

Operating Systems with Linux Internals- CSE3120

Module-1

Introduction to OS

Prepared By...

Shankar Rana

Asst. Prof., CSE, PU



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



Contents

Module 1: Introduction [9 Hrs.] [Knowledge Level (1)]

- Introduction to OS – Computer System Architecture, Operating System Structure, Operations – Different management activities handled by the OS, Computing environments, Operating System Services, User and OS interface, System Calls and its types, System Programs[loaders, linkers...],
- Overview of OS design and implementation.
- **Linux Operating System:** Introduction to Linux OS, Basic Commands of Linux OS

What is an Operating System?

- A program that acts as an intermediary between a **user** of a computer and the **computer hardware**
- Operating system goals:
 - Execute user programs and solving user problems easily
 - Make the computer system convenient to use
 - Use of computer hardware in an efficient manner



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



❑ 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

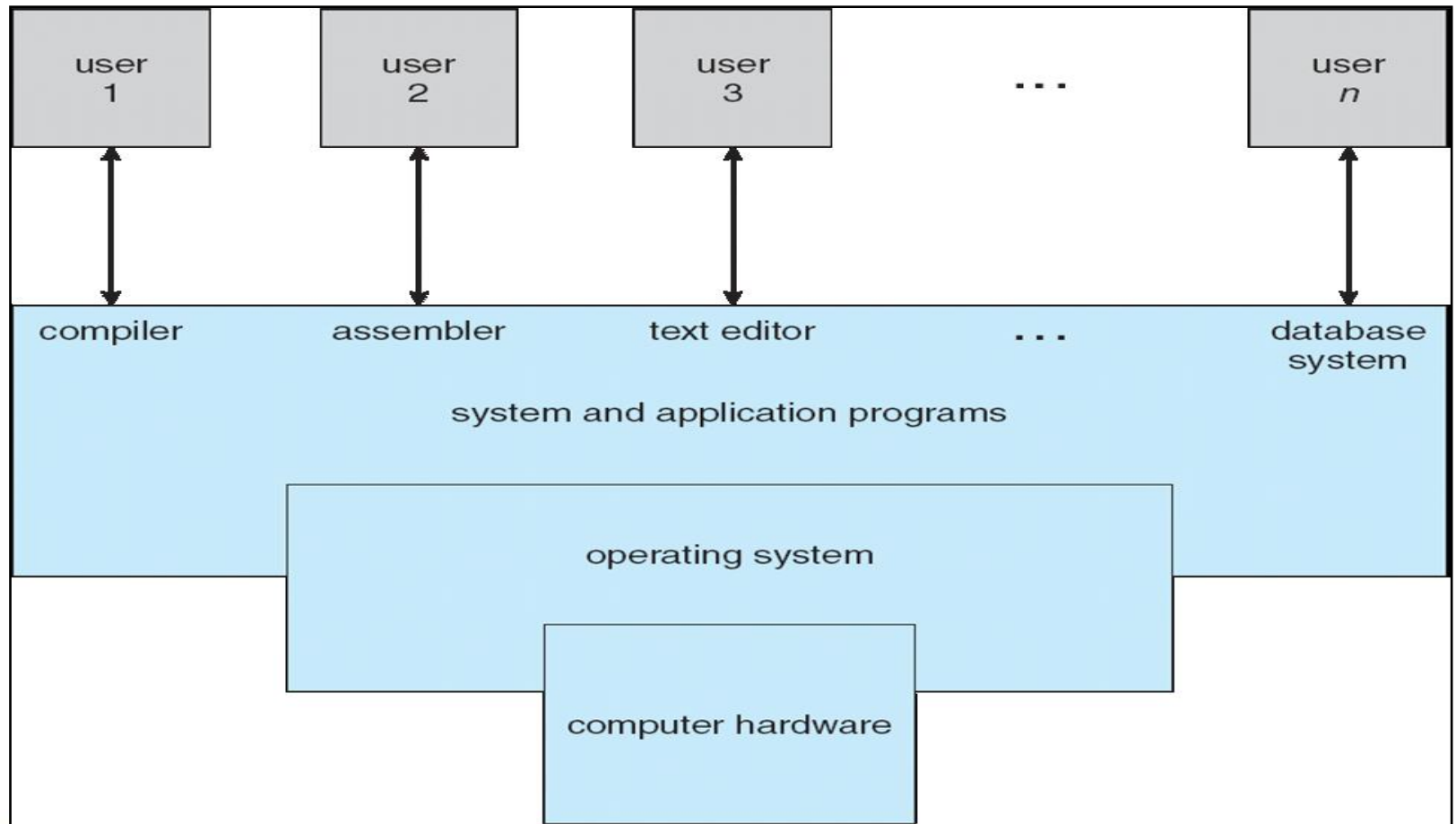


**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



Four Components of a Computer System



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



❑ What Operating Systems Do

- Depends on the point of view
- Users want **convenience**- **ease of use** and **good performance**
 - Don't care about **resource utilization**
- But shared computer such as **mainframe** or **minicomputer** must keep all users happy
- Users of dedicated systems such as **workstations** have dedicated resources but frequently use shared resources from **servers**
- **Handheld computers** are **resource poor**, optimized for usability and battery life
- Some computers have little or no user interface, **such as embedded computers** in devices and automobiles



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



• Operating System Definition

- OS is a **resource allocator**
 - Manages all resources
 - Decides between conflicting requests for efficient and fair resource use
- OS is a **control program**
 - Controls execution of programs to prevent errors and improper use of the computer



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



• Operating System Definition (Cont.)

- No universally accepted definition
- “Everything a vendor ships when you order an operating system” is a good approximation
 - But varies wildly
- “The one program running at all times on the computer” is the **kernel**
- Everything else is either
 - a system program (ships with the operating system) ,
 - or
 - an application program



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



Computer Startup

- **bootstrap program** is loaded at power-up or reboot
 - Typically stored in ROM or EPROM, generally known as **firmware**
 - Initializes all aspects of system
 - Loads operating system kernel and starts execution



**PRESIDENCY
UNIVERSITY**

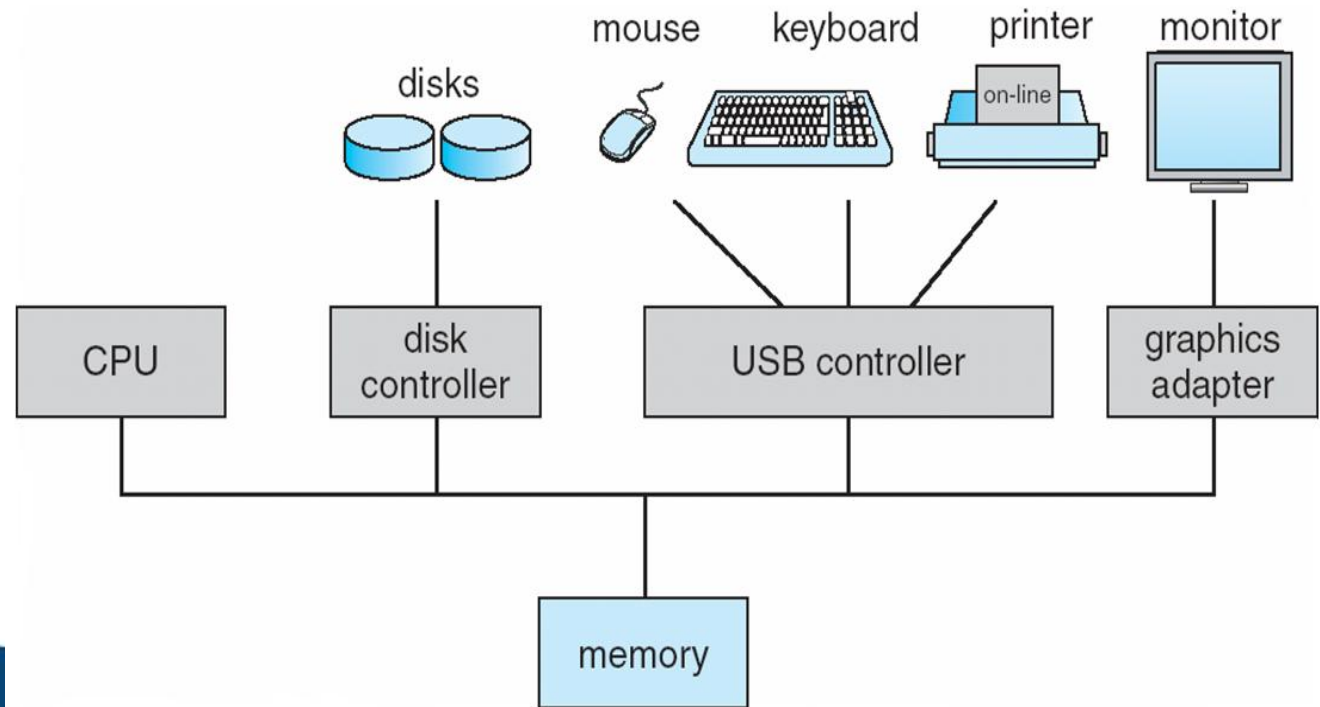
Private University Estd. in Karnataka State by Act No. 41 of 2013



Computer System Organization

- Computer-system operation

- One or more CPUs, device controllers connect through common bus providing access to shared memory
- Concurrent execution of CPUs and devices competing for memory cycles



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



Computer-System Operation

- I/O devices and the CPU can execute concurrently
- Each device controller **is in-charge of a particular device type**
- Each device controller has a **local buffer**
- CPU moves data from/to main memory to/from **local buffers**
- I/O is from the device to local buffer of controller
- Device controller informs CPU that it has finished its operation by causing an **Interrupt**



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



Storage Structure

- **Main memory** – only large storage media that the CPU can access directly
 - **Random access**
 - Typically **volatile**
- **Secondary storage** – extension of main memory that provides large **nonvolatile** storage capacity
- **Hard disks** – rigid metal or glass platters covered with magnetic recording material
 - Disk surface is logically divided into **tracks**, which are subdivided into **sectors**
 - The **disk controller** determines the logical interaction between the device and the computer
- **Solid-state disks** – **faster than hard disks, nonvolatile**
 - Various technologies
 - Becoming more popular



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



❑ Storage Hierarchy

- **Storage systems organized in hierarchy**
 - **Speed**
 - **Cost**
 - **Volatility**
- **Caching** – copying information into faster storage system;
 - **main memory can be viewed as a cache for secondary storage**
- **Device Driver** for each device controller to manage I/O
 - **Provides uniform interface between controller and kernel**

