Lecture – 6

Section-B

Theoretical concept of Unix Operating System

Reference Books: **(1) UNIX- Concepts & Applications by Sumitabha Das**

(2) Principles of Operating System- Galvin

Introduction

- Basic concepts of Operating System
- Features of OS
- Architecture of UNIX Operating System
- Features of UNIX Operating System

Basic Concepts of Operating System

- An operating system is a program that manages computer hardware; An operating system is an important part of almost every computer system.
- A computer system can be divided into four components :

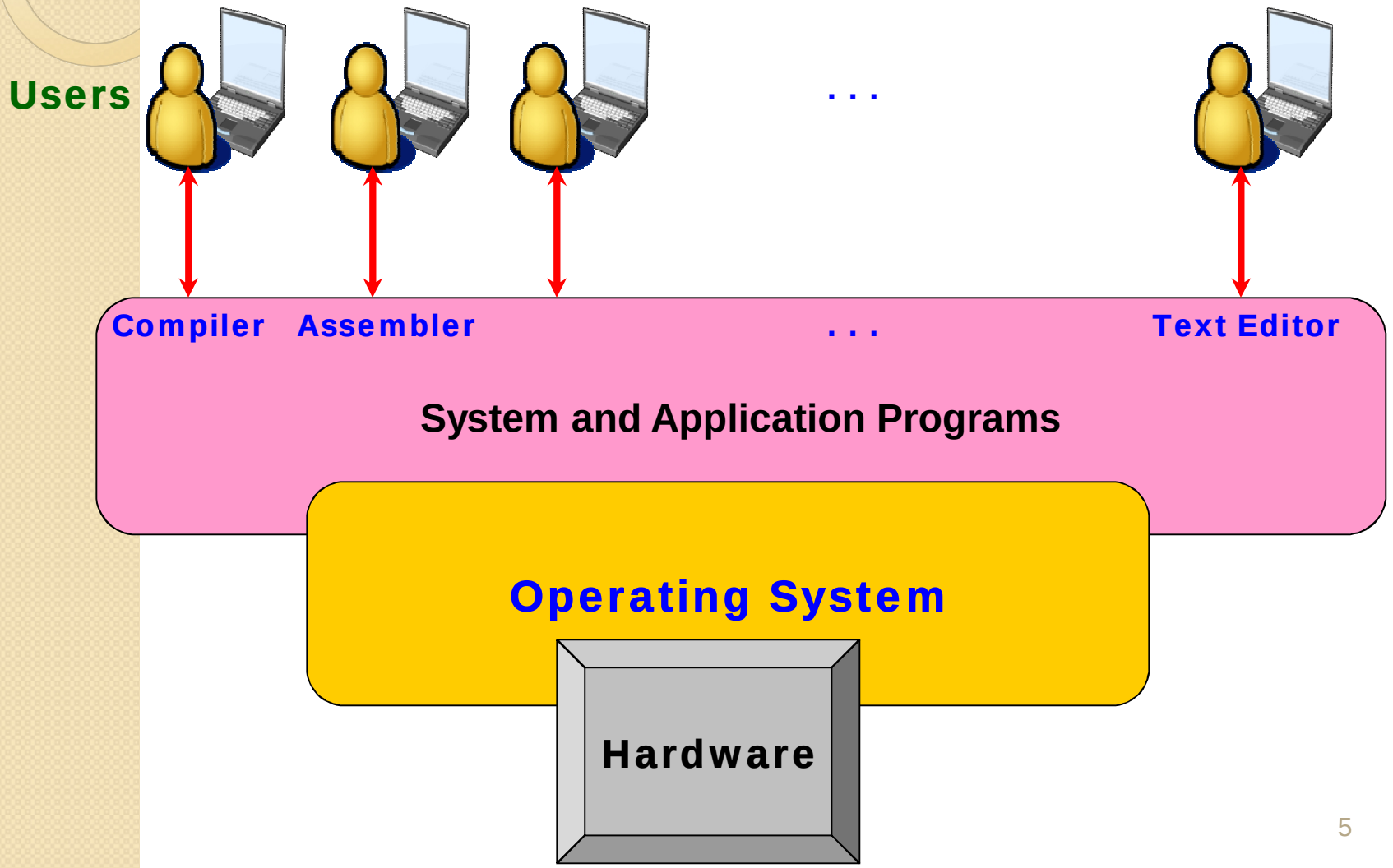
(A) The Hardware

(B) The Operating System

(C) The Application Programs

(D) And The Users

Computer System: Abstract View



- The “hardware” – the central processing unit(CPU), the memory, Input / Output devices provides the basic computing resources.

- The “application programs” – such as word processors, spreadsheets, compilers and assemblers, web-browsers – ***define the way in which these resources are utilized to solve computing problems of the user.***

- The “ operating system” – controls & coordinates the use of the hardware among the various application programs for the various users.

