Operating System Introduction

BIT 4th Semester (unit 1)

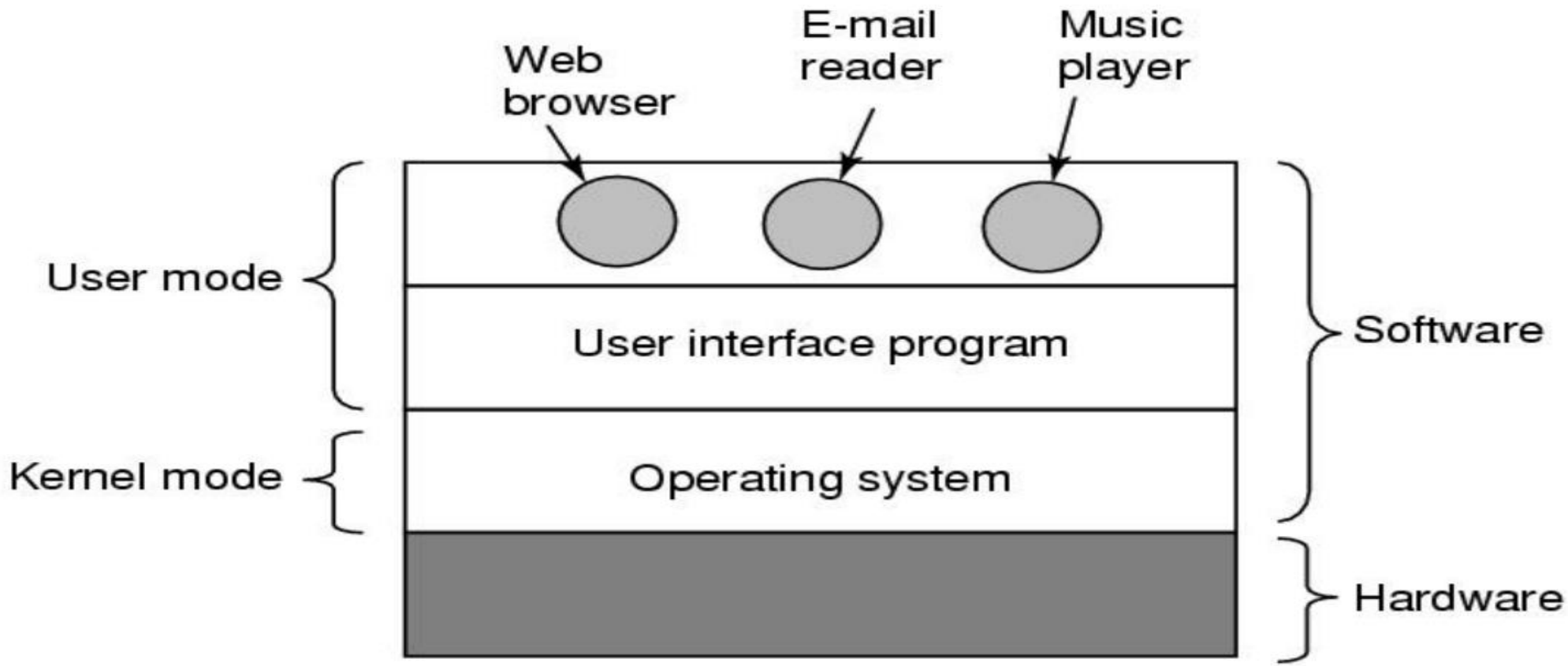
Prepared by:

Parichaya Adhikari

❑ Operating system introduction

- An operating system (OS) is a collection of system programs that together control the operation of a computer.
- OS manages all components computer system like one or more processors , main memory , Disks, printers and various input/ output device.
- Two modes of operation **kernel mode** (complete access to all the hardware and can execute any instruction the machine is capable of executing.) and **user mode** (Application software runs in user mode).

❑ Operating system introduction



❑ Operating system introduction

❖ **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.
- ➢ Provides an environment within which other programs can do useful work.

❑ Operating system

❖ **Two function of OS**

○ **OS as an extended Machine :**

➢ Disk driver , In OS deals with the hardware and provides and interface to read write disk blocks , without getting into details.

➢ OS creates higher-level abstraction (It is not necessary to show details of the various fields or indexes) for programmer.

➢ No any detail to deal OS shields the programmer from the disk hardware and presents a simple file oriented interface.

➢ Job of OS is to create good abstraction , implement and manage abstraction.

❑ Operating system

❖ **Two function of OS**

- **OS as an extended Machine :**

➢Example: (Floppy , disk I/O operation)

➢Disks contains a collection of named files

➢Each file must be open for READ/WRITE

➢After READ/WRITE complete close that file

❑ Operating system

❖ Two function of OS

○ OS as an extended Machine :

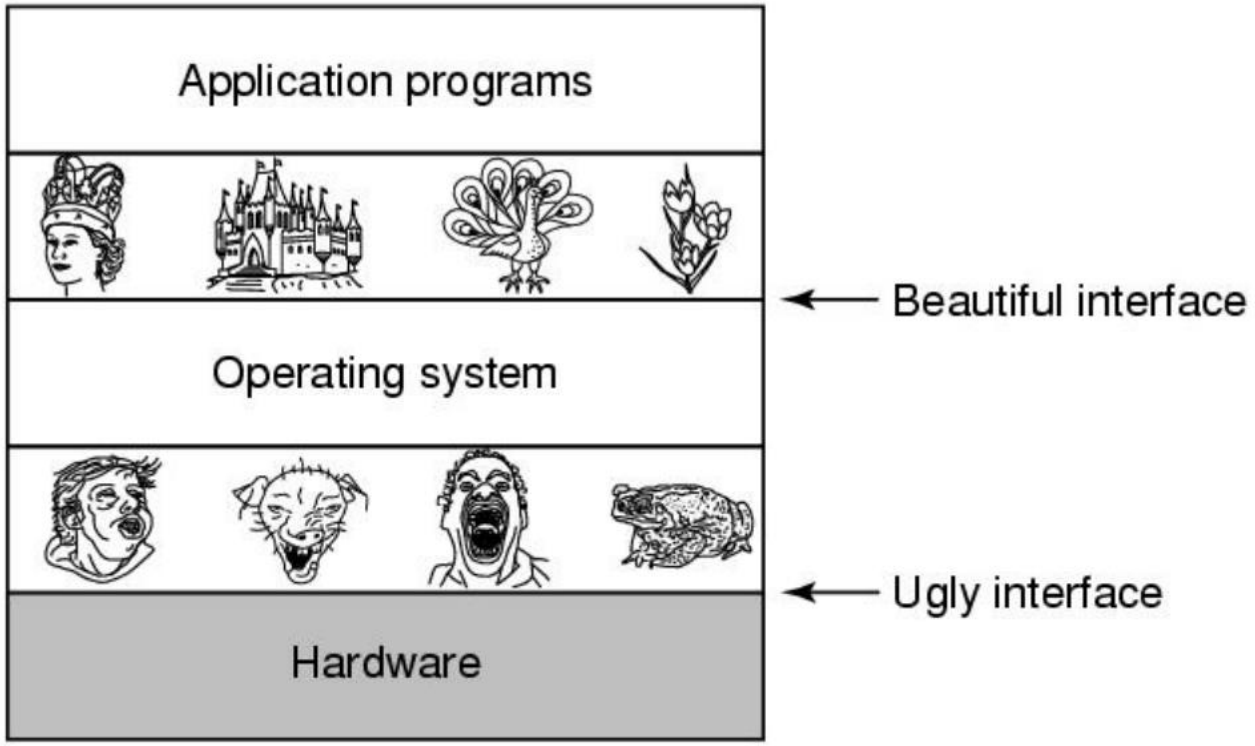


Figure 1-2. Operating systems turn ugly hardware into beautiful

❑ Operating system

❖ **Two function of OS**

- **OS as a Resource Machine.**

➢ What happens if three programs try to print their output on the same printer at the same time?

➢ What happens if two network users try to update a shared document at same time?

➢ Allow multiple programs to run at the same time.

➢ Manage and protect memory, I/O devices, and other resources.

➢ Includes multiplexing (sharing) resources in two different ways:

• In time • In space

➢ Virtualizes resource so multiple users/applications can share.

➢ Protect applications from one another.

➢ Provide efficient and fair access to resources.