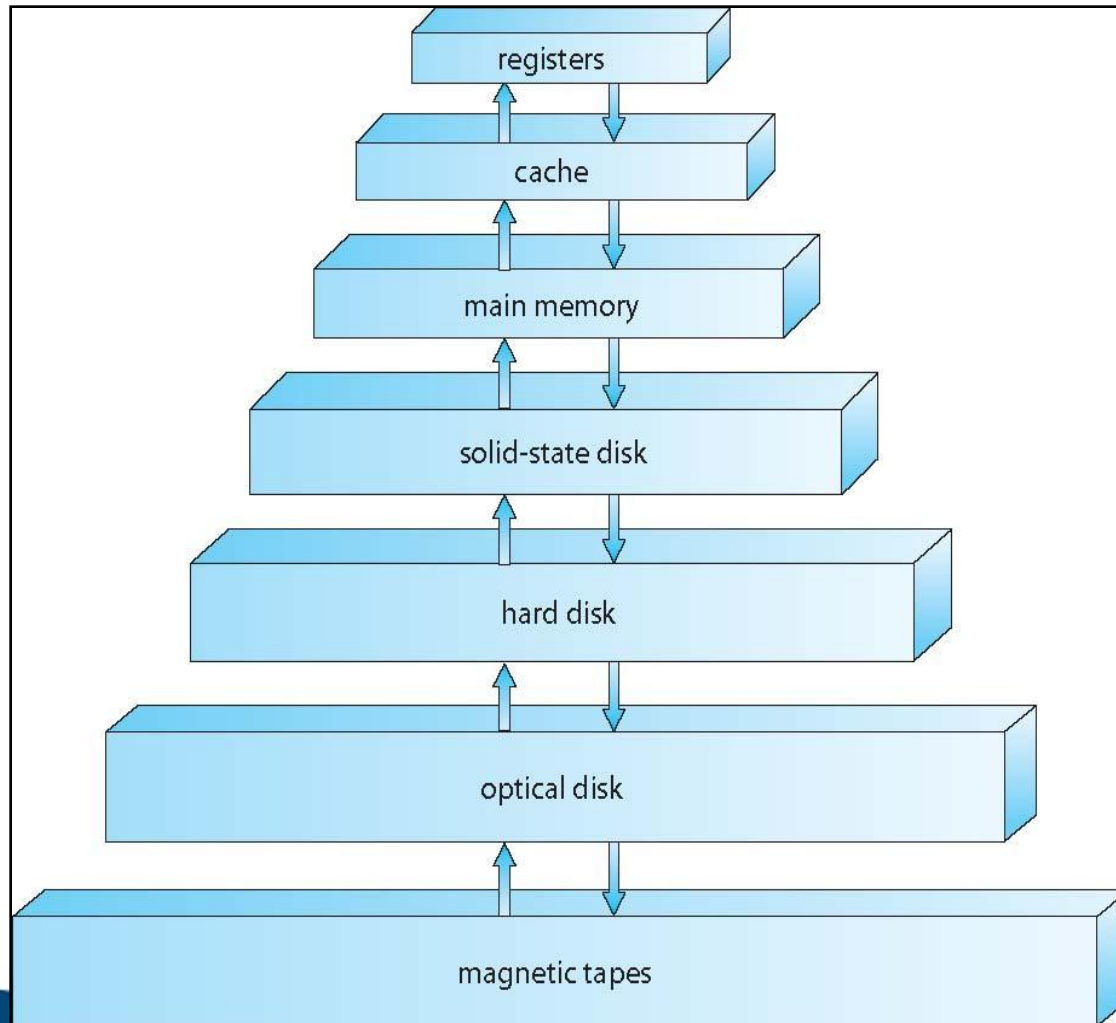


**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



Storage-Device Hierarchy

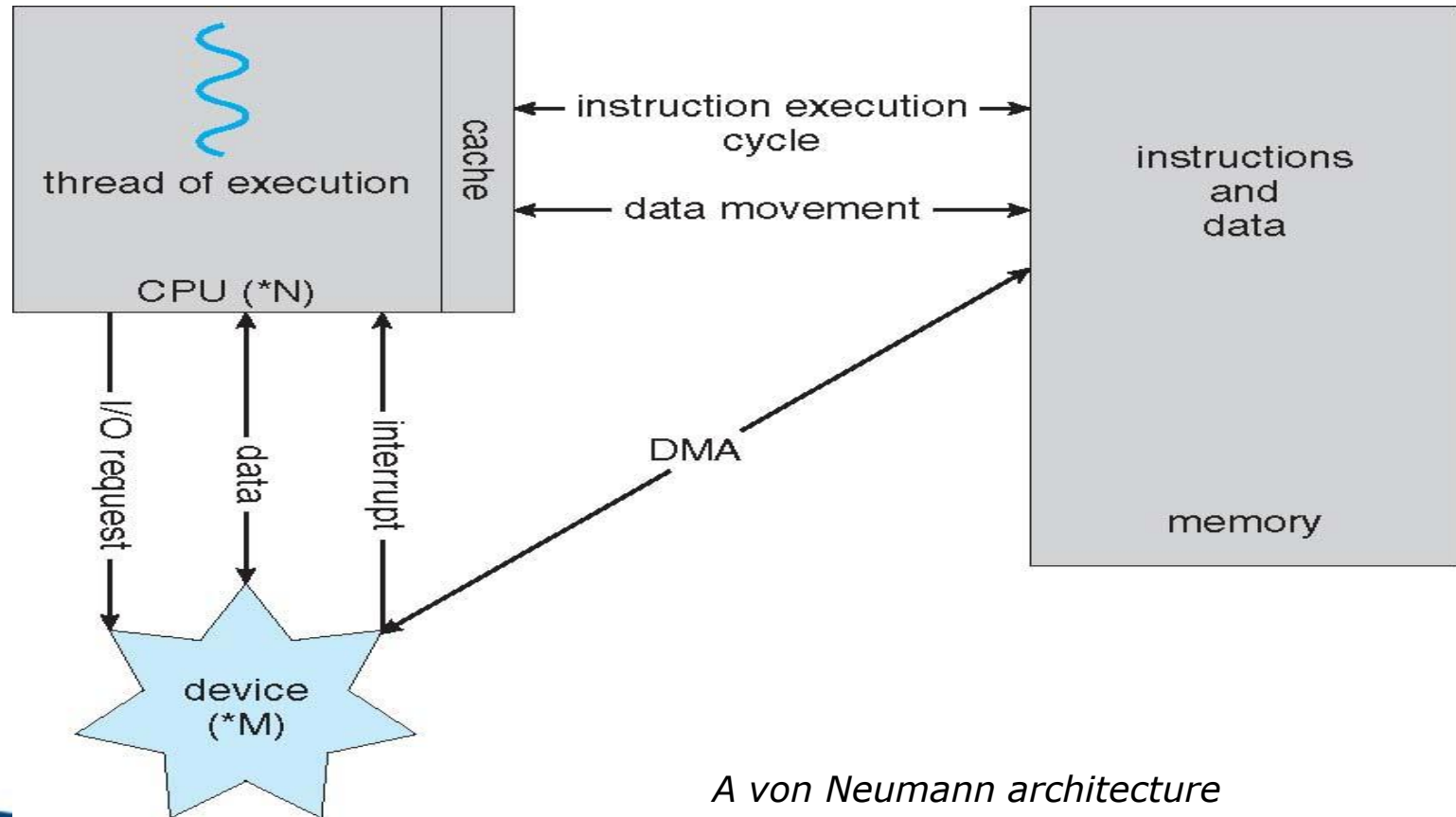


**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



How a Modern Computer Works



A von Neumann architecture



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



❑ Computer-System Architecture

1. Single General-Purpose processor- Mostly used systems

Most systems have special-purpose processors as well-

limited instruction set and do not run user processes

2. Multiprocessors systems- growing in use and importance

Also known as **parallel systems**, **tightly-coupled systems**

Advantages include:

1. **Increased throughput**
2. **Economy of scale**
3. **Increased reliability** – graceful degradation or fault tolerance

Two types:

1. **Asymmetric Multiprocessing** – each processor is assigned a specific task.
2. **Symmetric Multiprocessing** – each processor performs all tasks



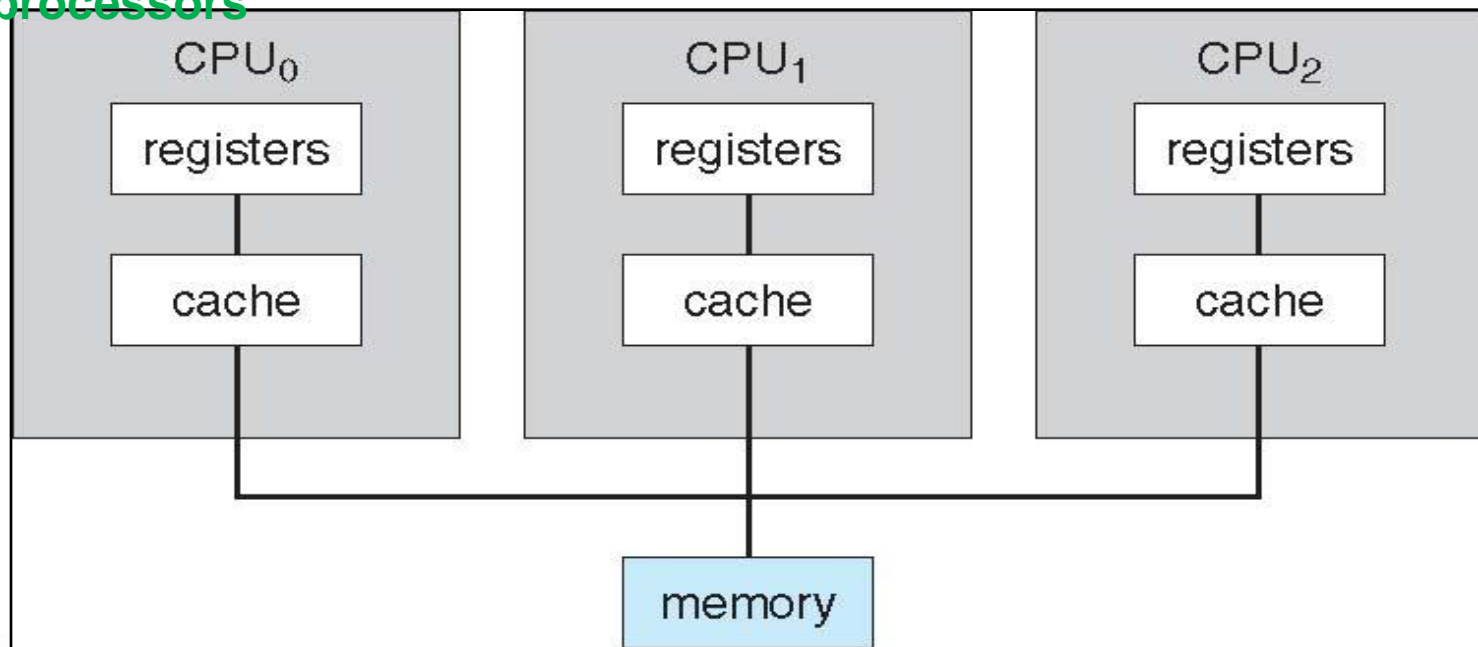
**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



▪ Symmetric Multiprocessing Architecture

- **boss** processor controls the system; the other processors either look to the boss for instruction or have predefined tasks
- **boss** processor schedules and allocates work to the **worker processors**



**PRESIDENCY
UNIVERSITY**

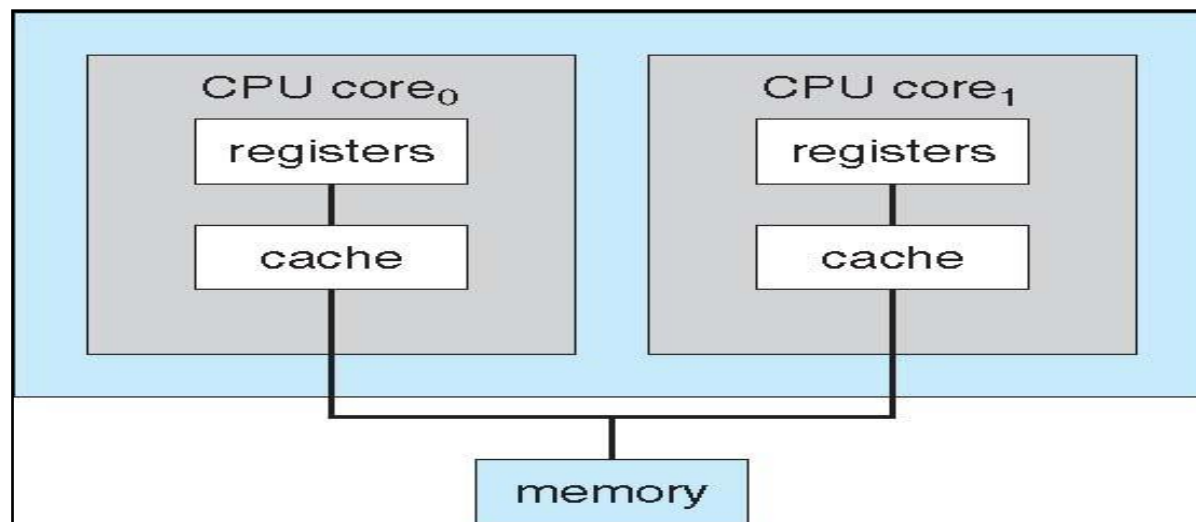
Private University Estd. in Karnataka State by Act No. 41 of 2013

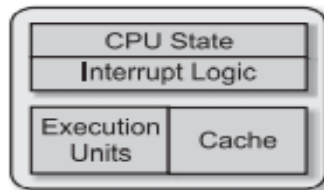


❑ Dual-Core Design

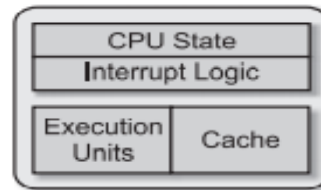
- Multi-chip and **multicore**
- Systems containing all chips
 - Chassis containing multiple separate systems

- **each core has its own register set as well as its own local cache**

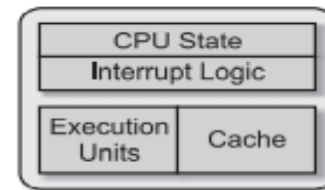




A) Single Core



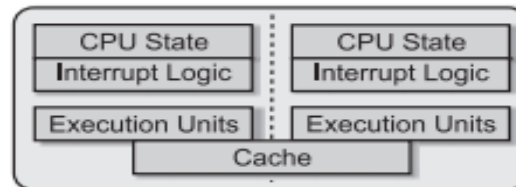
B) Multiprocessor



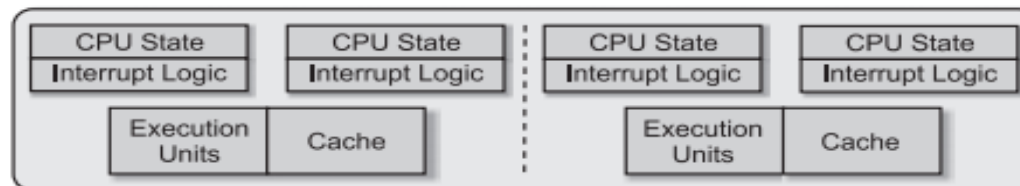
C) Hyper-Threading Technology



D) Multi-core



E) Multi-core with Shared Cache



F) Multi-core with Hyper-Threading Technology



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



❑ Clustered Systems

- Like multiprocessor systems, but multiple systems working together
- composed of two or more individual systems—or nodes—joined together (**loosely coupled**)
 - Usually sharing storage via a **storage-area network (SAN)**
 - Provides a **high-availability** service which survives failures
 - **Asymmetric clustering** has one machine in hot-standby mode
 - **Symmetric clustering** has multiple nodes running applications, monitoring each other- **more efficient**
 - Some clusters are for **high-performance computing (HPC)**
 - Applications must be written to use **parallelization**
 - Some have **distributed lock manager (DLM)** to avoid conflicting operations

