple nodes that are physically separate but linked together using the network. All the nodes in this system communicate with each other and handle processes in tandem. Each of these nodes contains a small part of the distributed operating system software.

Cloud Computing Environment

The computing is moved away from individual computer systems to a cloud of computers in cloud computing environment. The cloud users only see the service being provided and not the internal details of how the service is provided. This is done by pooling all the computer resources and then managing them using a software.

Cluster Computing Environment

The clustered computing environment is similar to parallel computing environment as they both have multiple CPUs. However a major difference is that clustered systems are created by two or more individual computer systems merged together which then work parallel to each other.

Operating System - Services

An Operating System provides services to both the users and to the programs.

- It provides programs an environment to execute.
- It provides users the services to execute the programs in a convenient manner.

Following are a few common services provided by an operating system −

- Program execution
- I/O operations
- File System manipulation
- Communication
- Error Detection
- Resource Allocation
- Protection

Program execution

Operating systems handle many kinds of activities from user programs to system programs like printer spooler, name servers, file server, etc. Each of these activities is encapsulated as a process.

A process includes the complete execution context (code to execute, data to manipulate, registers, OS resources in use). Following are the major activities of an operating system with respect to program management −

- Loads a program into memory.
- Executes the program.
- Handles program's execution.
- Provides a mechanism for process synchronization.
- Provides a mechanism for process communication.
- Provides a mechanism for deadlock handling.

I/O Operation

An I/O subsystem comprises of I/O devices and their corresponding driver software. Drivers hide the peculiarities of specific hardware devices from the users.

An Operating System manages the communication between user and device drivers.

- I/O operation means read or write operation with any file or any specific I/O device.
- Operating system provides the access to the required I/O device when required.

File system manipulation

A file represents a collection of related information. Computers can store files on the disk (secondary storage), for long-term storage purpose. Examples of storage media include magnetic tape, magnetic disk and optical disk drives like CD, DVD. Each of these media has its own properties like speed, capacity, data transfer rate and data access methods.

A file system is normally organized into directories for easy navigation and usage. These directories may contain files and other directions. Following are the major activities of an operating system with respect to file management −

- Program needs to read a file or write a file.
- The operating system gives the permission to the program for operation on file.
- Permission varies from read-only, read-write, denied and so on.
- Operating System provides an interface to the user to create/delete files.
- Operating System provides an interface to the user to create/delete directories.
- Operating System provides an interface to create the backup of file system.

Communication

In case of distributed systems which are a collection of processors that do not share memory, peripheral devices, or a clock, the operating system manages communications between all the processes. Multiple processes communicate with one another through communication lines in the network.

The OS handles routing and connection strategies, and the problems of contention and security. Following are the major activities of an operating system with respect to communication −

- Two processes often require data to be transferred between them
- Both the processes can be on one computer or on different computers, but are connected through a computer network.
- Communication may be implemented by two methods, either by Shared Memory or by Message Passing.

Error handling

Errors can occur anytime and anywhere. An error may occur in CPU, in I/O devices or in the memory hardware. Following are the major activities of an operating system with respect to error handling −

- The OS constantly checks for possible errors.
- The OS takes an appropriate action to ensure correct and consistent computing.

Resource Management

In case of multi-user or multi-tasking environment, resources such as main memory, CPU cycles and files storage are to be allocated to each user or job. Following are the major activities of an operating system with respect to resource management −

- The OS manages all kinds of resources using schedulers.
- CPU scheduling algorithms are used for better utilization of CPU.

Protection

Considering a computer system having multiple users and concurrent execution of multiple processes, the various processes must be protected from each other's activities.

Protection refers to a mechanism or a way to control the access of programs, processes, or users to the resources defined by a computer system. Following are the major activities of an operating system with respect to protection −

- The OS ensures that all access to system resources is controlled.
- The OS ensures that external I/O devices are protected from invalid access attempts.
- The OS provides authentication features for each user by means of passwords.

Print Page

What is the user interface?

