Operating System
KCS – 401

An Overview

Dr. Pankaj Kumar
Associate Professor – CSE
SRMGPC Lucknow

Outline of the Lecture

About the Course

Program Outcome

Objective of the Course

Scope of the Course

Course Outcome

Academic Formalities

Background Knowledge

References

Acknowledgement

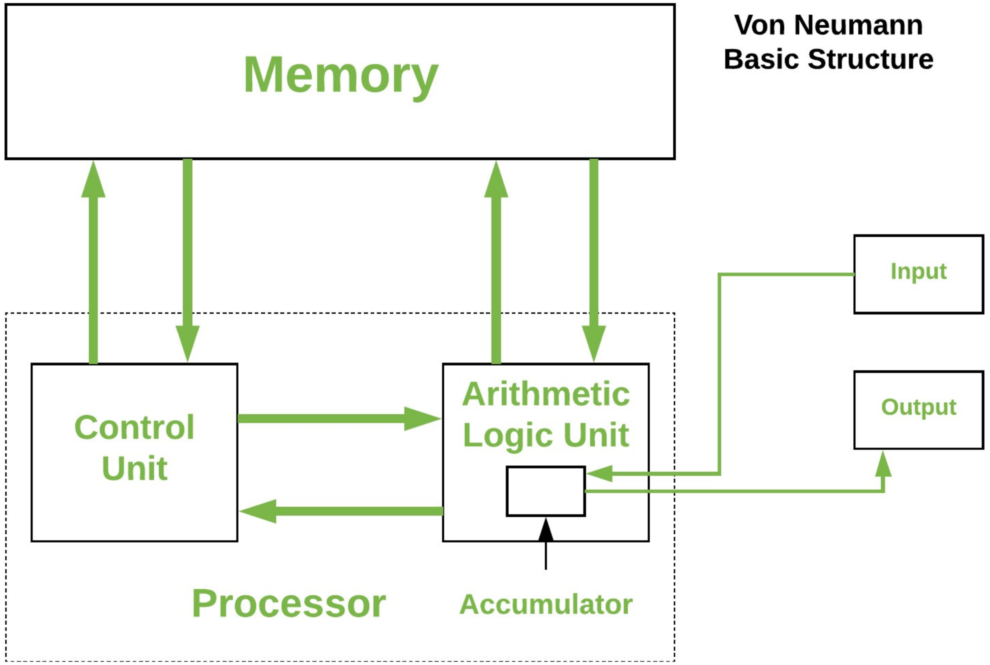
Program Outcome

1. Engineering knowledge
2. Problem analysis
3. Design/development of solutions
4. Conduct investigations of complex problems
5. Modern tool usage
6. The engineer and society
7. Environment and sustainability
8. Ethics
9. Individual and team work
10. Communication
11. Project management and finance
12. Life-long learning

About the Course

What is Computer :

A **computer** is an electronic device that manipulates information, or data. It has the ability to store, retrieve, and process data.



About the Course

What is Operating System :

The Operating System acts as a platform of information exchange between your computer's hardware and the applications running on it. Most people are familiar with the Windows Operating System family (like Windows 10, XP, or Vista) or Apple's suite of Operating Systems (like Catalina, Mojave, or Sierra),

This course tells the basic operating system abstractions, mechanisms, and their implementations.

Course has **Five** Unit.

1. **Introduction**
2. **Concurrent Processes**
3. **CPU Scheduling**
4. **Memory Management**
5. **I/O Management and Disk Scheduling**

Objective of the Course

1. To understand the services provided by and the design of an operating system.

2. To understand the structure and organization of the file system.

3. To understand what a process is and how processes are synchronized and scheduled.

4. To understand different approaches to memory management.

Scope of the Course

1. Students should be able to use system calls for managing processes, memory and the file system.
2. Students should understand the data structures and algorithms used to implement an OS.

Course Outcome

CO 1 K1, K2

Understand the structure and functions of OS

CO 2 K2, K6

Learn about Processes, Threads and Scheduling algorithms..

CO 3 K2

Understand the principles of concurrency and Deadlocks

CO 4 K2

Learn various memory management scheme

CO 5 K2, K4

Study I/O management and File systems.

Academic Formalities

- There are **five tutorials**. Each tutorial will be solved by you. Each tutorial consists of 50 marks. All will be submitted on google class room.
- Every week there will be a **surprise quiz.**
- There are **three quiz** having 30 marks.
- At the end of semester a **pre end semester** exam will be conducted.

Background Knowledge

- Strong programming skills
- Elementary data structures and algorithms
- Computer architecture
- Introduction to concurrency

References

TEXT BOOKS:

Ref. [ID]	Authors	Book Title	Publisher/Press	Edition &Year of Publication
[T1]	Silberschatz, Galvin and Gagne	Operating System Concept	Wiley	6th Ed.,2002
[T2]	D M Dhamdhere	Operating System: a concept-based approach	TMH	2nd Ed., 2006
[T3]	William Stalling	Operating System: Internal and Design Principles,	PEARSON	6TH, 2009
[T4]	A.S. Tanenbaum	Modern Operating System	PHI	2nd, 2002

REFERENCE BOOKS:

Ref. [ID]	Authors	Book Title	Publisher/Press	Edition &Year of Publication
[R1]	Sibsankar Halder and Alex A Aravind	Operating System	PEARSON	2nd , 2009

ONLINE/DIGITALREFERENCES:

Ref. [ID]	Source Name	Source Hyperlink
[D1]	Lecture Notes on Operating Systems	https://www.cse.iitb.ac.in/~mythili/os/

Acknowledgement

- Contents and examples will be taken from standard text books on the subject.
- For clearing the concepts Contents and examples will be taken from the other material on internet.

Outline of the Lecture

What is an Operating System

Computer-System Architecture

What Operating Systems Do

Operating System Definition

Computer-System Organization

What is an OS

A program that acts as an intermediary between a user of a computer and the computer hardware