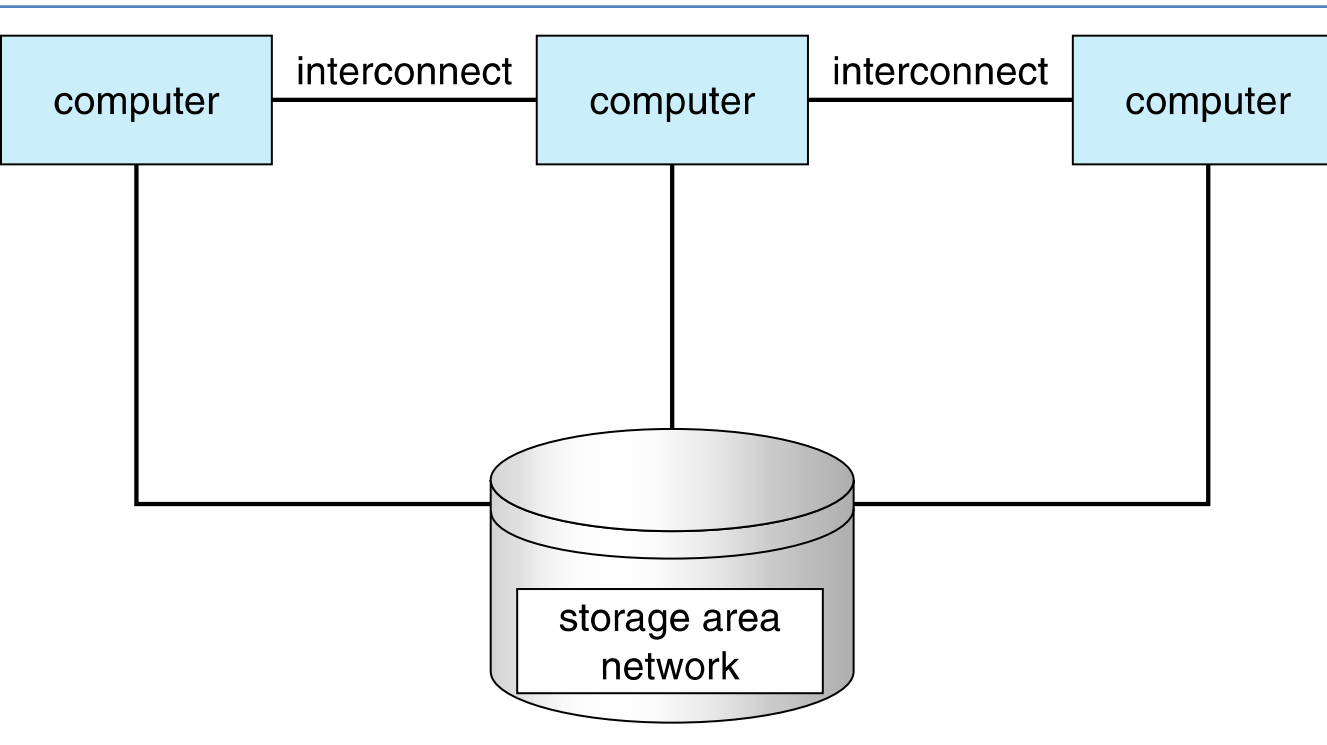


**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



• Clustered Systems



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013

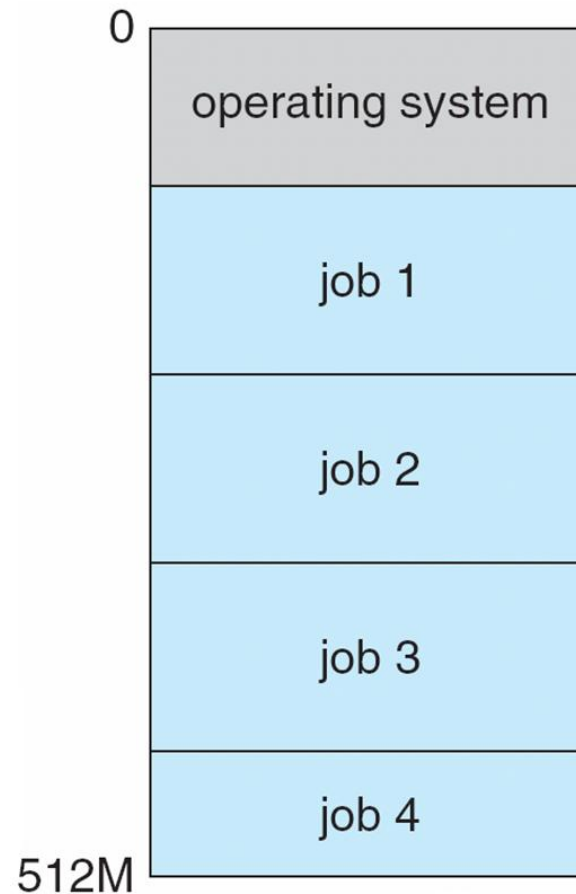


❑ Operating System Structure

- **Multiprogramming (Batch system)** needed for efficiency
 - Single user cannot keep CPU and I/O devices busy at all times
 - Multiprogramming organizes jobs (code and data) so CPU always has one to execute
 - A subset of total jobs in system is kept in memory
 - One job selected and run via **job scheduling**
 - When it has to wait (for I/O for example), OS switches to another job
- **Timesharing (multitasking):** logical extension of Multiprogramming
 - CPU switches jobs so frequently that users can interact with each job while it is running, creating **interactive** computing
 - **Response time** should be < 1 second
 - Each user has at least one program executing in memory ⇒ **process**
 - If several jobs ready to run at the same time ⇒ **CPU scheduling**
 - If processes don't fit in memory, **swapping** moves them in and out to run
 - **Virtual memory** allows execution of processes not completely in memory



Memory Layout for Multiprogrammed System



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



□ Operating-System Operations

- Modern operating systems are **Interrupt driven** (hardware and software)
 - **Hardware interrupt by one of the devices**
 - **Software interrupt (exception or trap):**
 - Software error (e.g., division by zero)
 - Request for operating system service
 - Other process problems include infinite loop, processes modifying each other or the operating system

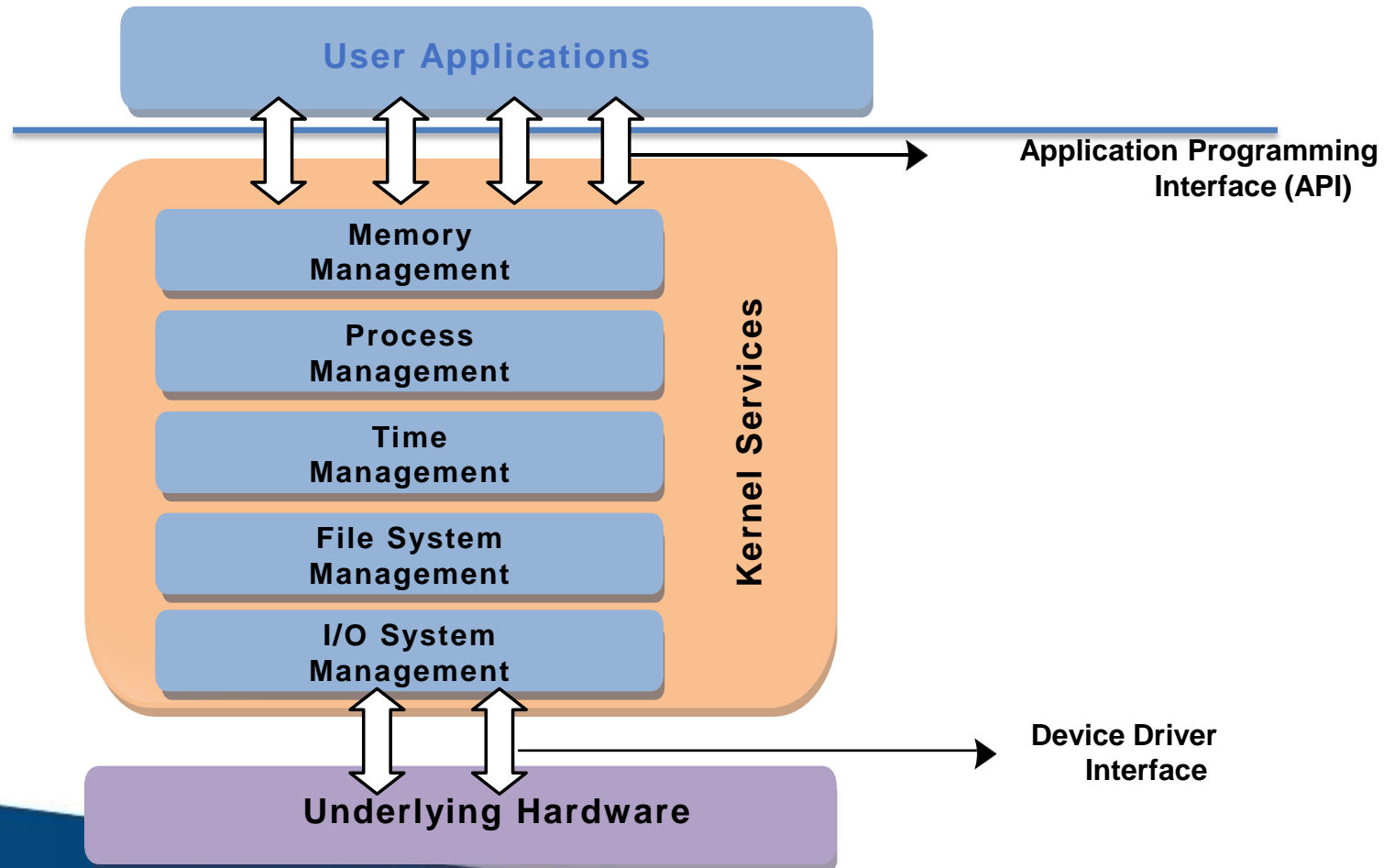


**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



Figure gives an insight into the basic components of an operating system and their interfaces with rest of the world



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



Operating-System Operations (cont.)

- **Dual-mode** operation allows OS to protect itself and other system components
 - **User mode** and **kernel mode**
 - **Mode bit** provided by hardware
 - Provides ability to distinguish when system is running **user code** or **kernel code**
 - Some instructions designated as **privileged**, only executable in kernel mode
 - **System call changes** mode to kernel, return from call resets it to user
- Increasingly CPUs support multi-mode operations
 - i.e. **virtual machine manager (VMM)** mode for guest **VMs**

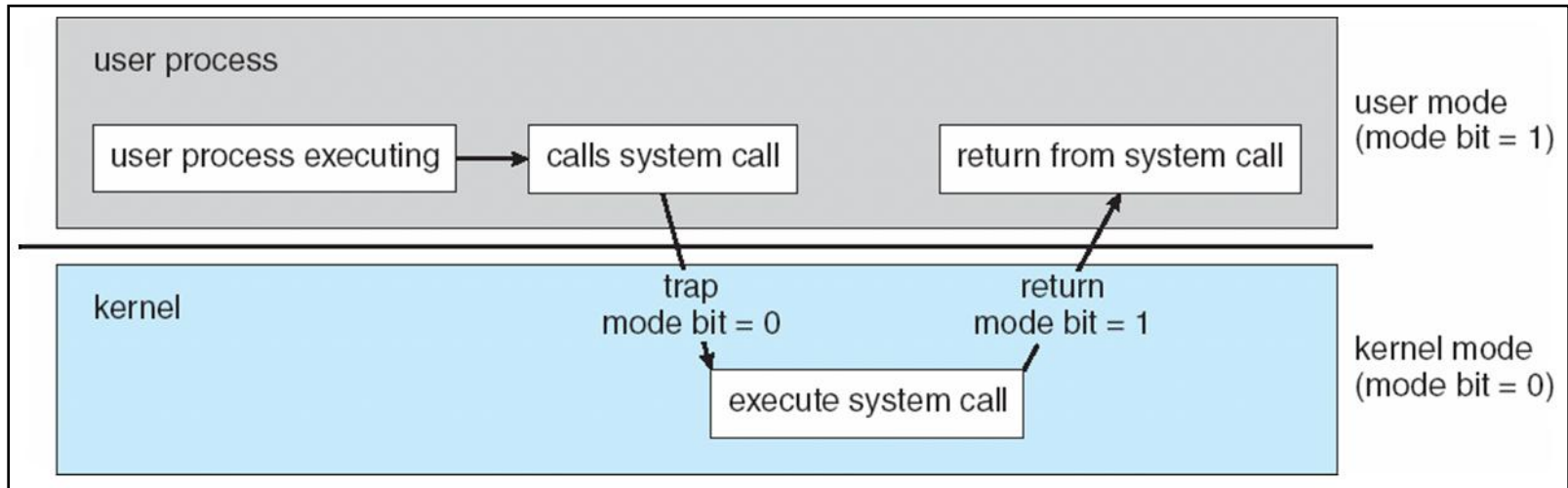


**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



❑ Transition from User to Kernel



- **Timer** to prevent infinite loop / process hogging resources
 - Timer is set to interrupt the computer after some time period
 - Keep a counter that is decremented by the physical clock.
 - Operating system set the counter (privileged instruction)
 - When counter zero generate an interrupt