A user and a computer can communicate in both directions. One of the operating system functions provides a ‘user interface,’ which allows a human to communicate with the computer’s hardware. When you acquire software, it will come with a user interface design that will allow you to access and use it. A user will provide data and instructions to a computer, and the machine will respond with information. The interface is how a computer and a user communicate. This can be described in a variety of ways. The Human-Computer Interface, or HCI, is another often-used phrase. If you describe the interface thoroughly, you must discuss the input devices, the software interface, and the output devices. We’ll focus on the nature of the software interface in this section. We’ll go through the many sorts of software interfaces that could be found in an operating system, as well as their characteristics.

Types of User interface

- Menu-based interfaces
- Command-Line Interfaces
- Form-based interfaces
- Graphical User Interfaces
- Natural language interfaces

So, let us elaborate on the user interface mentioned above.

- **Menu-based interface:** We use mobile phones and other screen-based devices every day. These gadgets make life easier for us. Menus are available on these devices to assist us in completing any task. On our mobile phones, we can check the weather, make calls, read emails, and so on. The smartphone screen leads us through the process by providing numerous menus. A screen can be defined as an interface, and a screen having multiple menus to conduct tasks is referred to as a menu console.
- **Command-Line Interfaces:** A command-line interface (CLI) is a computer interface in which text is used as the primary form of input and output. The CLI is a linear, keyboard-controlled interface. A command-line interface (CLI) is an alternative to a graphical user interface (GUI), which uses a mouse or pointing device to control input and output. Almost all operating systems still provide a command-line interface to communicate with the system, even if the GUI has superseded the CLI as the primary mode of interaction. Because CLI is easier to script/automate, it is available as an optional feature. Furthermore, the CLI has access to more advanced features that may not be available in GUIs aimed at beginner users.

- **Form-based interfaces:** Some operating systems are built for enterprises where employees must enter a large amount of data. Consider a paper-based form that you are requested to fill out, such as a club membership application or a driver's licence application. What you need to write down is specific. Instructions are provided and places where you may write or select information from a variety of options and boxes where you can just tick one of the options. A computer's form-based software interface is akin to a paper-based 'interface.' The data entered into the computer is predictable.
- **Graphic User Interface**: The graphical user interface, created by the Xerox Palo Alto research lab in the late 1970s and commercially available in Apple's Macintosh and Microsoft's Windows operating systems, was created in reaction to the inefficiency of early, text-based command-line interfaces for the typical user. The ability to rationally operate computers and other electronic devices by directly manipulating graphical icons such as buttons, scroll bars, windows, tabs, menus, cursors, and the mouse pointing device would become the standard of user-centred design in software application programming. Touchscreen and voice-command interaction features are common in modern graphical user interfaces.
- **Natural Language Interface:** Natural Language Interface may convert natural language questions into Boolean queries and augment them with possible combining and paraphrasing options. NLI can also add synonyms to the original queries. Still, because the total number of synonyms is limited and a list of synonyms suitable to a specific domain must be defined, this feature isn't available out of the box. Still, it can be added following a customization request.

System Calls in Operating System (OS)

A system call is a way for a user program to interface with the operating system. The program requests several services, and the OS responds by invoking a series of system calls to satisfy the request. A system call can be written in assembly language or a high-level language like **C** or **Pascal**. System calls are predefined functions that the operating system may directly invoke if a high-level language is used.

In this article, you will learn about the system calls in the operating system and discuss their types and many other things.

What is a System Call?

A system call is a method for a computer program to request a service from the kernel of the operating system on which it is running. A system call is a method of interacting with the operating system via programs. A system call is a request from computer software to an operating system's kernel.

The **Application Program Interface (API)** connects the operating system's functions to user programs. It acts as a link between the operating system and a process, allowing user-level programs to request operating system services. The kernel system can only be accessed using system calls. System calls are required for any programs that use resources.

