OVERVIEW OF COMPUTER WORKSHOP: OPERATING SYSTEMS

Adapted form Tanenbaum, Modern Operating Systems 3 e, (c) 2008 Prentice-Hall, Inc.

Dr.S.Manipriya & Dr. P.V. Arun

Contents

- What is OS?
- Need of OS
- Goals of OS
- Modes of OS
- Functions of OS
- History of OS
- Types of OS

What Is An Operating System?

Lots of hardware !!

- One or more processors
- Main memory
- Disks
- Printers
- Various input/output devices

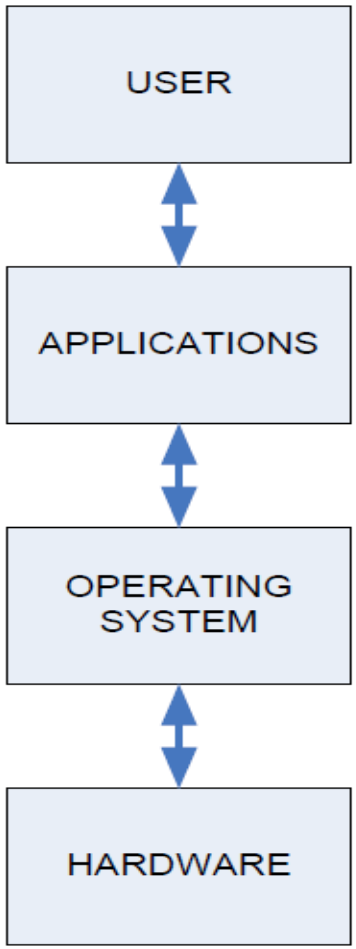
Managing all these components requires a layer of software – the **operating system**

What Is An Operating System?

An Operating System (OS) is a program that:

- Manages computer hardware
- Provides a basis for application programs
- Act as an intermediary between computer user and computer hardware

Computer System Structure

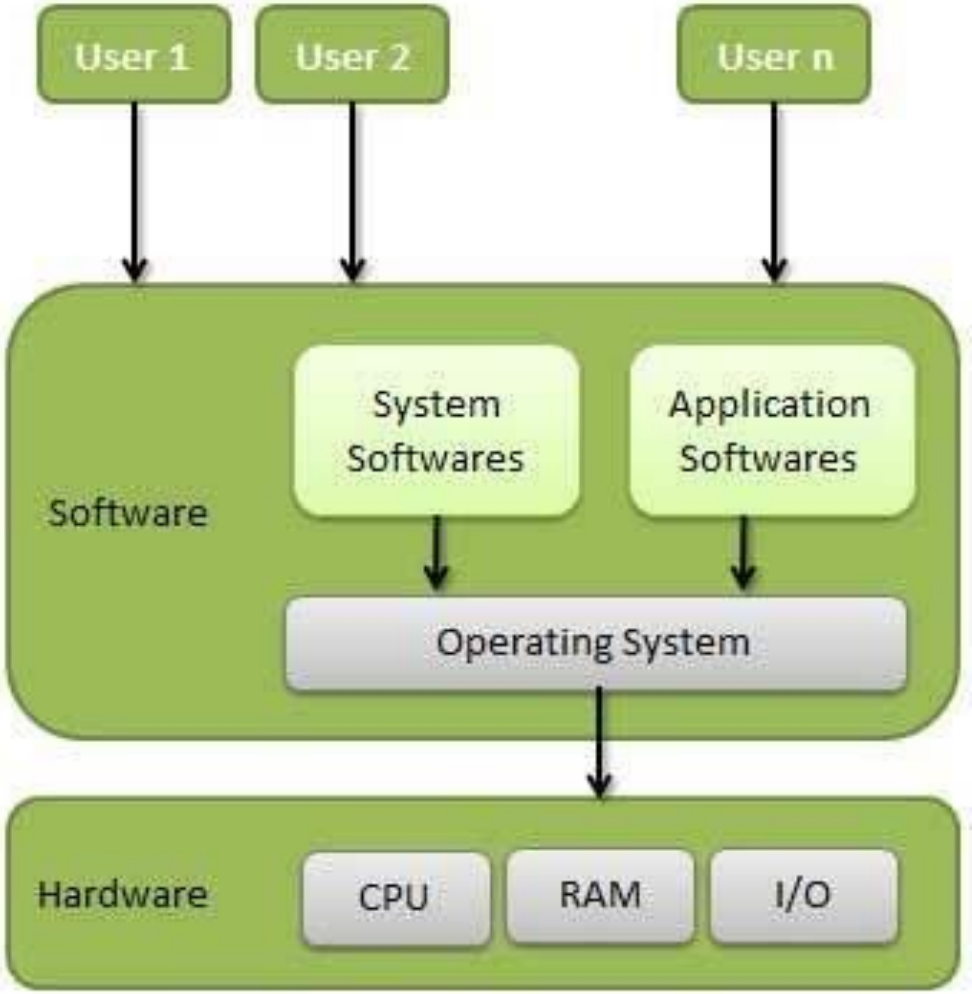


Computer System Structure

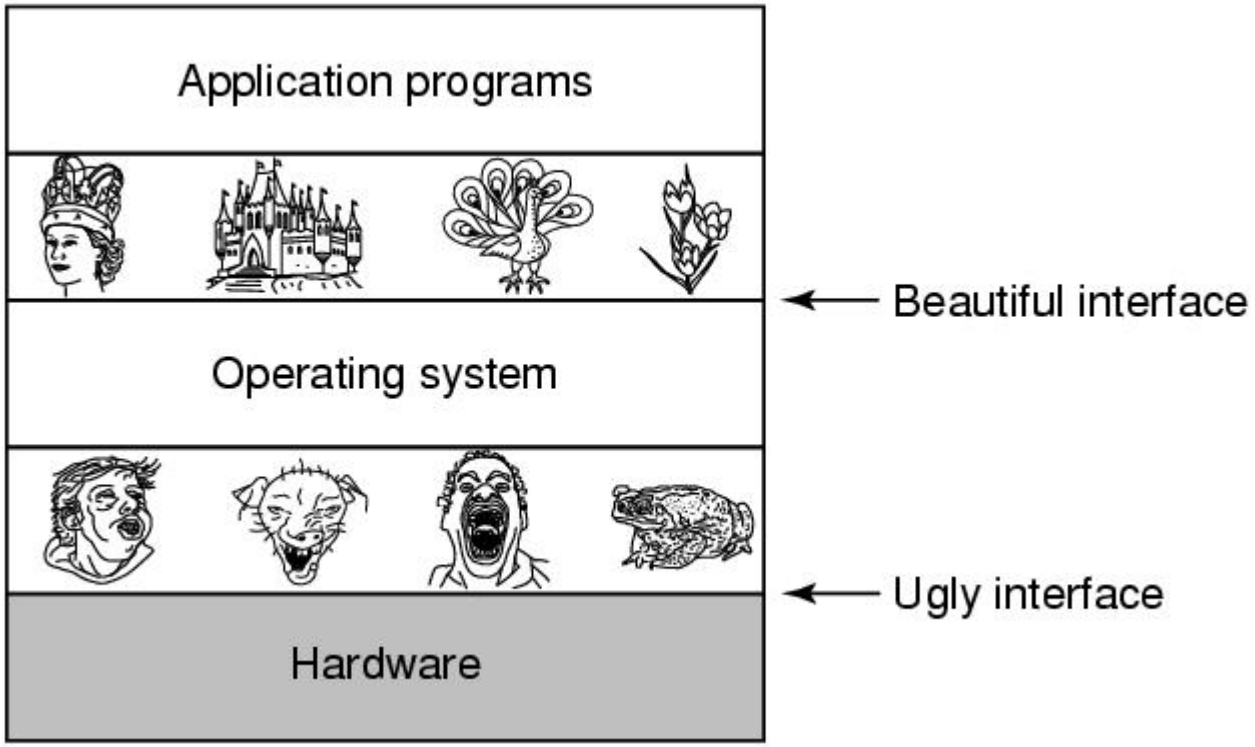
Computer system can be divided into four components:

- **Hardware** – provides basic computing resources
  CPU, memory, I/O devices
- **Operating system**
  Controls and coordinates use of hardware among various applications and users
- **Application programs** – define the ways in which the system resources are used to solve the computing problems of the users
  - Word processors, compilers, web browsers, database systems, video games
- **Users**
  People, machines, other computers

