History of O.S

- **1940s:** the earliest electronic digital computers did not include operating system. Machines of the time were so primitive (ancient).
- **1950s:** These computers were called single-stream batch-processing systems.
- **1960s:** It is also called the batch processing systems but using resources more efficiently by running several jobs at once. The systems improved resource utilization, Multiprogramming, Interactive users, Timesharing, Real-time systems, and reduce Turnaround time.
- **1970s:** Personal computers posted by early and continuing developments in microprocessor technology and using LAN with more problem in security.
- **1980s:** It was the decade of the personnel computers, Software such as spreadsheet programs, word processors, database packages and graphics packages and develop GUI, Distributed computing became wide spread under client/server model.
- **1990s:** Object technology: each software object encapsulates a set of attributes and methods. This allows applications to be built with components that can be reused in many applications. Open-source movement: open-source software is distributed with the source code, allowing individuals to examine and modify (Linux operating system). Operating system became increasingly user friendly (GUI features). Operating system designers developed techniques to protect computers from attacks and Microsoft became the dominant.
- **2000 and beyond:** Middleware: is a software that links two separate applications to communicate and exchange data via the internet, Massive parallelism, Computing on mobile devices which are used for e-mail, web browsing.

Definition of Operating System (OS)

An operating system acts as an intermediary between the 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 in a convenient and efficient manner.

Another definition is, O.S is a set of programs that controls effectively the computer resources and makes them conveniently available to users i.e easy to use. During computer operation, some basic O.S programs (Called O.S Core or Kernel) are resident in main memory.

Kernel: software that contains the core components of the operating system.

Operating System goals

1. Make S.W convenient to use.
2. Use the computer hardware in an efficient manner.
3. Provide a connection between the user and computer resources.

Computer System Components

Computer systems can be divided into four components.

1- Hardware (CPU, memory, I/O devices).

2- Operating system – Controls and coordinates use of hardware among various applications and users.

3- Application programs (compilers, assembler, text editor, database systems).

4- Users.

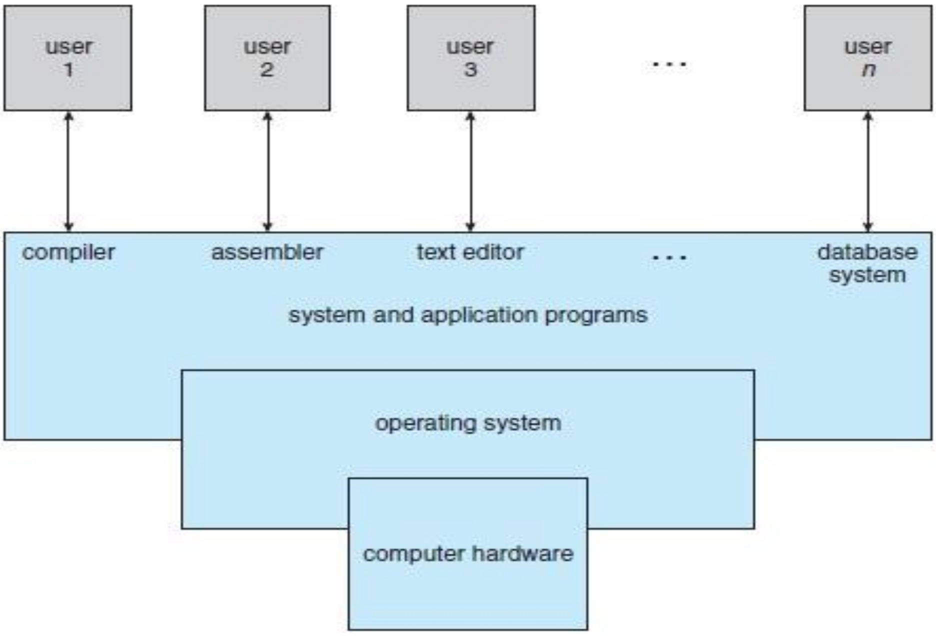

Figure 1: Abstract view of the components of a computer system.

Functions of OS

The functions can be summarized as follows (will be explained later in more details):

1- Management of computer resources (processors, memory, disks, I/O devices, programs, etc.).

2- Scheduling resources among users (time sharing).

3- Protection of programs being executed in memory from one another.

4- Providing a proper user interface e.g Graphics User Interface (GUI).

5- File management.

6- Network communication.

7- Many others.

Types of Operating Systems

Following are some of the most widely used types of Operating system.

1. **Batch System**

   The users of a batch operating system do not interact with the computer directly. The user has to submit a job (written on cards or tape) to a computer operator, then computer operator places a batch of several jobs on an input device. To speed up processing, jobs with similar needs were batch together and were run through the computer as a group. The programmers leave their programs with the operator and the operator then sorts the programs with similar requirements into batches. The major task of this system was to transfer control automatically from one job to the next. The problems with Batch Systems are as follows:

- There is no direct interaction between the user and the job.
- CPU is often idle, because the speed of the mechanical I/O devices is slower than the CPU.
- Difficult to provide the desired priority.