How System Calls Work

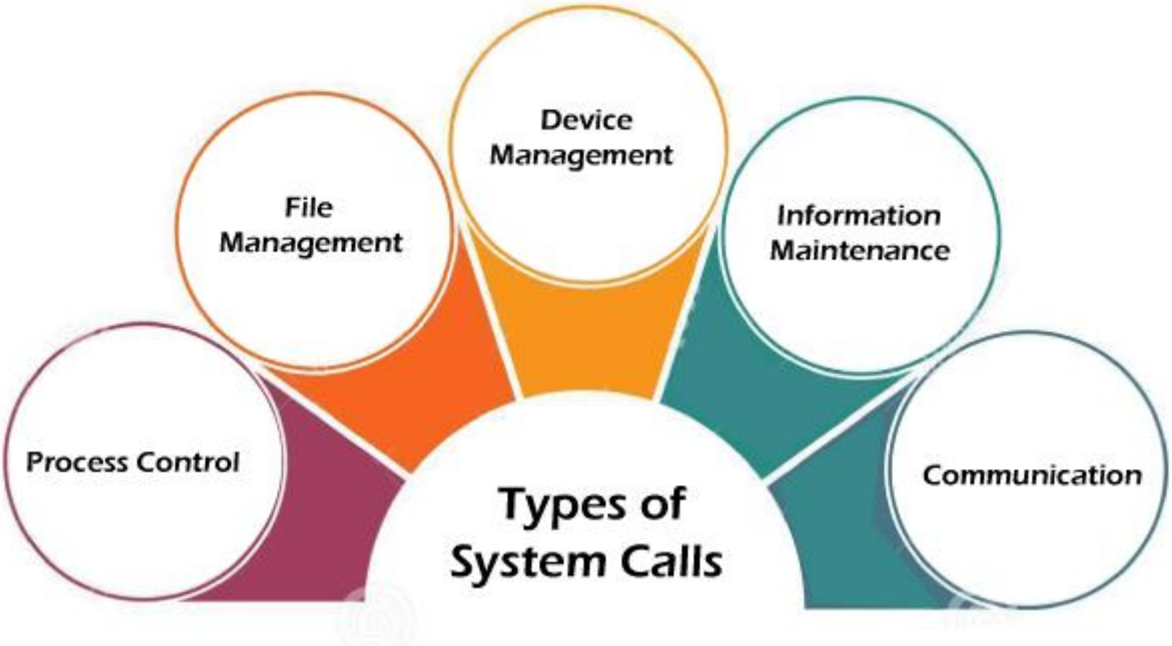
The Applications run in an area of memory known as user space. A system call connects to the operating system's kernel, which executes in kernel space. When an application creates a system call, it must first obtain permission from the kernel. It achieves this using an interrupt request, which pauses the current process and transfers control to the kernel.

If the request is permitted, the kernel performs the requested action, like creating or deleting a file. As input, the application receives the kernel's output. The application resumes the procedure after the input is received. When the operation is finished, the kernel returns the results to the application and then moves data from kernel space to user space in memory.

A simple system call may take few nanoseconds to provide the result, like retrieving the system date and time. A more complicated system call, such as connecting to a network device, may take a few seconds. Most operating systems launch a distinct kernel thread for each system call to avoid bottlenecks. Modern operating systems are multi-threaded, which means they can handle various system calls at the same time.

Types of System Calls

There are commonly five types of system calls. These are as follows:



1. **Process Control**
2. **File Management**
3. **Device Management**
4. **Information Maintenance**
5. **Communication**

Now, you will learn about all the different types of system calls one-by-one.

Process Control

Process control is the system call that is used to direct the processes. Some process control examples include creating, load, abort, end, execute, process, terminate the process, etc.

File Management

File management is a system call that is used to handle the files. Some file management examples include creating files, delete files, open, close, read, write, etc.

Device Management

Device management is a system call that is used to deal with devices. Some examples of device management include read, device, write, get device attributes, release device, etc.

Information Maintenance

Information maintenance is a system call that is used to maintain information. There are some examples of information maintenance, including getting system data, set time or date, get time or date, set system data, etc.