Operating System as an Interface



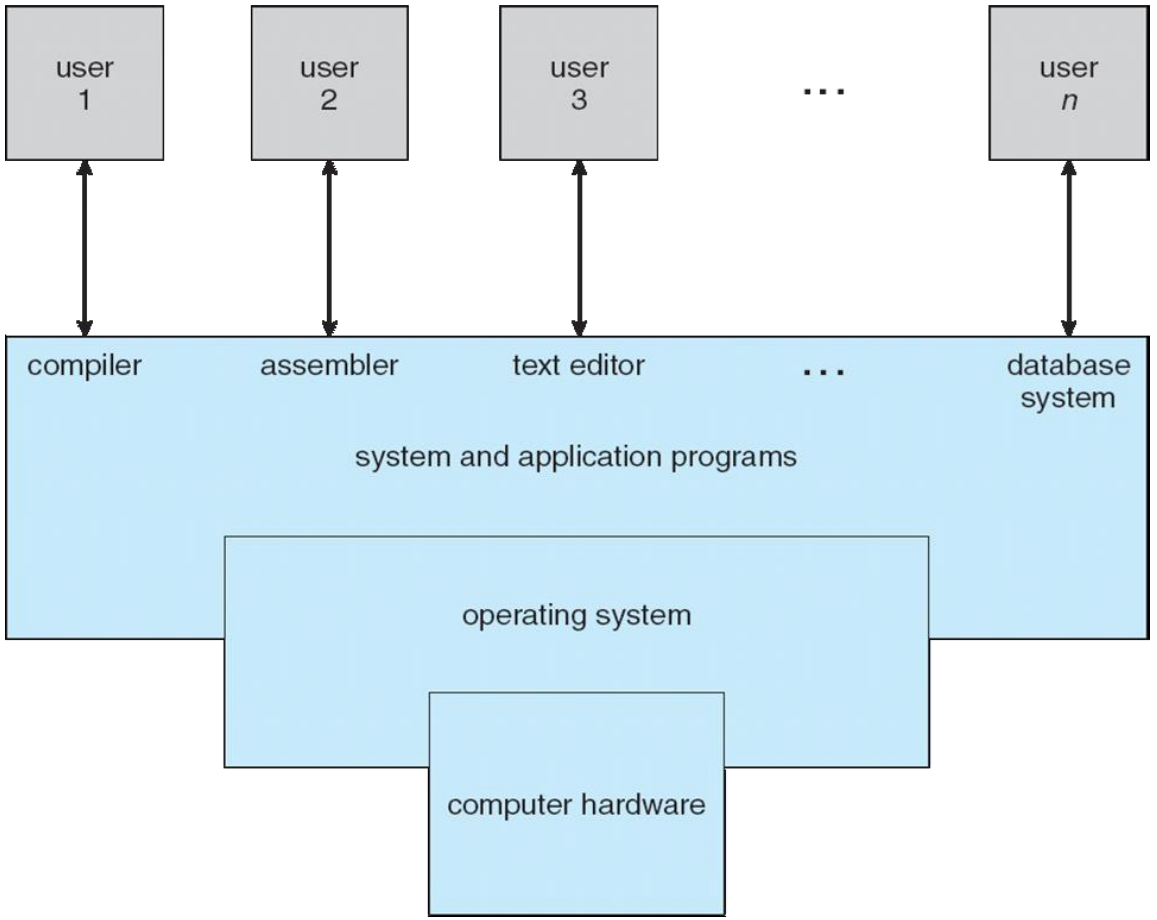
The Operating System as an Extended Machine



The Operating System as a Resource Manager

- Allow multiple programs to run at the same time
- Manage and protect memory, I/O devices, and other resources
- Multiplexes (shares) resources in two different ways:
  - In time
  - In space

The Operating System as a Resource Manager



Operating Systems

- Software for overall operation of computer
- User can store and retrieve files
- Request the execution of programs, and provides the environment necessary to execute the programs requested

Goals of An Operating System

- Operating system goals:
  - Execute user programs and make solving user problems easier
  - Make the computer system convenient to use
  - Use the computer hardware in an efficient manner

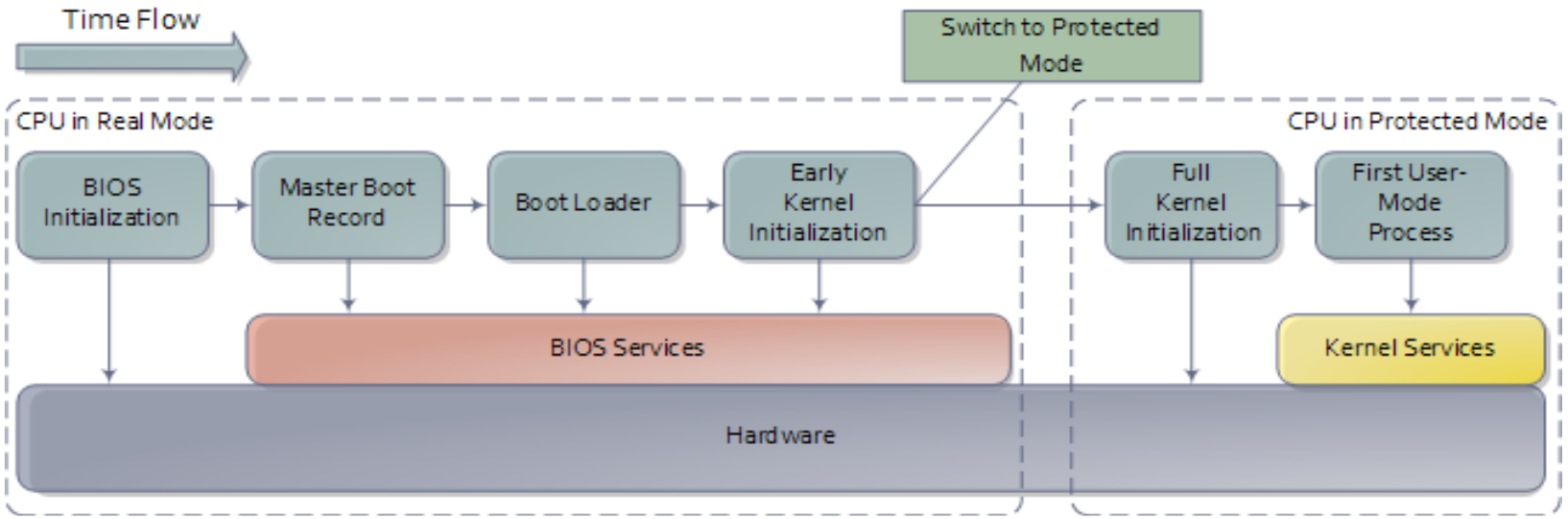
Operating System & CPU

- CPU is the brain of your computer.
- It performs all of the calculations.
- In order to do its job, the CPU needs commands to perform, and data to work with.
- The instructions and data travel to and from the CPU on the system bus.
- The operating system provides rules for how that information gets back and forth, and how it will be used by the CPU.

Operating System & RAM

- This is like a desk, or a workspace, where your computer temporarily stores all of the information (data) and instructions (software or program code) that it is currently using.
- Each RAM chip contains millions of address spaces.
- Each address space is the same size, and has its own unique identifying number (address).
- The operating system provides the rules for using these memory spaces, and controls storage and retrieval of information from RAM.
- Device drivers for RAM chips are included with the operating system.

Starting an Operating System(Booting)



- ✓ Power On Switch sends electricity to the motherboard on a wire called the *Voltage Good* line.
- ✓ If the power supply is good, then the BIOS (Basic Input/Output System) chip takes over.
- ✓ In Real Mode, CPU is only capable of using approximately 1 MB of memory built into the motherboard.
- ✓ The BIOS will do a Power-On Self Test (POST) to make sure that all hardware are working.

- ✓ BIOS will then look for a small sector at the very beginning of your primary hard disk called MBR.
- ✓ The MBR contains a list, or map, of all of the partitions on your computer‘s hard disk (or disks).
- ✓ After the MBR is found the Bootstrap Loader follows basic instructions for starting up the rest of the computer, including the operating system.
- ✓ In Early Kernel Initialization stage, a smaller core of the Kernel is activated.
- ✓ This core includes the device drivers needed to use computer‘s RAM chips.

