SOFTENGER
Strive to Excel

OS fundamental and Unix Introduction

Operating system

SOFTENGER
Strive to Excel

- An Operating System (OS) is an interface between a computer user and computer hardware.
- An operating system is a software which performs all the basic tasks like file management, memory management, process management, handling input and output, and controlling peripheral devices such as disk drives and printers.
- Some popular Operating Systems include Linux Operating System, Windows Operating System, VMS, OS/400, AIX, z/OS, etc.
- Definition

  An operating system is a program that acts as an interface between the user and the computer hardware and controls the execution of all kinds of programs.

Operating System

OS are classified into different types depending on their capability of processing

- Single user
- Multiuser
- Multitasking
- Multiprocessing
- Real time
- Embedded

Single User and Single Task OS:

- It is for use by a single user for a standalone single computer for performing a single task
- Operating system for Personal Computers (PC) are single-user OS
- Single user OS are simple operating system designed to manage one task at a time.
- Example: MS-DOS

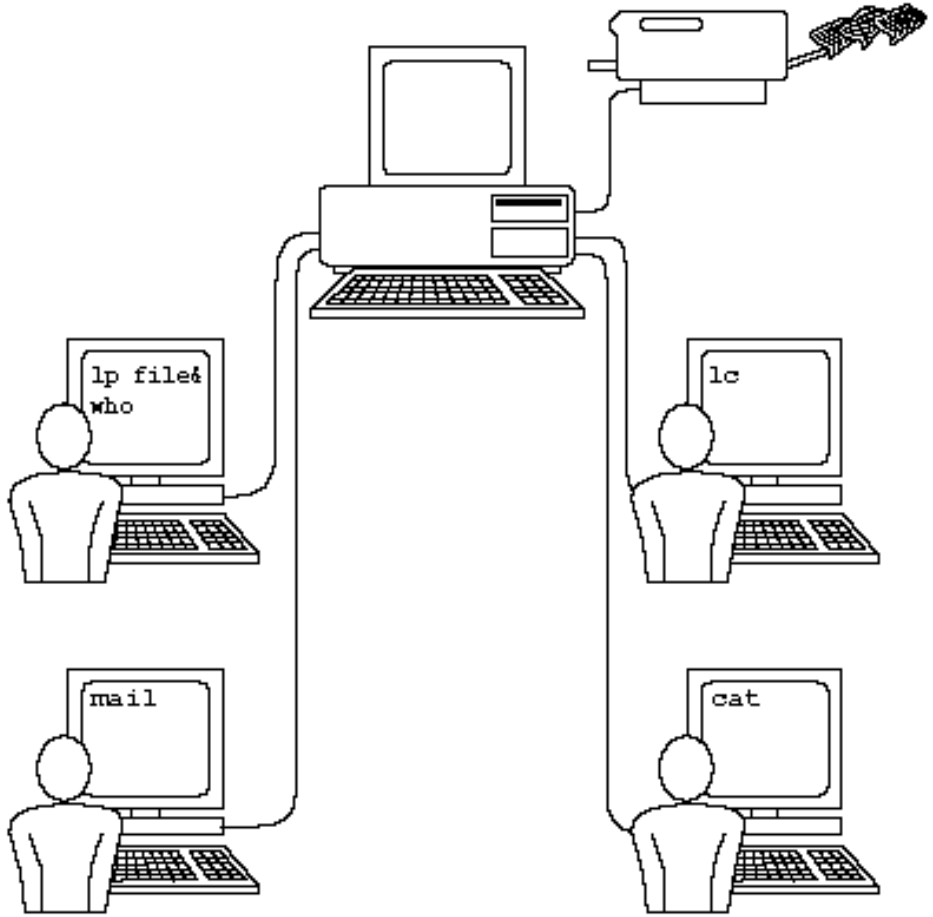
SOFTENGER
Strive to Excel

Single User and Multitasking OS

- Allows execution of more than one task or process concurrently
- The processor time is divided amongst different tasks
- This division of time is also called time sharing.
- The processor switches rapidly between processes.
- For example, the user can listen to music on the computer while writing
- an article using a word processor software.

Multiuser OS

- is used in computer networks that allow same data and applications to be accessed by multiple users at the same time
- The users can also communicate with each other.
- Linux, UNIX, and Windows 7 are examples of multiuser OS.



- A multi-user operating system allows more than one user to share the same computer system at the same time.
- It does this by time-slicing the computer processor at regular intervals between the various users.

Multiprocessing OS

- have two or more processors for a single running process. Processing takes place in parallel and is also called parallel processing.
- Each processor works on different parts of the same task, or, on two or more different tasks.
- Since execution takes place in parallel, they are used for high speed execution, and to increase the power of computer.
- Linux, UNIX and Windows 7 are examples of multiprocessing OS.