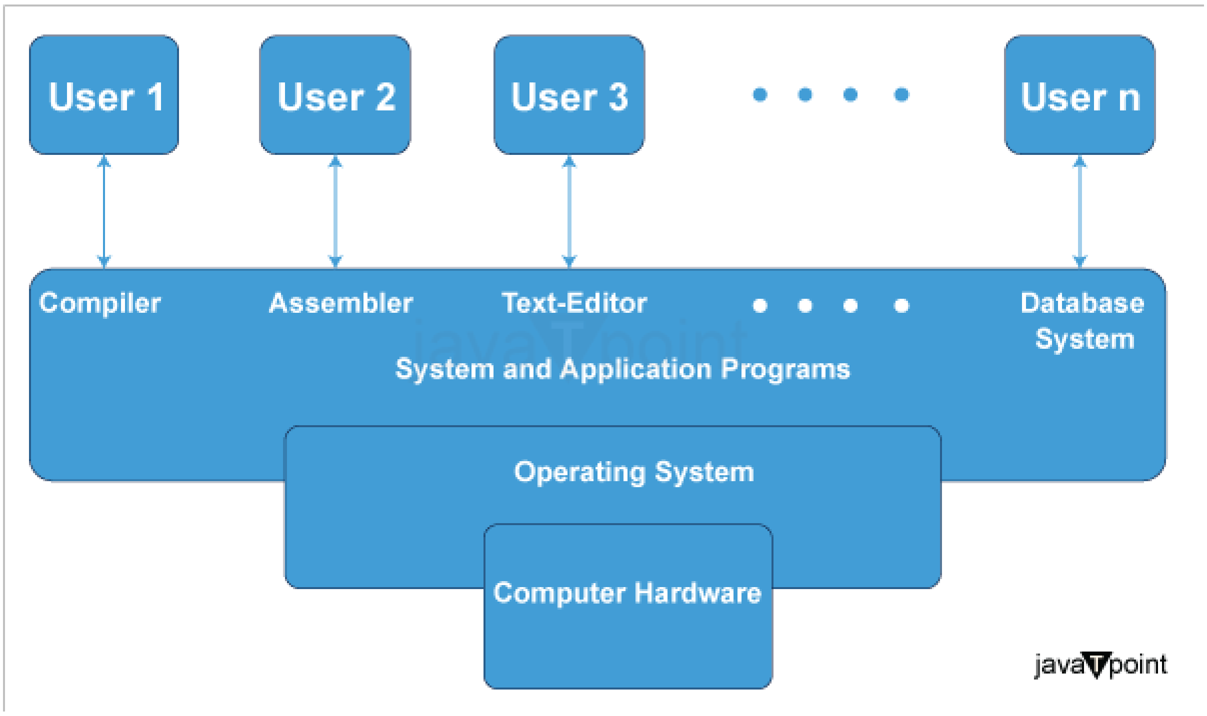
Communication

Communication is a system call that is used for communication. There are some examples of communication, including create, delete communication connections, send, receive messages, etc.

What is a System program in an Operating System?

In this article, we will discuss the system program in operating systems with its types and functions.

The performance of the entire ***computer's application*** software is controlled by the ***system program***, which is one of several sorts of system programs that users of operating systems can use.



System calls define several sorts of system programs for various activities, and system calls can be used to access these programs. System programs are responsible for the ***creation*** and ***execution*** of a program.

Functions of the System Program

There are several functions of the system program in the operating system. These are as follows:

- **File management:**

The ***files*** and ***directory*** are generally manipulated by these programs, which also ***create, delete, copy, rename, print***, and ***exit***.

1. In the computer system, adding new files and putting them in certain places can be helpful.
2. It aids in ***swiftly*** and ***simply*** locating these files within the computer system.
3. It effectively simplifies and streamlines the process of exchanging files among users.
4. Files should be kept in ***distinct directories***, which are ***folders***.
5. Users can manage files according to their intended usage or conduct rapid searches of files using these directories.
6. The ability to change a file's data or a directory's file names is helpful to users.