BIOS

- BIOS firmware is stored in a ROM/EPROM (Erasable Programmable Read-Only Memory) chip known as **firmware** on the PC motherboard.
- BIOS can be accessed during the initial phases of the boot procedure by pressing del, F2 or F10.
- Finally, the firmware code cycles through all storage devices and looks for a boot-loader. (usually located in first sector of a disk which is 512 bytes)
- If the boot-loader is found, then the firmware hands over control of the computer to it.

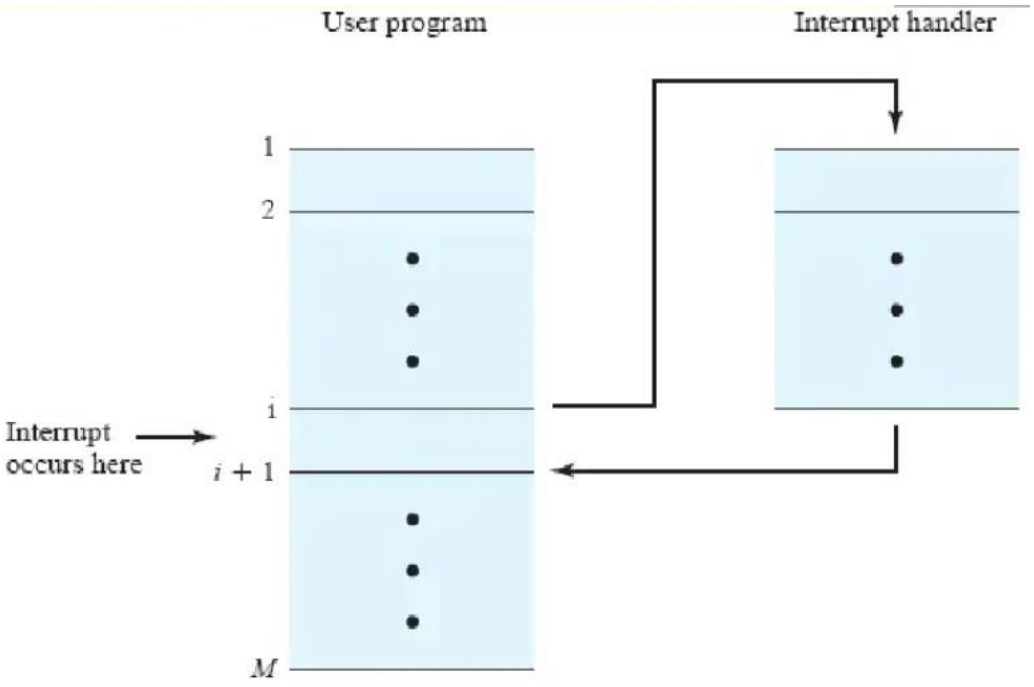
Interrupts

- Interrupt transfers control to the interrupt service routine generally, through the **interrupt vector**, which contains the addresses of the service routines
- Hardware can interrupt the CPU
- Software interrupts are called system calls
- Interrupt architecture must save the address of the interrupted instruction
- Once the interrupt is handled, CPU may continue its current operation

OS & Interrupt Handling

- An operating system is **interrupt driven**
- The operating system preserves the state of the CPU by storing registers and the program counter
- Separate segments of code determine what action should be taken for each type of interrupt

Interrupt Handling



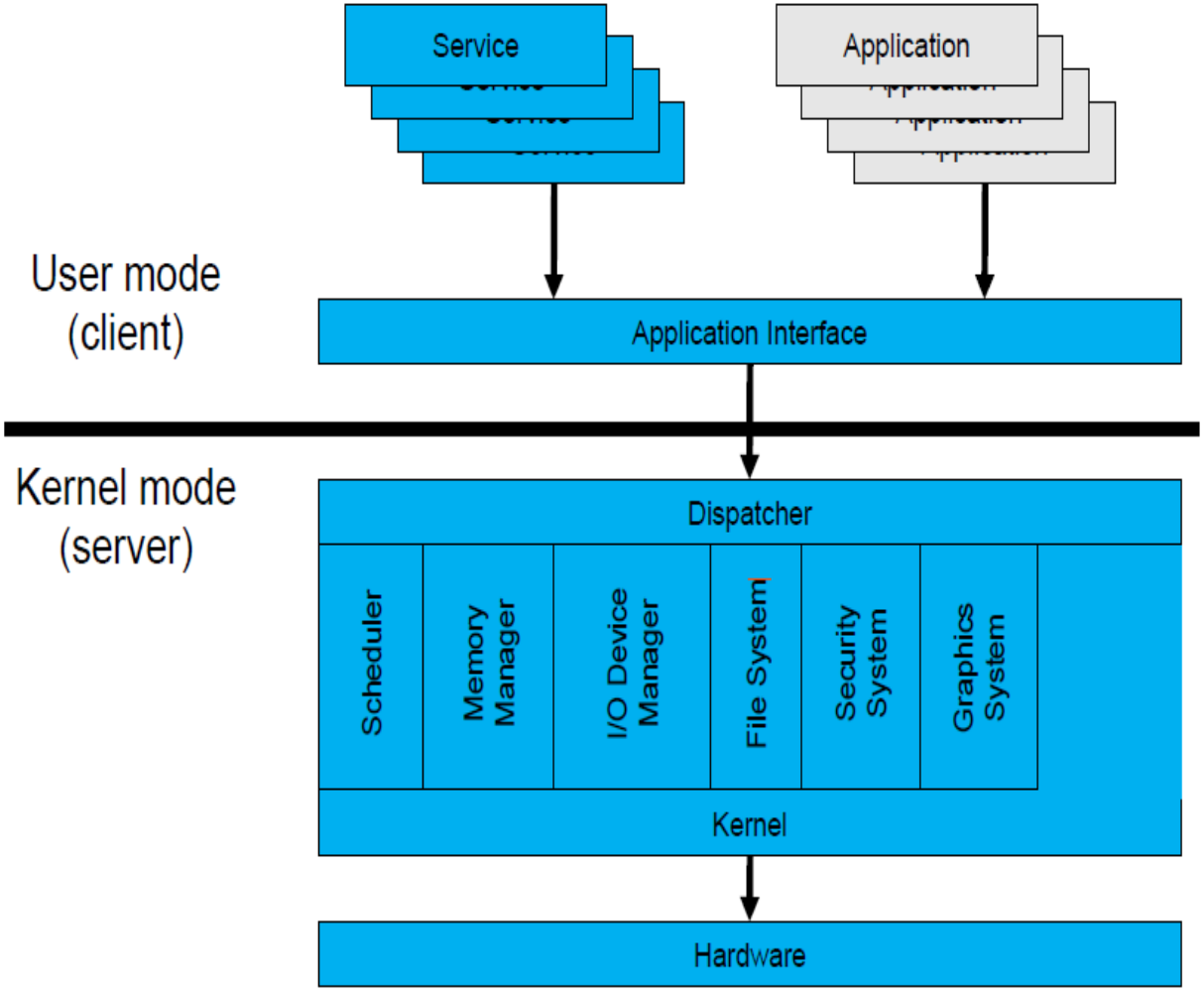
Different Modes of Operating System

- The *User Mode* is concerned with the actual interface between the user and the system.
- It controls things like running applications and accessing files.