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



□ Process Management

- A **process** is a program in execution
 - It is a unit of work within the system.
- Program is a *passive entity*, process is an *active entity*.
- Process needs resources to accomplish its task
 - CPU, memory, I/O, files
 - Initialization data
- **Process termination** requires reclaim of any reusable resources



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



□ Process Management

- Single-threaded process has one **program counter** specifying location of next instruction to execute
 - Process executes instructions sequentially, one at a time, until completion
- Multi-threaded process has **one program counter per thread**
- Typically system has many processes, some user, some operating system running concurrently on one or more CPUs
 - Concurrency by multiplexing the CPUs among the processes / threads



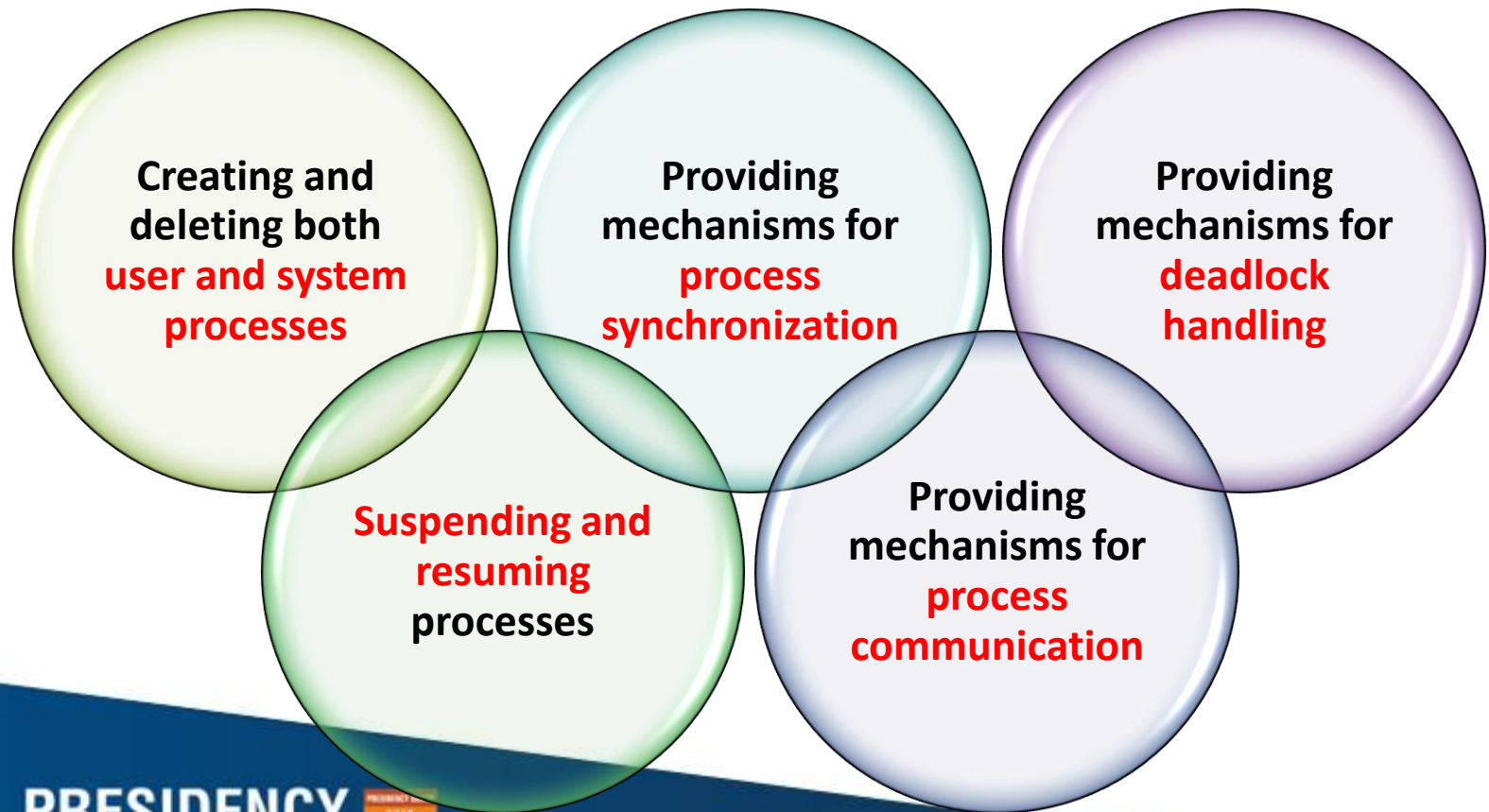
**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



❑ Process Management Activities

- OS is responsible for the following activities in connection with process management:



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



□ Memory Management

- To execute a program all (or part) of the **instructions** must be in memory
- All (or part) of the **data** that is needed by the program must be in memory.
- Memory management determines what is in memory and when
 - Optimizing CPU utilization and computer response to users

Memory
management
activities

- Keeping track of which parts of memory are currently **being used and by whom**
- Deciding which processes (or parts thereof) and data to **move into and out** of memory
- Allocating and deallocating memory space as needed



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



□ Storage Management

- OS provides uniform, logical view of information storage
 - Abstracts physical properties to logical storage unit - **file**
 - **File- Collection of related information defined by its creator**
- The operating system maps files onto physical media and accesses these files via the storage devices
- **File-System management**
 - Files usually organized into directories
 - Access control on most systems to determine who can access what
 - **OS activities include**
 - Creating and deleting files and directories
 - Primitives to manipulate files and directories
 - Mapping files onto secondary storage
 - Backup files onto stable (non-volatile) storage media



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



□ Mass-Storage Management

- Usually disks used to store data that does not fit in main memory or data that must be kept for a “**long**” period of time
- Proper management is of central importance
- Entire speed of computer operation hinges on disk subsystem and its algorithms

OS activities

- Free-space management
- Storage allocation
- Disk scheduling
- **Some storage need not be fast**
 - Tertiary storage includes optical storage, magnetic tape
 - Still must be managed – by OS or applications
 - Varies between WORM (write-once, read-many-times) and RW (read-write)



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



Performance of Various Levels of Storage

Level	1	2	3	4	5
Name	registers	cache	main memory	solid state disk	magnetic disk
Typical size	< 1 KB	< 16MB	< 64GB	< 1 TB	< 10 TB
Implementation technology	custom memory with multiple ports CMOS	on-chip or off-chip CMOS SRAM	CMOS SRAM	flash memory	magnetic disk
Access time (ns)	0.25 - 0.5	0.5 - 25	80 - 250	25,000 - 50,000	5,000,000
Bandwidth (MB/sec)	20,000 - 100,000	5,000 - 10,000	1,000 - 5,000	500	20 - 150
Managed by	compiler	hardware	operating system	operating system	operating system
Backed by	cache	main memory	disk	disk	disk or tape

Movement between levels of storage hierarchy can be explicit or implicit



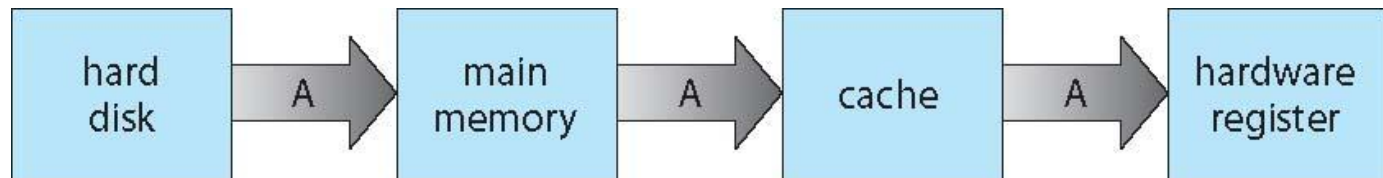
**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



❑ Migration of data “A” from Disk to Register

- Multitasking environments must be careful to use most recent value, no matter where it is stored in the storage hierarchy



Multiprocessor environment must provide **cache coherency** in hardware such that all CPUs have the most recent value in their cache

- Distributed environment situation even more complex



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



I/O Subsystem

- One purpose of OS is to **hide peculiarities** of hardware devices from the user
- I/O subsystem responsible for
 - **Memory management of I/O including buffering, caching, spooling** (the overlapping of output of one job with input of other jobs)
 - **General device-driver interface**
 - **Drivers for specific hardware devices**



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



□ Protection and Security

- **Protection** – mechanism for **controlling access of processes** or users to resources defined by the OS
- **Security** – defense of the system against internal and external attacks
 - Huge range, including denial-of-service, worms, viruses, identity theft, theft of service
- **Systems generally first distinguish among users, to determine who can do what**
 - User identities (**user IDs**, security IDs) include name and associated number, one per user
 - User ID then associated with all files, processes of that user to determine access control
 - Group identifier (**group ID**) allows set of users to be defined and controls managed, then also associated with each process, file
 - **Privilege escalation** allows user to change to effective ID with more rights



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



❑ Computing Environments

- TRADITIONAL

- Stand-alone general purpose machines
- But **blurred** as most systems interconnect with others (i.e., the Internet)
- **Portals** provide web access to internal systems
- **Network computers (thin clients)** are like Web terminals
- Mobile computers interconnect via **wireless networks**
- Networking becoming ubiquitous – even home systems use **firewalls** to protect home computers from Internet attacks



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



❑ COMPUTING ENVIRONMENTS – MOBILE

- Handheld smartphones, tablets, etc
- What is the functional difference between them and a “**traditional**” laptop?
- Extra feature – more OS features (GPS, Gyroscope)
- Allows new types of apps like *augmented reality*
- Use IEEE 802.11 wireless, or cellular data networks for connectivity
- Leaders are **Apple iOS** and **Google Android**



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013

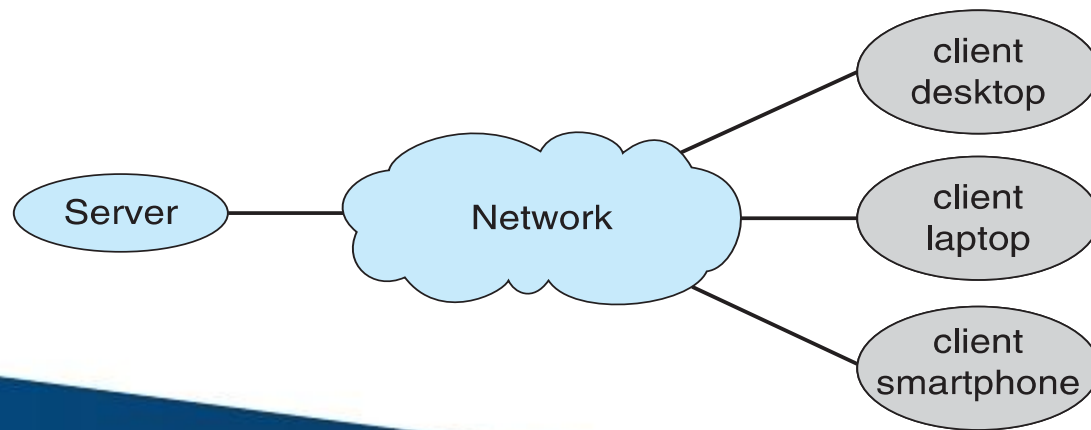


- **Distributed computing**
 - Collection of separate, possibly heterogeneous, systems networked together
 - **Network** is a communications path, **TCP/IP** most common
 - **Local Area Network (LAN)**
 - **Wide Area Network (WAN)**
 - **Metropolitan Area Network (MAN)**
 - **Personal Area Network (PAN)**
 - **Network Operating System** provides features between systems across network
 - Communication scheme allows systems to exchange messages
 - Illusion of a single system



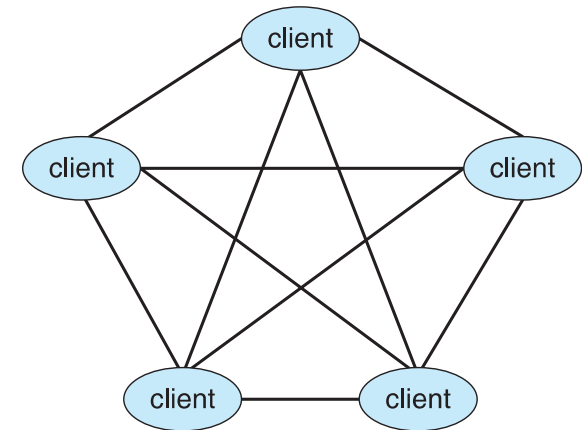
■ Client-Server Computing

- Dumb terminals supplanted by smart PCs
- Many systems now **servers**, responding to requests generated by **clients**
 - ▶ **Compute-server system** provides an interface to client to request services (i.e., database)
 - ▶ **File-server system** provides interface for clients to store and retrieve files