SOFTENGER
Strive to Excel

Real Time OS

- are designed to respond to an event within a predetermined time.
- These operating systems are used to control processes.
- Processing is done within a time constraint.
- OS monitors the events that affect the execution of process and respond accordingly.
- They are used to respond to queries in areas like medical imaging system, industrial control systems etc.

Embedded OS

- is embedded in a device in the ROM.
- They are specific to a device and are less resource intensive.
- They are used in appliances like microwaves, washing machines, traffic control systems etc.

Functions of OS

- Operating system is a large and complex software consisting of several components.
- Each component of the operating system has its own set of defined inputs and outputs.
- Different components of OS perform specific tasks to provide the overall functionality of the operating system.

Process Management

- The process management activities handled by the OS are.
- control access to shared resources like file, memory, I/O and CPU
- control execution of applications.
- create, execute and delete a process.
- cancel or resume a process
- schedule a process
- synchronization, communication and deadlock handling for processes.

Memory Management

- The activities of memory management handled by OS are
- allocate memory
- free memory
- re-allocate memory to a program when a used block is freed
- keep track of memory usage.

File Management

- The activities of memory management handled by OS are
- create and delete both files and directories
- provide access to files
- allocate space for files
- keep back-up of files
- secure files

Device Management

- The Device management tasks handled by OS are:
- open, close and write device drivers,
- communicate, control and monitor the device driver.

Protection and Security

- OS protects the resources of system.
- User authentication.
- file attributes like read, write, encryption, and back-up of data are used by OS to provide basic protection.

SOFTENGER
Strive to Excel

User Interface or Command Interpreter

- Operating system provides an interface between the computer user and the computer hardware.
- The user interface is a set of commands or a graphical user interface via which the user interacts with the applications and the hardware.

Unix History

SOFTENGER
Strive to Excel

- 1964 : OS called MULTICS (Multiplexed Information and Computing System) was developed by Bell Labs, MIT & General Electric.
- 1969 : Ken Thompson (System programmer of Bell Labs) wrote OS on PDP -7 Computer (Program Data Processor Model - 7), assembler and few utilities, this is know as Unix.
- 1971 : Moves UNIX from the PDP-7 to the PDP-11. (This version of Unix was not portable.)
- 1973 : Rewrites UNIX in Dennis Ritchie's C language Unix written in 'C' is portable. It means Unix can run on variety of Hardware platform.
- 1973 : Rewrites portions of UNIX to include Doug McIlroy's concept of pipes 1975: UNIX leaves home. Also widely known as Version 6, this is the first to be widely available out side of Bell Labs. The first BSD version (1.x) was derived from V6.
- 1979 : It was a "improvement over all preceding and following Unices" [Bourne]. It had C, UUCP and the Bourne shell. It was ported to the VAX and the kernel was more than 40 Kilobytes (K).

- 1980 : Microsoft introduces Xenix. 32V and 4BSD introduced.
- 1982 : AT&T's UNIX System Group (USG) release System III, the first public release outside Bell Laboratories. SunOS 1.0 ships. HP-UX introduced. Ultrix-11 Introduced.
- 1983 : Computer Research Group (CRG), UNIX System Group (USG) and a third group merge to become UNIX System Development Lab. AT&T announces UNIX System V, the first supported release. Installed base 45,000.
- 1984 : University of California at Berkeley releases 4.2BSD, includes TCP/IP, new signals and much more. X/Open formed.
- 1984 : System V Release 2 introduced. At this time there are 100,000 UNIX installations around the world.
- 1986 : 4.3BSD released, including internet name server. SVID introduced. NFS shipped. AIX announced. Installed base 250,000

Unix

- Unix is a layered operating system. The innermost layer is the hardware that provides the services for the OS.
- The operating system, referred to in Unix as the kernel, interacts directly with the hardware and provides the services to the user programs.
- These user programs don't need to know anything about the hardware. They just need to know how to interact with the kernel and it's up to the kernel to provide the desired service.
- Most well written user programs are independent of the underlying hardware, making them readily portable to new systems.
- User programs interact with the kernel through a set of standard system calls.
- These system calls request services to be provided by the kernel.

Such services would include

- accessing a file: open close, read, write, link, or execute a file;
- starting or updating accounting records;
- changing ownership of a file or directory; changing to a new directory;
- creating, suspending, or killing a process;
- Enabling access to hardware devices;
- And setting limits on system resources.