- **Status information:** ***Status information*** is the knowledge of the ***input, output, storage***, and CPU usage processes and how the process will compute how much memory is needed to complete a task.

- **A programming language supports:** ***Compiler, Assembler***, and ***Interrupt*** are examples of programming language tools used in the operating system of a computer for certain functions.

- **Programming Loading and Execution:** It performs ***system calls*** with the aid of system programs to do two tasks: ***enter the program*** and ***execute the output*** of the program when the program has loaded.

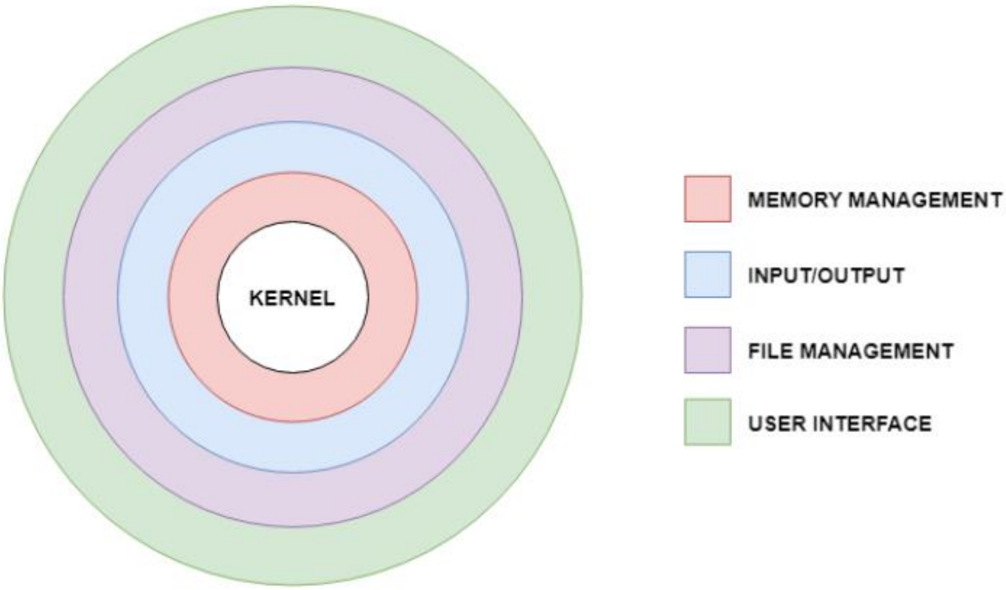
- **Communication:** The user offers these services because the operating system requires that a large number of devices be able to communicate with one another via ***wireless*** or other devices.

- **Background services:** In the ***operating system***, there are several services that may be used for communication, and a ***background service*** can be used to change the ***backdrop*** of your window and search for and identify computer viruses.

Operating System Design and Implementation

An operating system is a construct that allows the user application programs to interact with the system hardware. Operating system by itself does not provide any function but it provides an atmosphere in which different applications and programs can do useful work.

There are many problems that can occur while designing and implementing an operating system. These are covered in operating system design and implementation.



Layered Operating System Design

Operating System Design Goals

It is quite complicated to define all the goals and specifications of the operating system while designing it.The design changes depending on the type of the operating system i.e if it is batch system, time shared system, single user system, multi user system, distributed system etc.

There are basically two types of goals while designing an operating system. These are −

User Goals

The operating system should be convenient, easy to use, reliable, safe and fast according to the users. However, these specifications are not very useful as there is no set method to achieve these goals.

System Goals

The operating system should be easy to design, implement and maintain. These are specifications required by those who create, maintain and operate the operating system. But there is not specific method to achieve these goals as well.

Operating System Implementation

The operating system needs to be implemented after it is designed. Earlier they were written in assembly language but now higher level languages are used. The first system not written in assembly language was the Master Control Program (MCP) for Burroughs Computers.

