Types of Operating Systems

Bhavneet Soni

Author's Note

Operating Systems are the heart of any device, without an OS a computer or a device will just be brick with shinny lights. Advancements in computer architecture has led to need for and development of various different types of Oss. In this report, we look at various OS available today from personal computers to network servers, mobile devices to real time OS used to operate machines and all.

Types of Operating Systems

To understand different types of operating systems available, how they work and differ from each other we need to understand what is an OS and why do we need it. Operating system is a software abstraction layer over hardware. It provides an interface between the hardware and the application or program you are using. Figure 1.0 shows how OS behaves like an abstraction layer between the application and the hardware. Now question arises why do we need an OS and why the application can't directly communicate with the hardware.

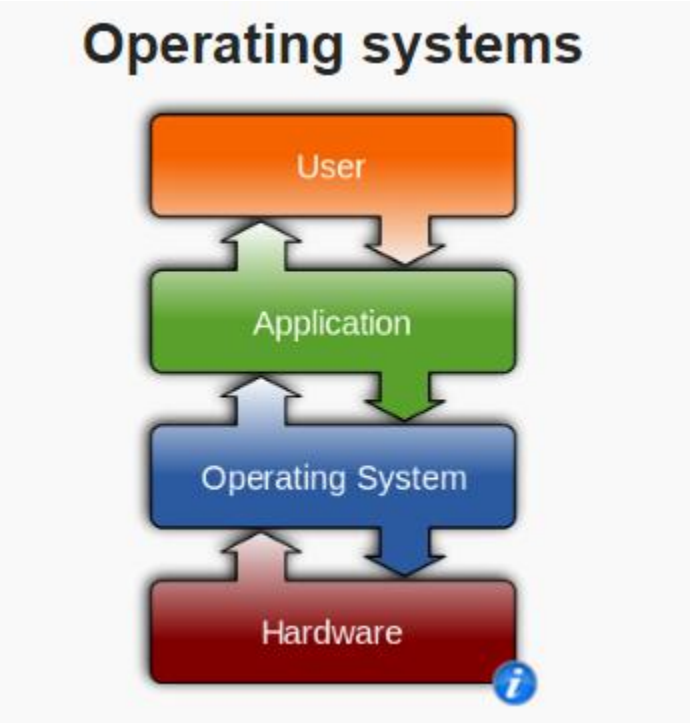

Figure 1 Abstraction layers between user and the hardware. (www.wikipedia.org, 2016)

For application to interact with the hardware it should be able to handle how different pieces of hardware interact with each other. How the hardware understands the user inputs, access and store data into the storage. Application will have to keep track of the time, which is required for all the computing operations. Application will also have to manage how to show output from the application, keeping track and interacting with devices such as monitor printer speakers etc.

Apart from it application will have to communicate with other computers network devices etc. Its not an ideal scenario in which a developer has to take care of all these issues when designing a new application. In addition, applications will have to be designed differently to run on different computer architectures. Hence, there is a need to have an abstraction layer that can provide an environment for applications to run smoothly. This is where operating system comes in, OS handles all these tasks such as scheduling memory management device drivers freeing user and application developer from the worries of hardware level details. The OS handles routing and connection strategies, and the problems of contention and security. To enumerate following are some of the functions an operating system do but not limited to

1. Loads a program into memory.
2. Executes the program.
3. Handles program's execution.
4. Provides a mechanism for process synchronization.
5. Provides a mechanism for process communication.
6. Provides a mechanism for deadlock handling.
7. I/O Operations – communication between user and devices, storage etc
8. Lets user create and delete files, directories and manage file systems.
9. Operating System provides an interface to the user to create/delete directories.
10. Provide an interface to create the backup of file system.
11. Error handling for any I/O devices
12. CPU management

There is a plethora of operating systems are available basaed on the application and devices, below are mentioned a few of the operating systems and how they differ from each other

Types Of Operating Systems available

1. **Batch Operating System**

In batch operating systems, users prepares his job on an offline device such as puch cards and submits it to the computer operator. Similar jobs are batched together and run in a group. These OS use using JCL (Job Control Language) and submit jobs for execution to another program called JES (Job Entry Sub-System). Running the jobs as a group gives an advantage of more time efficient use of the computer resource. However, the lack of end user directly with the computer and are dependent on a computer operator. CPU is often idle between the different batches being run. It’s often difficult to prioritize the jobs as they are all run in batches. Some of the examples of batch OS is IBM's MVS Operating system.

2. **Time Sharing Operating Systems**

Another technique to interact with computer systems is Time-sharing, this technique enables many people, located at various terminals, to use a particular computer system at the same time. Time-sharing extends the multiprogramming or multitasking as processor's time is shared among multiple users simultaneously. It differs from multi-programmed Batch Systems is that batch systems try to maximize processor use, whereas in Time-Sharing Systems, the objective is to minimize response time. The CPU executes multiple jobs by switching between them frequently, thus, giving out immediate response.

For example, in a transaction processing, the processor executes each user program in a short burst or quantum of computation, giving each user a time quantum. Time sharing OS utilizes CPU scheduling and multiprogramming to provide each user with a small portion of a time. Computer systems that were designed primarily as batch systems have been modified to time-sharing systems. While time sharing systems have security and integrity of user programs and data, reliability issues and data communication. However they provide a quicker response, avoid duplication of software and reduces CPU idle time. BSD Unix is one such OS that is used widely in academic environment.