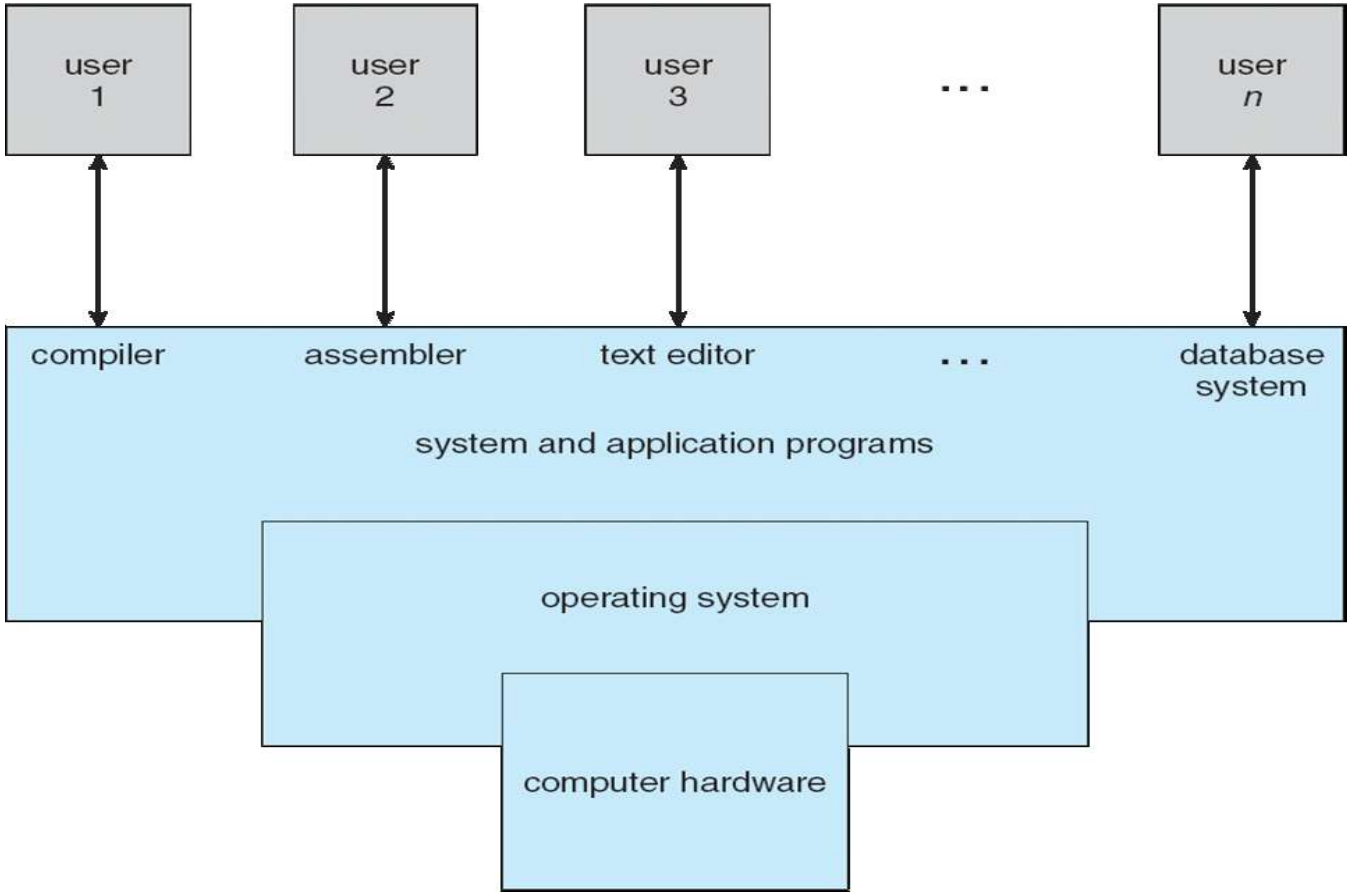
Operating system goals:

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

Computer-System Architecture

- 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

Computer-System Architecture



Four Component of Computer System

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 dedicate 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

Operating System Definition

- OS is a resource allocator
    - Manages all resources
    - Decides between conflicting requests for efficient and fair resource use

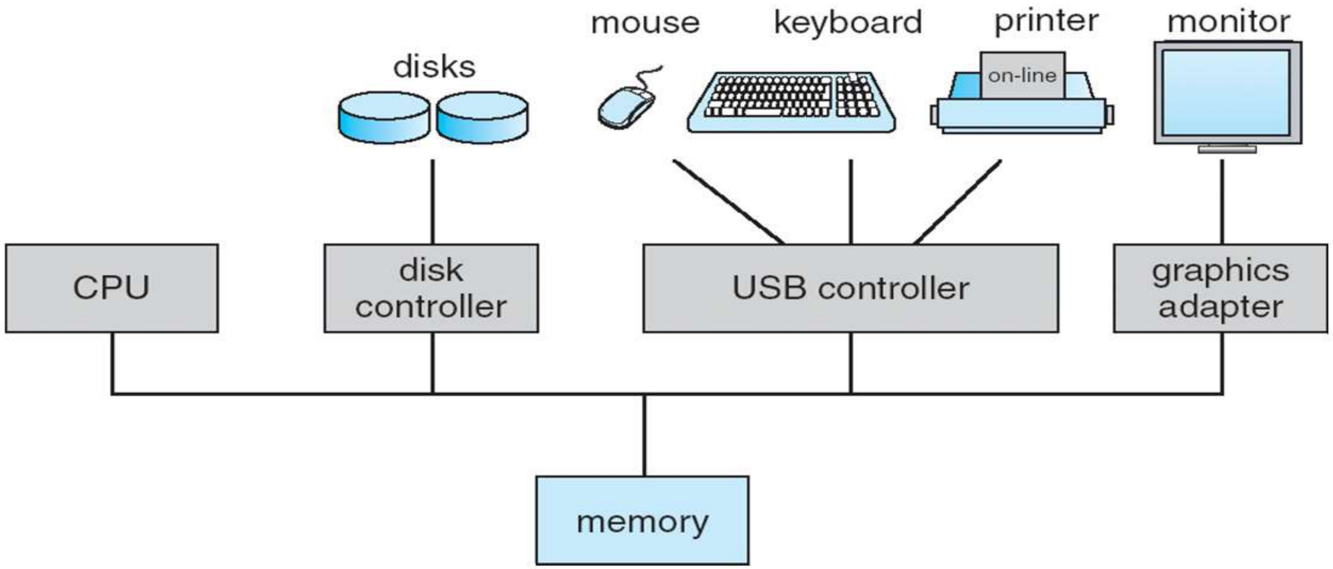
- OS is a control program
    - Controls execution of programs to prevent errors and improper use of the computer

Operating System Definition

- 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.

