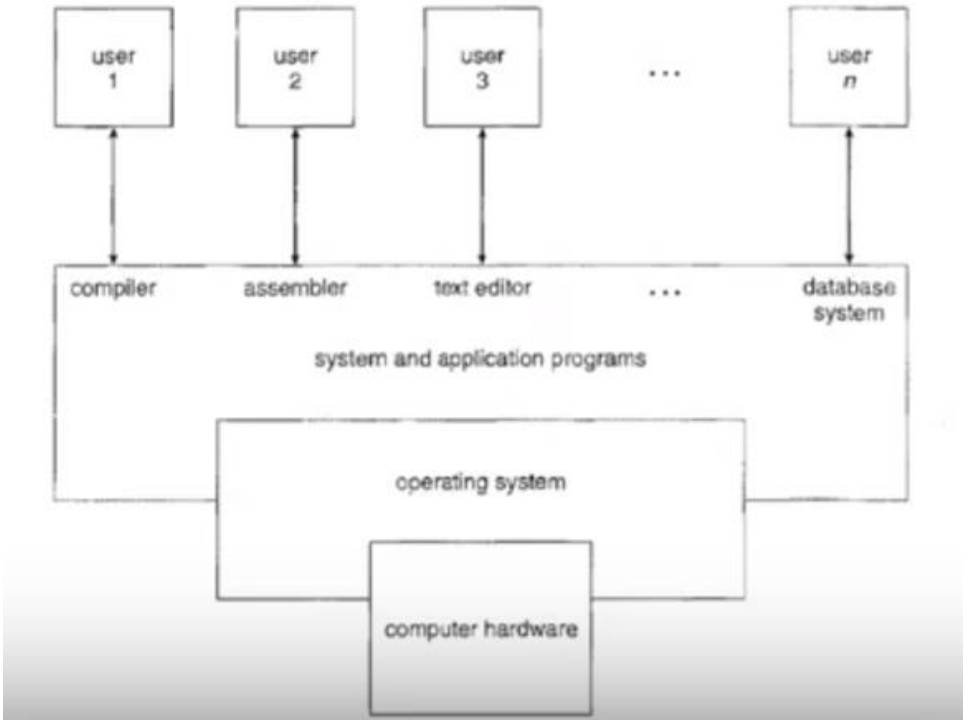
Chapter 4

Operating Systems

An operating system is the software that manages and controls the hardware and other software found on the computer. A main aspect of the operating system is not only to control the workings of a computer, but to provide an interface for users to be able to interact with it. Being able to interact with a computer will allow a user to create, modify, remove data, and execute a limitless number of programs.



In the previous diagram, at the base we have the computer system's hardware, which includes the CPU, I/O devices, printer and memory. The operating system contains functions which makes the hardware usable. For example, if you want to read something from memory or write something to the memory there are functions already written and it comes from the layer of system applications

program, which may include Microsoft Word, Excel, compiler, assembler or database systems.

As we know computers have different resources, some examples of resources are CPU, I/O, memory space, and system storage. When the user makes a request to use the CPU or I/O, the OS will allocate resources to fulfill the request. If multiple users have requests, then the OS's task is to decide how to order the tasks.

There are many different operating systems in existence; all have many differences and similarities. When comparing one to another, they may share everything except the visual presentation of the system, or they may look very similar but be completely different in how they operate. Every operating system has its purpose and reason for being. An operating system that may be made to satisfy a user at home may not be well suited for a computer used in a datacenter or business environment. All operating systems have their niche.

Popular Operating Systems

- Microsoft Windows
- Mac OS
- Linux
- Unix
- Apple IOS
- Android

History of Operating Systems

The most primitive of operating systems were introduced in the 1950's and early 1960's. Over the decades, operating systems have evolved several features to enhance the performance of computers they run on, better manage programs that run on them, and create an overall better user experience.

At first, computers were created to solve only a single or very limited number of problems/tasks(**single-tasking operating system**). As computers became more complex, and as operating systems were introduced, the ability to concurrently solve multiple complex and changing problems was possible(**multi-tasking operating system**). Multi-tasking systems accomplish this by using **multiprogramming**. Multiprogramming is a method to split up a task into small time allotments for use on the processor. The size of time allotments may be dependent on an external trigger to switch the process or by the operating system's task scheduler. Because many tasks are split up, the illusion of many programs operating at the same time is created. An overly broad example would be to concurrently run a text editor, virus scan, and system update.