❑ Evolution of Operating system

❖ Computer Generation, Component and OS Types

1 st	(1945-55)	Vacuum Tubes	User Driven Source:
2 nd	(1955-65)	Transistor	Batch
3 rd	(1965-80)	IC	Multiprogramming
4 th	(1980-present)	PC	Client Server/Distributed
5th			

❑ Evolution and types of Operating system

❖ First Generation (1945-55) Vacuum Tubes.

➢Human operators scheduled jobs for execution and supervised the use of the computer's resources.

1. Serial Processing

➢ Early computer from late 1940 to the mid 1950.

➢The programmer interacted directly with the computer hardware.

➢Don't have OS.

➢Every computer system is programmed in it's language.

❑ Evolution and types of Operating system

❖ **Second Generation (1955-65) . Transistor**

2. Batch Processing OS

- The wasted time due to scheduling and setup time in serial processing .
- To improve utilization , the concept of a batch operating system was developed.
- Batch is defined as a group of jobs with similar needs.
- OS allows user to form batches .
- Computer execute each batch sequentially , processing all jobs of a batch as a single process called batch processing.
- Largely programmed in FORTRAN and assembly language.
- Eg: FORTRAN card , telling the operating system to load the **FORTRAN** compiler from the system tape.

❑ Evolution and types of Operating system

❖ Second Generation (1955-65) . Transistor

2. Batch Processing OS

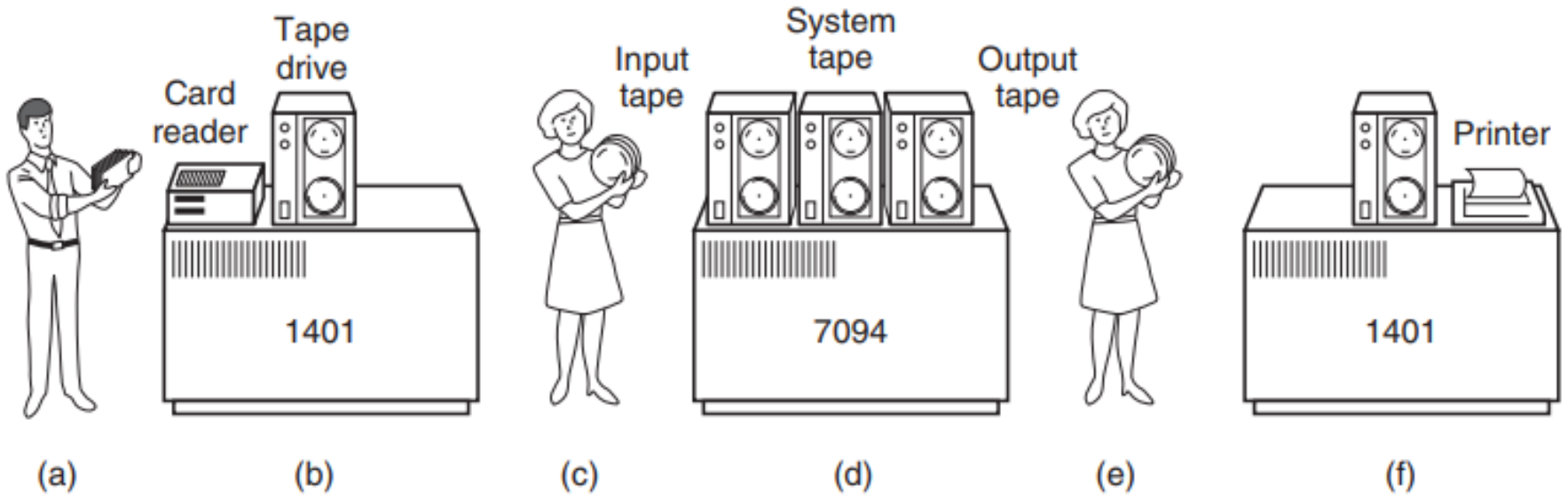


Figure 1-3. An early batch system. (a) Programmers bring cards to 1401. (b) 1401 reads batch of jobs onto tape. (c) Operator carries input tape to 7094. (d) 7094 does computing. (e) Operator carries output tape to 1401. (f) 1401 prints output.

❑ Evolution and types of Operating system

❖ Third Generation (1955-65) . IC and Multiprogramming

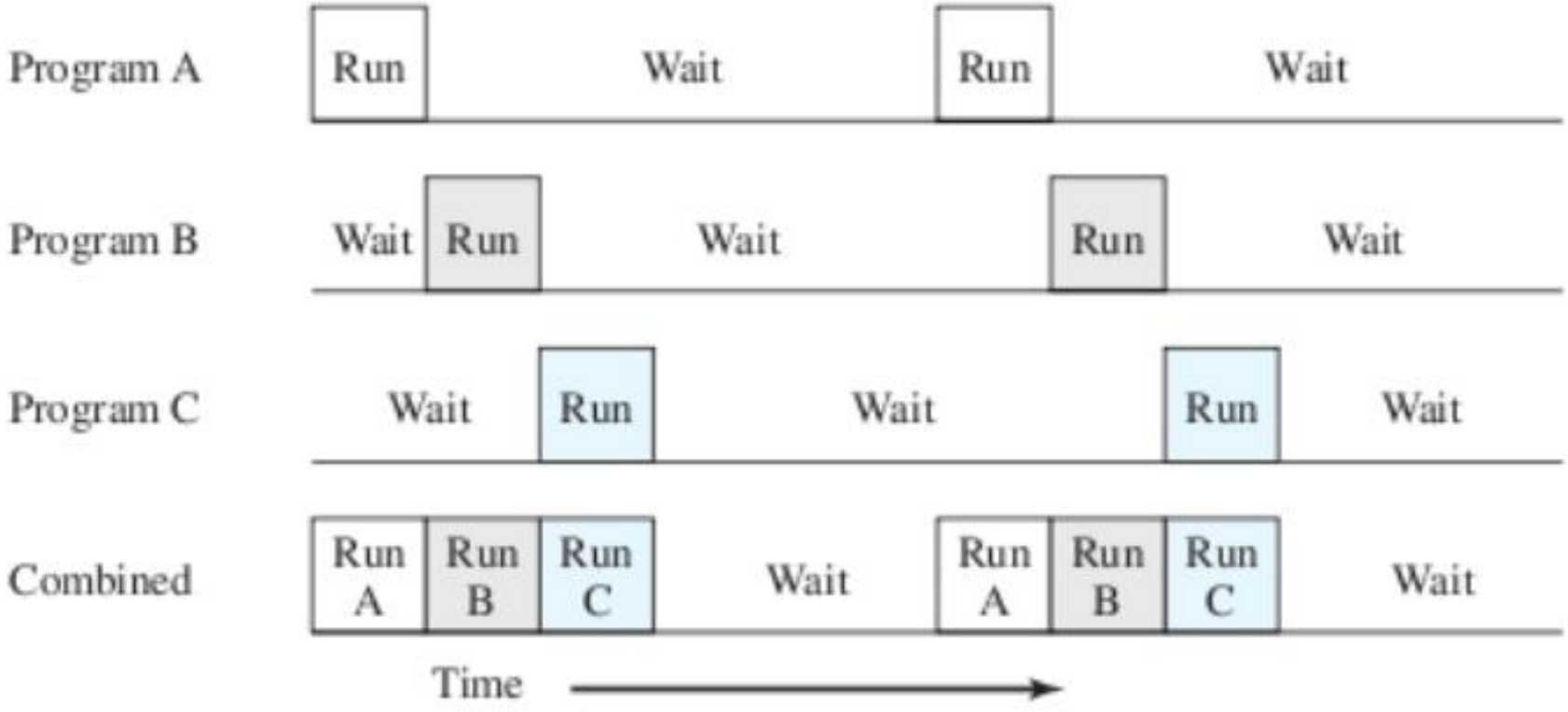
3. Multiprogramming OS .

- Operating systems that are able to execute more than one program simultaneously on a processor.
- A single program cannot keep CPU or I/O devices busy at all times.
- Increases CPU utilization by organizing jobs .
- One program is waiting for an I/O transfer , another program can use the processor.

❑ Evolution and types of Operating system

❖ Third Generation (1955-65) . IC and Multiprogramming

3. Multiprogramming OS .



(c) Multiprogramming with three programs

❑ Evolution and types of Operating system

❖ Third Generation (1955-65) . IC and Multiprogramming

4. Multitasking or Time sharing OS

- Processor time is shared among multiple users simultaneously.
- Multitasking is a logical extension of multiprogramming that provide user interaction.
- There are more than one user interacting the system at the same time.
- Multitasking system uses CPU scheduling and multiprogramming to provide each user with a small portion of time shared computer.