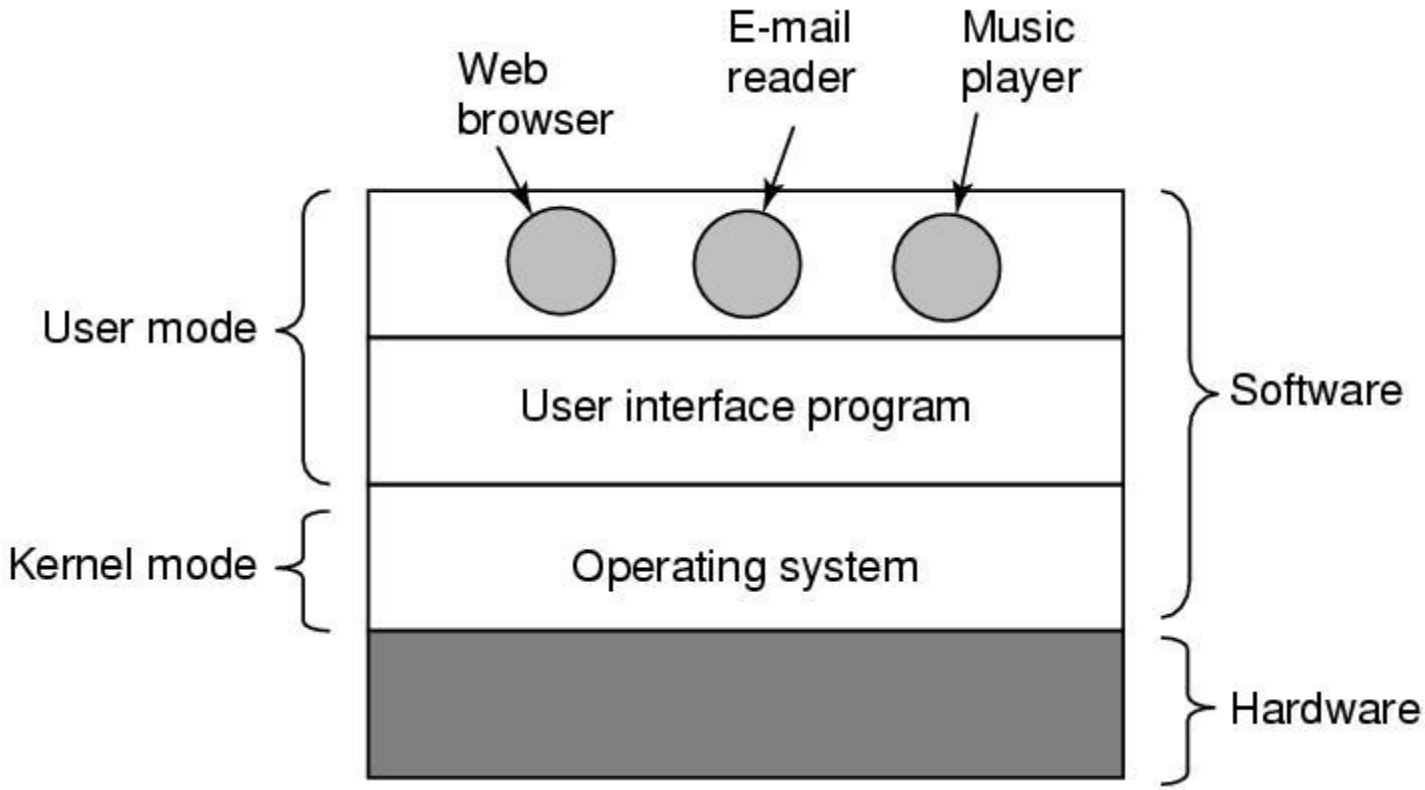
Different Modes of Operating System

- The *Kernel Mode* is concerned with everything running in the background.
- It controls things like accessing system resources, controlling hardware functions and processing program instructions.
- System calls are used to change mode from User to Kernel.

Different Modes of Operating System



Different Modes of Operating System



Where the operating system fits in.

System Call

- **System call** is the programmatic way in which a computer program/user application requests a service from the kernel of the operating system on which it is executed.
- Application program is just a user-process. Due to security reasons, user applications are not given access to privileged resources(the ones controlled by OS).
- When they need to **do any I/O** or have **some more memory** or **spawn a process** or wait for **signal/interrupt,** it requests operating system to facilitate all these. This **request is made through System Call.**
- System calls are also called **software-interrupts.**

System Calls

- Interface between user programs and OS
- Varies from OS to OS
- System call issued by user program
- Call uses a library call of the same name
- Library routine puts machine into kernel modes (by issuing a special instruction)
- Finds actual routine for system call in a table
- Does the work involved in the call
- Returns to user program

Kernel

- Kernel is a software code that reside in central core of OS. It has complete control over system.
- When operation system boots, kernel is first part of OS to load in main memory.
- Kernel remains in main memory for entire duration of computer session. The kernel code is usually loaded in to protected area of memory.
- Kernel performs it's task like executing processes and handling interrupts in kernel space.
- User performs it's task in user area of memory.
- This memory separation is made in order to prevent user data and kernel data from interfering with each other.
- Kernel does not interact directly with user, but it interacts using SHELL and other programs and hardware.

Kernel cont…

- Kernel includes:-

  1. Scheduler: It allocates the Kernel's processing time to various processes.

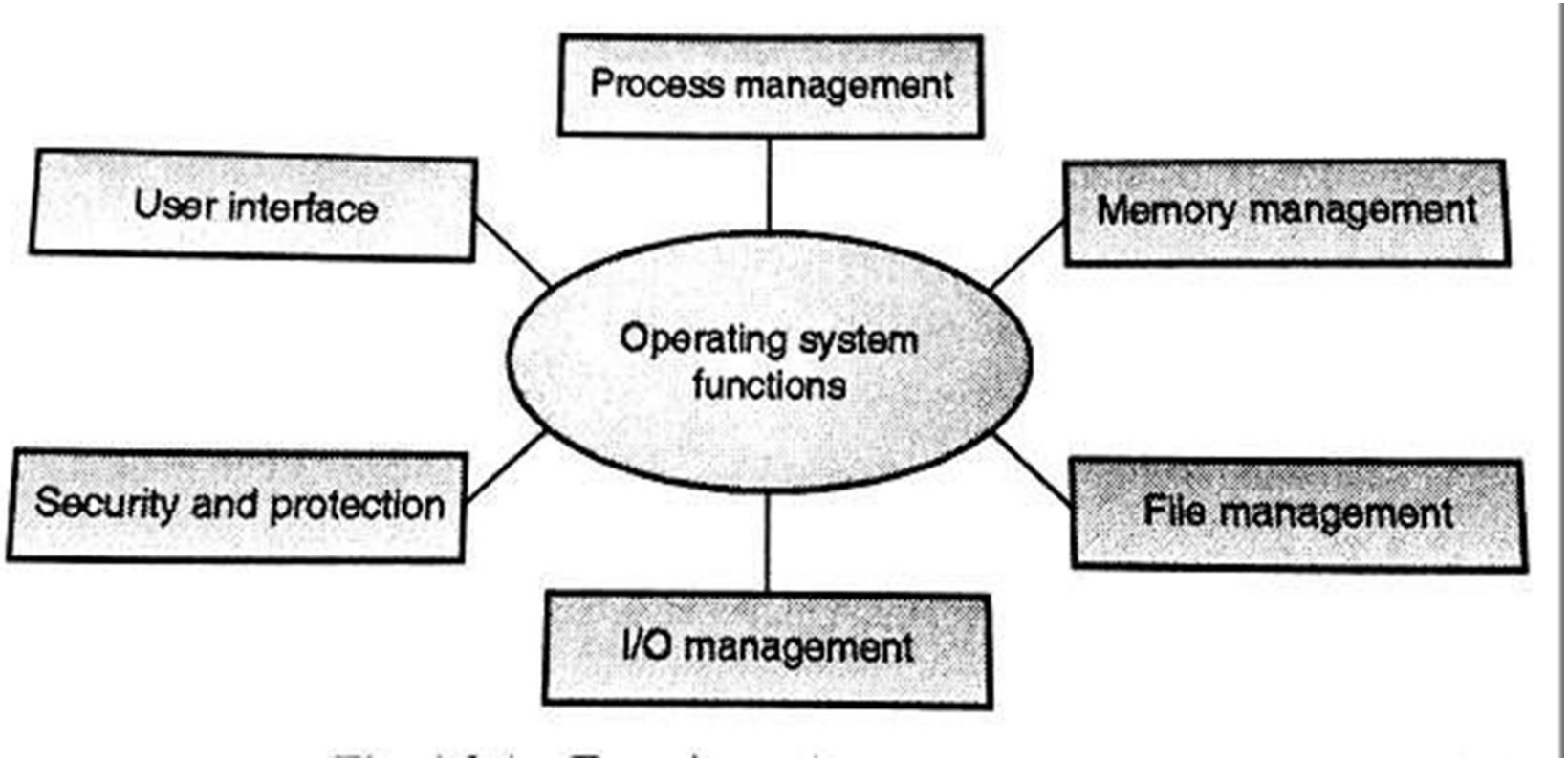
  2. Supervisor: It grants permission to use computer system resources to each process.

  3. Interrupt handler : It handles all requests from the various hardware devices which compete for kernel services.

  4. Memory manager : allocates space in memory for all users of kernel service.

- kernel provides services for process management, file management, I/O management, memory management.
- System calls are used to provide these type of services.