3. **Distributed Operating Systems**

While batch and time-shared OS work on one Central CPU, Distributed systems use multiple central processors to serve multiple real-time applications and multiple users. CPUs communicate with each other via high-speed buses or telephone lines and distribute Data processing jobs among the processors. These systems may vary in size and function, the processors are referred as sites, nodes, computers, and so on. In a distributed OS, the kernel often supports a minimal set of functions, including low-level address space management, thread management, and inter-process communication (IPC). Some of the advantages provided by these systems are access to remote resources, high-speed data exchange, redundancy if one system fails remaining can continue to operate and better load distribution. Some of the examples include UNIX®, Linux®, and the Windows® operating system, these are installed on multiple servers to communicate with each other.

4. **Network Operating System**

Another type of OS is Network Operating System that runs on a server and is responsible for managing data, user groups, security, applications and other networking

functions. The main use of the network OS is to allow for sharing of resources and data among multiple computers on a network like local area network (LAN), a private network or to other networks. Advantages of network OS is it provides highly stable and secure solution. Upgradation to new technologies and hardware is easily managed and integrated in to the system additional benefit of remote access from different locations and types of systems. Its provides some challenges as well, apart from being quite expensive its highly centralized and requires regular maintenance and updates. Examples of network operating systems include Microsoft Windows Server, UNIX, Linux, Mac OS X, Novell NetWare, and BSD.

5. **Real Time operating**

A real-time OS system processes and responds to inputs in near real time and controls the environment. It has a relatively very short response time, response time can be defined as the time taken by the system to respond to an input and display of required updated information. These systems are widely used when the time required operation of a processor or the flow of data and real-time systems are required. Such as operating a a control device in a dedicated application. These OS have to adhere to well-defined, fixed time constraints, otherwise the system will fail. Examples of such OS are the ones used in 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***

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. In these systems, virtual memory is almost never found.

b. ***Soft real time***

Soft real-time systems are less restrictive. 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 and planetary rovers, etc.

Modern computers unsed on personal comptuers are also known as GUIs – Graphical user interface. OS are being segregated based on mobile vs personal computer OS, but that line is thining and converging into single OS for both mobile devices and desktop computers. Below is a list of O Systems used on PCs and mobiles.

Table 1List of Operating Systems over the time

Operating system	Platform	Developer
AIX and AIXL	Various	IBM
AmigaOS	Amiga	Commodore
Android	Mobile	Google
BSD	Various	BSD
Caldera Linux	Various	SCO
Corel Linux	Various	Corel
CP/M	IBM	CP/M
Debian Linux	Various	GNU

DUnix	Various	Digital
DYNIX/ptx	Various	IBM
HP-UX	Various	Hewlett Packard
iOS	Mobile	Apple
IRIX	Various	SGI
Kondara Linux	Various	Kondara
Linux	Various	Linus Torvalds
MAC OS 8	Apple Macintosh	Apple
MAC OS 9	Apple Macintosh	Apple
MAC OS 10	Apple Macintosh	Apple
MAC OS X	Apple Macintosh	Apple
Mandrake Linux	Various	Mandrake
MINIX	Various	MINIX
MS-DOS 1.x	IBM	Microsoft
MS-DOS 2.x	IBM	Microsoft
MS-DOS 3.x	IBM	Microsoft
MS-DOS 4.x	IBM	Microsoft
MS-DOS 5.x	IBM	Microsoft
MS-DOS 6.x	IBM	Microsoft
NEXTSTEP	Various	Apple
OS/2	IBM	IBM
OSF/1	Various	OSF
QNX	Various	QNX
Red Hat Linux	Various	Red Hat
SCO	Various	SCO
Slackware Linux	Various	Slackware
Sun Solaris	Various	Sun
SuSE Linux	Various	SuSE
Symbian	Mobile	Nokia

System 1	Apple Macintosh	Apple
System 2	Apple Macintosh	Apple
System 3	Apple Macintosh	Apple
System 4	Apple Macintosh	Apple
System 6	Apple Macintosh	Apple
System 7	Apple Macintosh	Apple
System V	Various	System V
Tru64 Unix	Various	Digital
Turbolinux	Various	Turbolinux
Ultrix	Various	Ultrix
Unisys	Various	Unisys
Unix	Various	Bell labs
UnixWare	Various	UnixWare
VectorLinux	Various	VectorLinux
Windows 2000	IBM	Microsoft
Windows 2003	IBM	Microsoft
Windows 3.X	IBM	Microsoft
Windows 7	IBM	Microsoft
Windows 8	IBM	Microsoft
Windows 95	IBM	Microsoft
Windows 98	IBM	Microsoft
Windows 10	IBM	Microsoft
Windows CE	PDA	Microsoft
Windows ME	IBM	Microsoft
Windows NT	IBM	Microsoft
Windows Vista	IBM	Microsoft
Windows XP	IBM	Microsoft
Xenix	Various	Microsoft

References

(2016, Oct). Retrieved from www.wikipedia.org: https://en.wikipedia.org/wiki/Operating_system