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



Outline of the Lecture

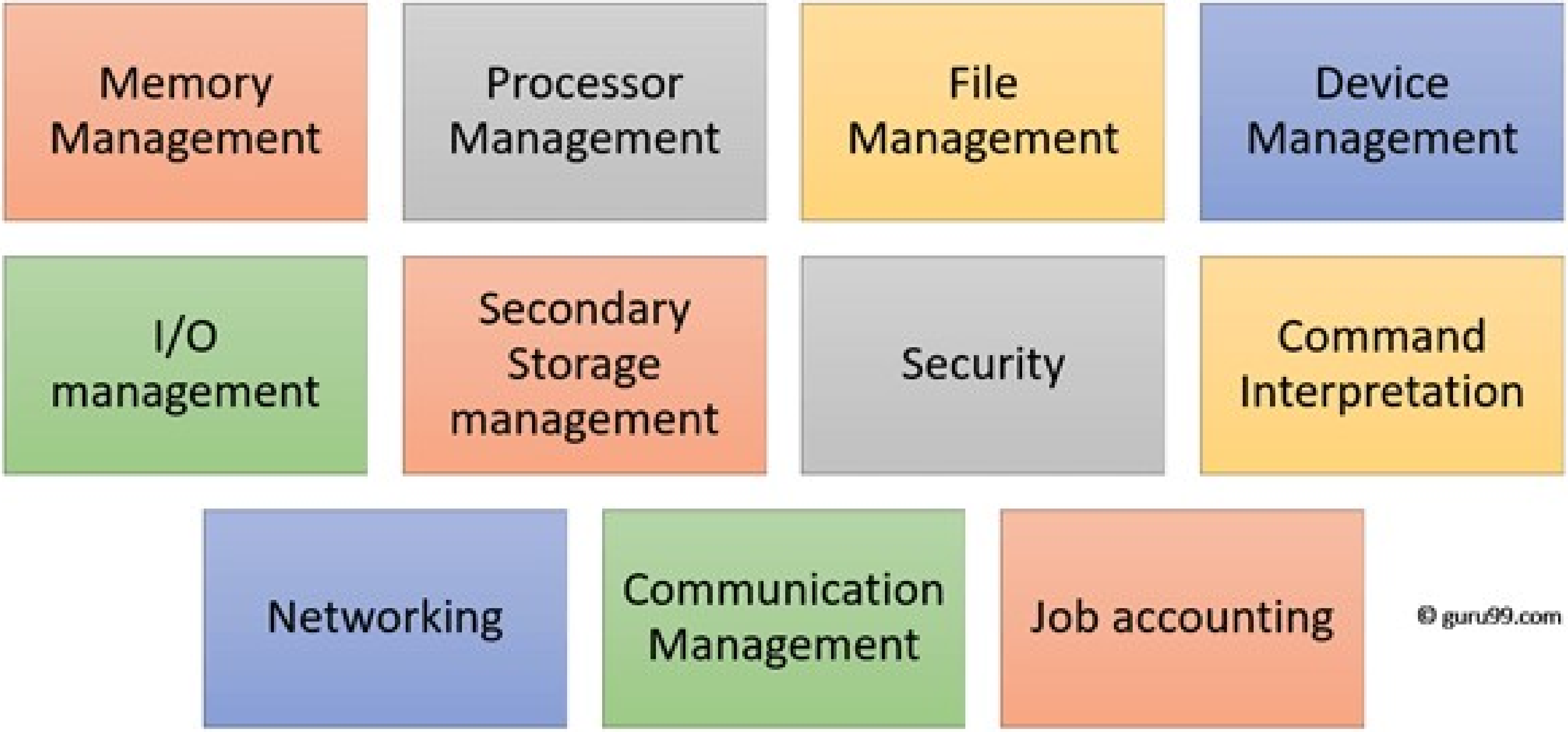
Function of Operating System

Feature of Operating System

Advantage of OS

Disadvantage of OS

Function of OS



Memory Management

Processor Management

File Management

Device Management

I/O management

Secondary Storage management

Security

Command Interpretation

Networking

Communication Management

Job accounting

© guru99.com

Function of OS

- **Process management**:- Process management helps OS to create and delete processes. It also provides mechanisms for synchronization and communication among processes.

- **Memory management:-** Memory management module performs the task of allocation and de-allocation of memory space to programs in need of this resources.

- **File management**:- It manages all the file-related activities such as organization storage, retrieval, naming, sharing, and protection of files.

Function of OS

- **Device Management**: Device management keeps tracks of all devices. This module also responsible for this task is known as the I/O controller. It also performs the task of allocation and de-allocation of the devices.

- **I/O System Management:** One of the main objects of any OS is to hide the peculiarities of that hardware devices from the user.

- **Secondary-Storage Management**: Systems have several levels of storage which includes primary storage, secondary storage, and cache storage. Instructions and data must be stored in primary storage or cache so that a running program can reference it.

Function of OS

- **Security**:- Security module protects the data and information of a computer system against malware threat and authorized access.

- **Command interpretation**: This module is interpreting commands given by the and acting system resources to process that commands.

- **Networking:** A distributed system is a group of processors which do not share memory, hardware devices, or a clock. The processors communicate with one another through the network.

- **Job accounting:** Keeping track of time & resource used by various job and users.

- **Communication management**: Coordination and assignment of compilers, interpreters, and another software resource of the various users of the computer systems.

Feature of OS

- Protected and supervisor mode
- Allows disk access and file systems Device drivers Networking Security
- Program Execution
- Memory management Virtual Memory Multitasking
- Handling I/O operations
- Manipulation of the file system
- Error Detection and handling
- Resource allocation
- Information and Resource Protection

Advantage of OS

- Allows you to hide details of hardware by creating an abstraction
- Easy to use with a GUI
- Offers an environment in which a user may execute programs/applications
- The operating system must make sure that the computer system convenient to use
- Operating System acts as an intermediary among applications and the hardware components
- It provides the computer system resources with easy to use format
- Acts as an intermediator between all hardware's and software's of the system

Disadvantage of OS

- If any issue occurs in OS, you may lose all the contents which have been stored in your system
- Operating system's software is quite expensive for small size organization which adds burden on them. Example Windows
- It is never entirely secure as a threat can occur at any time

Outline of the Lecture

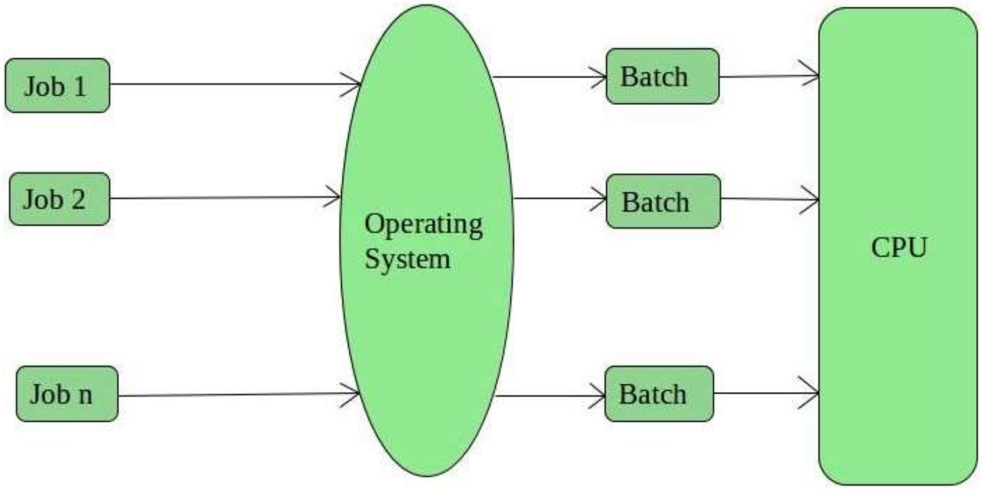
Classification of OS

- Batch
- Interactive
- Time Sharing
- Distributed
- Real Time system
- Embedded
- Network
- Mobile

Classification of OS

Batch Operating System

This type of operating system does not interact with the computer directly. The similar jobs are grouped together into batches with the help of some operator and these batches are executed one by one. It is the responsibility of the operator to sort jobs with similar needs.