□ Computing Environments – Peer-to-Peer

- Another model of distributed system
- P2P does not distinguish clients and servers
 - All nodes are considered peers
 - May each act as client, server or both
 - Node must join P2P network
 - Registers its service with **central lookup service on network**, or
 - Broadcast request for service and respond to requests for service via **discovery protocol**
 - Examples include Napster and Gnutella, **Voice over IP (VoIP)** such as Skype



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



□ Computing Environments – **Virtualization**

- Allows operating systems to **run applications within other OSes**
 - **Vast and growing industry**
- **Emulation** used when source CPU type is **different** from target type (i.e. PowerPC to Intel x86)
 - **Generally slowest method**
 - **When computer language not compiled to native code – Interpretation**
- **Virtualization** – OS natively compiled for CPU, running **guest OSes** also natively compiled
 - Consider VMware running WinXP guests, each running applications, all on native WinXP **host** OS
 - **VMM** (virtual machine Manager) provides virtualization services

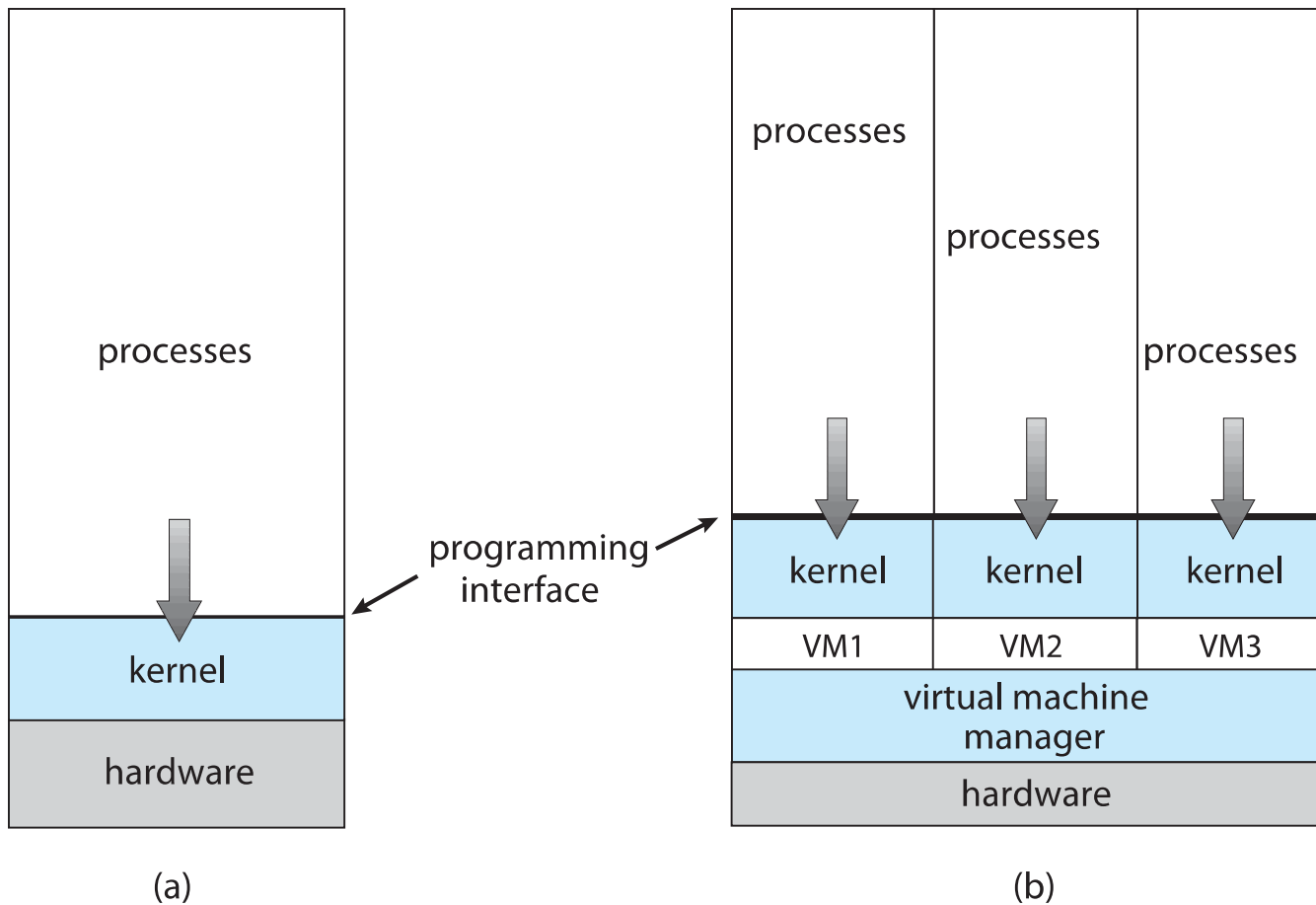


**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



Computing Environments - **Virtualization**



Computing Environments – Cloud Computing

- **Delivers computing, storage, even apps as a service across a network**
- Logical extension of virtualization
- Many types
 - **Public cloud** – available via Internet to anyone willing to pay
 - **Private cloud** – run by a company for the company's own use
 - **Hybrid cloud** – includes both public and private cloud components
 - **Software as a Service (SaaS)** – one or more applications available via the Internet (i.e., word processor)
 - **Platform as a Service (PaaS)** – **software stack ready** for application use via the Internet (i.e., a database server)
 - **Infrastructure as a Service (IaaS)** – servers or storage available over Internet (i.e., storage available for backup use)}

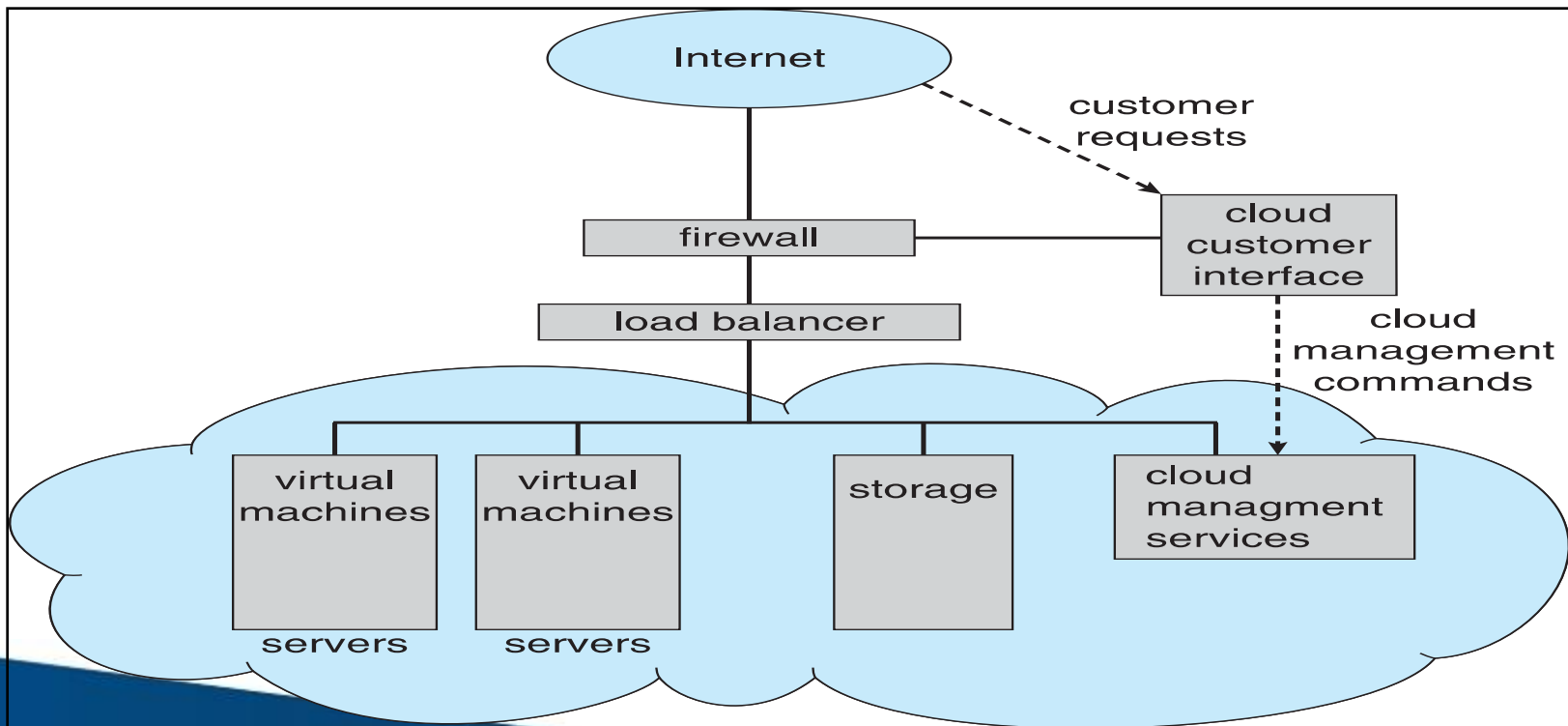


**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



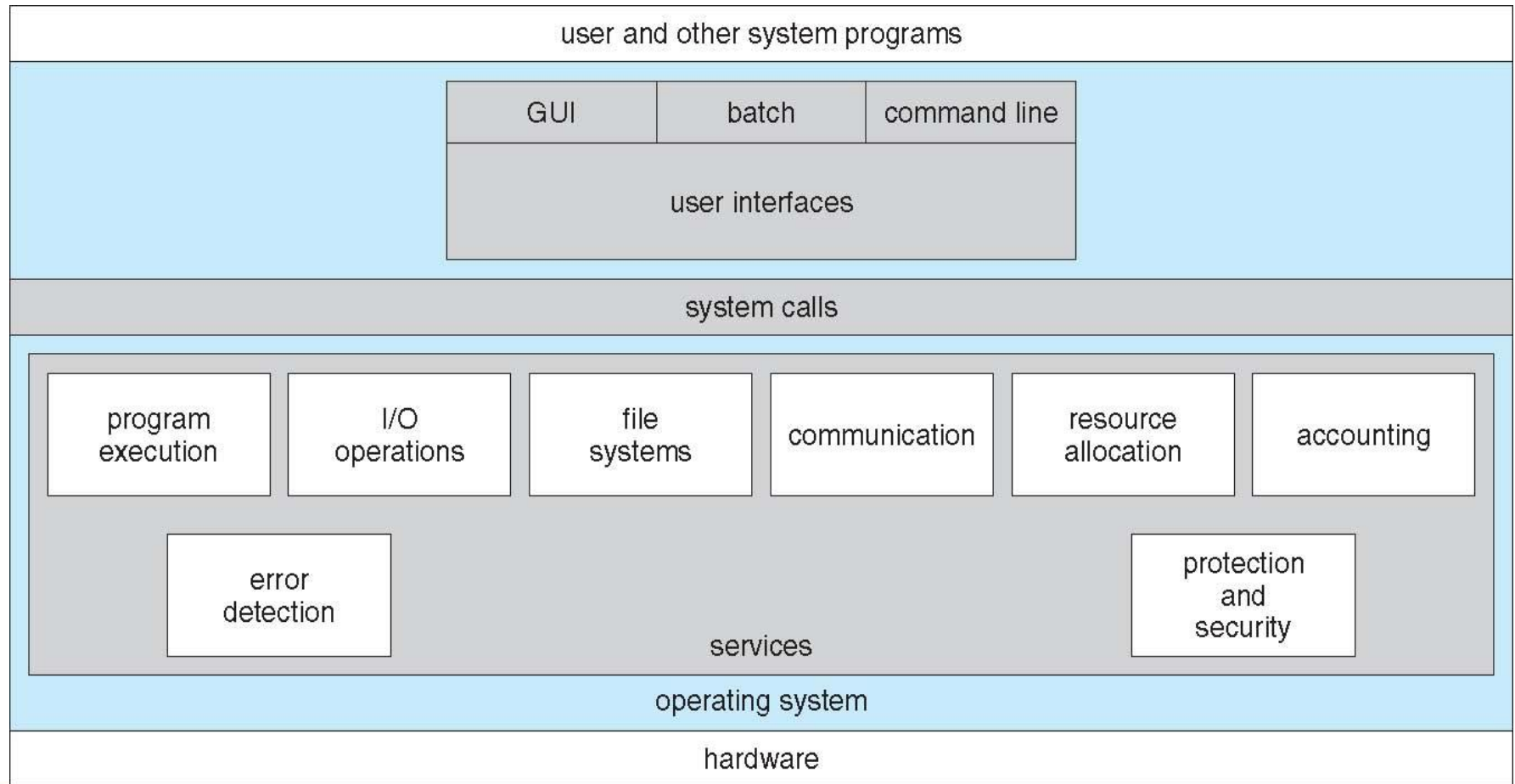
- Cloud computing environments composed of traditional OSES, plus VMMs, plus cloud management tools
 - Internet connectivity requires security like firewalls
 - Load balancers** spread traffic across multiple applications