❑ Evolution and types of Operating system

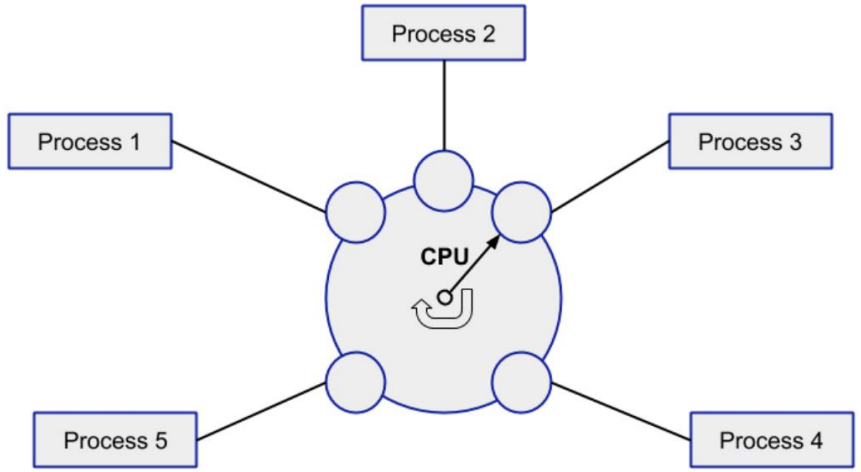
❖ Third Generation (1955-65) . IC and Multiprogramming

4. Multitasking or Time sharing OS

- The objective of multi programmed is to maximize the CPU use , Where as in time sharing system is to minimize response time.
- Uses CPU scheduling and multiprogramming to provide each user with a small portion of a time.

For eg: Unix , Linux ,

windows 2000 server.



❑ Evolution and types of Operating system

❖ Third Generation (1955-65) . IC and Multiprogramming

5. Real time operating system.

- Any system in which a timely response by the computer.
- Real time operating system must perform tasks in well defined and fixed time constraints otherwise system will fail.
- example of some real time system are as follow:

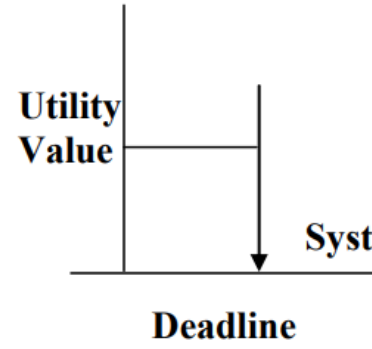
Microwave oven , Industrial process control, Missile tracking system, Air traffic control system etc.

❑ Evolution and types of Operating system

❖ Third Generation (1955-65) . IC and Multiprogramming

5. Real time operating system.

➢ There are two types of real time operating system.

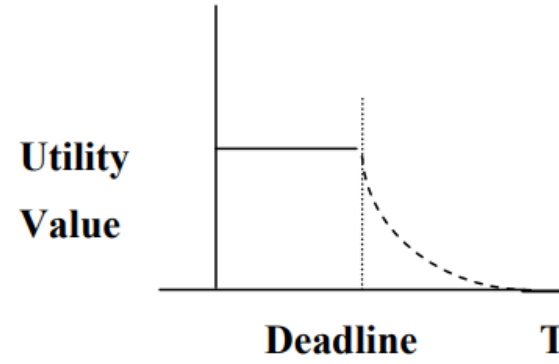


a. Hard real time system:

➢A system crashes if the deadline is not met by the task.

b. Soft real time system:

➢Here utility value becomes less/drops with time.



❑ Evolution and types of Operating system

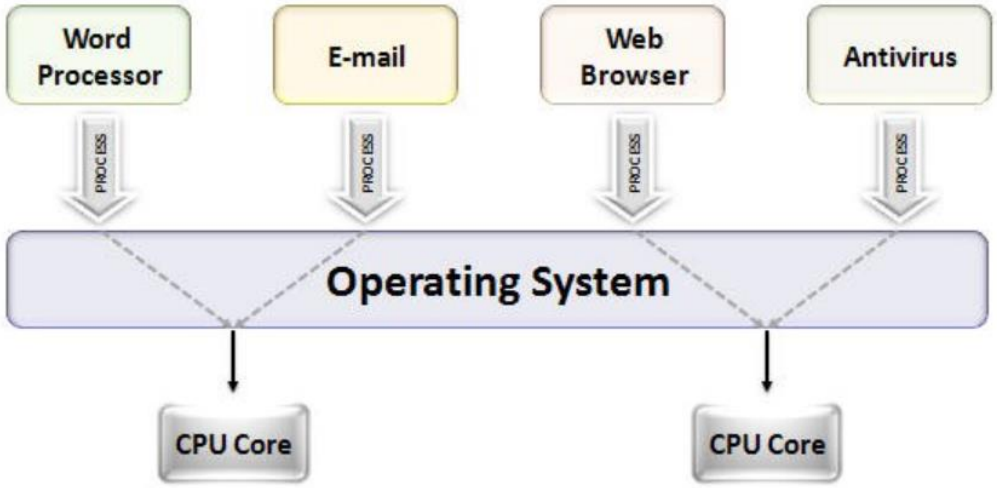
❖ Third Generation (1955-65) . IC and Multiprogramming

- **Multiprocessing Operating system:**

➢ There are more than one processor which work parallel to perform the required operations.

➢ Multiple program are executed in different processor or CPUs at the same time , then it is called multiprocessing.

➢ The main motive of multiprocessing is to increase computation speed.



❑ Evolution and types of Operating system

- **The Fourth Generation (1980 to present) , PC , Microprocessor.**
- **Distributed Operating system:**

➢ In distributed operating system, have various system and all these systems have their own CPU, main memory, secondary Memory and resources. .

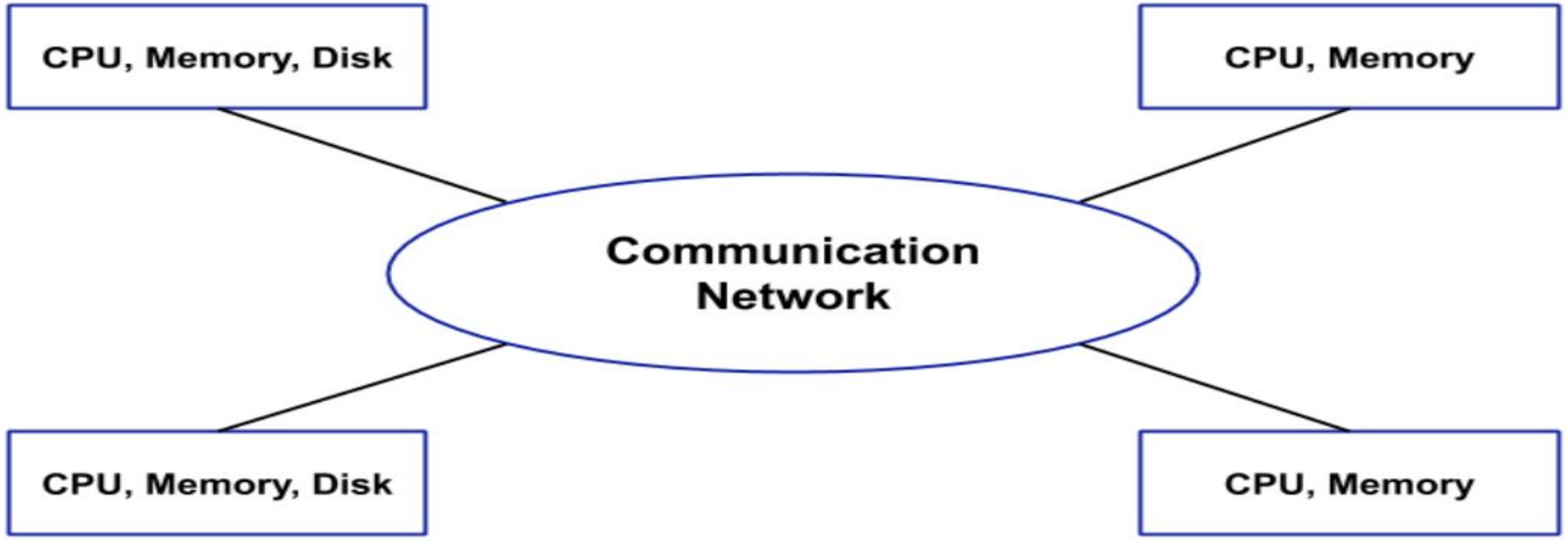
➢ These Systems are connected to each other using a shared communication network. .