Examples of Batch based Operating System:

Payroll System,

Bank Statements, etc.

Classification of OS

Batch Operating System

Advantages:

- It is very difficult to guess or know the time required for any job to complete. 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:

- 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

Classification of OS

Interactive Operating System

In Interactive Operating System there is a direct interaction between the user and the computer. Mostly all the Personal Computers are Interactive Operating System.

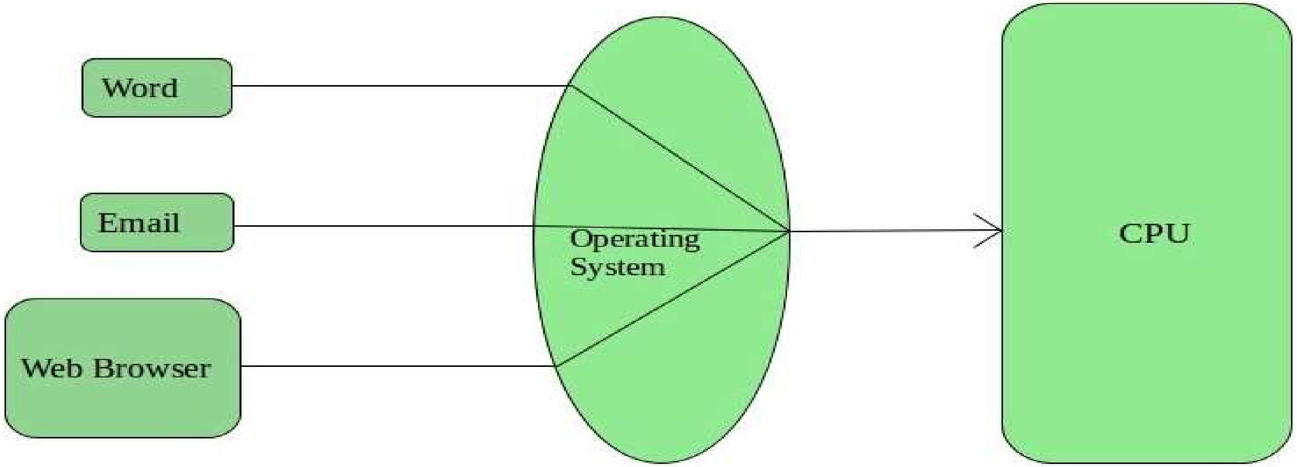
In this kind of Operating system, the user input some command in the system and the system works according to it.

Classification of OS

Time Shared Operating System

Each task is given some time to execute so that all the tasks work smoothly. Each user gets the time of CPU as they use a single system. These systems are also known as Multitasking 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.



Classification of OS

Time Shared Operating System

Advantages:

- Since equal time quantum is given to each process, so each process gets equal opportunity to execute.
- The CPU will be busy in most of the cases and this is good to have case. (CPU idle time can be reduced)
- Fewer chances of duplication of software

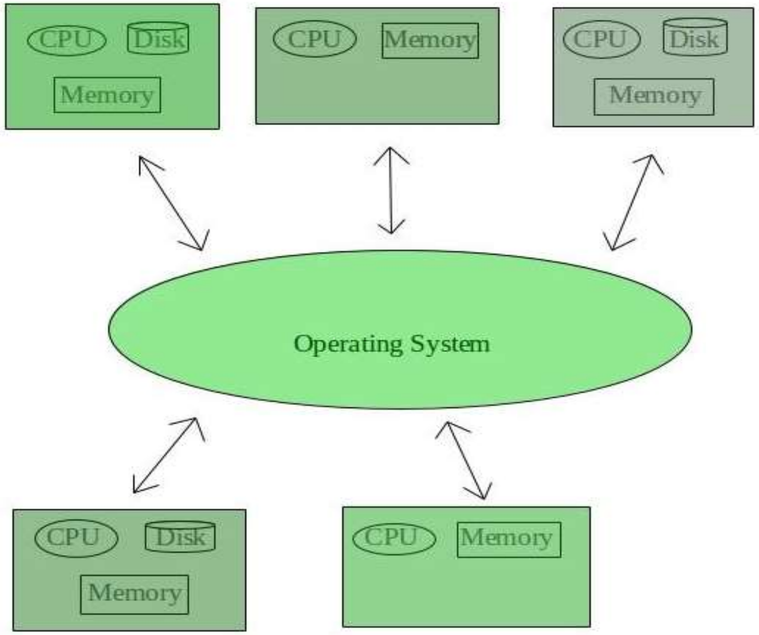
Disadvantages:

- Process having higher priority will not get the chance to be executed first because the equal opportunity is given to each process.
- Reliability problem
- One must have to take care of the security and integrity of user programs and data
- Data communication problem

Classification of OS

Distributed Operating System

In a Distributed Operating System, we have various systems 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. Here, each system can perform its task individually. The best part about these Distributed Operating System is remote access i.e. one user can access the data of the other system and can work accordingly. So, remote access is possible in these distributed Operating Systems. Example:- LOCUS, etc



Classification of OS

Distributed Operating System

Advantages:

- Since the systems are connected with each other so, the failure of one system can't stop the execution of processes because other systems can do the execution.
- Resources are shared between each other.
- The load on the host computer gets distributed and this, in turn, increases the efficiency.
- Since resources are being shared, computation is highly fast and durable

Disadvantages:

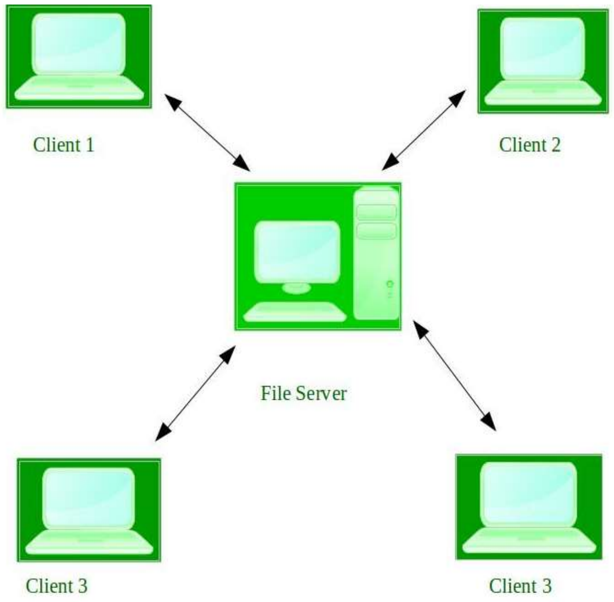
- Since the data is shared among all the computers, so to make the data secure and accessible to few computers, you need to put some extra efforts.
- If there is a problem in the communication network then the whole communication will be broken.