Features of an Operating System:

- **Process Management**:

A program does nothing unless its instructions are executed by CPU. A process is a program in execution.

The operating system is responsible for the following activities in connection with process Management:

- **Creating and deleting both user & system processes**
- **Suspending & resuming processes.**

- **Main Memory management:**

The operating system is responsible for the following activities in connection with memory management :

- **Keeping track of which part of memory are currently being used & by whom.**
- **Deciding which processes are to be loaded into memory when memory space become available.**

File Management :

A file is a collection of related information defined by its creator. Commonly, files represents programs (both source & object forms) & data.

Data files maybe numeric, alphabetic or alphanumeric.

Operating system is responsible for following activities in connection with file management.

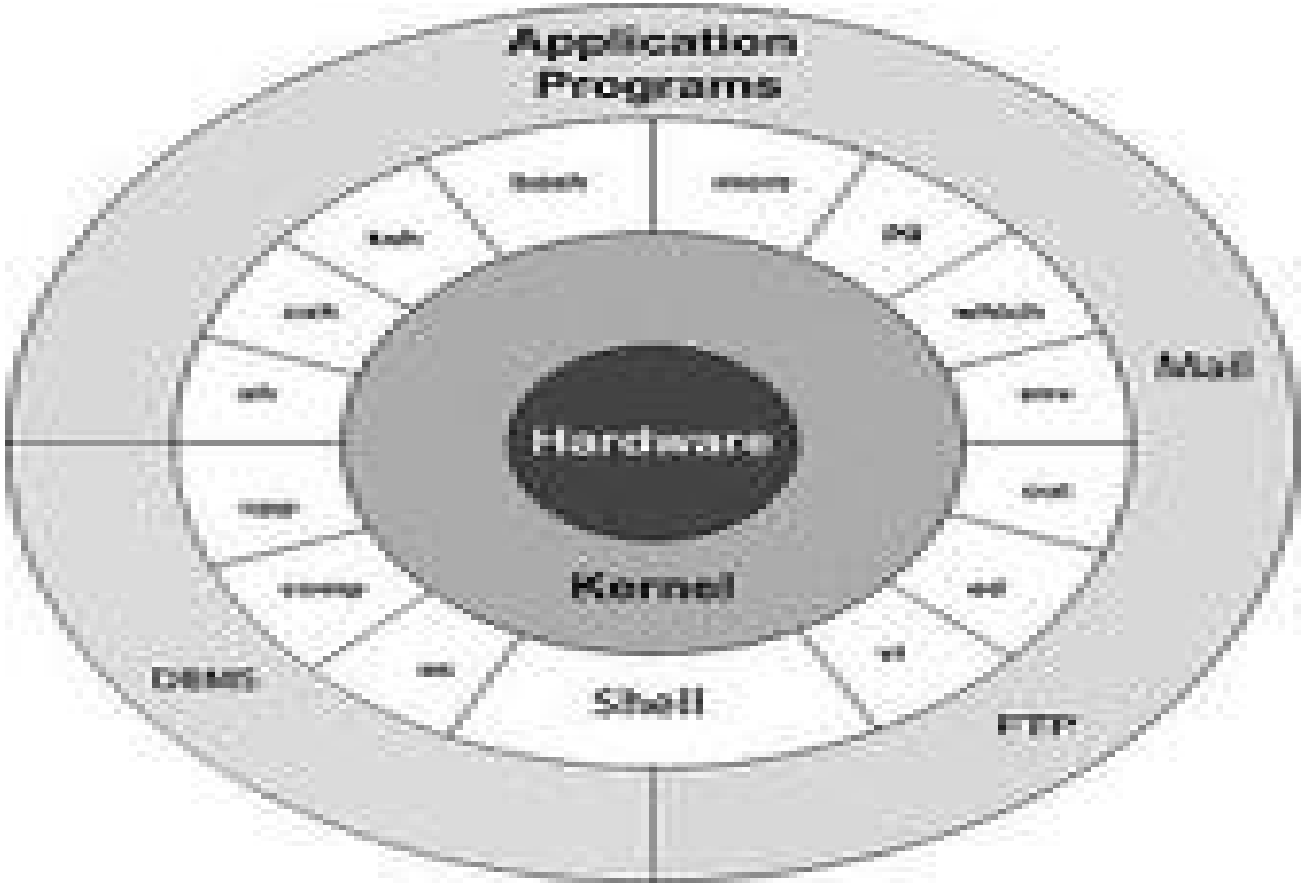
- **Creating & deleting files**
- **Creating & deleting directories.**