➢ Each system can perform its task individually.

➢ One user can access the data of the other system and can work accordingly , so remote access is possible in these Distributed operating system.

❑ Evolution and types of Operating system

- **The Fourth Generation (1980 to present) , PC , Microprocessor.**
- **Distributed Operating system:**

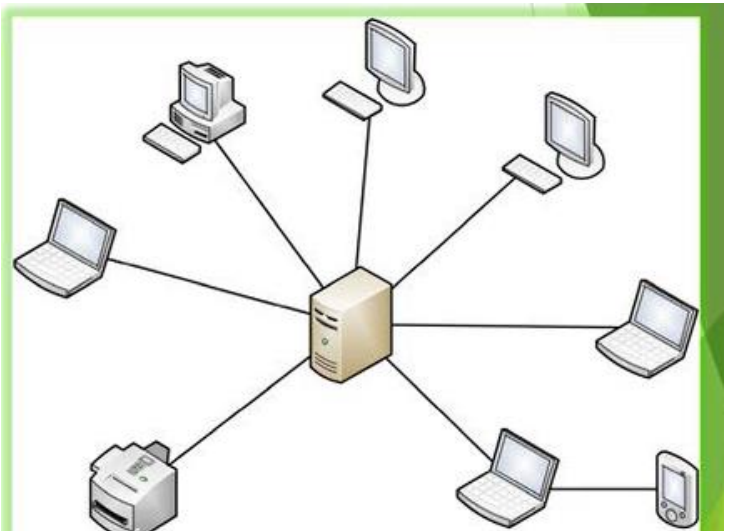


❑ Evolution and types of Operating system

- **The Fourth Generation (1980 to present) , PC , Microprocessor.**
- **Network Operating system:**

➢ Designed for the sole purpose of supporting workstations, database sharing, application sharing and file and printer access sharing among multiple computers in network

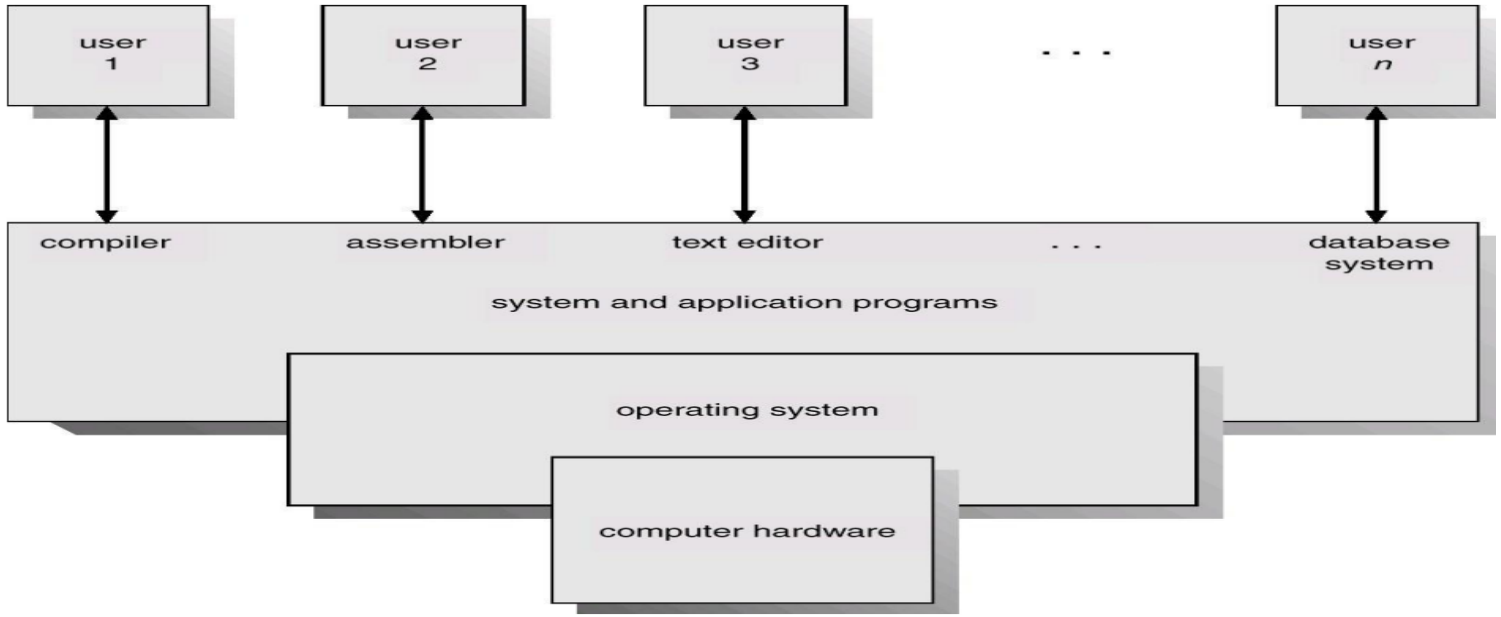
➢ Some of the most well known network operating system include Microsoft windows server 2003, Microsoft window server 2008, Linux and mac OS x.



❑ Computer system overview

❖**Computer system**.

➢ Computer system provide a capability for gathering data , performing computations, storing information , communicating with other computer system and gathering output.



Abstract view of computer system

❑ Computer system overview

1. **Hardware:** Provides basic computing resources (CPU , Memory, I/O, Devices).
2. **Operating system:** Controls and co-ordinates the use of the hardware among the various application programs for the various users.
3. **Applications Programs:** Define the ways in which the system resources are used to solve the computing problems of the users (Compilers , database system , video games , business programs).
4. **Users:** (People , machine , other computers)