Classification of OS

Network Operating System

The primary purpose of the network operating system is to allow shared file and printer access among multiple computers in a network, typically a local area network (LAN), a private network or to other networks. A Network Operating System runs on a server and provides the server the capability to manage data, users, groups, security, applications, and other networking functions.

To function network operating system requires an existing operating system.



Example : Microsoft Windows Server 2003, Microsoft Windows Server 2008, UNIX, Linux, Mac OS X, and Novell NetWare

Classification of OS

Network Operating System

Advantage:

- Centralized servers are very highly stabled.
- Server managed the security concerns.
- Remote access to servers is possible from different locations and types of systems.
- It can manage multiple requests (inputs) simultaneously & also provides security in a multi-user environment.

Disadvantage:

- High cost of buying and running a server.
- Dependency on a central location for most operations.
- Regular maintenance and updates are required.

Classification of OS

Difference between Network and Distributed Operating System

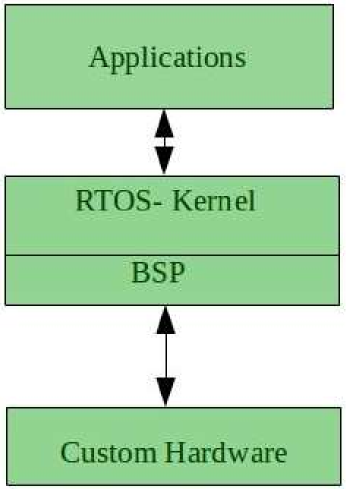
Main objective is to provide the local services to remote client.	Main objective is to manage the hardware resources.
In Network Operating System, Communication takes place on the basis of files.	In Distributed Operating System, Communication takes place on the basis of messages and shared memory.
Network Operating System is more scalable than Distributed Operating System.	Distributed Operating System is less scalable than Network Operating System.
In Network Operating System, fault tolerance is less.	While in Distributed Operating System, fault tolerance is high.
Rate of autonomy in Network Operating System is high.	While The rate of autonomy in Distributed Operating System is less.
Ease of implementation in Network Operating System is also high.	While in Distributed Operating System Ease of implementation is less.
In Network Operating System, All nodes can have different operating system.	While in Distributed Operating System, All nodes have same operating system.

Classification of OS

Real Time Operating System

A real-time operating system is a type of operating system that processes data within a very short amount of time. It means that a real-time operating system processes the data immediately without any delay and then produces the output. If any input is their it responds very quickly to it. The time taken by the system to process and respond to inputs is termed as response time.

The time taken by the system to respond to an input and display of required updated information is termed as the **response time**.



Example : Scientific experiments, medical imaging systems, industrial control systems, weapon systems, robots, air traffic control systems, etc.

Classification of OS

Real Time Operating System

Advantage:

- It is an error-free system.
- This system focuses on running applications. Memory allocation is best maintained in them.
- Maximum consumption of devises & system, execution of more outputs from all the resources.
- The time assigned for shifting tasks in these systems are very less.

Disadvantage:

- Very few tasks can run at the same time.
- System resources are more expensive.
- Complex algorithm & difficult for the designers to write on.

Classification of OS

Embedded Operating System

Embedded Systems are a specially designed computer system that essentially contains software and hardware for performing specific tasks. As the name suggests Embedded Operating System is an Embedded System's Operating System.

It has limited features. It is usually designed for some particular operations to control an electronic device.

Embedded Operating System is defined as a combination of hardware and software. Embedded operating systems runs on **embedded processors**.

Classification of OS

Embedded Operating System

Advantages:

1. Since it is dedicated to a particular job, so it is fast.
2. Low cost.
3. These consume less memory and other resources.

Disadvantages:

1. Only one job can be performed.
2. It is difficult to upgrade or is nearly scalable.

Classification of OS

Mobile Operating System

Mobile operating systems are those OS which is especially that are designed to power smartphones, tablets, and wearables devices.

Some most famous mobile operating systems are Android and iOS, but others include BlackBerry, Web, and watchOS.

Outline of the Lecture

Different Aspect of OS:

Multiprogramming System

Multitasking

Multiprocessing/Multiprocessor Systems

Multiuser Systems

Multithreaded Systems

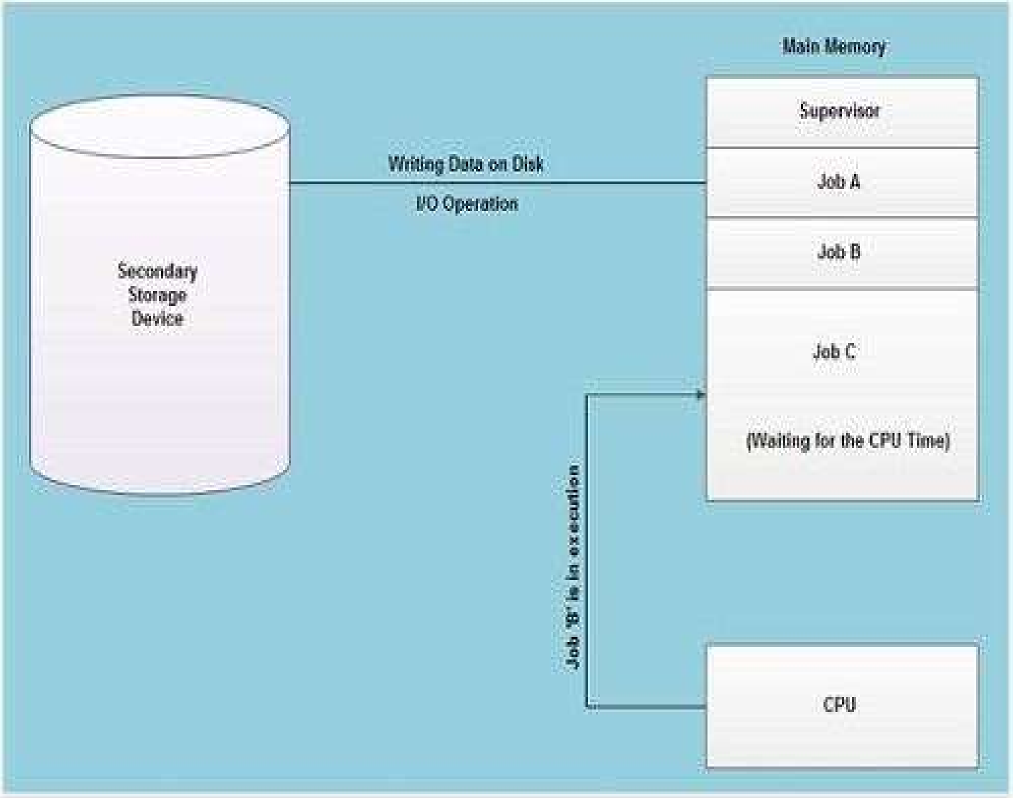
Aspect of OS

Multiprogramming

Single user cannot keep CPU and I/O devices busy at all times.