Each task, known as a **job**, was able to be completed one at a time. As more and more tasks were needing to be completed, computer operators of **single-user computer systems** would develop a queue of what needed to be ran/solved on the computer for themselves and others; this is known as a **job queue**, a list of tasks.

Once computers were able to accommodate multiple users at the same time, the concept of **time-sharing** was created, where multiple users were able to 'share' CPU time.

Components of an Operating System

The most familiar and identifiable component of an operating is the **GUI** (Graphical User Interface). Whether talking about the operating system or any program running on it, the GUI is the graphical presentation of software. Before GUIs came around, computer users operated computers using a text-based command-line **shell interface**, a user interface used to interact with an operating system. On Microsoft Windows, two command line interfaces are Command and PowerShell; PowerShell is the newest shell interface offered. All operating systems come with a command-line shell interface for users to interact with it, most likely both a GUI and command line interface will be provided. An important quality of command-line shell interfaces is that they are very powerful (they can do more), much more so than what is capable with a GUI.

At the lowest level (closest to the CPU) of an operating system lies the kernel. The kernel is the most important part of an operating system. It provides control over the computer's hardware devices for the operating system, memory management, virtual memory, and scheduling of tasks.

Device drivers, a component of the kernel, are specific to controlling a single piece of hardware. There are drivers for mice, keyboards, printers, monitors, video cards, hard drive disks, etc. Device drivers may be updated automatically via system updates, by software running on your computer, or manually in a system's device driver management window. Without device drivers we would not be able to effectively use many components of a modern computer and its peripheral devices.

Memory management conducted by the kernel determines where data will be stored in memory, transfer it when it is called, and replace it when no longer needed. When the total amount of memory needed exceeds available space in RAM, data will be stored in virtually memory located on a hard drive disk; this is called **paging**.

Multitasking is possible thanks to the scheduling program of the kernel. When multiple programs need to run concurrently, the scheduler is what gets it done. The scheduler accomplishes this by determining how much time to allot to each process and the order of the process queue.

File Systems

Many different file systems have been created over the decades. The purpose of a file system is to create a standard way of storing information on a hard drive, usually in the form of files. By having information consolidated into an isolated file, information is much easier to keep track of and identify than if all the data were one long continuous string of data. Common Microsoft and Linux operating systems usually only support limited number of file systems such as FAT or NTFS and may have a preference. If the current operating system does not support a file system, an extra driver may be installed to be able to manage it.

Security

An important function of operating systems is to provide users with basic security. This is accomplished several ways. The most recognizable security feature, though you may not consider it first, is the user system. Operating systems, when initially set up/installed, create a couple of basic accounts (administrator and guest) and usually offer the installer an option to create an additional user account. Each user is then assigned to any number of specific groups, which then can perform specific functions like accessing, reading, writing, and deleting files. Groups may be created, modified, and removed by working with an operating system's group policy editor or by using a command-line interface. Users assigned to an administrator group almost always have an unfettered access to all files and settings, while regular users have controlled access to the file system and settings. Depending on the computer's environment, it may be best to provide users with a quite restrictive system access. Along side users being assigned to a group, passwords are used. A password provides a simple layer of protection to restrict and deter unauthorized access to a user account, be it a regular user or an administrator.

A second type of protection that may be offered by an operating system is a built-in anti-virus, anti-malware, and/or firewall. By having an application running in the background, a system will be monitored with minimal user input. These three types of service are a method to control different types of attacks that may occur against a system and, most likely, will all be packaged together. By utilizing these tools, system data is much more likely to remain secure, though not guaranteed due to the ever-evolving methods of attack.

The system's CPU and memory manager are integral to foiling unauthorized access to data and applications. This is done by controlling and cordoning areas in memory in which a process has access to. If a process has access to an area of memory outside of its domain, that is said to be insecure and the memory manager is meant to protect against that as part of its many duties. Because faults are routinely identified, updates are regularly pushed out to address this. Another method related to the CPU is the use of privilege levels. Anytime a process acquires CPU time, its job is started at a nonprivileged level by default. If a process attempts to execute a privileged level instruction, it is interrupted and essentially shut down.