Unix, Linux runs on different hardware platform :

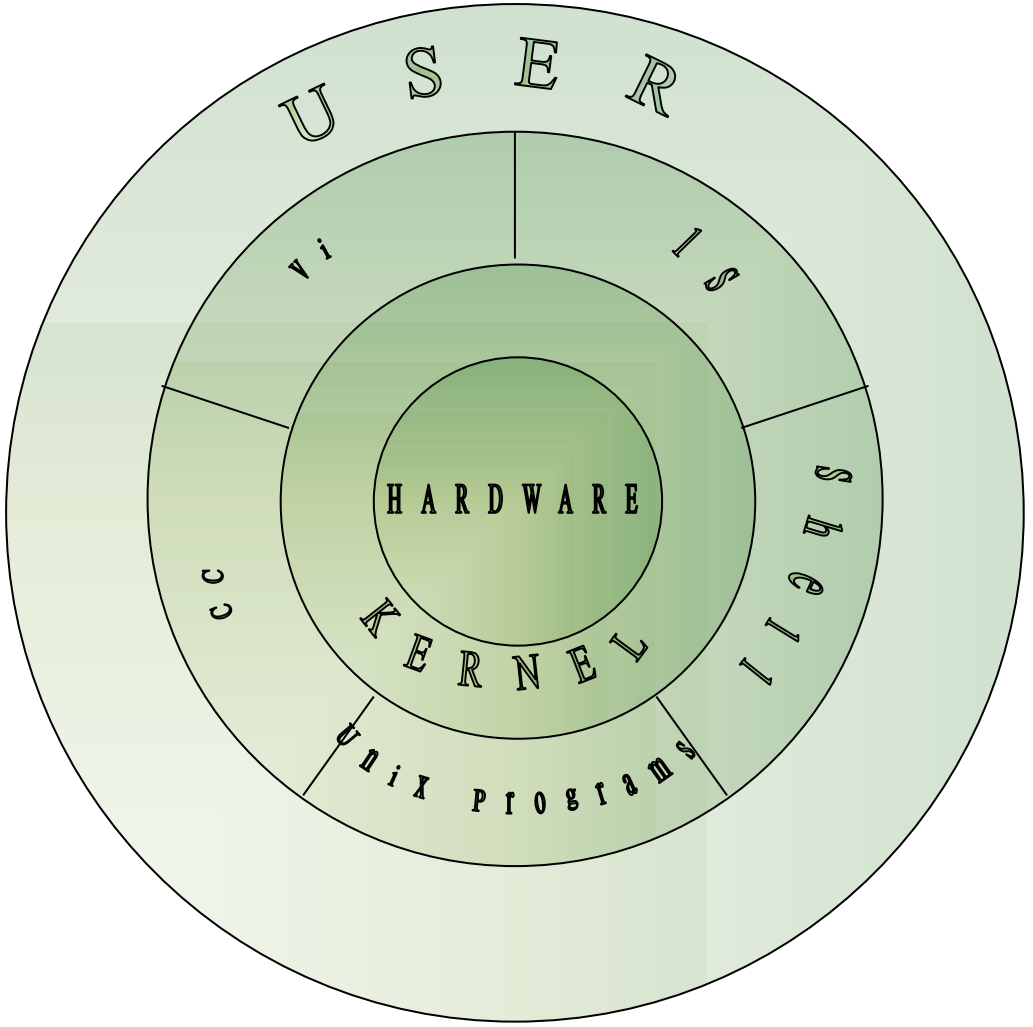
- Intel x86 processor (Celeron/PII/PIII/PIV/Old- Pentiums/80386/80486)
- Macintosh PC's
- Cyrix processor
- AMD processor
- Sun Microsystems Sparc processor
- Alpha Processor (Compaq)