Sharing the processor, when two or more programs reside in memory at the same time, is referred as **multiprogramming**. Multiprogramming assumes a single shared processor. Multiprogramming increases CPU utilization by organizing jobs so that the CPU always has one to execute.

Multiprogramming operating systems monitor the state of all active programs and system resources using memory management programs to ensures that the CPU is never idle, unless there are no jobs to process.



Aspect of OS

Multiprogramming

Advantages

- High and efficient CPU utilization.
- User feels that many programs are allotted CPU almost simultaneously.

Disadvantages

- CPU scheduling is required.
- To accommodate many jobs in memory, memory management is required.

Aspect of OS

Multitasking : Timesharing

is logical extension of multiprogramming in which CPU switches jobs so frequently that users can interact with each job while it is running, creating **interactive** computing

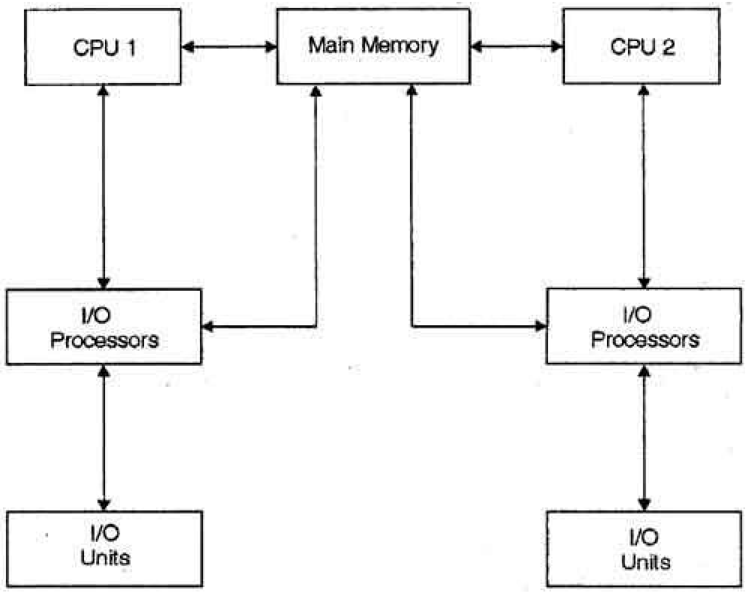
- **Response time** should be < 1 second
- Each user has at least one program executing in memory ⇨**process**
- If several jobs ready to run at the same time ⇨ **CPU scheduling**
- If processes don't fit in memory, **swapping** moves them in and out to run
- **Virtual memory** allows execution of processes not completely in memory

Aspect of OS

Multiprocessing/Multiprocessor

Multiprocessing is the use of two or more CPUs (processors) within a single Computer system. The term also refers to the ability of a system to support more than one processor within a single computer system.

Now since there are multiple processors available, multiple processes can be executed at a time. These multi processors share the computer bus, sometimes the clock, memory and peripheral devices also.



Aspect of OS

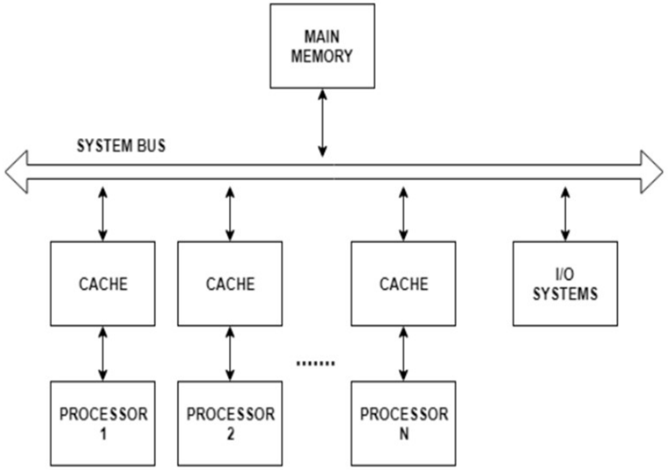
Multiprocessing/Multiprocessor

There are mainly two types of multiprocessor systems.

- Symmetric Multiprocessor System
- Asymmetric Multiprocessor System

Symmetric Multiprocessor System

In symmetric multiprocessing, multiple processors share a common memory and operating system. All of these processors work in tandem to execute processes. The operating system treats all the processors equally, and no processor is reserved for special purposes.

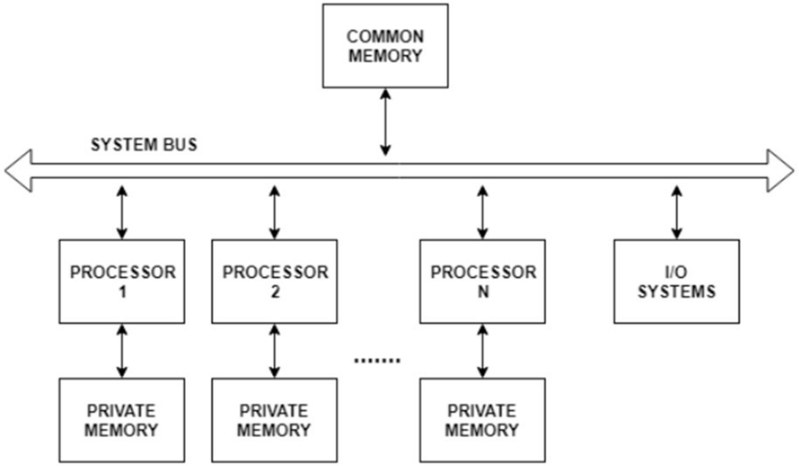


Aspect of OS

Multiprocessing/Multiprocessor

Asymmetric Multiprocessor System

Asymmetric multiprocessor systems are a part of multiprocessor systems along with symmetric multiprocessor systems. Multiprocessor systems have multiple processors working in parallel that share the computer clock, memory, bus, peripheral devices etc.



Aspect of OS

Multiprocessing/Multiprocessor

Asymmetric Multiprocessing	Symmetric Multiprocessing
In asymmetric multiprocessing, the processors are not treated equally.	In symmetric multiprocessing, all the processors are treated equally.
Tasks of the operating system are done by master processor.	Tasks of the operating system are done individual processor
No Communication between Processors as they are controlled by the master processor.	All processors communicate with another processor by a shared memory.
In asymmetric multiprocessing, process are master-slave.	In symmetric multiprocessing, the process is taken from the ready queue.
Asymmetric multiprocessing systems are cheaper.	Symmetric multiprocessing systems are costlier.
Asymmetric multiprocessing systems are easier to design	Symmetric multiprocessing systems are complex to design

Aspect of OS

Multiprocessing/Multiprocessor

Advantages

In a multiprocessor system, even if one processor fails, the system will not halt. This ability to continue working despite hardware failure is known as graceful degradation.