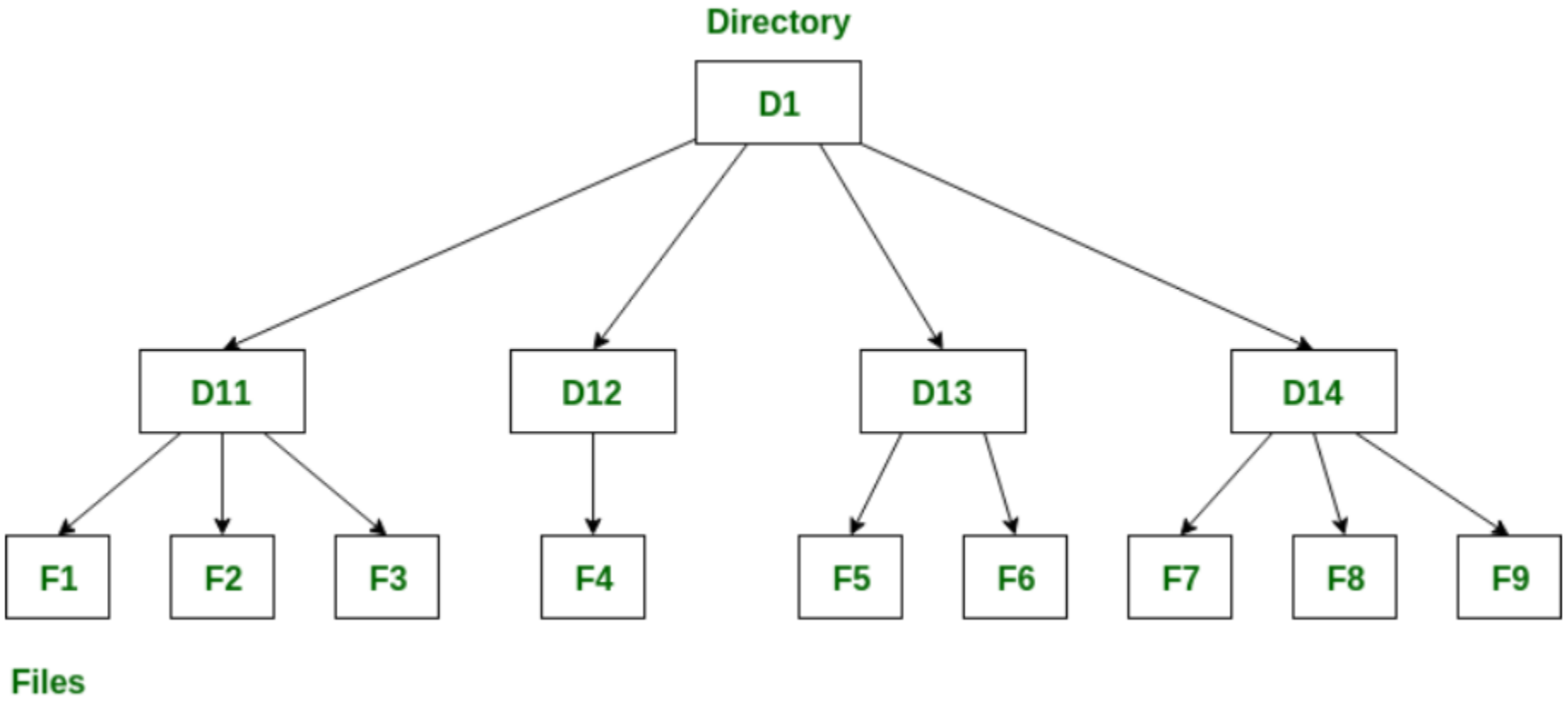
❑ Operating system concepts

❖Files

- The major function of the OS is to hide the internal details of the disks and other I/O devices and present the programmer with a nice , clean abstract model of device independent files.
- Every file within the directory hierarchy can be specified by giving it's path name from the top of the directory hierarchy. For eg: c:\user\Desktop>
- System calls are obviously needed to create files , remove files , read files and write files.

❑ Operating system concepts

❖Files



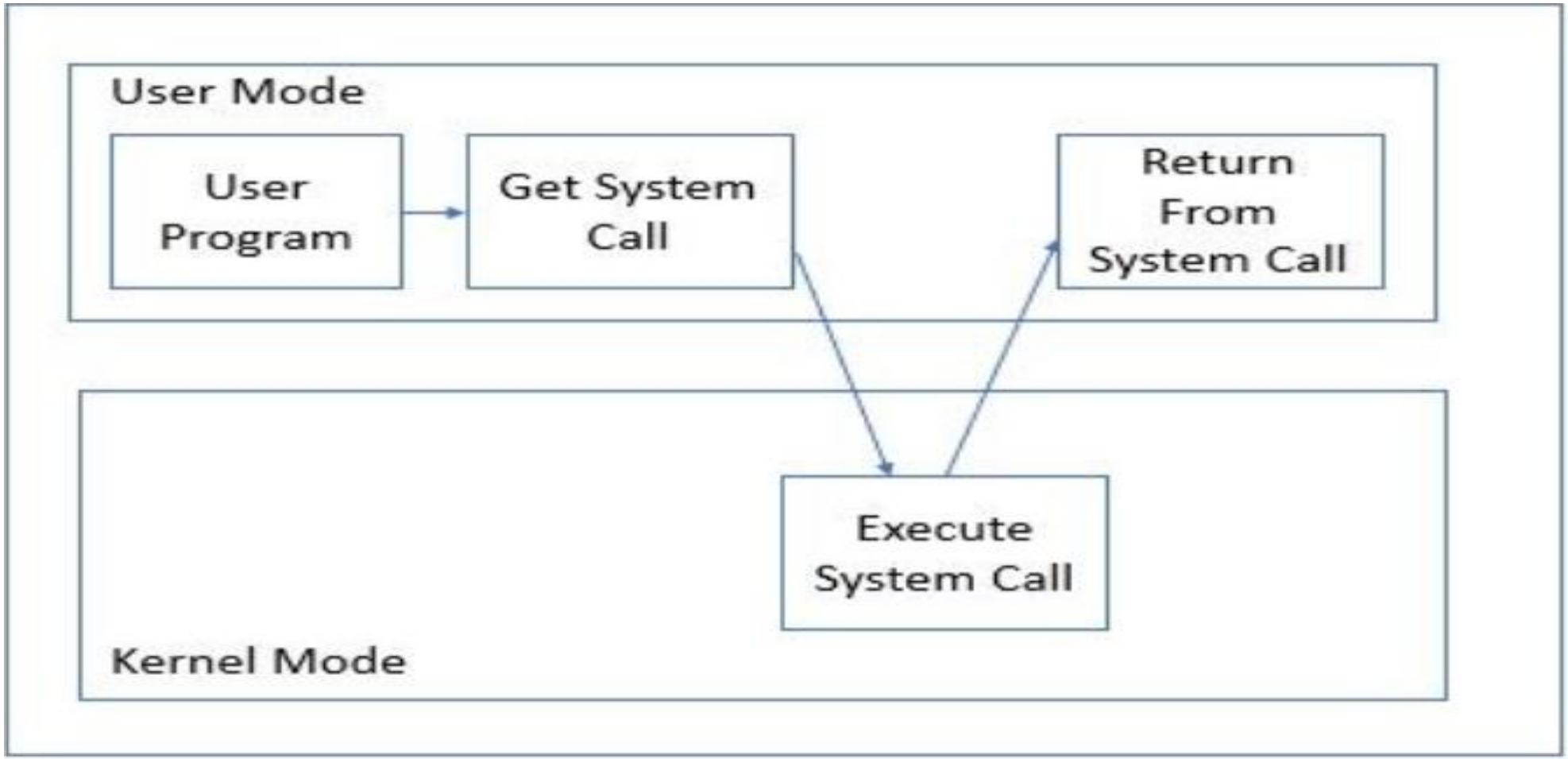
❑ Operating system concepts

❖**System calls**

- A system call is **how a program requires a service from an operating system** kernel.
- System call provide the interface between a **process** and the **operating system.**
- This may include hardware related services (Eg accessing the harddisk) , Creating and executing new processes and communicating with kernel services(Like CPU scheduling).
- Some popular system calls are open , read , write, close wait , exit , fork , kill etc.

❑ Operating system concepts

❖System calls



❑ Operating system concepts

❖**System calls**

- **In general , system calls are required in the following situations.**

➢ If a file system requires the **creation or deletion of files** , reading and writing from files also require a system call.

➢Creation and management of new processes.

➢Network connections also require system calls. This includes sending and receiving packets.

➢Access to a hardware device such as a printer , Scanner etc. requires a system call.

❑ Operating system concepts

❖**Types of system Calls.**

- There are mainly five types of system calls.

➢**Process control :** These system calls deals with processes such as process creation , process termination , load , execute process etc.

➢**File management:** These system calls are responsible for file manipulation such as creating a file , reading a file , writing into a file delete file etc.

➢**Device management:** These system calls are responsible for device manipulation such as reading from device buffers , writing into device buffers , request device , release device .

❑ Operating system concepts

❖**Types of system Calls.**

- There are mainly five types of system calls.

➢**Information Maintenance : ** These system calls handle information and its transfer between the operating system and the user program for eg: get/set time or date

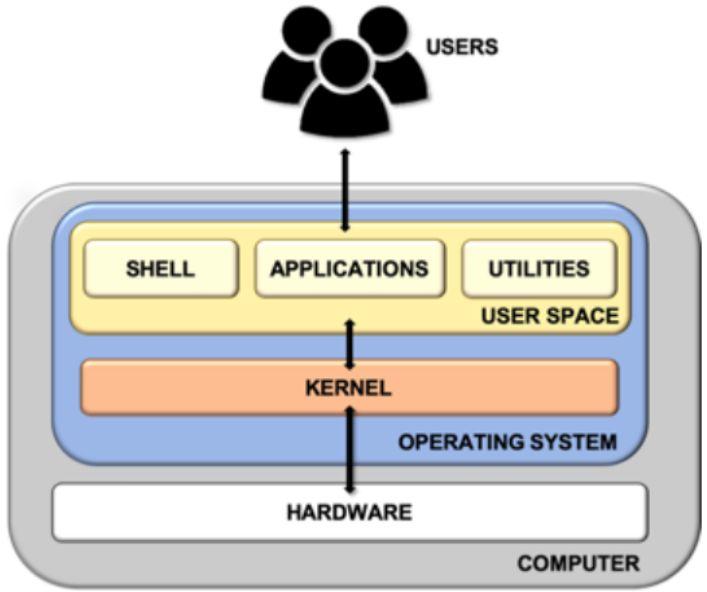
➢**Communication:** These system calls are useful for inter process communication. They also deal with creation and deleting a communication connection for eg : send and receive message , Transfer status information.