Advantages of Higher Level Language

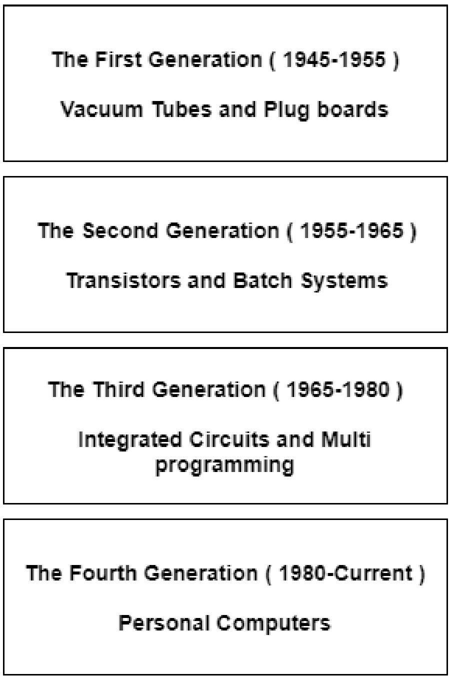
There are multiple advantages to implementing an operating system using a higher level language such as: the code is written more fast, it is compact and also easier to debug and understand. Also, the operating system can be easily moved from one hardware to another if it is written in a high level language.

Disadvantages of Higher Level Language

Using high level language for implementing an operating system leads to a loss in speed and increase in storage requirements. However in modern systems only a small amount of code is needed for high performance, such as the CPU scheduler and memory manager. Also, the bottleneck routines in the system can be replaced by assembly language equivalents if required.

Operating System Generations

Operating Systems have evolved over the years. So, their evolution through the years can be mapped using generations of operating systems. There are four generations of operating systems. These can be described as follows –



OPERATING SYSTEM GENERATIONS

The First Generation (1945 - 1955): Vacuum Tubes and Plugboards

Digital computers were not constructed until the second world war. Calculating engines with mechanical relays were built at that time. However, the mechanical relays were very slow and were later replaced with vacuum tubes. These machines were enormous but were still very slow.

These early computers were designed, built and maintained by a single group of people. Programming languages were unknown and there were no operating systems so all the programming was done in machine language. All the problems were simple numerical calculations.

By the 1950's punch cards were introduced and this improved the computer system. Instead of using plugboards, programs were written on cards and read into the system.

The Second Generation (1955 - 1965): Transistors and Batch Systems

Transistors led to the development of the computer systems that could be manufactured and sold to paying customers. These machines were known as mainframes and were locked in air-conditioned computer rooms with staff to operate them.

The **Batch System** was introduced to reduce the wasted time in the computer. A tray full of jobs was collected in the input room and read into the **magnetic tape**. After that, the tape was rewound and mounted on a tape drive. Then the batch operating system was loaded in which read the first job from the tape and ran it. The output was written on the second tape. After the whole batch was done, the input and output tapes were removed and the output tape was printed.

The Third Generation (1965 - 1980): Integrated Circuits and Multiprogramming

Until the 1960's, there were two types of computer systems i.e., the scientific and the commercial computers. These were combined by IBM in the System/360. This used integrated circuits and provided a major price and performance advantage over the second-generation systems.

The third-generation operating systems also introduced **multiprogramming**. This meant that the processor was not idle while a job was completing its I/O operation. Another job was scheduled on the processor so that its time would not be wasted.

The Fourth Generation (1980 - Present): Personal Computers

Personal Computers were easy to create with the development of large-scale integrated circuits. These were chips containing thousands of transistors on a square centimetre of silicon. Because of these, microcomputers were much cheaper than minicomputers and that made it possible for a single individual to own one of them.

The advent of personal computers also led to the growth of networks. This created network operating systems and distributed operating systems. The users were aware of a network while using a network operating system and could log in to remote machines and copy files from one machine to another.

System Boot : Case Study (Assignment)