

Last Edit: 24-Jul-09

Further
Inside**Introduction to Operating System :**

An Operating System is defined as a set of system programs that control and coordinate the operation of a computer system

At This
Level

An *operating system* is a program that acts as an intermediary between a user of a computer and the computer hardware. The purpose of an operating system is to provide an environment in which a user can execute programs. The primary goal of an operating system is thus to make the computer system convenient to use. A secondary goal is to use the computer hardware in an *efficient* manner.

An operating system is an important part of almost every computer system. A computer system can be divided roughly into 4 components : the hardware, the operating system, the applications programs, and the users.

The hardware - the central processing unit (CPU), the memory, and the input/output(I/O) devices - provides the basic computing resources.

The application program - such as compilers, database systems, games, and business programs - define the ways in which these resources are used to solve the computing problems of the users.

The operating systems controls and coordinates the use of hardware among the various application programs for the various uses.

We can view an operating system as a *resource allocator*. A computer system has many resources (h/w and s/w) that may be required to solve a problem: CPU time, memory space, file storage space, I/O devices and so on. The operating system acts as the manager of these resources and allocates them to specific programs and users as necessary for tasks. Since there may be many requests of resources , the operating systems decide which requests are allocated resources to operate the computer system efficiently and fairly.

It is easier to define operating systems by what they *do* than by what they *are*. The primary goal of an operating system is *convenience for the user*. Operating systems exist because they are supposed to make it easier to compute with them than without them.

Every operating system has a *kernel* or a *nucleus* which permanently resides in the main memory of a computer to perform some of the basic functions of OS and to access other portions as and when they are needed. The remaining parts of an OS are normally stored in a disk ready to be loaded into the main memory when required and ordered to do so by the kernel.

Following are the services which are provided for the convenience of the programmer , to make the programming task easier.

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

2) I/O operations : A running program may require I/O. This I/O may involve a file or an I/O device. For specific devices , special functions may be desired. For efficiency and protection , users usually cannot control I/O devices directly. Therefore the operating system must provide some means to do I/O.

3) File-system manipulation : The file system is of particular interest. It should be obvious that program need to read and write files. They also need to create and delete files by name.

4) Communications : There are many circumstances in which one process needs to exchange informations with another process. There are 2 major ways in which such communications can occur. The first takes place between processes executing on the same computer; the second takes place between processes executing on different computer systems that are tied together by a computer network. Communications may be implemented via *shared memory*, or by the technique of *message passing*, in which packets of information are moved between processes by the operating system.

5) Error detection : The OS constantly needs to be aware of possible errors. Errors may occur in the CPU and memory hardware ,in I/O devices or in the user program. For each type of error ,the operating system should take the appropriate action to ensure correct and consistent computing.

6) Resource allocation : When there are multiple users or multiple jobs running at the same

time, resources must be allocated to each of them. Many different types of resources are managed by the operating system. Some may have special allocation code, whereas others may have much more general request and release code. For instance, in determining how best to use the CPU, operating systems have CPU-scheduling routines that take into account the speed of the CPU, the jobs that must be executed, the number of registers available and other factors.

Within the broad family of operating systems, there are generally four types, categorized based on the types of computers they control and the sort of applications they support. The broad categories are:

- **Real-time operating system (RTOS)** - Real-time operating systems are used to control machinery, scientific instruments and industrial systems. An RTOS typically has very little user-interface capability, and no end-user utilities, since the system will be a "sealed box" when delivered for use. A very important part of an RTOS is managing the resources of the computer so that a particular operation executes in precisely the same amount of time every time it occurs. In a complex machine, having a part move more quickly just because system resources are available may be just as catastrophic as having it not move at all because the system is busy.
- **Single-user, single task** - As the name implies, this operating system is designed to manage the computer so that one user can effectively do one thing at a time. The Palm OS for Palm handheld computers is a good example of a modern single-user, single-task operating system.
- **Single-user, multi-tasking** - This is the type of operating system most people use on their desktop and laptop computers today. Windows 98 and the MacOS are both examples of an operating system that will let a single user have several programs in operation at the same time. For example, it's entirely possible for a Windows user to be writing a note in a word processor while downloading a file from the Internet while printing the text of an e-mail message.
- **Multi-user** - A multi-user operating system allows many different users to take advantage of the computer's resources simultaneously. The operating system must make sure that the requirements of the various users are balanced, and that each of the programs they are using has sufficient and separate resources so that a problem with one user doesn't affect the entire community of users. Unix, VMS, and mainframe operating systems, such as MVS, are examples of multi-user operating systems.