❑ Operating system concepts

❖Shell

- The shell is command line interpreter (It translates commands entered by the user and converts them into a language that is understood by the kernel).
- A shell is an interface between the user and the internal part of the operating system.
- Shell uses the system kernel to execute

Program , create file etc.



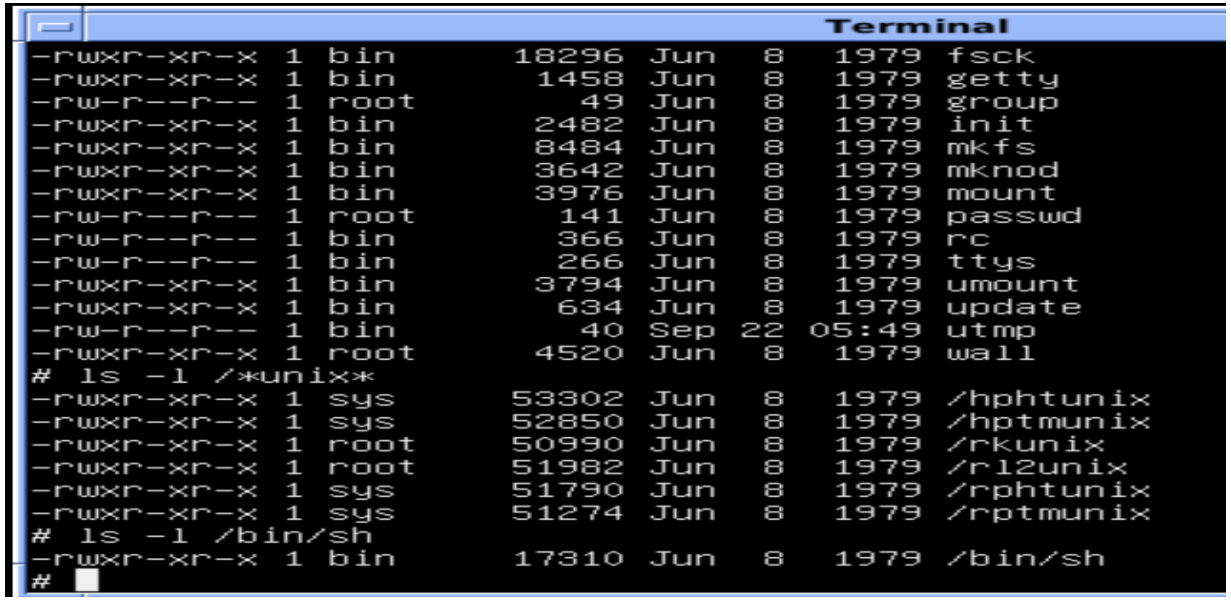
❑ Operating system concepts

❖**Shell Example**

❑ **Bourne Shell**

➢Used by UNIX operating system.

➢ It provides command based programming to interpret and execute user commands.

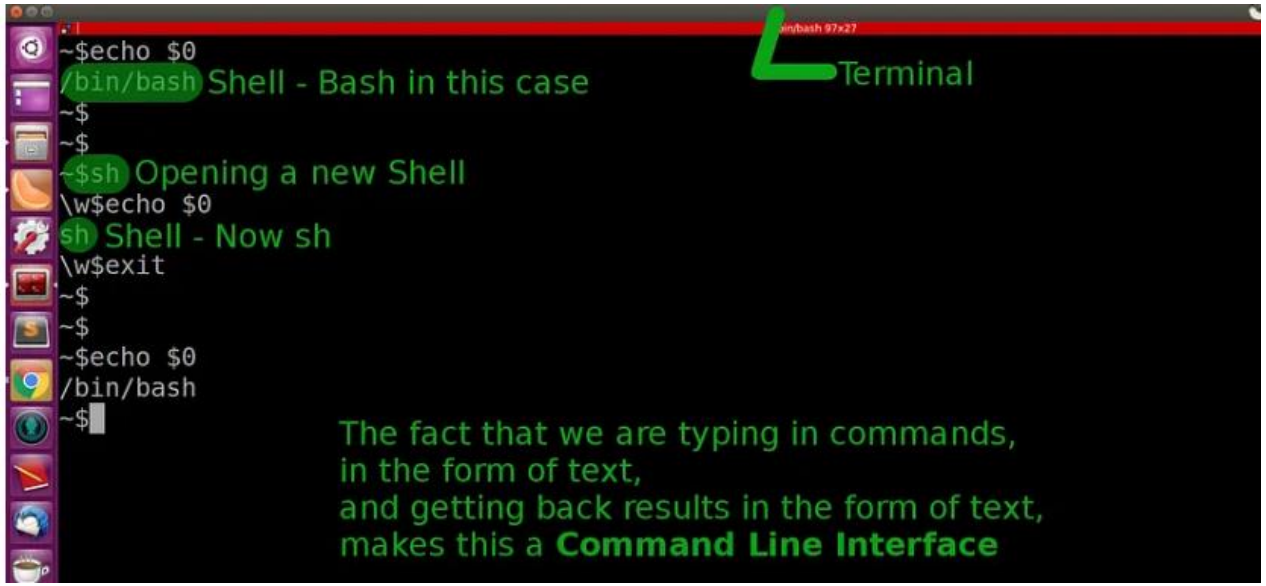


❑ Operating system concepts

❖**Shell Example**

❑**Bash Shell (Bourne Again Shell).**

- Bash shell is the free and enhanced version of the Bourne shell.
- Command language interpreter for the GNU .
- It currently runs on nearly every version of UNIX and a few other OS. (MS-DoS , Windows platform etc).



❑ Operating system concepts

❖**Shell Example**

❑ **Korn Shell**

➢Korn Shell or KSH which attempts to integrate the features of other shells like C shell, Bourne Shell, etc.

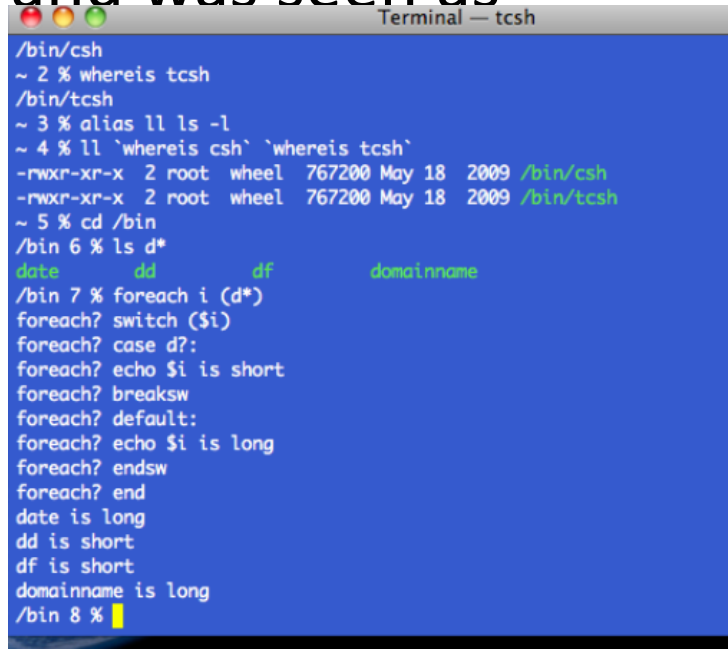
➢It provides more programming features than Bash.

❑ Operating system concepts

❖Shell Example

❑C shell

- ➢ The C shell is a command processor which is typically run in a text window, allowing the user to type and execute commands.
- ➢ The overall style of the language looked more like C and was seen as more readable.
- ➢ For eg such as macOS and Red Hat Linux csh is used



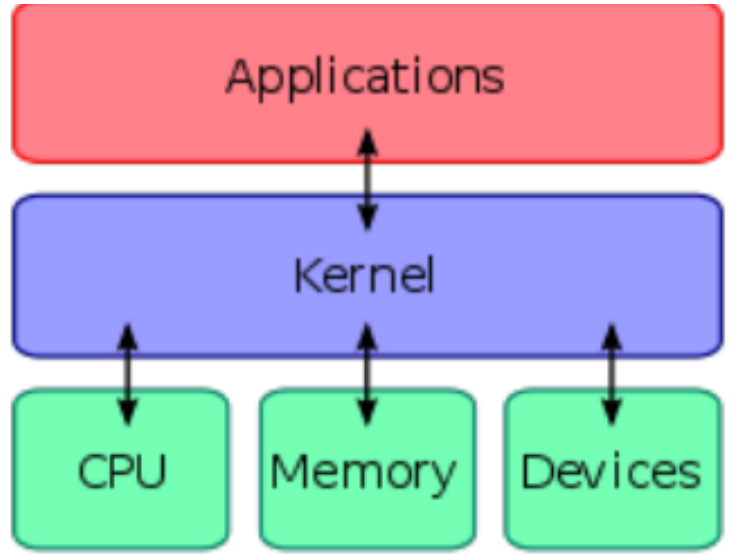
❑ Operating system concepts

❖**Kernel**

➢ Kernel is the main component of Operating System.