- Real-time embedded systems most prevalent form of computers
 - Vary considerable, special purpose, limited purpose OS, **real-time OS**
 - Usages are expanding
- Many other special computing environments as well
 - Some have OSes, some perform tasks without an OS
- Real-time OS has well-defined fixed time constraints
 - Processing **must be done within constraint**
 - Correct operation only if constraints met



❑ A View of Operating System Services



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



□ System Calls

- Programming interface to the services **provided by the OS**
- Typically written in a high-level language (C or C++)
- Mostly accessed by programs via a high-level **Application Programming Interface (API)** rather than direct system call use
- Three most common APIs are
 - Win32 API for Windows,
 - POSIX API for POSIX-based systems
 - Java API for the Java virtual machine (JVM)



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



Example of Standard API

EXAMPLE OF STANDARD API

As an example of a standard API, consider the `read()` function that is available in UNIX and Linux systems. The API for this function is obtained from the `man` page by invoking the command

```
man read
```

on the command line. A description of this API appears below:

```
#include <unistd.h>

ssize_t      read(int fd, void *buf, size_t count)
```

return value	function name	parameters
-----------------	------------------	------------

A program that uses the `read()` function must include the `unistd.h` header file, as this file defines the `ssize_t` and `size_t` data types (among other things). The parameters passed to `read()` are as follows:

- `int fd`—the file descriptor to be read
- `void *buf`—a buffer where the data will be read into
- `size_t count`—the maximum number of bytes to be read into the buffer

On a successful read, the number of bytes read is returned. A return value of 0 indicates end of file. If an error occurs, `read()` returns `-1`.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



❑ System Call Implementation

- Typically, a number associated with each system call
 - **System-call interface** maintains a table indexed according to these numbers
- The system call interface invokes the intended system call in OS kernel and returns status of the system call and any return values
- The caller need know nothing about how the system call is implemented
 - **Just needs to obey API and understand what OS will do as a result call**
 - **Most details of OS interface hidden from programmer by API**
 - Managed by run-time support library (set of functions built into libraries included with compiler)

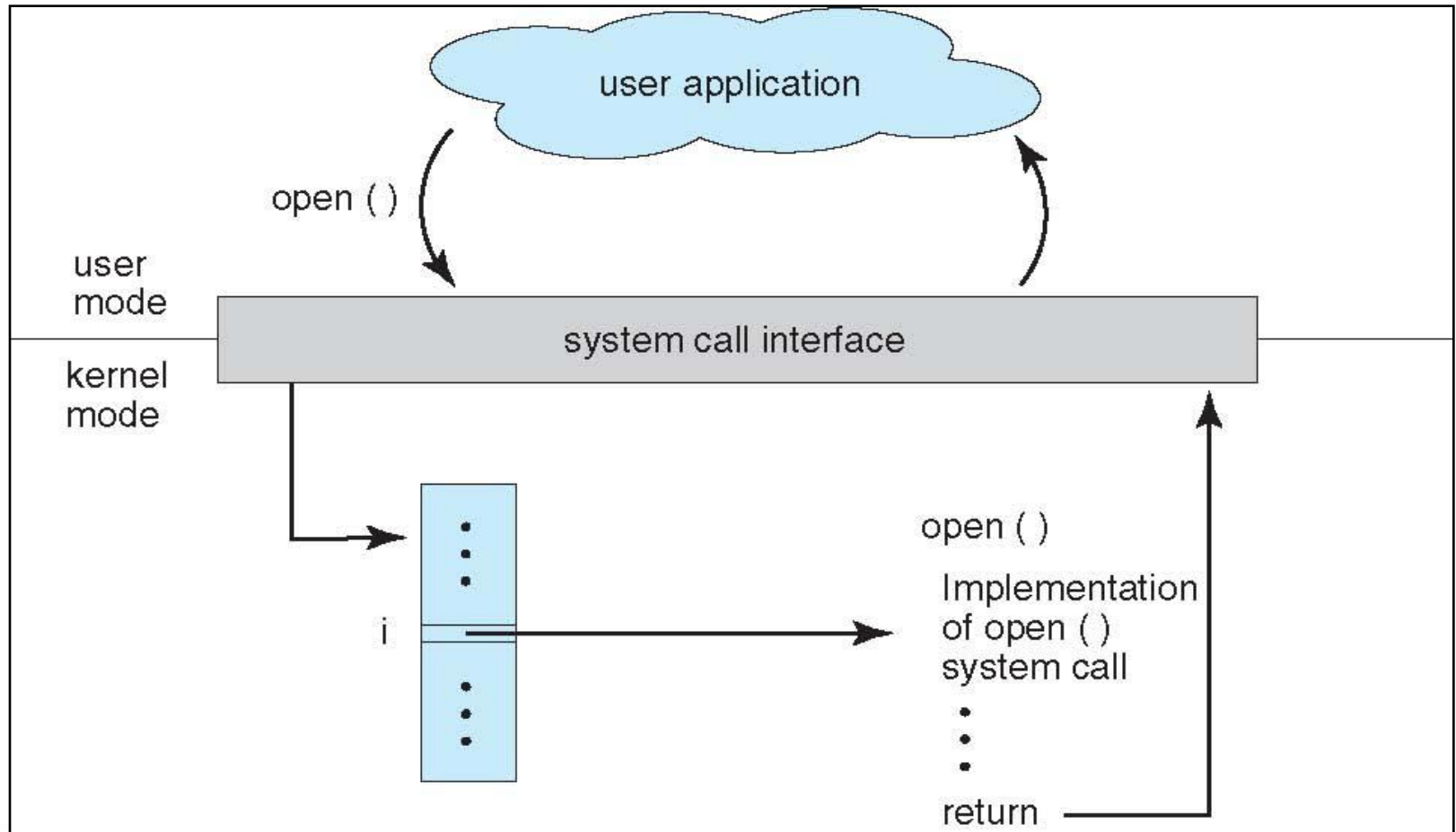


**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



API – System Call – OS Relationship



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



▪ Types of System Calls

• Process control

- create process, terminate process
- end, abort
- load, execute
- get process attributes, set process attributes
- wait for time
- wait event, signal event
- allocate and free memory
- Dump memory if error
- **Debugger** for determining **bugs, single step** execution
- **Locks** for managing access to shared data between processes



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



- File management
 - create file, delete file
 - open, close file
 - read, write, reposition
 - get and set file attributes
- Device management
 - request device, release device
 - read, write, reposition
 - get device attributes, set device attributes
 - logically attach or detach devices



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



- Information maintenance
 - get time or date, set time or date
 - get system data, set system data
 - get and set process, file, or device attributes
- Communications
 - create, delete communication connection
 - send, receive messages if **message passing model** to **host name** or **process name**
 - From **client** to **server**
 - **Shared-memory model** create and gain access to memory regions
 - transfer status information
 - attach and detach remote devices



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



- Protection
 - Control access to resources
 - Get and set permissions
 - Allow and deny user access



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



Examples of Windows and Unix System Calls

	Windows	Unix
Process Control	CreateProcess() ExitProcess() WaitForSingleObject()	fork() exit() wait()
File Manipulation	CreateFile() ReadFile() WriteFile() CloseHandle()	open() read() write() close()
Device Manipulation	SetConsoleMode() ReadConsole() WriteConsole()	ioctl() read() write()
Information Maintenance	GetCurrentProcessID() SetTimer() Sleep()	getpid() alarm() sleep()
Communication	CreatePipe() CreateFileMapping() MapViewOfFile()	pipe() shmget() mmap()
Protection	SetFileSecurity() InitializeSecurityDescriptor() SetSecurityDescriptorGroup()	chmod() umask() chown()



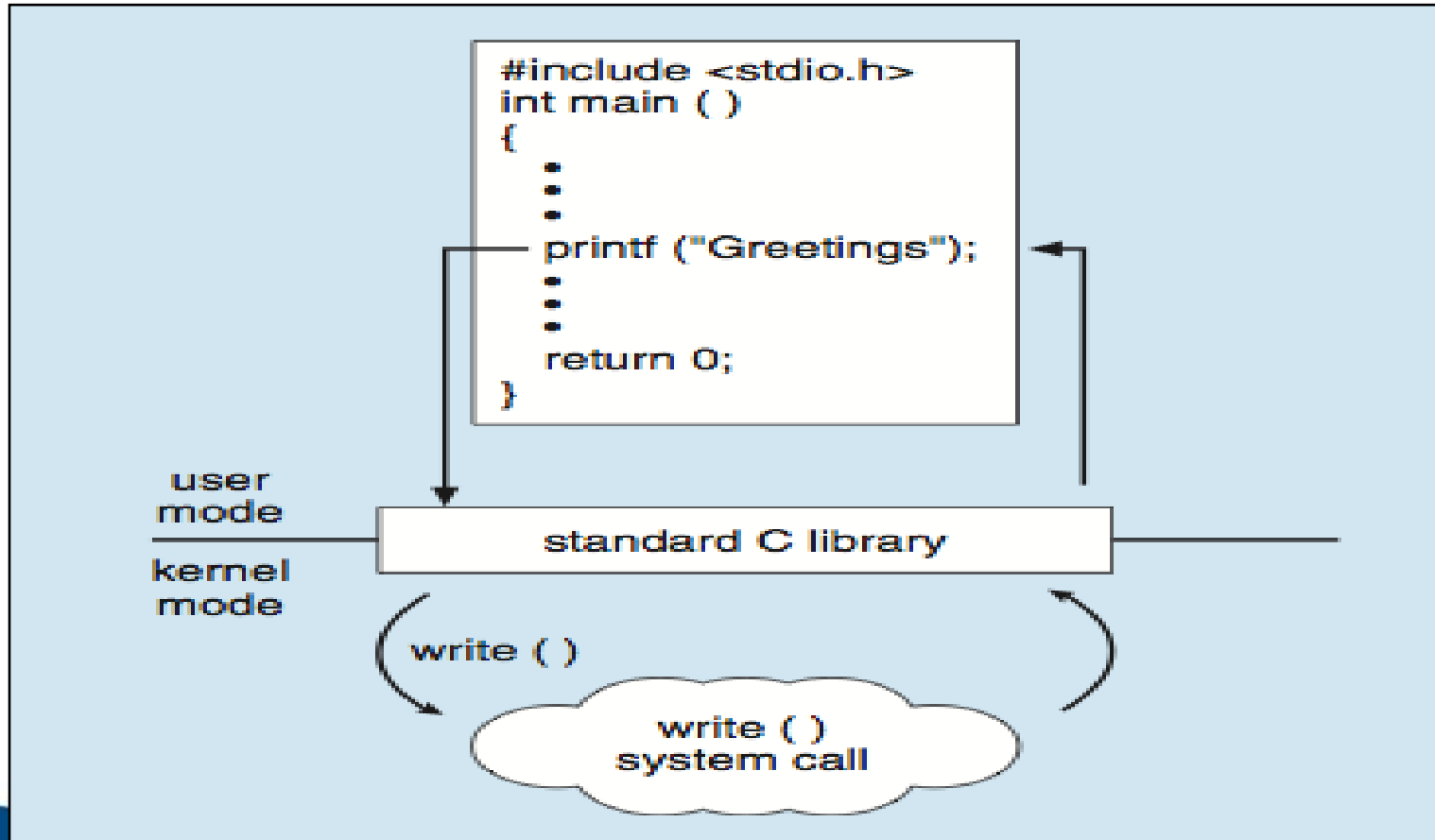
**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



Standard C Library Example

C program invoking **printf()** library call, which calls **write()** system call



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



❑ Operating System Structure

- General-purpose OS is **very large program**
- Various ways to structure ones
 - Simple structure – MS-DOS
 - More complex -- UNIX
 - Layered – an abstraction
 - Microkernel -Mach



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013

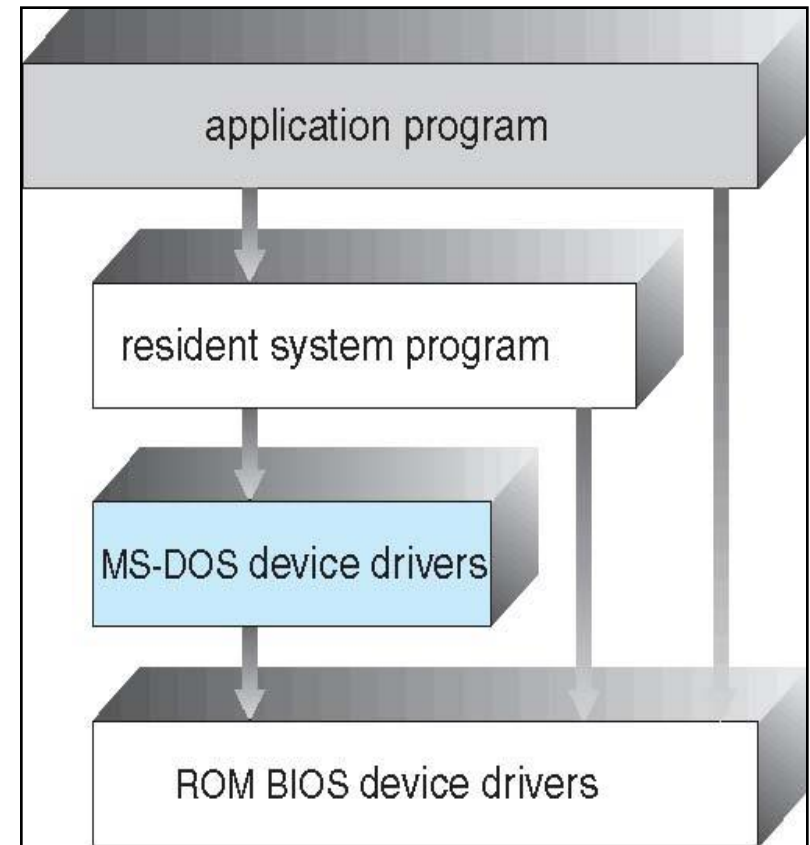


❑ Simple Structure -- MS-DOS

MS-DOS – written to provide the most functionality in the least space

Not divided into modules

Although MS-DOS has some structure, its interfaces and levels of functionality are not well separated



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



❑ Non Simple Structure -- UNIX

UNIX – limited by hardware functionality, the original UNIX operating system had limited structuring.