Function of OS



1. Process Management

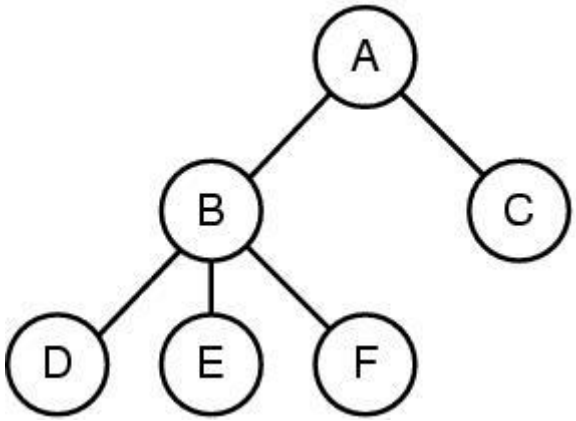
Processes

- Program in execution
- Lives in address space

Process Management (Contd)

A Process Tree

Process creates child processes



Process Management (Contd)

Process

- Can communicate with one another
- Have UID's and group ID's (GID)

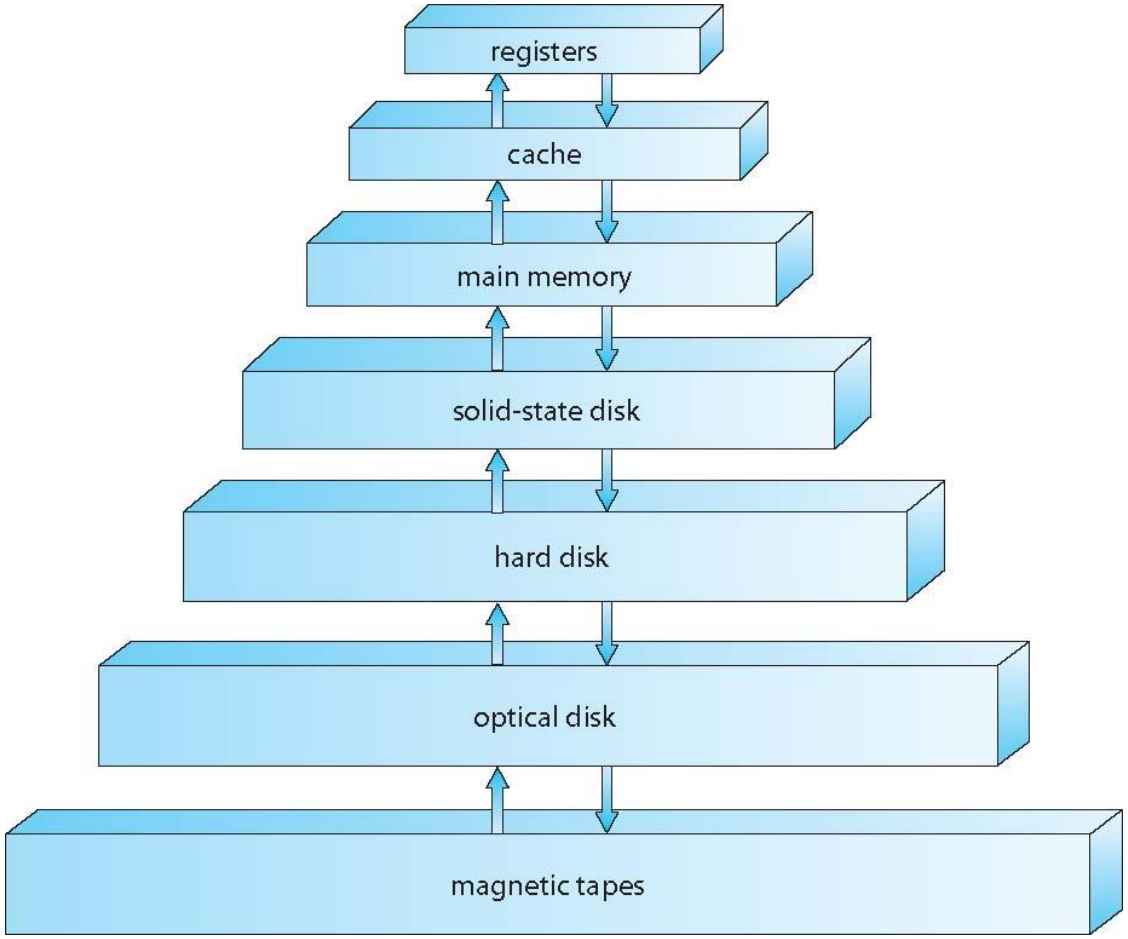
Process Management (Contd)

- A process needs certain resources, including CPU time, memory, files, and I/O devices to accomplish its task.
- Simultaneous execution leads to multiple processes. Hence creation, execution and termination of a process are the most basic functionality of an OS

Process Management (Contd)

- If processes are dependent, then they may try to share same resources. Thus task of process synchronization comes to the picture.
- If processes are independent, then a due care needs to be taken to avoid their overlapping in memory area.
- Based on priority, it is important to allow more important processes to execute first than others.

2. Memory management

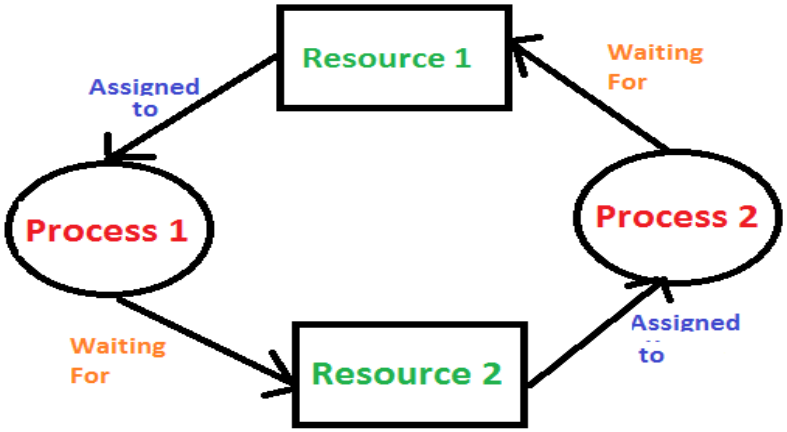


Memory management contd…

- Memory is a large array of words or bytes, each with its own address.
- It is a repository of quickly accessible data shared by the CPU and I/O devices.
- Main memory is a **volatile** storage device. When the computer is turned off everything stored in RAM will be erased automatically.
- In addition to the physical RAM installed in your computer, most modern operating systems allow your computer to use a *virtual memory system. Virtual memory allows your computer to use part of a permanent storage device (such as a hard disk) as extra memory.*
- The operating system is responsible for the following activities in connections with memory management:
  - Keep track of which parts of memory are currently being used and by whom.
  - Decide which processes to load when memory space becomes available.
  - Allocate and de-allocate memory space as needed.

3. Device Management or I/O Management

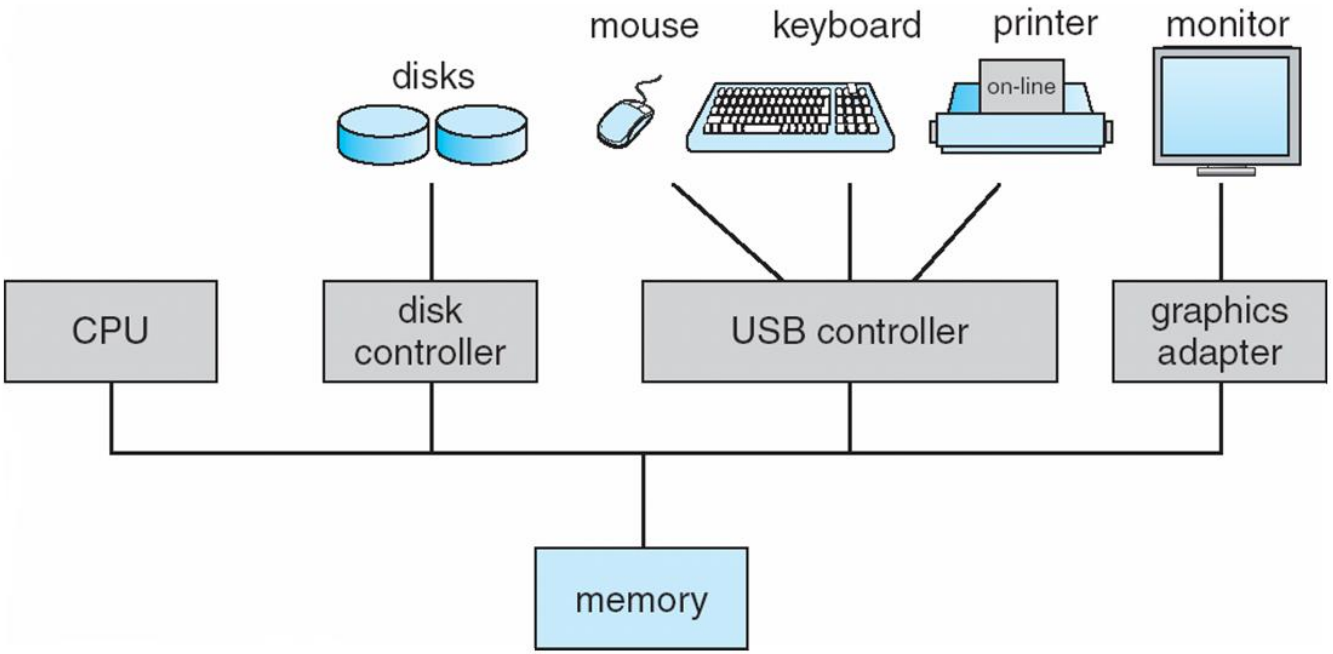
- *Device controllers are components on the motherboard (or on expansion cards) that act as an interface between the CPU and the actual device.*
- *Device drivers, which are the operating system software components that interact with the devices controllers.*
- A special device (inside CPU) called the Interrupt Controller handles the task of receiving interrupt requests and prioritizes them to be forwarded to the processor.
- Deadlocks can occur when two (or more) processes have control of different I/O resources that are needed by the other processes, and they are unwilling to give up control of the device.