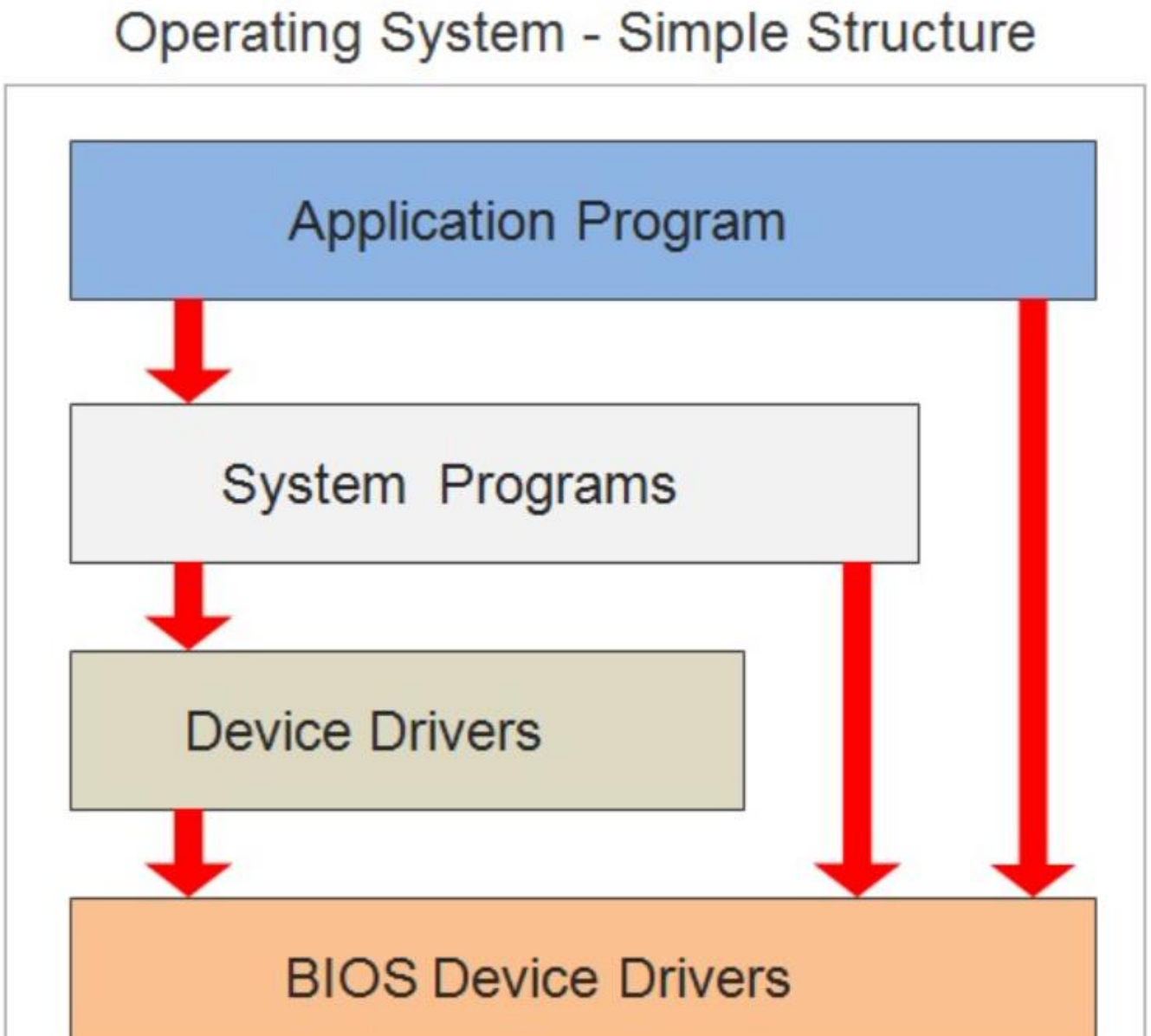
➢By using the Interprocess communication and system calls , it acts as a bridge between Hardware and software component.

➢Kernel manages the system resources (Task , Memory , disks) etc.



❑ Operating system Structure

1. Simple Structure



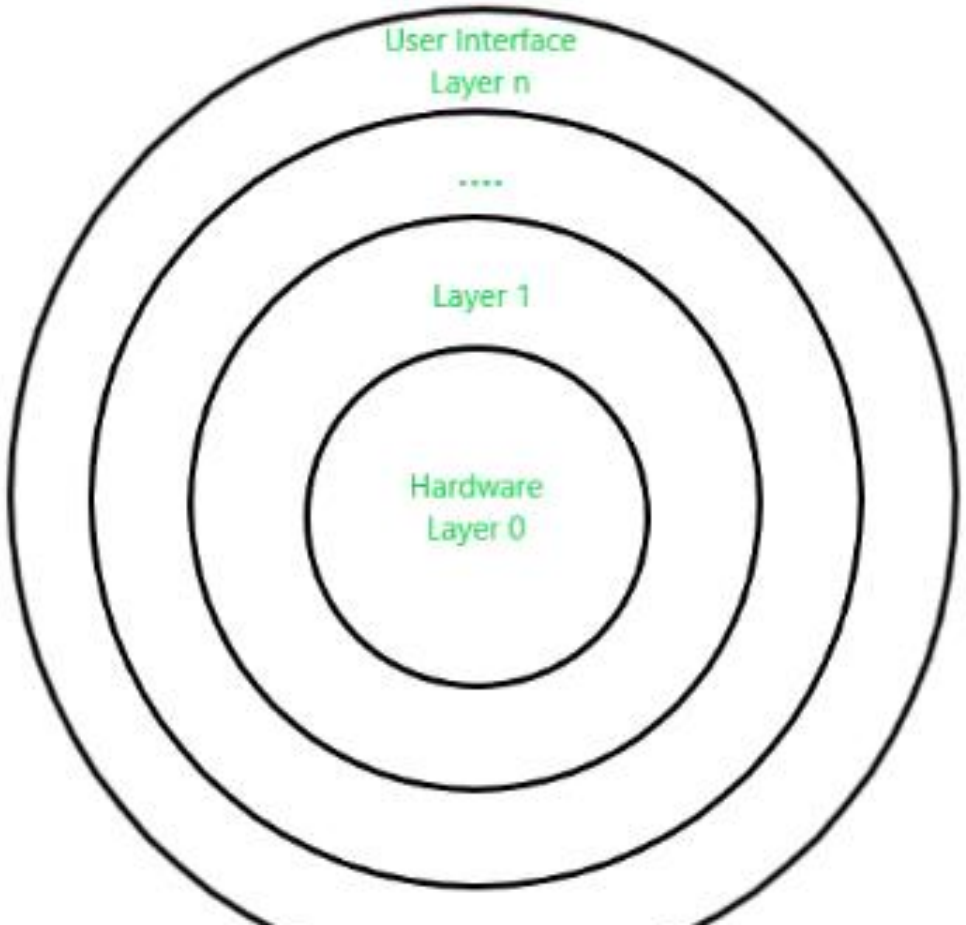
❑ Operating system Structure

1. Simple Structure

- The interface and the levels of functionality are not well separated.
- Application programs are able to access the basic I/O routines to write directly to the device drivers.
- Every layer can access the base hardware layer.
- So this looks like that they are kind of a layered structure but not actually.
- Interfaces and function are not well defined.
- Causing the entire system to crash when the user program fails.