Memory management in batch system is very simple. Memory is usually divided into two areas: Operating system and user program area as shown in figure 2.

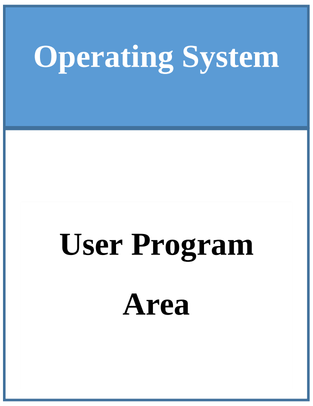

Figure 2: Memory Layout for a Batch System

Performance Development

OS attempt to schedule computational activities to ensure good performance, where many facilities had been added to OS some of these are:

a- Spooling

Spooling stands for simultaneous peripheral operation on-line. Spooling refers to putting jobs in a buffer, a special area in memory or on a disk where a device can access them when it is ready. The I/O devices can't match with the speed of a CPU. Hence, the output from the CPU will be stored in this spool (buffer) and the I/O devices can take the output from this buffer as and when required according to their speed. The CPU is hence not bound to this I/O device and can perform other operations. So, spooling keeps both the CPU and the I/O devices working at high rates without any waiting.

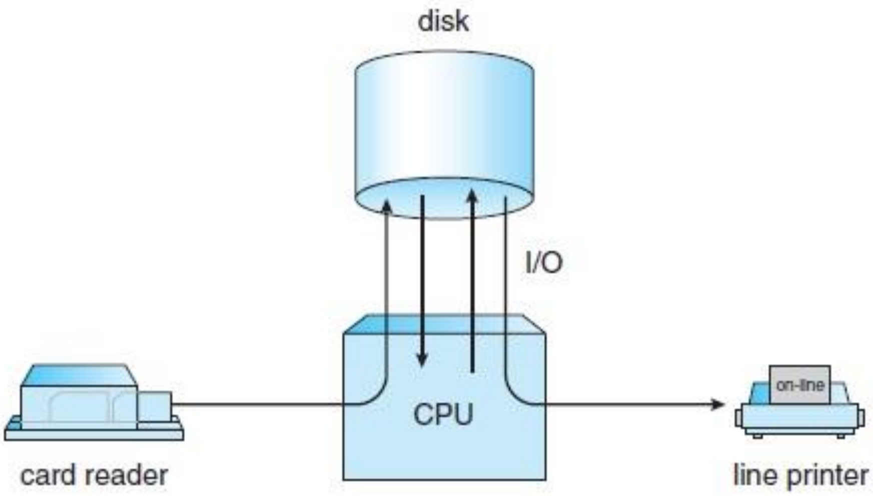

Figure 3: The Spooling Layout

b- Multiprogramming batch system

In a multiprogramming system there are one or more programs loaded in main memory which are ready to execute. Only one job at a time is able to get the CPU for executing its instructions while all the others are waiting their turn. The main idea of multiprogramming is to maximize the use of CPU time. In this system when the job may have wait for any reason, the OS simply switches to executes another job, when the second job need to wait the CPU is switches to another job and so on. Then the CPU will never be idle.

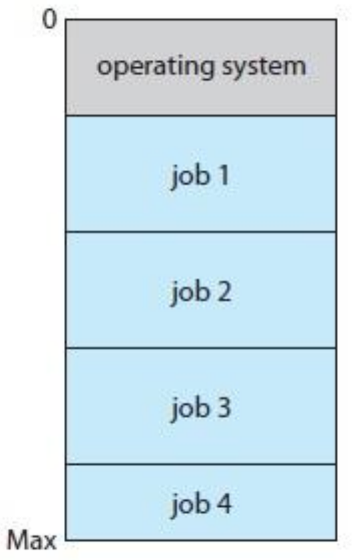

Figure 4: Memory Layout in Multiprogramming Environment

2- Time Sharing System

This type of OS provides on-line communication between the user and the system, where the user will give instruction to O.S directly (usually from terminal) and receive an immediate response, therefore some time called an interactive system.

The Time sharing system allows many users to share the computer resources simultaneously, where little CPU time is needed for each user. For example a mainframe computer that has many users logged on to it. Each user uses the resources of the mainframe -i.e. memory, CPU etc. The users feel that they are exclusive user of the CPU, even though this is not possible with one CPU i.e. shared among different users.

- Advantage: reduce the CPU idle time.
- Disadvantage: more complex.

3- Parallel System

a. Most systems previously were simple processor systems that is they have one main CPU.

b. There is a trend to have multiprocessor system, where such systems have more than one processor in close communication sharing the computer Bus, the clock, and some memory and peripheral devices.

c. Parallel operating systems are break large tasks into smaller pieces that are done at the same time in different places and by different mechanisms. They are sometimes also described as “multi-core” processors.