Multi user operating system works in 4 different ways as :

- a) *Multitasking OS*
- b) *Timesharing OS*
- c) *Real time OS*
- d) *Distributed OS*

Multitasking OS :The OS which monitors multiple task simultaneously with in the stipulated time with perfection is referred as a Multitasking OS.E.g Windows NT,Linux,SCO Unix.

Timesharing OS : In a large network (mostly server centric) the server is equipped with this kind of OS which attends the request from the individual nodes in a time bound manner. The approach of the server to the request made by the individual node referred as a Round Robin League.E.g.Unix,Linux,Solaris,Windows NT.

A time shared operating system allows a user to interact with the computer and it facilitates program development. The response time of a good computer system to users' requests is of the order of 1or 2 seconds. The availability of large disk storage in such systems makes it feasible to store large volumes of data and to retrieve them fast. This facility is very useful in implementing systems such as an airline reservation system. In such a system ,the response time should be very short because a customer's reservation to be done while he waits.

****Time-sharing** systems were developed to provide interactive use of a computer system at a reasonable cost.A time-shared operating systems uses CPU scheduling and multiprogramming to provide each user with a small portion of a time-shared computer.It allows the many users to *share* the computer simultaneously. Since each action or command in a time shared system tends to be short ,only a little CPU time is needed for each user.

Real-Time OS: There are applications in which a computer is expected to control the operation of a physical system. For example, a satellite in orbit may be controlled by a computer. The position, velocity, acceleration and spin information of the satellite may be fed to a computer which may be programmed to compute the orbit and give instructions to rocket motors to correct the orbit. In such an application the operation is in "real time", that is, the control has to be exercised during the actual functioning of the system. Any delay beyond that specified for control would be disastrous. Real time OS have to work within strict time limits for critical jobs. Critical jobs are locked in memory and receive the highest priority.

**A real-time system is used when there are rigid time requirements on the operation of a processor or the flow of data, and thus often used as a control device in a dedicated application. A real-time task cannot be kept waiting indefinitely for the kernel to run it.

Distributed OS :

The distributed operating system supports the distributed computing environments which consists distributed databases system. This OS coordinates the activities occurring on remote terminals. Example: IBM OS/2.

It improves the performance of the data mines by providing the proper dead locks.

It's important to differentiate here between multi-user operating systems and single-user operating systems that support networking. Windows 2000 and Novell Netware can each support hundreds or thousands of networked users, but the operating systems themselves aren't true multi-user operating systems. The system administrator is the only "user" for Windows 2000 or Netware. The network support and all of the remote user logins the network enables are, in the overall plan of the operating system, a program being run by the administrative user.

With the different types of operating systems in mind, it's time to look at the basic functions provided by an operating system.

Wake-Up Call

When the power to a computer is turned on, the first program that runs is usually a set of instructions kept in the computer's **read-only memory (ROM)** that examines the system hardware to make sure everything is functioning properly. This **power-on self test (POST)** checks the CPU, memory, and basic input-output systems (BIOS) for errors and stores the result in a special memory location. Once the POST has successfully completed, the software loaded in ROM (sometimes called **firmware**) will begin to activate the computer's disk drives. In most modern computers, when the computer activates the hard disk drive, it finds the first piece of the operating system: the bootstrap loader.

This site is authored and maintained by M/s Cyber Planet for Abhinav Vidyalay & Jr. College. The said institution provides educational services in the area of primary, secondary & higher secondary education and thus the content is the intellectual property of the staff and students of this institution. No part of it may be reproduced without the consent of the institution.