Device Management or I/O Management

- OS performs the following activities for device management.
  - Keeps tracks of all devices connected to system.
  - Interacts with the program responsible for every device known as Input/output controller.
  - Decides which process gets access to a certain device and for how long.
  - Allocates devices in an effective and efficient way.
  - Deallocates devices when they are no longer required.

Device Management or I/O Management



Device Management or I/O Management

- Controller runs a device-accepts commands from the OS and executes them
- Complicated business
  - Eg. Gets command to read sector x on disk y. Must convert to (cylinder, sector, head) address, move arm to correct cylinder, wait for sector to rotate under the head, read and store bits coming off the drive, compute checksum, store bits as words in memory
  - Controller contains a computer to run its device

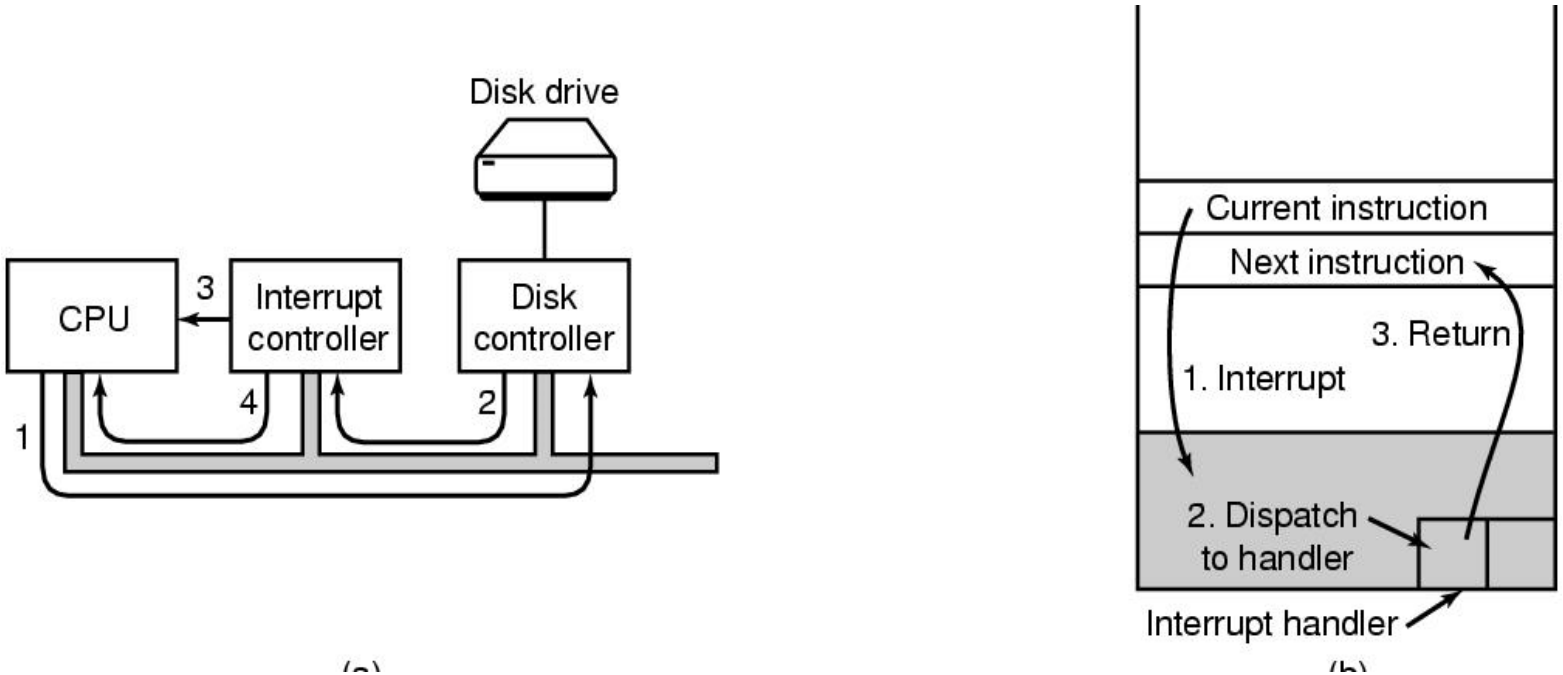
Device Driver

- OS software that talks to controller-gives commands, accepts responses
- Each controller manufacturer supplies a driver for each OS
- Driver runs in kernel mode
- Controller has registers which are used to communicate with the driver
- Three modes of communication
  - Polling
  - Interrupts
  - DMA

I/O by polling device

- Driver issues command to controller
- Driver polls device until it is ready
- E.g. Send character to printer controller and poll until it is ready to accept the next character
- Big use of CPU
- Called programmed I/O-not really used any more

I/O by Interrupts



Generate an interrupt when I/O is finished.

Eg When character is finished being printed, interrupt CPU. Allows CPU to do something else while character is being printed

I/O by DMA

- Special (controller) chip
- Avoids using the CPU as part of the transfer to/from memory
- CPU tells chip to set up transfer and take care of it
- Chip does as it is told and interrupts CPU when it is finished

4. File Management

- A file is a collection of related information defined by its creator.
- *File systems provide the conventions for the encoding, storage and management of data on a storage device such as a hard disk.*
  - FAT12 (floppy disks)
  - FAT16 (DOS and older versions of Windows)
  - FAT32 (older versions of Windows)
  - NTFS (newer versions of Windows)
  - EXT3 (Unix/Linux)
  - HFS+ (Max OS X)

File Management

- The operating system is responsible for the following activities in connections with file management:
  - ✦ File creation and deletion.
  - ✦ Directory creation and deletion.
  - ✦ Support of primitives for manipulating files and directories.
  - ✦ Mapping files onto secondary storage.
  - ✦ File backup on stable (nonvolatile) storage media.

5. Security & Protection

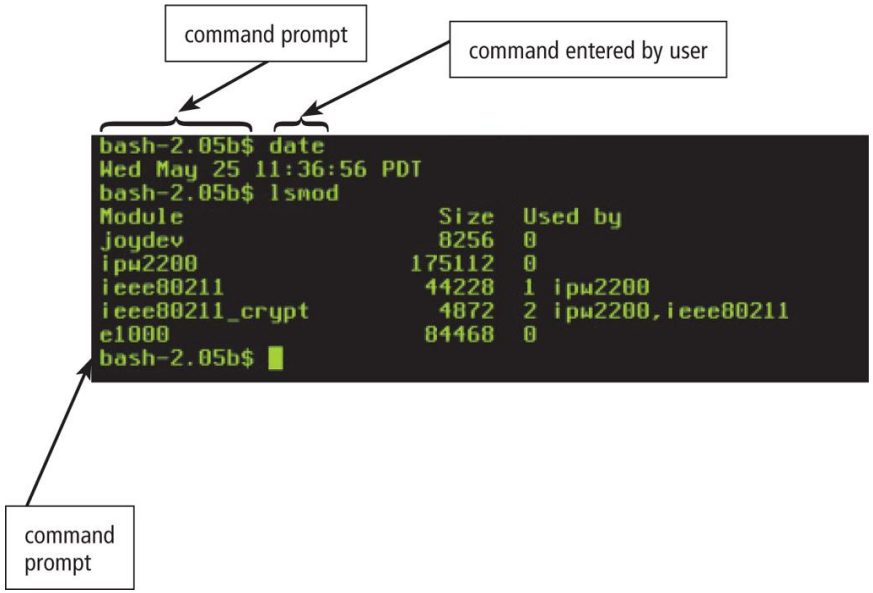
- The operating system uses password protection to protect user data and similar other techniques.
- It also prevents unauthorized access to programs and user data by assigning access right permission to files and directories.
- The owners of information stored in a multiuser or networked computer system may want to control use of that information, concurrent processes should not interfere with each other.