Execution of several tasks by different processors concurrently, increases the system's throughput without speeding up the execution of a single task.

Disadvantages

All the processors in the multiprocessor system share the memory. So a much larger pool of memory is required as compared to single processor systems.

Aspect of OS

Multiuser

Multi-user operating system is a computer operating system which allows multiple users to access the single system with one operating system on it. In the multi-user operating system, different users connected at different terminals and we can access, these users through the network.

USER 1
USER 2
USER 3
USER 4
USER 5

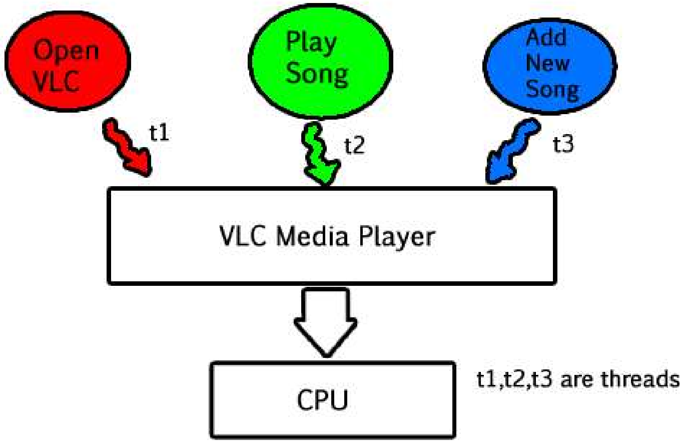
A multi-user operating system (OS) is one that can be used by more than one person at a time while running on a single machine. Different users access the machine running the OS through networked terminals. The OS can handle requests from users by taking turns among connected users.

Aspect of OS

Multithreading

A thread is a basic unit of CPU utilization. Multi threading is an execution model that allows a single process to have multiple code segments (i.e., threads) running concurrently within the “context” of that process.

Multi threading is the ability of a process to manage its use by more than one user at a time and to manage multiple requests by the same user without having to have multiple copies of the program.



Aspect of OS

Multithreading

Advantages

- Threads minimize the context switching time.
- Use of threads provides concurrency within a process.
- Efficient communication.
- It is more economical to create and context switch threads.
- Threads allow utilization of multiprocessor architectures to a greater scale and efficiency.

Outline of the Lecture

OS Structure

Simple Structure

Monolithic Structure

Layered Structure

Micro Kernal

Modules

Operating System Structure

Since the operating system is such a complex structure, it should be created with utmost care so it can be used and modified easily. An easy way to do this is to create the operating system in parts. Each of these parts should be well defined with clear inputs, outputs and functions.

Operating system can be implemented with the help of various structures. The structure of the OS depends mainly on how the various common components of the operating system are interconnected and melded into the kernel. Depending on this we have following structures of the operating system:

Simple Structure

Monolithic Strcuture

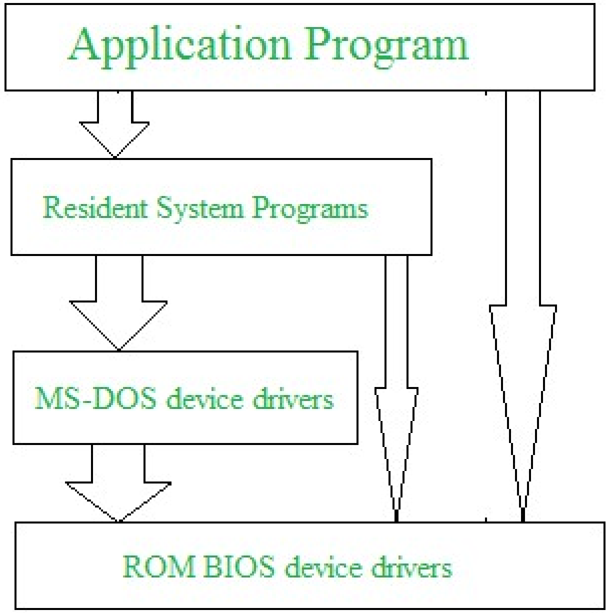
Layered Structure

Micro Kernal

Modules

Simple Structure

Such operating systems do not have well defined structure and are small, simple and limited systems. The interfaces and levels of functionality are not well separated. MS-DOS is an example of such operating system. It was written to provide the most functionality in the least space, so it was not divided into modules carefully..

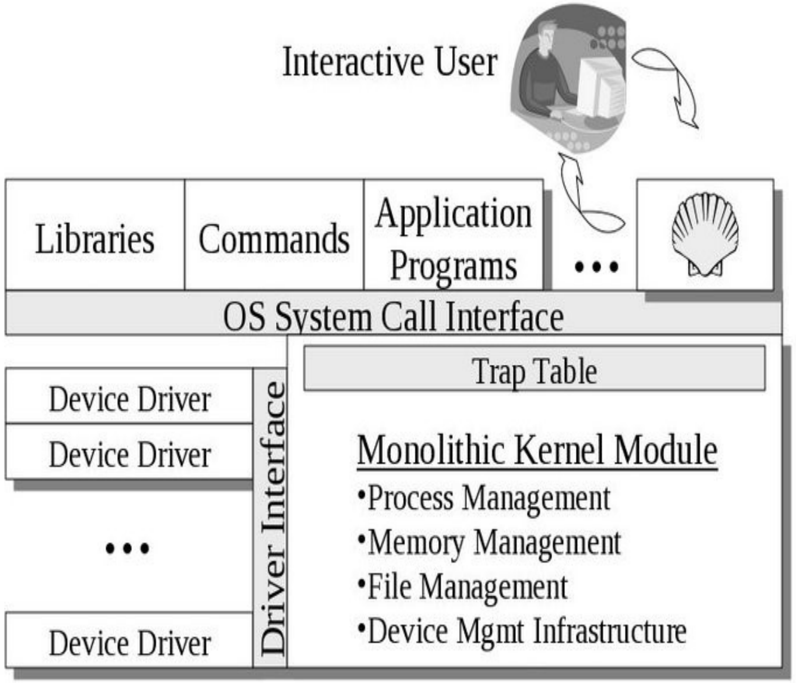


In this structure the interfaces and levels of functionality are not well separated. For instance application programs are able to access the basic I/O routine to write directly to the display and disk drives. Such freedom leaves MS-DOS vulnerable to errant (or malicious) programs, causing entire system crashes when user programs fail.