Secondary-Storage Management:

The operating system is responsible for the following activities in connection with disk management:

- Free space management
- Storage allocation
- Disk scheduling.

The Unix Architecture



Pl. Note : Its just a sample architecture: For complete diagram, kindly refer the book… **Sumitabha Das** for better understanding of the architecture. During exams I'd be expecting you people to design the architecture available in your book.

- The “kernel”– is the core of the operating system – its collection of routines mostly written in C.
- **It is loaded into memory when the system is booted and communicates directly with the hardware.**
- User Programs(the applications) that need to access the hardware (like the hard-disk or terminal) use the services of the kernel, which performs the job on user’s behalf.

- **General task performed by Kernel**

– manages the system's memory.

– schedules processes

– decide their priorities

- Computers don't have any inherent capability of translating commands into actions. That requires a "command interpreter" a job that is handled by the "outer part" of operating – the shell.
- **Shell :**It is actually the interface between the user & kernel.
- Even though there's only one kernel running on the system, there could be several shells in action – one for each user who is logged in.
- When you enter a command through the keyboard, the shell thoroughly examines the keyboard input for special characters.
- If it finds any, it rebuilds a command line & finally communicates with the kernel to see that the command is executed.

Example:

- Unix allows us to customize the prompt by using following – "$,#, %" -- all these are Unix prompt.

Features of Unix Operating System:

Unix – A Multi-user System:

Unix is a multi-programming system; it permits multiple programs to run & compete for the attention of the CPU.

This can happen in two ways:

1. Multiple users can run separate jobs
2. A single user can also run multiple jobs.

- In UNIX, the resources are actually shared between all users ; UNIX is also a multi-user system.
- For creating an Illusionary effect, the computers breaks up a unit of time into several segments, and each user is allotted a segment. So at any point in time, the machine will be doing the job of a single user.
- The moment the allocated time expires, the previous job is kept on hold & the next job is taken up.
- The process goes on until the clock has turned full circle & the first user's job is taken up once again.

UNIX – A Multi-tasking system too:

- A single user can also run multiple tasks concurrently; Unix is a multi-tasking system.
- It is useful for a user to edit a file, print another file on the printer, send email to a friend & browse the world wide web – all without leaving any of the applications.
- Kernel is designed to handle user's multiple needs.