6. User Interface Mechanism

- A **user interface** (**UI**) controls how you enter data and instructions and how information is displayed on the screen
- There are two types of user interfaces
  1. Command Line Interface
  2. Graphical user Interface

6.1. Command-line interface

- In a command-line interface, a user types commands represented by short keywords or abbreviations or presses special keys on the keyboard to enter data and instructions



6.2. Graphical User Interface

- With a graphical user interface (GUI), you interact with menus and visual images

History of Operating System

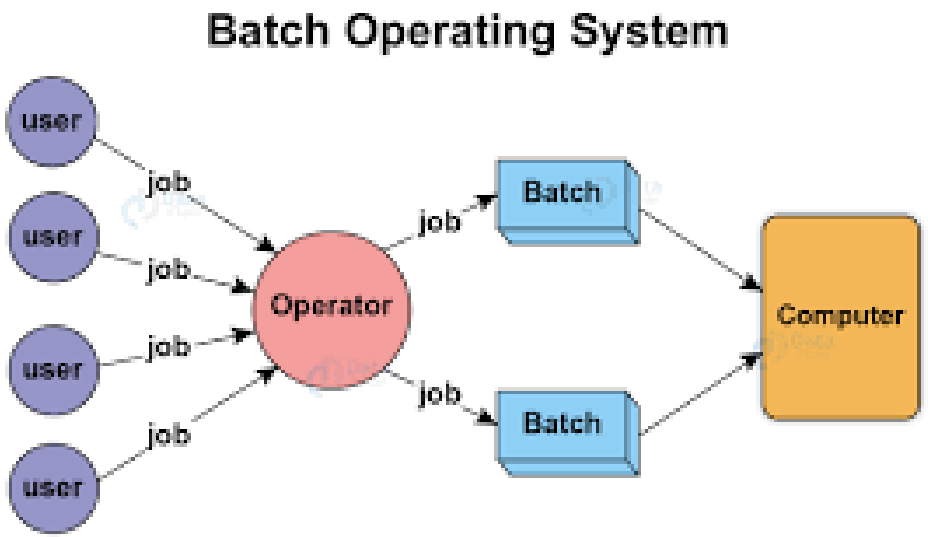
- **The First Generation (1940's to early 1950's)**
  - No Operating System
  - All programming was done in absolute machine language, often by wiring up plug-boards to control the machine's basic functions.
- **The Second Generation (1955-1965)**
  - First operating system was introduced in the early 1950's.It was called GMOS
  - Created by General Motors for IBM's machine the 701.
  - Single-stream batch processing systems
- **The Third Generation (1965-1980)**
  - Introduction of multiprogramming
  - Development of Minicomputer
- **The Fourth Generation (1980-Present Day)**
  - Development of PCs
  - Birth of Windows/MaC OS

Types of Operating Systems

1. Batch Operating System
2. Multiprogramming Operating System
3. Time-Sharing OS
4. Multiprocessing OS
5. Distributed OS
6. Network OS
7. Real Time OS
8. Embedded OS

1. Batch Operating System

- The users of this type of operating system does not interact with the computer directly.
- Each user prepares his job on an off-line device like punch cards and submits it to the computer operator
- There is an operator which takes similar jobs having the same requirement and group them into batches.



1. Batch Operating System cont..

Advantages of Batch Operating System:

- Processors of the batch systems know how long the job would be when it is in queue
- Multiple users can share the batch systems
- The idle time for the batch system is very less
- It is easy to manage large work repeatedly in batch systems

Disadvantages of Batch Operating System:

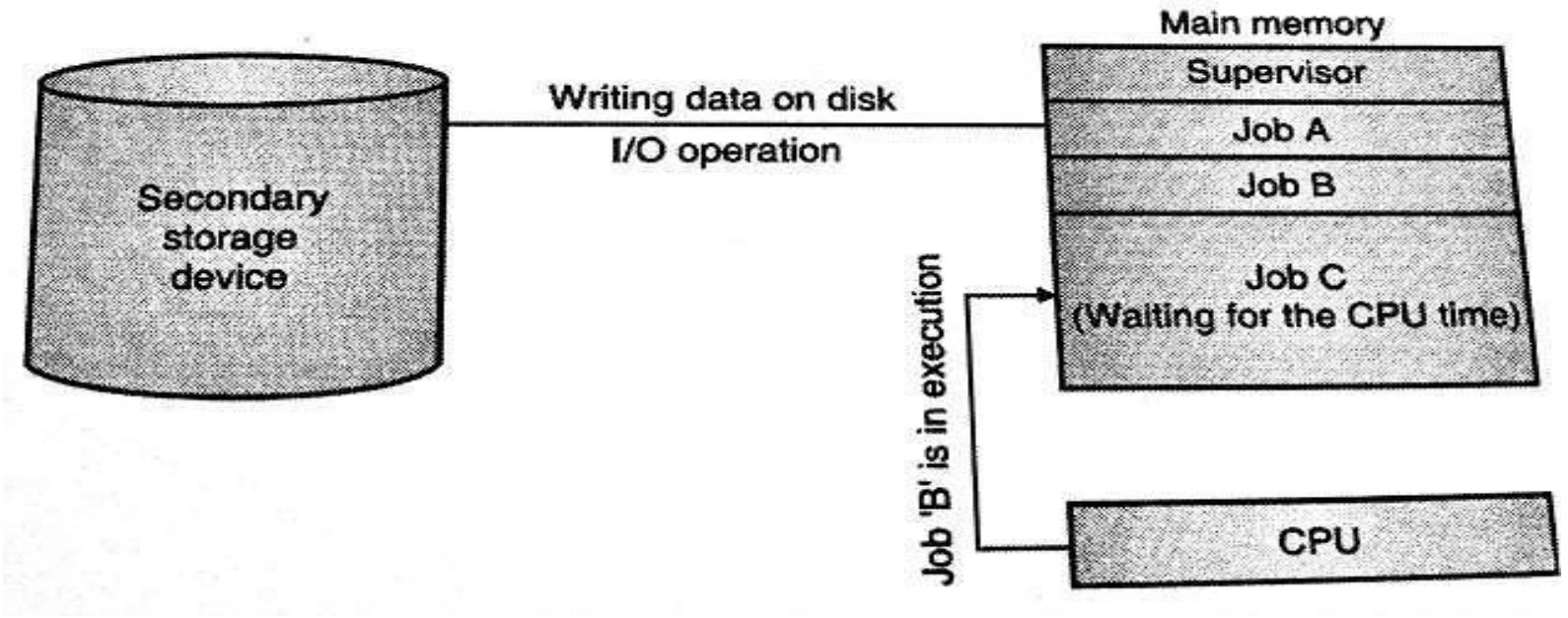
- The computer operators should be well known with batch systems
- Batch systems are hard to debug
- It is sometimes costly
- The other jobs will have to wait for an unknown time if any job fails

Examples of Batch based Operating System:

