Computer OS's

Common OS's you will have heard of :-

MAC Os !
Windows from 98 XP and beyond ???
Linux
DOS

OS you may have forgo ton

CPM
DOS
Amiga OS
Sinclair BASIC
Commodore BASIC 2.0

Others ?

Not so common but in use today on millions of places
Java – Mobile Phones
Unix – Internet, Communications
Windows CE – Phones PDA's
Palm OS – Phones PDA's
ROM based – ADSL routers
Games Consoles etc

Can you name any others ?

An Operating System, or OS, is a software program that enables the computer hardware to communicate and operate with the computer software. Without a Operating System, the hardware would be useless, the OS performs basic tasks such as controlling and allocating memory, prioritizing system requests, controlling input and output devices, facilitating networking, and managing files. It also may provide a graphical user interface for higher level functions.
At the simplest level, an operating system does two things:

1. It manages the hardware and software resources of the system. In a desktop computer, these resources include such things as the processor, memory, disk space, etc. (On a cell phone, they include the keypad, the screen, the address book, the phone dialer, the battery and the network connection.)
2. It provides a stable, consistent way for applications to deal with the hardware without having to know all the details of the hardware.

The first task, managing the hardware and software resources, is very important, as various programs and input methods compete for the attention of the **central processing unit** (CPU) and demand memory, storage and input/output (I/O) bandwidth for their own purposes. In this capacity, the operating system plays the role of the good parent, making sure that each application gets the necessary resources while playing nicely with all the other applications, as well as husbanding the limited capacity of the system to the greatest good of all the users and applications.

The second task, providing a consistent application interface, is especially important if there is to be more than one of a particular type of computer using the operating system, or if the hardware making up the computer is ever open to change. A consistent **application program interface** (API) allows a software developer to write an application on one computer and have a high level of confidence that it will run on another computer of the same type, even if the amount of memory or the quantity of storage is different on the two machines.

Even if a particular computer is unique, an operating system can ensure that applications continue to run when hardware upgrades and updates occur. This is because the operating system and not the application is charged with managing the hardware and the distribution of its resources. One of the challenges facing developers is keeping their operating systems flexible enough to run hardware from the thousands of vendors manufacturing computer equipment. Today's systems can accommodate thousands of different printers, disk drives and special peripherals in any possible combination.

What Kinds Are There?

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. Microsoft's *Windows* and Apple's *MacOS* platforms are both examples of operating systems 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.

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

When you turn on the power to a computer, the first program that runs is usually a set of instructions kept in the computer's read-only memory (ROM). This code 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 the BIOS or **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**.

The bootstrap loader is a small program that has a single function: It loads the operating system into memory and allows it to begin operation. In the most basic form, the bootstrap loader sets up the small driver programs that interface with and control the various hardware subsystems of the computer. It sets up the divisions of memory that hold the operating system, user information and applications. It establishes the data structures that will hold the myriad signals, flags and semaphores that are used to communicate within and between the subsystems and applications of the computer. Then it turns control of the computer over to the operating system.

The operating system's tasks, in the most general sense, fall into six categories:

- Processor management
- Memory management
- Device management
- Storage management

- Application interface
- User interface

While there are some who argue that an operating system should do more than these six tasks, and some operating-system vendors do build many more utility programs and auxiliary functions into their operating systems, these six tasks define the core of nearly all operating systems.

The heart of managing the processor comes down to two related issues:

- Ensuring that each process and application receives enough of the processor's time to function properly.
- Using as many processor cycles for real work as is possible.

The basic unit of software that the operating system deals with in scheduling the work done by the processor is either a **process** or a **thread**, depending on the operating system.

When an operating system manages the computer's memory, there are two broad tasks to be accomplished:

1. Each process must have enough memory in which to execute, and it can neither run into the memory space of another process nor be run into by another process.
2. The different types of memory in the system must be used properly so that each process can run most effectively.

Device Management

The path between the operating system and virtually all hardware not on the computer's motherboard goes through a special program called a driver. Much of a driver's function is to be the translator between the electrical signals of the hardware subsystems and the high-level programming languages of the operating system and application programs. Drivers take data that the operating system has defined as a file and translate them into streams of bits placed in specific locations on storage devices, or a series of laser pulses in a printer.

Application Interface

Just as drivers provide a way for applications to make use of hardware subsystems without having to know every detail of the hardware's operation, **application program interfaces** (APIs) let application programmers use functions of the computer and operating system without having to directly keep track of all the details in the CPU's operation.

User Interface

Just as the API provides a consistent way for applications to use the resources of the computer system, a **user interface** (UI) brings structure to the interaction between a user and the computer. In the last decade, almost all development in user interfaces has been in the area of the **graphical user interface** (GUI), with two models, Apple's *Macintosh* and Microsoft's *Windows*, receiving most of the attention and gaining most of the market share. The popular, open-source Linux operating system also supports a graphical user interface.

Tonight this short demo will deal with PC's but as described before OS.s can come in many shapes and forms most electronic equipment will have some sort of code that enables it to communicate with its hardware.

Most of you with a few exceptions will use Windows XP 98 etc. WHY ? Because it is available of the shelf you can turn it on and it works ? you can buy or download readily available software for it. TRUE !

However this is true for most of the above os's

All of the os's do the same thing surf the net, send an email, gather your digital cam pics, write a letter, scan and print, amateur radio progs etc which is in truth the main thing a home computer does ? So why do we all choose Windows ? When there are alternatives.

I have on the table a PC
A Solaris X86 Laptop
Same PC that was running Windoze running Linux
MAC Osx Laptop

All these systems will do the same thing some may do it faster some may do it more reliably and some may do it more securely. They will all communicate with each other over TCPIP, some may be subject to vulnerabilities over the Internet some not so.

The main problem with choosing a new os if the steep learning curve and being able to get you accessories to work, Drivers etc.

However with Solaris,Linux and MAC OS and to some extent Windows the “Opensource” network has become a easy way of find applications, drivers etc for the above OS.s and most common pieces of hardware work on all of the above.

So why would you want to change well the good news you do not have to ! If you want to try Linux you can with out having any effect on your current OS on your PC sorry David.

You can use a “Live CD” the one I have tonight is from AI9NL and is amateur radio orientated, just place in you CD / DVD drive and boot from it, you may have to set your BIOS to boot from CD 1st.

This will give you a fully functioning “OS” with Packet, APRS,packet, Data modes SSTV using your soundcard. If you are connected to Broadband via a router it will also connect to the Internet and collect your email and browse the net, once finished turn it off and go back to your original OS or if you like it install it. An alternative to overwriting your current OS is to add a 2nd hard drive and install the new OS on to that.

Getting back to why change,,,,,

How many have computers running anything other than Windows

from the people who are left what OS.s do you use ?

If you went to a Shop to buy a new PC would you consider looking at anything other than a Windows PC ?

Average price of a Windows PC £350 -500
Average price of a Linux PC £150 - £200

The choice is yours ! But a few of you can try with a live CD if you wish.

OS.s are varied but they are all tied to the hardware it will be used on.
Windows runs on a Pc based hardware, however MAC OSX, Unix and linux can also run on PC based hardware, however PowerPC hardware is limited to MACOS and Unix, Sparc Based is limited to Solaris (unix) etc this ty in also ties the applications to the OS hence the ability to easily access s/w for the PC and MACOS to some extent.

Microsoft Office runs on Microsoft windows, however Openoffice which is compatible with Microsoft windows runs on Linux Solaris MACOS and Windows and it free it is part of the opensource licence.