Monolithic Structure

It consists of two separable parts: the kernel and the system programs.

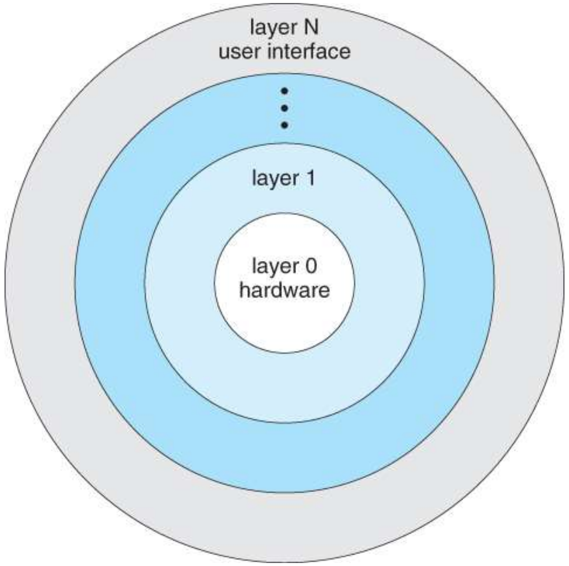
The kernel is further separated into a series of interfaces and device drivers, which have been added and expanded over the years as UNIX has evolved. Everything below the system call interface and above the physical hardware is the kernel. The kernel provides the file system, CPU scheduling, memory management, and other operating-system functions through system calls. Taken in sum, that is an enormous amount of functionality to be combined into one level. This monolithic structure was difficult to implement and maintain.



Layered Structure

With proper hardware support, operating systems can be broken into pieces that are smaller and more appropriate than those allowed by the original MS-DOS or UNIX systems. The operating system can then retain much greater control over the computer and over the applications that make use of that computer.

In this structure the OS is broken into number of layers (levels). The bottom layer (layer 0) is the hardware and the topmost layer (layer N) is the user interface. These layers are so designed that each layer uses the functions of the lower level layers only.



- This allows implementers to change the inner workings, and increases modularity.
- As long as the external interface of the routines don’t change, developers have more freedom to change the inner workings of the routines.
- With the layered approach, the bottom layer is the hardware, while the highest layer is the user interface.

Layered Structure

This approach simplifies debugging and system verification.

The first layer can be debugged without any concern for the rest of the system, because, by definition, it uses only the basic hardware (which is assumed correct) to implement its functions.

Once the first layer is debugged, its correct functioning can be assumed while the second layer is debugged, and so on.

If an error is found during the debugging of a particular layer, the error must be on that layer, because the layers below it are already debugged. Thus, the design and implementation of the system is simplified.

Each layer is implemented with only those operations provided by lower level layers.

A layer does not need to know how these operations are implemented; it needs to know only what these operations do. Hence, each layer hides the existence of certain data structures, operations, and hardware from higher-level layers.

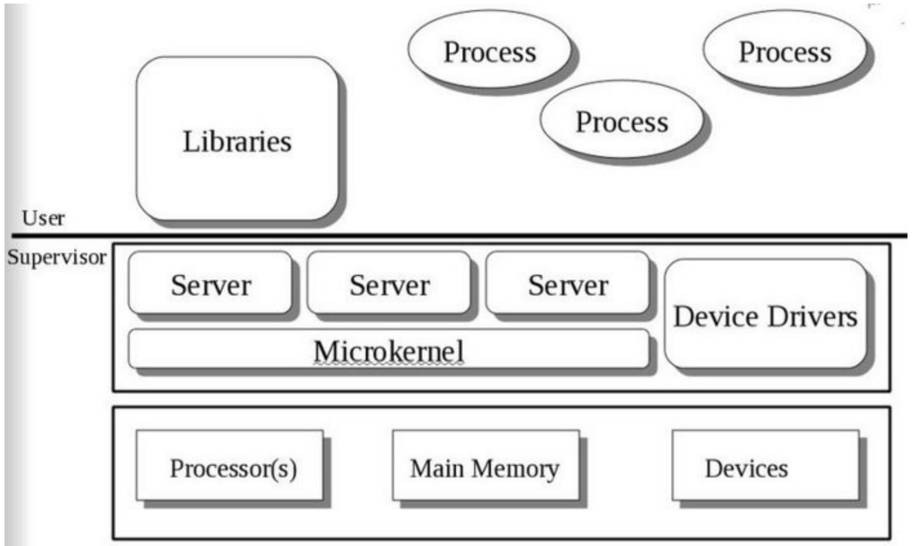
The major difficulty with the layered approach involves appropriately defining the various layers. Because a layer can use only lower-level layers, careful planning is necessary.

Micro Kernel Structure

This structure designs the operating system by removing all non-essential components from the kernel and implementing them as system and user programs. This result in a smaller kernel called the micro-kernel.

- Generally they provide minimal process and memory management, and a communications facility.
- Communication between components of the OS is provided by message passing.

Advantages of this structure are that all new services need to be added to user space and does not require the kernel to be modified. Thus it is more secure and reliable as if a service fails then rest of the operating system remains untouched. **Mac OS is an example of this type of OS.**

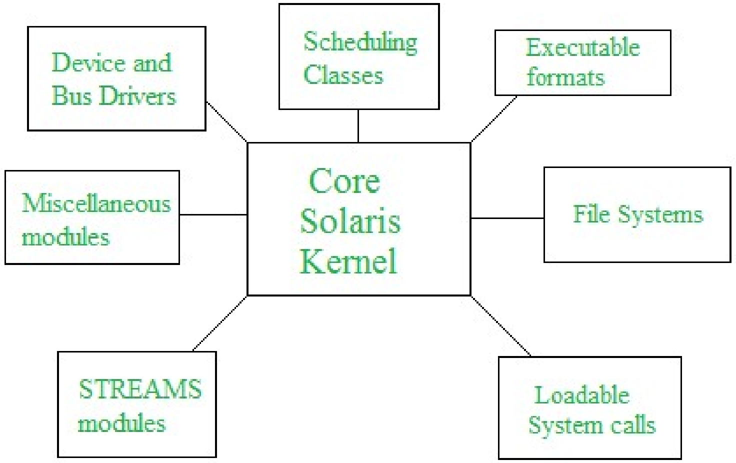


Main *disadvantage* is poor performance due to increased system overhead from message passing.

Module Structure

It is considered as the best approach for an OS. It involves designing of a modular kernel. The kernel has only set of core components and other services are added as dynamically loadable modules to the kernel either during run time or boot time. It resembles layered structure due to the fact that each kernel has defined and protected interfaces but it is more flexible than the layered structure as a module can call any other module. For example Solaris OS

Such a design allows the kernel to provide core services yet also allows certain features to be implemented dynamically. For example, device and bus drivers for specific hardware can be added to the kernel, and support for different file systems can be added as loadable modules.



Queries

???????