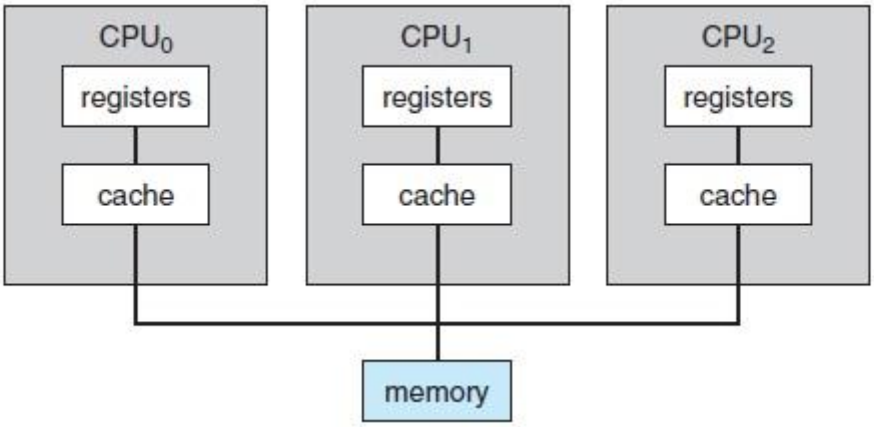

Figure 5: Parallel System Layout

d. The advantages of this type of systems are:

1. Increase throughput.
2. The cost.
3. Increase reliability.

There are two types of processors in multiprocessors systems:

1. **Asymmetric multiprocessor:** in which each processor is assigned a specific task. A boss processor controls the system; the other processors either look to the boss for instruction or have predefined tasks. This scheme defines a boss–worker relationship. The boss processor schedules and allocates work to the worker processors.

2. **Symmetric multiprocessor (SMP)**: in which each processor performs all tasks within the operating system. SMP means that all processors are peers; no boss–worker relationship exists between processors.

4. Distributed Systems

A distributed operating system is an operating system that runs on several machines whose purpose is to provide a useful set of services. The processors communicate with one another through various communication lines (such as high-speed buses or telephone lines) these are called distributed systems.

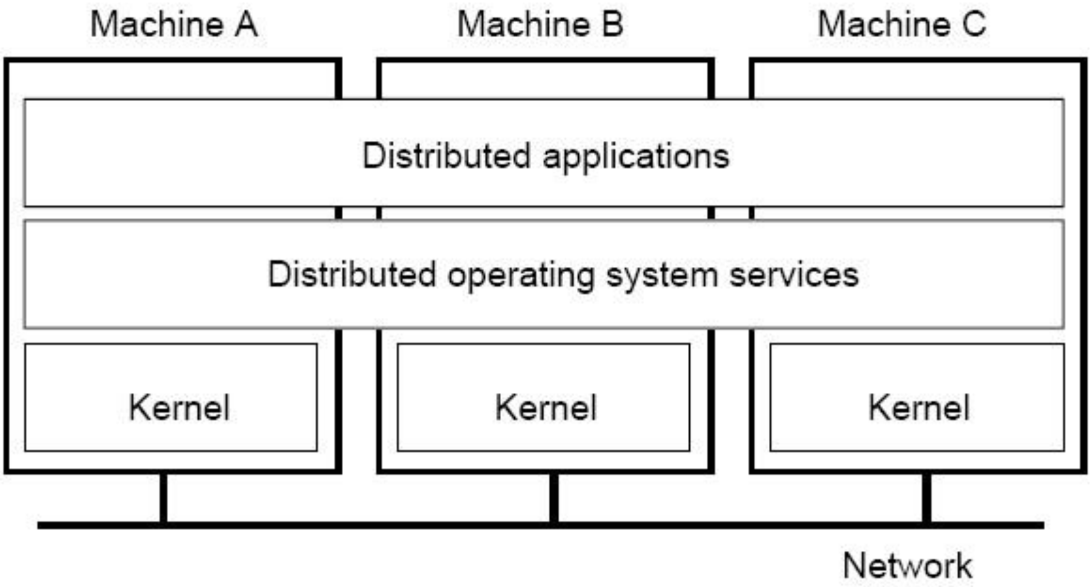

Figure 6: Distributed operating system

There are many reason to build this type of system:

1- Resource sharing.
2- Computation speed up.
3- Reliability
4- Communication.

5- Real Time System

Real-time systems are used when there are rigid time requirements on the operation of a processor or the flow of data. A real-time operating system must have well-defined, fixed time constraints, otherwise the system will fail. For example, scientific experiments, medical imaging systems, industrial control systems, weapon systems, robots, air traffic control systems, etc.

There are two types of real-time operating systems.

a. **Hard real-time systems:** guarantee that critical tasks complete on time. In hard real-time systems, secondary storage is limited or missing and the data is stored in ROM.

b. **Soft real-time systems:** A critical real-time task gets priority over other tasks and retains the priority until it completes. Soft real-time systems have limited utility than hard real-time systems. For example, multimedia, virtual reality, Advanced Scientific Projects like undersea exploration, etc.