The UNIX OS consists of two separable parts

- I. **Systems programs**
- II. **The kernel**
 - Consists of everything below the system-call interface and above the physical hardware
 - Provides the **file system, CPU scheduling, memory management**, and other operating-system functions; a large number of functions for one level



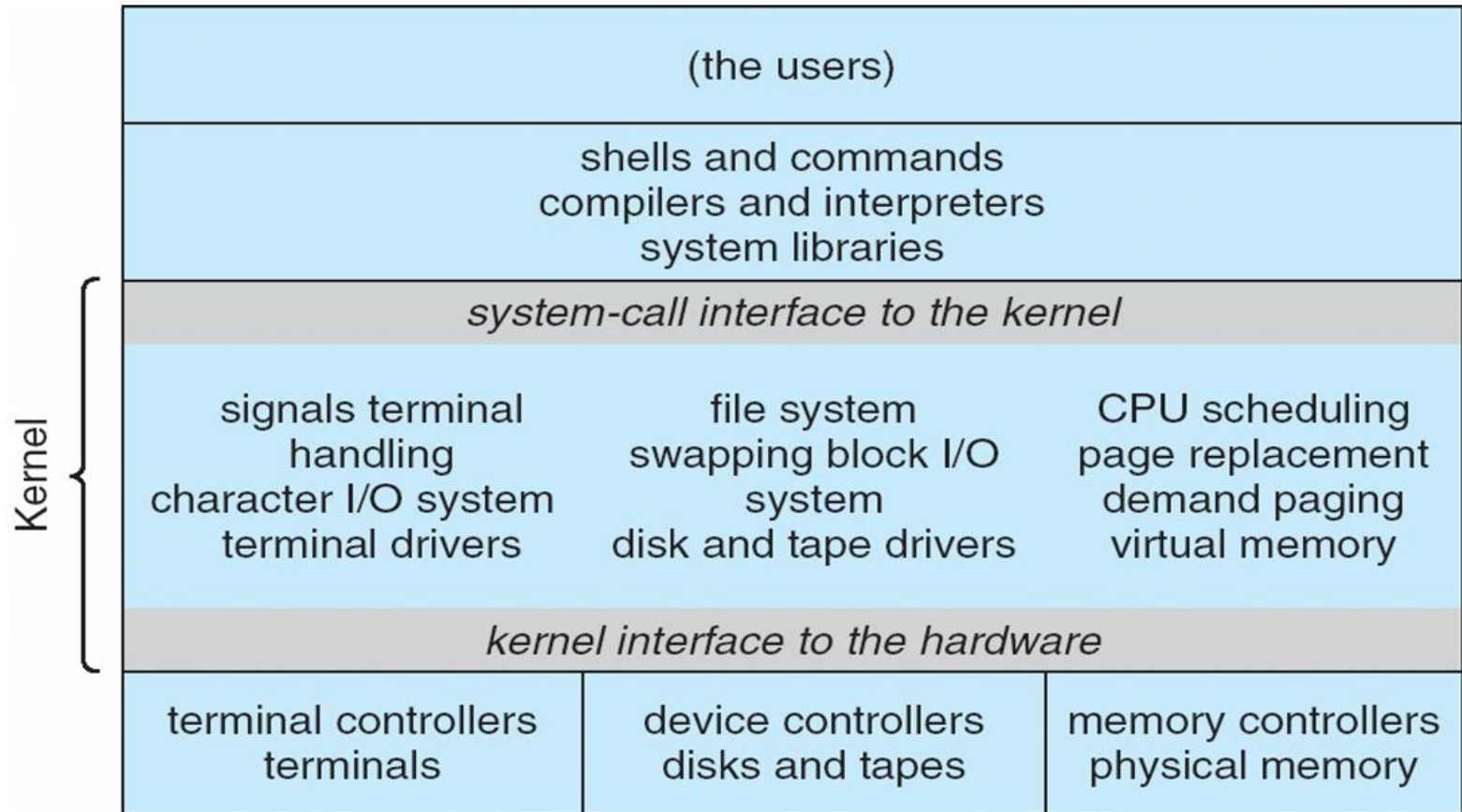
**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



❑ Traditional UNIX System Structure

Beyond simple but not fully layered



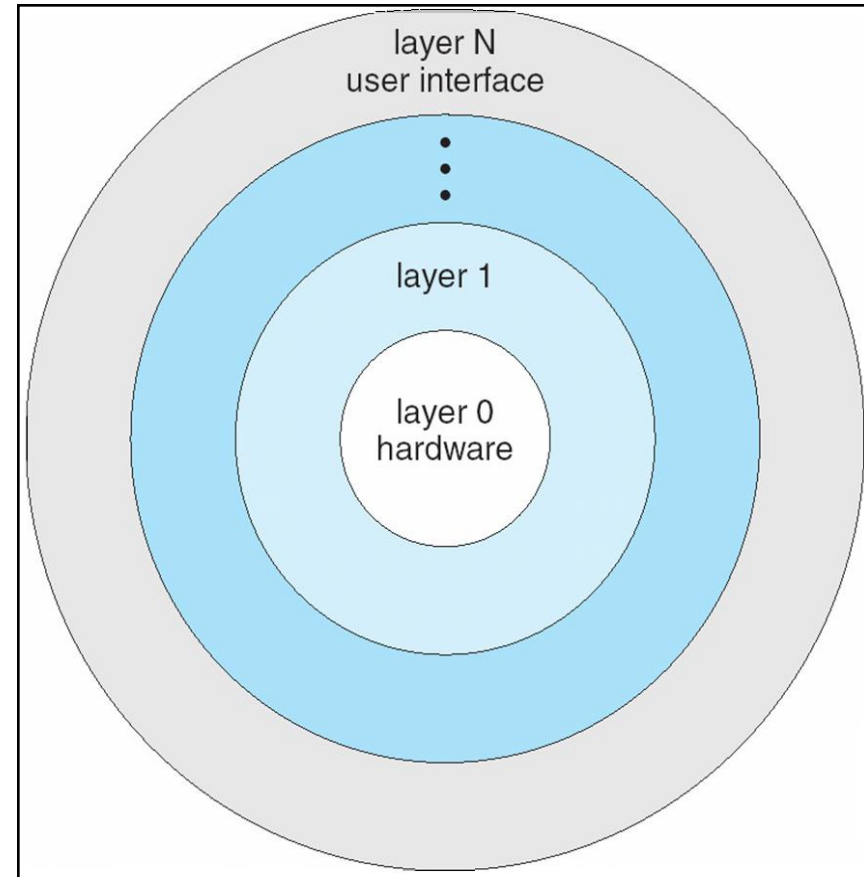
**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



❑ Layered Approach

- The operating system is divided into a number of layers (levels), each built on top of lower layers.
 - The bottom layer (layer 0), is the **hardware; the highest (layer N) is the user interface**
- With modularity, layers are selected such that each uses functions (operations) and services of only lower-level layers



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



❑ Microkernel System Structure

- Moves as much as from the **kernel** into **user space**
- **Mach** example of **microkernel**
 - Mac OS X kernel (**Darwin**) partly based on Mach
- Communication takes place between user modules using **message passing**
- Benefits:
 - Easier to extend a microkernel
 - Easier to port the operating system to new architectures
 - More reliable (less code is running in kernel mode)
 - More secure
- **Detriments:**
 - Performance overhead of user space to kernel space communication