IBM's MVS

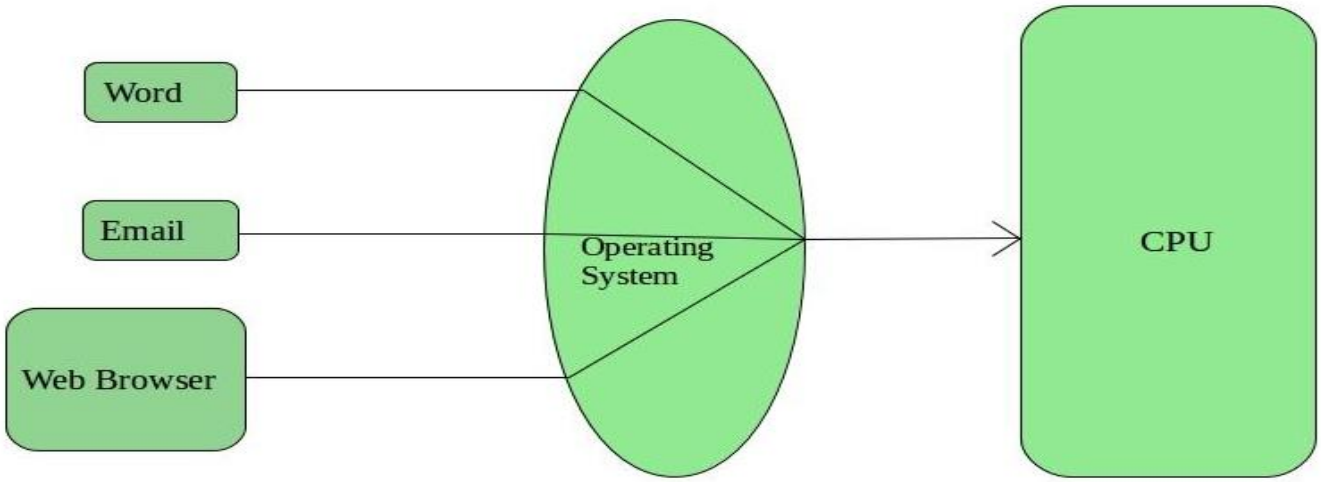
2. Multiprogramming Operating System:

- This type of OS is used to execute more than one jobs simultaneously by a single processor.
- It increases CPU utilization by organizing jobs so that the CPU always has one job to execute.
- Multiprogramming operating systems use the mechanism of job scheduling and CPU scheduling.



3. Time-Sharing Operating Systems

- Each task is given some time to execute so that all the tasks work smoothly.
- These systems are also known as **Multi-tasking Systems.**
- The task can be from a single user or different users also.
- The time that each task gets to execute is called quantum.
- After this time interval is over OS switches over to the next task.



3. Time-Sharing Operating Systems cont..

- **Advantages of Time-Sharing OS:**
  - Each task gets an equal opportunity
  - Fewer chances of duplication of software
  - CPU idle time can be reduced
- **Disadvantages of Time-Sharing OS:**
  - Reliability problem
  - One must have to take care of the security and integrity of user programs and data
  - Data communication problem
- **Examples of Time-Sharing Oss**

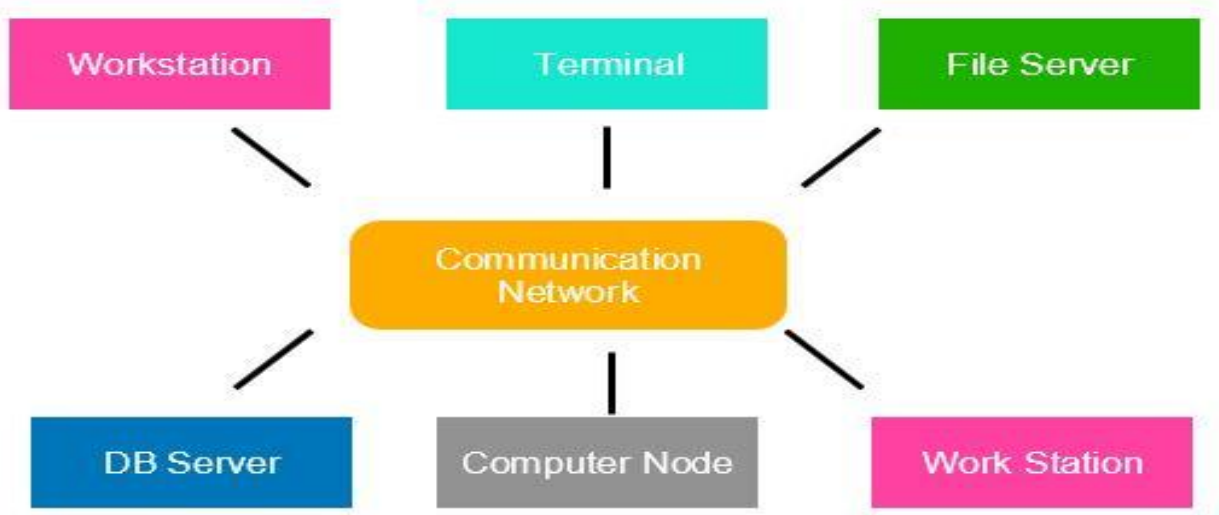
  Multics, Unix, etc.

4. Multiprocessor operating systems

- Multiprocessor operating systems are also known as parallel OS or tightly coupled OS.
- Such operating systems have more than one processor in close communication that sharing the computer bus, the clock and sometimes memory and peripheral devices.
- It executes multiple jobs at the same time and makes the processing faster.
- It supports large physical address space and larger virtual address space.
- If one processor fails then other processor should retrieve the interrupted process state so execution of process can continue.
- Inter-processes communication mechanism is provided and implemented in hardware.

5. Distributed Operating System

- Various autonomous interconnected computers communicate with each other using a shared communication network.
- Independent systems possess their own memory unit and CPU.
- These are referred to as **loosely coupled systems.**
- Examples:- Locus, DYSEAC



6. Network Operating System

- These systems run on a server and provide the capability to manage data, users, groups, security, applications, and other networking functions.
- These types of operating systems allow shared access of files, printers, security, applications, and other networking functions over a small private network.
- The “ other" computers arc called client computers, and each computer that connects to a network server must be running client software designed to request a specific service.
- popularly known as **tightly coupled systems.**

6. Network Operating System

Advantages of Network Operating System:

- Highly stable centralized servers
- Security concerns are handled through servers
- New technologies and hardware up-gradation are easily integrated into the system
- Server access is possible remotely from different locations and types of systems

Disadvantages of Network Operating System:

- Servers are costly
- User has to depend on a central location for most operations
- Maintenance and updates are required regularly

Examples of Network Operating System are:

Microsoft Windows Server 2003/2008/2012, UNIX, Linux, Mac OS X, Novell NetWare, and BSD, etc.

7. Real-Time Operating System

- These types of OSs serve real-time systems.
- The time interval required to process and respond to inputs is very small.
- This time interval is called **response time**.
- **Real-time systems** are used when there are time requirements that are very strict like
  - missile systems,
  - air traffic control systems,
  - robots, etc.

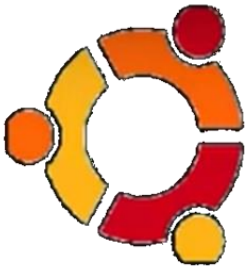
8. Embedded Operating System

- An embedded operating system is one that is built into the circuitry of an electronic device.
- Embedded operating systems are now found in automobiles, bar-code scanners, cell phones, medical equipment, and personal digital assistants.
- The most popular embedded operating systems for consumer products, such as PDAs, include the following:
  - Windows XP Embedded
  - Windows CE .NET:- it supports wireless communications, multimedia and Web browsing. It also allows for the use of smaller versions of Microsoft Word, Excel, and Outlook.
  - Palm OS:- It is the standard operating system for Palm-brand PDAs as well as other proprietary handheld devices.
  - Symbian:- OS found in " smart" cell phones from Nokia and Sony Ericsson

Popular Operating Systems

Windows

Linux



Ubuntu

Mac OS X
iOS

Android

VIRTUAL MACHINE AND VIRTUALBOX

Adapted from the slides of Dr. Cliff Zou, University of Central Florida

Virtualization Software

- Runs operating systems in fully emulated environment
  - Vmware (Vmware Inc.)
  - VirtualBox (Oracle)
  - Virtual PC (Microsoft)
  - Xen (open source project)

Virtualization Terminology

- ***Host OS*** – running on physical computer
  - Only one host OS may run at a time
  - "Hosts" the other running operating systems
- ***Guest OS –*** running in emulated environment
  - Can run multiple guests at the same time
  - Guest ***thinks*** it is running on actual hardware
- ***Virtual machine*** – set of files that make up a guest OS

Virtual Machine Advantages

- Can distribute a pre-configured OS
  - Run VM, install/configure it, then export to another VM image
- Easy to create multiple snapshots
  - If something goes wrong, roll-back to a previously saved snapshot
- Portable
  - Run on any host OS
  - Store on portable hard drive or laptop

Virtual Machine Advantages

- Sandbox
  - Does not affect anything on host OS
- Networked
  - Can access over the network