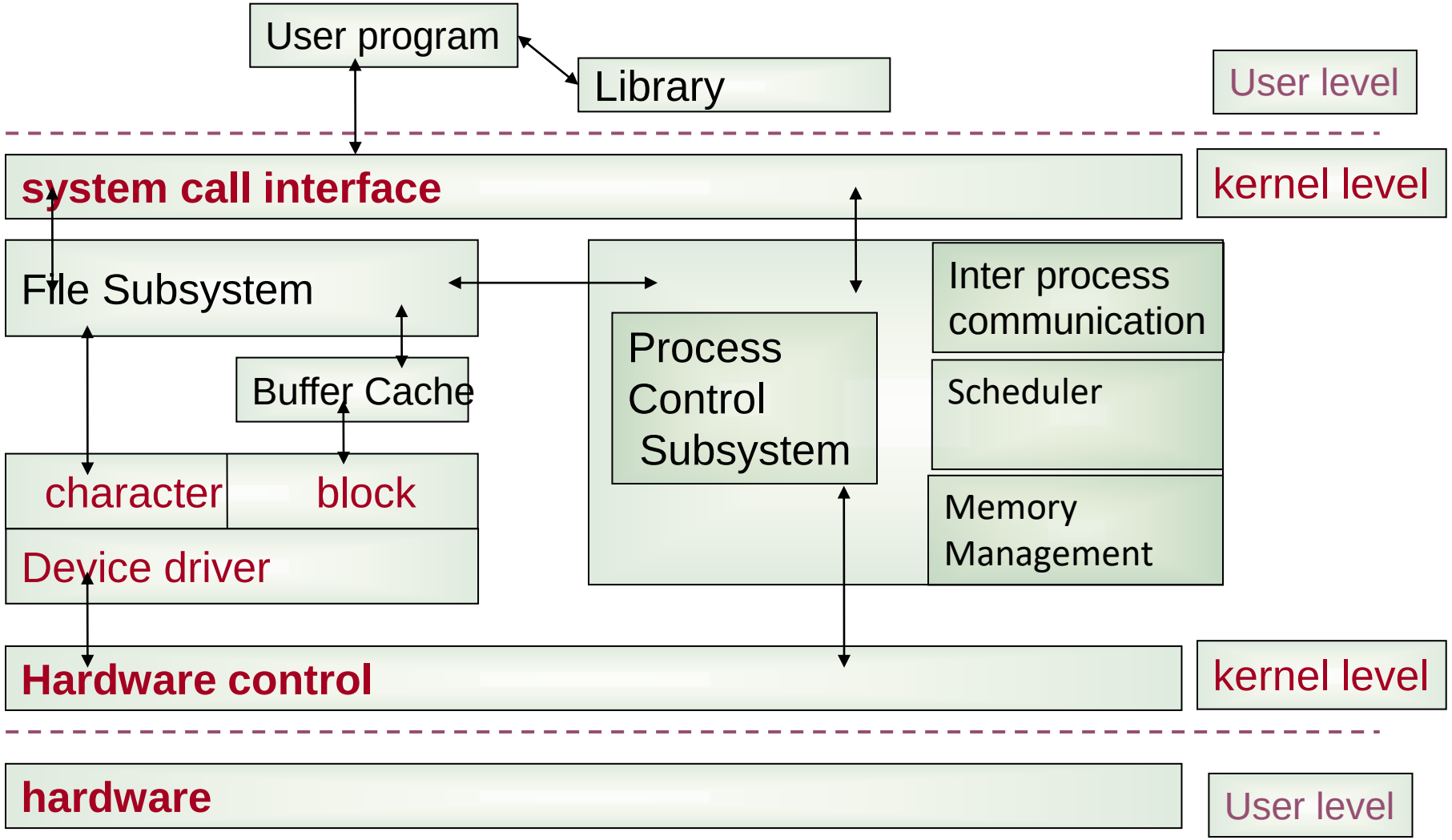
Minimum hardware required :

- Intel x86 processor (Celeron/PII/PIII/PIV/Old-Pentiums/80386/80486)
- 4 MB RAM
- 80 MB hard disk)

Unix Architecture

- **Kernel** – The kernel is the heart of the operating system. It interacts with the hardware and most of the tasks like memory management, task scheduling and file management.
- As an illustration of the way that the shell and the kernel work together, suppose a user types **rm myfile** (which has the effect of removing the file **myfile**). The shell searches the filestore for the file containing the program **rm**, and then requests the kernel, through system calls, to execute the program **rm** on **myfile**.




User program
Library
User level
system call interface
kernel level
File Subsystem
Buffer Cache
character
block
Device driver
Process Control Subsystem
Inter process communication
Scheduler
Memory Management
Hardware control
kernel level
hardware
User level

- When the process **rm myfile** has finished running, the shell then returns the UNIX prompt % to the user, indicating that it is waiting for further commands.
- **Shell** – The shell is the utility that processes your requests. When you type in a command at your terminal, the shell interprets the command and calls the program that you want. The shell uses standard syntax for all commands. C Shell, Bourne Shell and Korn Shell are the most famous shells which are available with most of the Unix variants.
- **Commands and Utilities** – There are various commands and utilities which you can make use of in your day to day activities. **cp**, **mv**, **cat** and **grep**, etc. are few examples of commands and utilities. There are over 250 standard commands plus numerous others provided through 3rd party software. All the commands come along with various options.
- **Files and Directories** – All the data of Unix is organized into files. All files are then organized into directories. These directories are further organized into a tree-like structure called the **filesystem**.

- When you enter a command through the keyboard , the shell thoroughly examines the keyboard input for special characters.
- If it finds it rebuilds a simplified command line and finally communicates with the kernel to see that the command is executed.
- Ex. $ echo Red Hat Linux

  Red Hat Linux
- While processing the shell compress all multiple contiguous spaces in the above command to the single one.

- Computers don’t have the inherent capability of translating commands into action.
- They require an interpreter .
- This job is done by the “outer part” of the operating system – the **shell** .
- It’s the interface between the User & the Kernel .