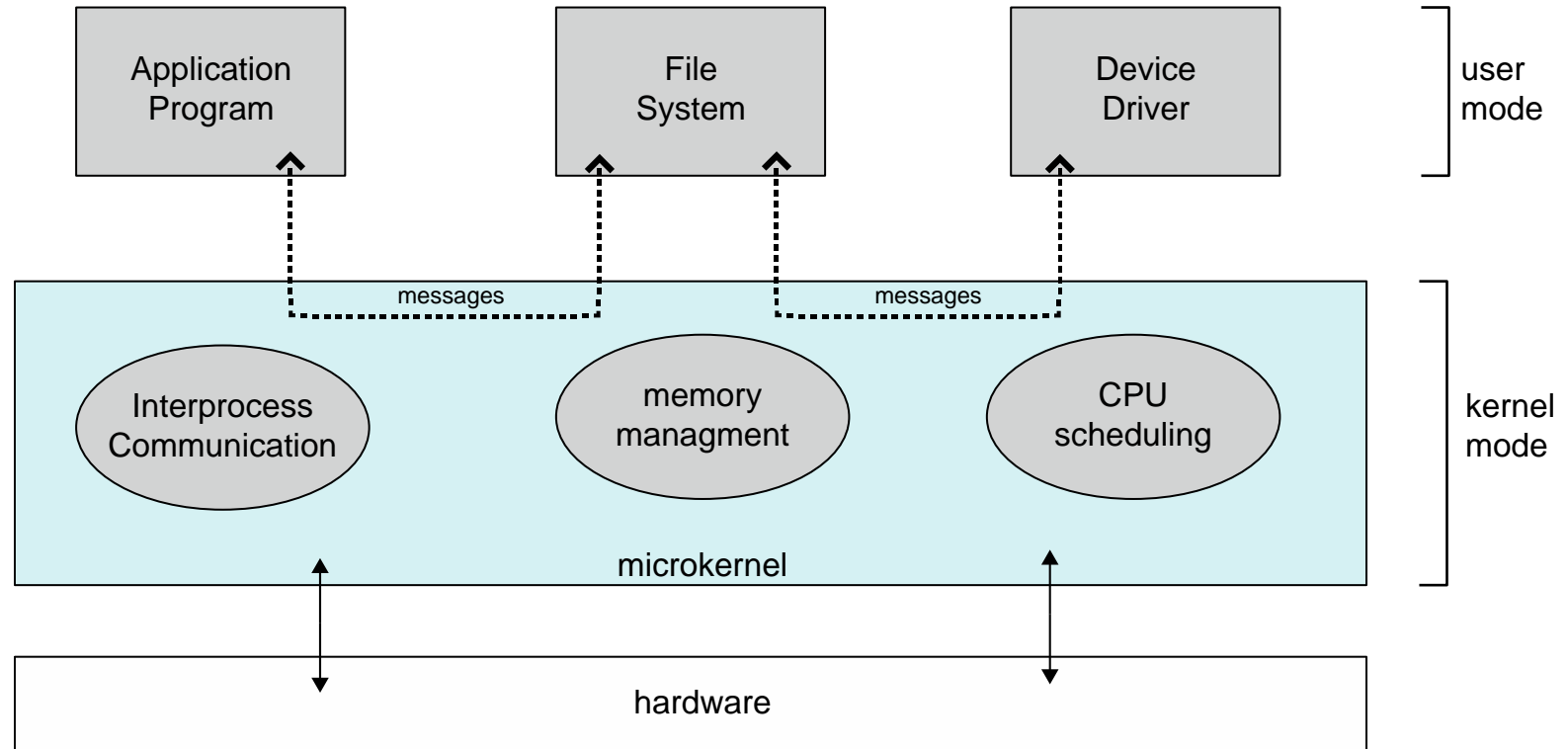


**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



❑ Microkernel System Structure

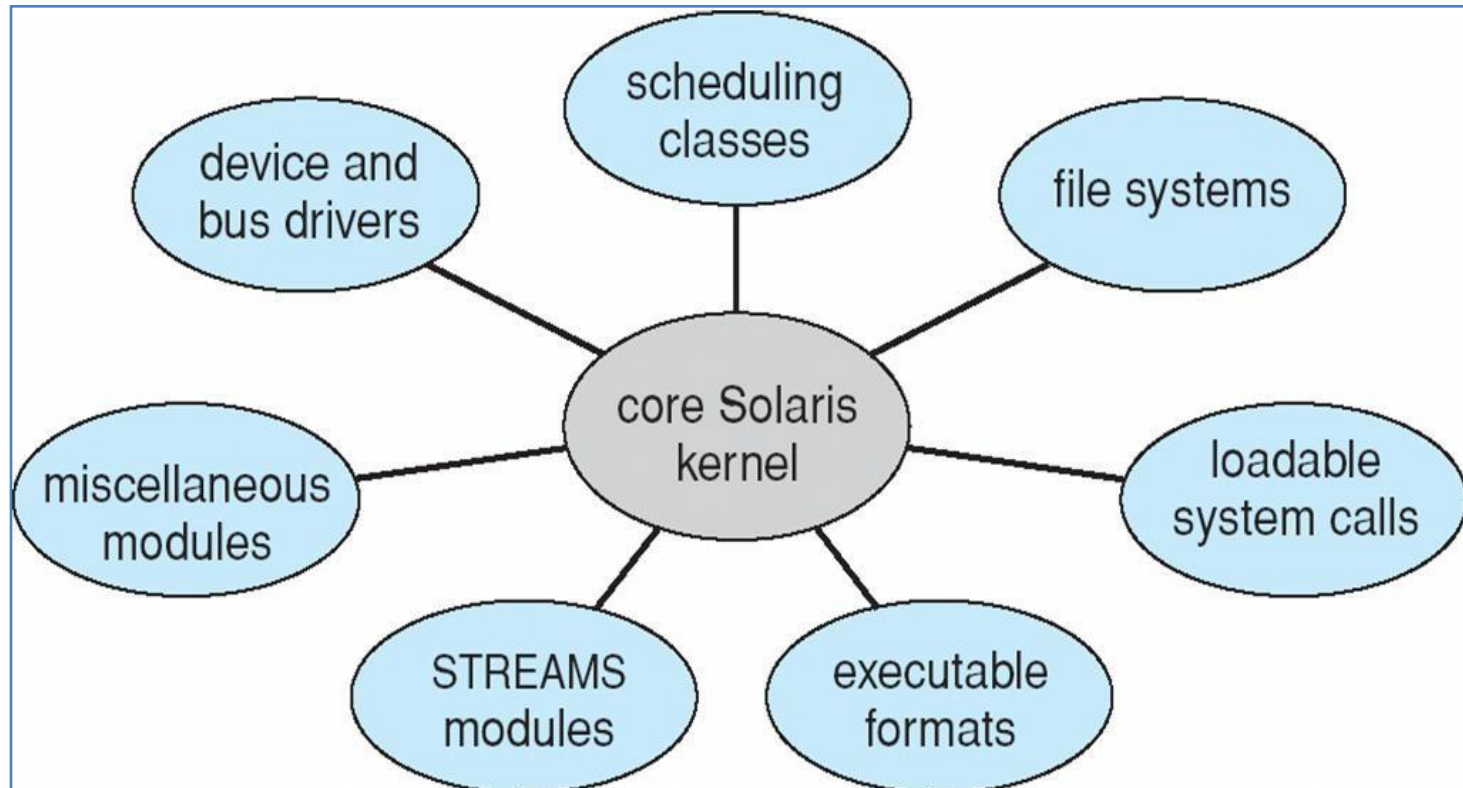


**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



❑ Solaris Modular Approach



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



- Apple mobile OS for *iPhone, iPad*
 - Structured on Mac OS X, added functionality
 - Does not run OS X applications natively
 - Also runs on different CPU architecture (ARM vs. Intel)
 - **Cocoa Touch** Objective-C API for developing apps
 - **Media services** layer for graphics, audio, video
 - **Core services** provides cloud computing, databases
 - Core operating system, based on Mac OS X kernel

Cocoa Touch

Media Services

Core Services

Core OS



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



□ Android

- Developed by Open Handset Alliance (**mostly Google**)
 - Open Source
- Similar stack to iOS
- Based on Linux kernel but modified
 - Provides process, memory, device-driver management
 - Adds power management
- Runtime environment includes core set of libraries and Dalvik virtual machine
 - Apps developed in Java plus Android API
 - Java class files compiled to Java bytecode then translated to executable then runs in Dalvik VM
- Libraries include frameworks for web browser (webkit), database (SQLite), multimedia, smaller libc



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



Android Architecture

Application Framework

Libraries

SQLite

openGL

surface
manager

media
framework

webkit

libc

Android runtime

Core Libraries

Dalvik
virtual machine



**PRESIDENCY
UNIVERSITY**

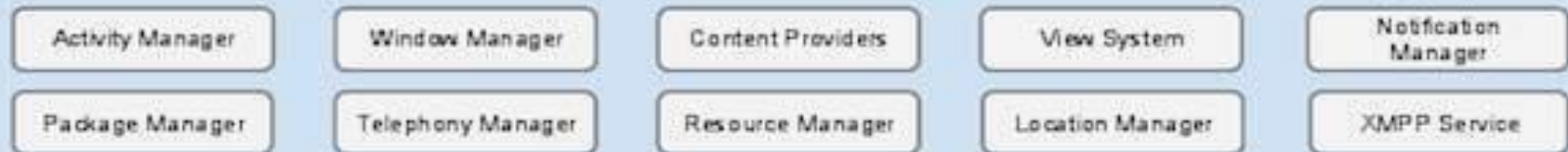
Private University Estd. in Karnataka State by Act No. 41 of 2013



Applications



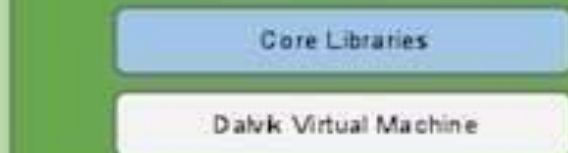
Application Framework



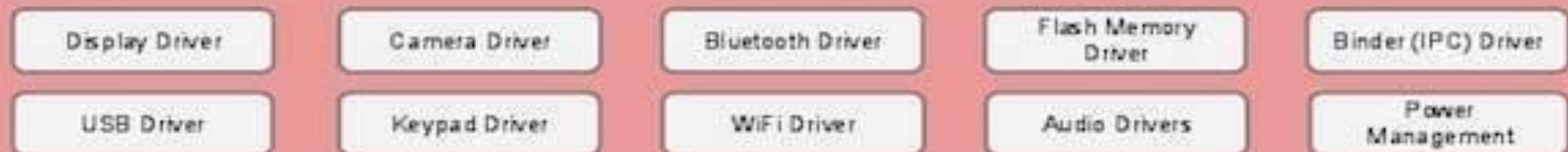
Libraries



Android Runtime



Linux Kernel



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



❑ System Programs[loaders, linkers...],

- System programs are
 - software programs to manage the computer's resources and provide services to other software programs or users.
 - System programs are an essential part of an operating system and include a wide range of utilities
 - **managing files,**
 - **allocating memory,**
 - **scheduling tasks, and**
 - **providing security**
- Loaders and linkers are two important types of system programs that are responsible for loading and linking programs into memory for execution.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



❑ System Programs[loaders, linkers...]

- **Loaders:**

- Loaders are programs that load executable files into memory for execution.
- When a user wants to run a program, the loader reads the program's executable file and transfers it from the storage device into memory.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



- **Linkers:**

- Linkers are programs that link object files together to create an executable file.
- Object files are produced by the compiler and contain the compiled code for a single module of the program.
- Linkers take these object files and combine them into a single executable file.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



• Compiler:

- software program that translates human-readable source code written in a high-level programming language into machine-readable code (object code) that a computer executes
 - compiling involves several stages, including lexical analysis, syntax analysis, semantic analysis, code generation, and optimization.
- Compilers are also used to create libraries, operating systems, and other software systems.
- Some examples of popular compilers include GCC, Clang, Microsoft Visual C++, and Java Compiler.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



❑ Introduction to Linux OS

- Linux is a free and open-source operating system (OS) based on the Unix operating system. It was created in 1991 by Linus Torvalds,
- widely used in a variety of applications, from servers to desktops to embedded systems.
- One of the key features of Linux is its modularity and flexibility.
- Linux **kernel is the core of the operating system**,
 - and it can be customized and configured in many different ways to meet the needs of specific applications
- Linux is also known for its stable, robustness and reliability.
- key advantage of Linux is its **security**



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



❑ Introduction to Linux OS

- Some of the most popular Linux distributions include
 - **Ubuntu,**
 - **Debian,**
 - **CentOS,**
 - **Fedora, and**
 - **Red Hat Enterprise Linux.**
- Linux is also used in a variety of other applications, including embedded systems, smartphones, and other mobile devices



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



❑ Linux ARCHITECTURE

- The Linux operating system has a modular architecture consisting of several layers, each with its own set of functions and responsibilities.
- main components of the Linux architecture:
- **Kernel:** The kernel is the core of the Linux operating system.
 - responsible for managing system resources such as memory, CPU, and I/O devices.
 - also provides an interface for system calls and manages processes, threads, and synchronization.
- **Shell:** The shell is the command-line interface (CLI) of the Linux system.
 - It is a program that allows users to interact with the operating system by typing commands.
 - also provides a scripting language that allows users to automate tasks and create complex programs.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



- **Libraries:** Linux includes a set of libraries that provide common functionality to applications.
 - libraries contain pre-written code that can be called by applications, such as functions for handling input/output or networking.
- **Applications:** Applications are user-level programs that run on top of the Linux operating system.
 - Can range from simple command-line utilities to complex graphical applications.
- **File system:** Linux file system is hierarchical and organized in a tree-like structure.
 - The root directory is denoted by a forward slash (/), and all other directories and files are organized under it.
 - Linux supports many file systems, like **ext4**, **XFS** and etc

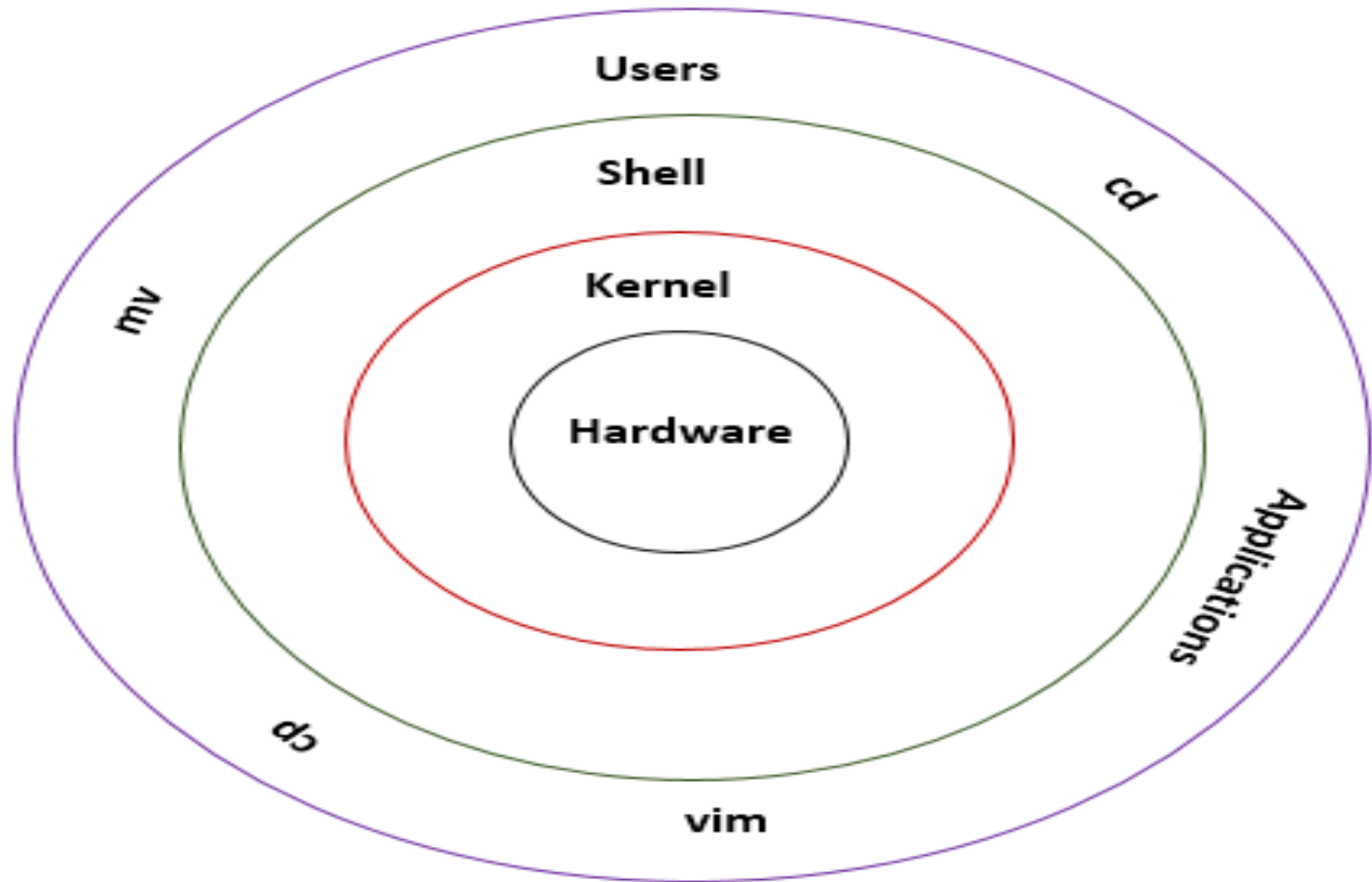


**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013



Device drivers: Device drivers are responsible for managing hardware devices such as disk drives, network cards, and printers.



**PRESIDENCY
UNIVERSITY**

Private University Estd. in Karnataka State by Act No. 41 of 2013