Programming Facility:

- The Unix shell is also a programming language; it was designed for a programmer; not a casual end user.
- It has all the necessary ingredients like control structure, loops, variables, that establish.
- It is a powerful programming language in its own right.

Pattern Matching :

- $ ls

  READ ME

  chap0

  chap1

  chap2

  chap3

  $ ls chap*

  chap0

  chap1

  chap2

  chap3

- Here we listed the chapters of the text by using “ls” command with an unusual argument (chap*) instead of explicitly specifying all file names.
- The * (known as metacharacter) isn’t the only character used by the Unix operating system; there are several others.
- Unix features elaborated pattern matching schemes that use several characters from this meta character set.

Some APPLICATIONS of Unix O.S.

- It is written in high-level language, ‘C’ making it easy to port to different configurations.
- It is a good operating system, especially, for programs. UNIX programming environment is unusually rich and productive. It provides features that allow complex programs to be built from simpler programs.
- It uses a hierarchical file system that allow easy maintenance and efficient implementation.
- It uses consistent format for files, the byte stream, making application programs easier to write.
- It hides the machine architecture from the user, making it easier to write programs that run on different hardware implementation.

RESEARCH

- Each version of the **UNIX** **Time-Sharing System** evolved from the version before, with version one evolving from the prototypal **Unics**. Not all variants and descendants are displayed.

Research Unix

- **Ken's new system** (→**Unix**) (1969)
- **UNIX Time-Sharing System v1** (1971)
- **UNIX Time-Sharing System v2** (1972)
- **UNIX Time-Sharing System v3** (1973)
- **UNIX Time-Sharing System v4** (1973)
- **UNIX Time-Sharing System v5** (1974)
  - **UNSW 01** (1978)
- UNIX Time-Sharing System v6 (1974)
  - **MINI-UNIX** (1977)
  - PWB/UNIX **1.0** (1977)
    - **USG 1.0**
      - CB Unix **1**
- UNIX Time-Sharing System v7 (1979)
  - **Unix** System III (1981)
- UNIX Time-Sharing System v8 (1985)
- UNIX Time-Sharing System v9 (1986)
- UNIX Time-Sharing System v10 (1989)
- After the release of Version 10, the Unix research team at Bell Labs turned its focus to **Plan 9 from Bell Labs**, a distinct operating system that was first released to the public in 1993.

- **AT&T UNIX Systems and descendants**
- Each of the systems in this list is evolved from the version before, with **Unix System III** evolving from both the **UNIX Time-Sharing System v7** and the descendants of the **UNIX Time-Sharing System v6**.
- **Unix** System III (1981)
- **Unix System IV** (1982)
- **Unix** System V (1983)
  - **Unix System V Release 2** (1984)
  - **Unix System V Release 3.0** (1986)
  - **Unix System V Release 3.2** (1987)
  - **Unix System V Release 4** (1988)
  - **Unix System V Release 4.2** (1992)
- UnixWare **1.1** (1993)
  - **UnixWare 1.1.1** (1994)
- **UnixWare 2.0** (1995)
  - **UnixWare 2.1** (1996)
    - **UnixWare 2.1.2** (1996)
- **UnixWare 7 (System V Release 5)** (1998)
  - **UnixWare 7.0.1** (1998)
- **UnixWare 7.1** (1999)
  - **UnixWare 7.1.1** (1999)
  - **UnixWare NSC 7.1**+IP (2000)
  - **UnixWare NSC 7.1**+LKP (2000)
  - **UnixWare NSC 7.1**DCFS (2000)
- Open Unix **8** (UnixWare 7.1.2) (2001)
  - **Open Unix 8MP1** (2001)
  - **Open Unix 8MP2** (2001)
  - **Open Unix 8MP3** (2002)
  - **Open Unix 8MP4** (2002)
- SCO **UnixWare 7.1.3** (2002)
  - **SCO UnixWare 7.1.3 Update Pack 1** (2003)
  - **SCO UnixWare 7.1.4** (2004)