❑ Operating system Structure

2. Layered Structure



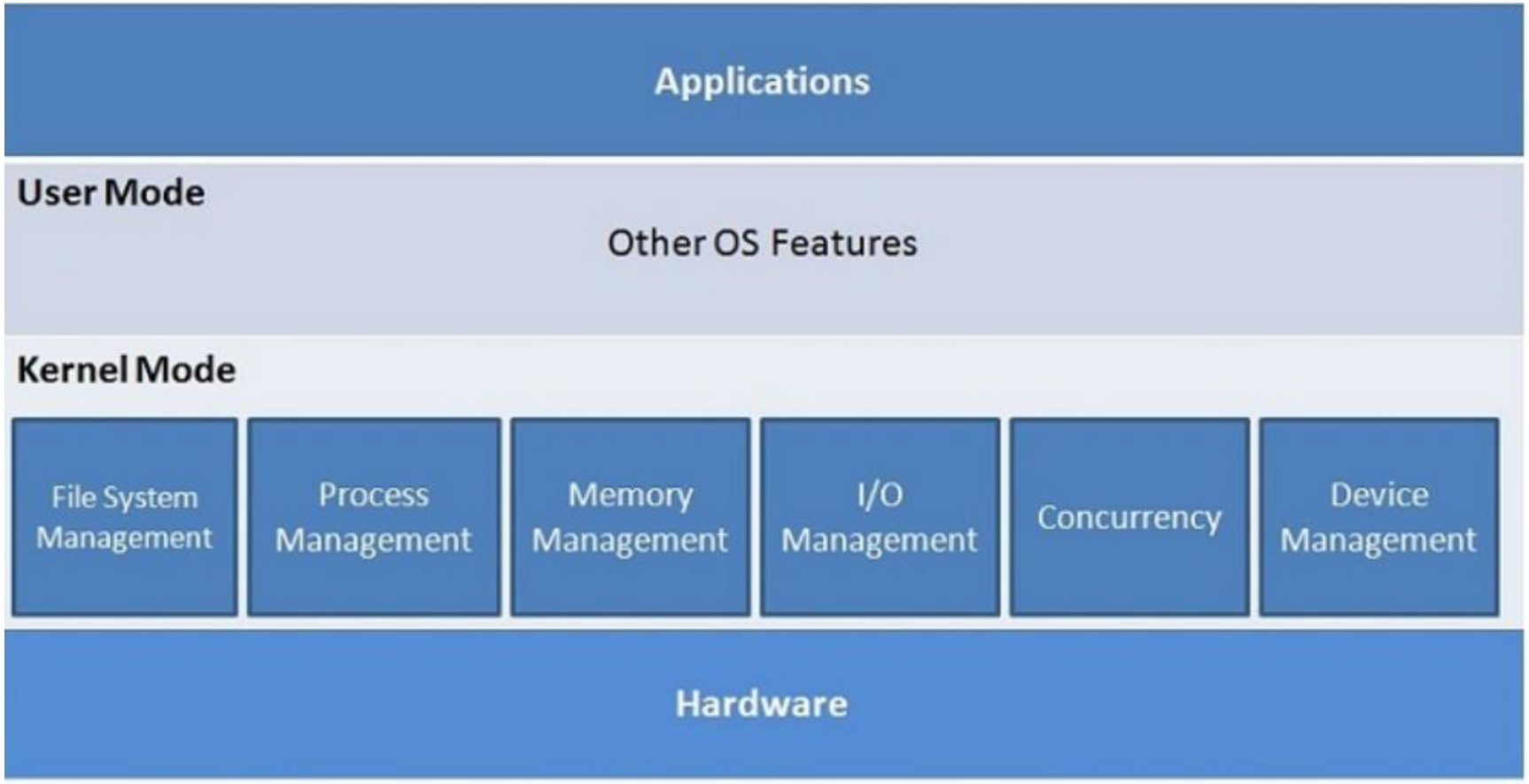
❑ Operating system Structure

2. Layered Structure

- Consists of breaking the operating system into the number of layers.
- The OS components are grouped in layers based on the functionality of these components.
- The bottom layer (Layer 0) is the hardware layer and the topmost layer (n) is user interface.
- For eg: Windows NT uses this layers.

❑ Operating system Structure

3. Monolithic Structure



Monolithic Kernel Architecture

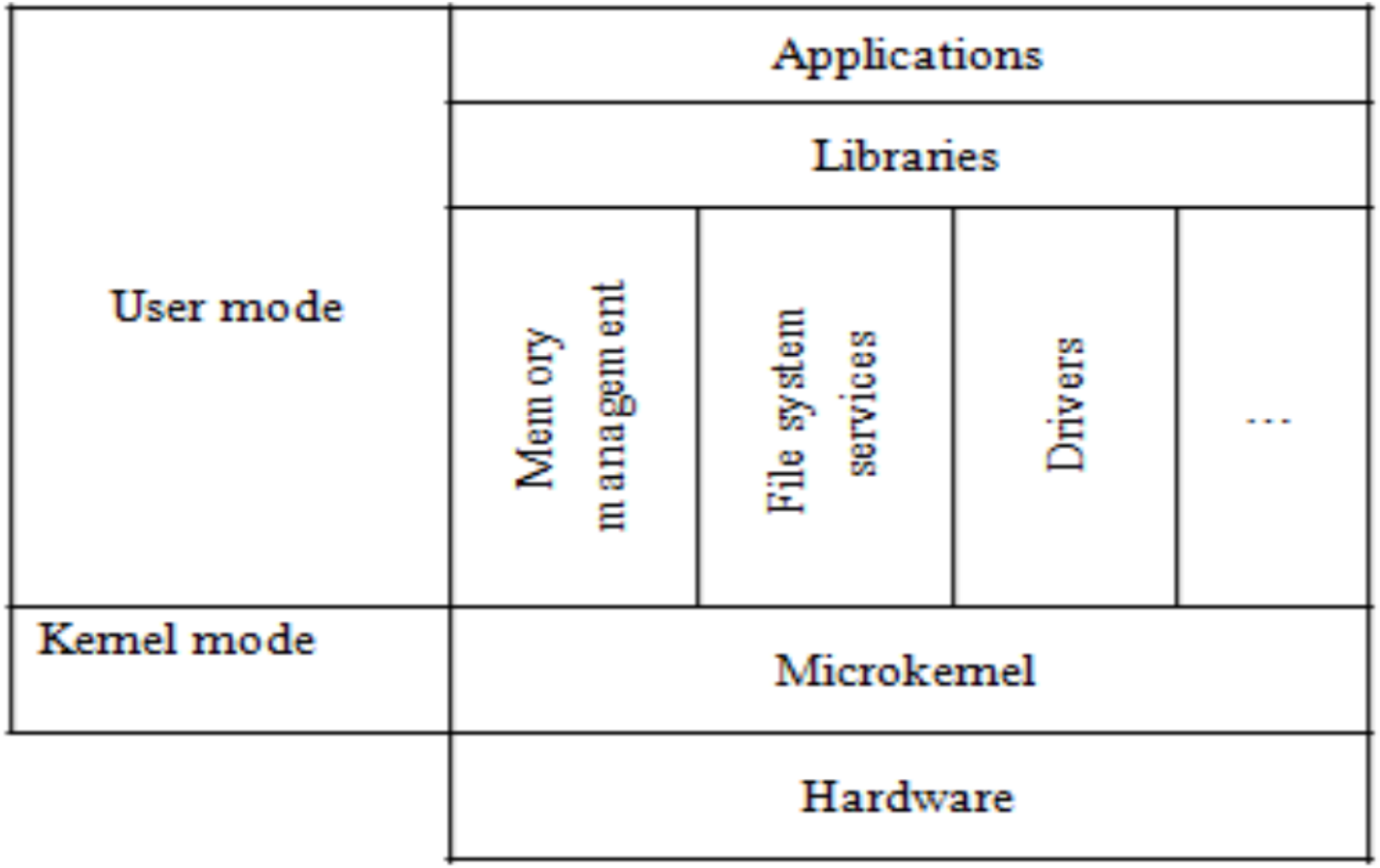
❑ Operating system Structure

3. Monolithic Structure

- Every component of OS contained in the kernel and can directly communicate with any other . (By using Function call).
- Monolithic application describe a single tiered software application in which different components combined into a single program from a single platform.
- Difficult to maintain and implemented.

❑ Operating system Structure

4. Microkernel Structure.



❑ Operating system Structure

4. Microkernel Structure

- Most of the function are executed on the user mode.
- In this structure remove non essential form the kernel and implement them as system and user level program.
- For eg : like the device driver , the file server, the process server , the virtual memory (All these services are implemented as a user level system program).
- Kernel just provide a communication between the user and hardware. (Message passing).