WEEK 1 : LECTURE 1 - INTRODUCTION

Bachelor's Degree in Information Technology
BITOS4111, OPERATING SYSTEMS
Trimester: 05
From **July 2024** to **October 2024**

OBJECTIVES/TOPICS

- Introduction to Operating Systems.
- Types of Operating Systems.
- Operating System Structures.
- Operating System Services.
- System Calls.
- Virtual Machines.
- Operating System Design and Implementation.

INTRODUCTION TO OPERATING SYSTEMS

- An operating system (OS) is software that manages a computer's hardware
  - It also provides a basis for application programs & acts as an intermediary between the computer user & the computer hardware.

- Operating systems are everywhere:
  - In cars
  - In "Internet of Things" devices aka home appliances
  - On Smart phones, personal & enterprise computers
  - On cloud computing environments.

INTRODUCTION TO OPERATING SYSTEMS

- In order to explore the role of an operating system in a modern computing environment, it is important first to understand the organization and architecture of computer hardware.
  - This includes the CPU, memory, and I/O devices, as well as storage

- A fundamental responsibility of an operating system is to allocate these resources to programs.

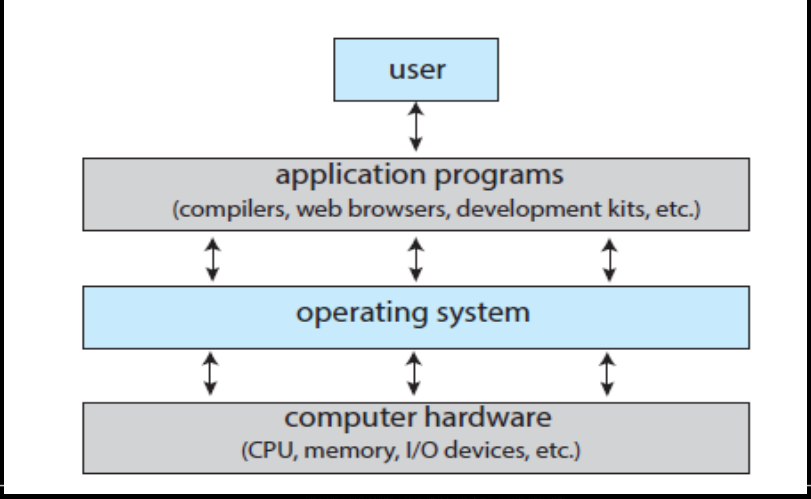
INTRODUCTION TO OPERATING SYSTEMS

- Because an operating system is large and complex, it must be created piece by piece.

- Each of these pieces should be a well-delineated portion of the system, with carefully defined inputs, outputs, and functions.
  - Making up the different functions provided by operating systems

INTRODUCTION TO OPERATING SYSTEMS

- A computer system can be divided roughly into four components (shown in the diagram below):
  - the hardware
  - the operating system
  - the application programs
  - the user



INTRODUCTION TO OPERATING SYSTEMS

- The hardware, which provides the basic computing resources for the system includes:
  - the central processing unit (CPU)
  - the memory
  - The input/output (I/O) devices

- The application programs define the ways in which these resources are used to solve users' computing problems.
  - word processors
  - spreadsheets
  - compilers, and web browsers

INTRODUCTION TO OPERATING SYSTEMS

- The operating system controls the hardware & coordinates its use among the various application programs for the various users.

- We can also view a computer system as consisting of hardware, software, and data.
  - The operating system provides the means for proper use of these resources in the operation of the computer system

INTRODUCTION TO OPERATING SYSTEMS

- A more common definition, and the one that we usually follow, is that the operating system is the one program running at all times on the computer—usually called the kernel.
- Along with the kernel, there are two other types of programs:
  - System programs, which are associated with the operating system but are not necessarily part of the kernel
  - application programs, which include all programs not associated with the operation of the system.

INTRODUCTION TO OPERATING SYSTEMS

- Today, however, if we look at operating systems for mobile devices, we see that once again the number of features constituting the operating system is increasing.
  - Mobile operating systems often include not only a core kernel but also middleware—a set of software frameworks that provide additional services to application developers.
  - E.g., each of the two most prominent mobile operating systems—Apple's iOS and Google's Android—features a core kernel along with middleware that supports databases, multimedia, and graphics

TYPES OF OPERATING SYSTEMS

- 1. Batch Operating Systems
  - The batch operating system does not have a direct link with the computer.
  - A different system divides and allocates similar tasks into batches for easy processing and faster response.
  - The batch operating system is appropriate for lengthy and time-consuming tasks.
  - These systems are used for tasks such as managing payroll systems, data entry and bank statements.

TYPES OF OPERATING SYSTEMS

- 1. Batch Operating Systems

Advantages	Disadvantages
Many users can share batch systems. There is little idle time for batch operating systems.	Batch operating systems are challenging to debug.
It becomes possible to manage large workloads.	Any failure of the system creates a backlog.
It's easy to estimate how long a task will take to be completed.	It may be costly to install and maintain good batch operating systems.

TYPES OF OPERATING SYSTEMS

- 2. Time-sharing or multitasking OS
  - Works by allocating time to a particular task and switching between tasks frequently.
  - Unlike the batch system, the time-sharing system allows users to complete their work in the system simultaneously.
  - Examples of time-sharing operating systems include Multics and Unix.

TYPES OF OPERATING SYSTEMS

- 2. Time-sharing or multitasking OS

Advantages	Disadvantages
There's a quick response during task performance.	The user's data security might be a problem.
It minimizes the idle time of the processor.	System failure can lead to widespread failures.
All tasks get an equal chance of being accomplished.	Problems in data communication may arise.
It reduces the chance of software duplication.	The integrity of user programs is not assured.

TYPES OF OPERATING SYSTEMS

- 3. Distributed OS
  - This system is based on autonomous but interconnected computers communicating with each other via communication lines or a shared network.
  - Each autonomous system has its own processor that may differ in size and function.
  - These operating systems are often used for tasks such as telecommunication networks, airline reservation controls and peer-to-peer networks.
  - A distributed operating system serves multiple applications and multiple users in real time

TYPES OF OPERATING SYSTEMS

- 3. Distributed OS

Advantages	Disadvantages
They allow remote working.	If the primary network fails, the entire system shuts down.
They allow a faster exchange of data among users.	They're expensive to install.
Failure in one site may not cause much disruption to the system.	They require a high level of expertise to maintain.
They reduce delays in data processing.	
They minimize the load on the host computer.	
They enhance scalability since more systems can be added to the network.	

TYPES OF OPERATING SYSTEMS

- 4. Network OS
  - Network operating systems are installed on a server providing users with the capability to manage data, user groups and applications.
  - This operating system enables users to access and share files and devices such as printers, security software and other applications, mostly in a local area network.
  - Examples of network operating systems include Microsoft Windows, Linux and macOS X

TYPES OF OPERATING SYSTEMS

- 4. Network OS

Advantages	Disadvantages
Centralized servers provide high stability.	They require regular updates and maintenance.
Security issues are easier to handle through the servers.	Servers are expensive to buy and maintain.
It's easy to upgrade and integrate new technologies.	Users' reliance on a central server might be detrimental to workflows.
Remote access to the servers is possible.	

TYPES OF OPERATING SYSTEMS

- 5. Real-time OS
  - Real-time operating systems provide support to real-time systems that require observance of strict time requirements.
  - They are typically used for tasks such as scientific experiments, medical imaging, robotics and air traffic control operations.
  - The response time between input, processing and response is tiny, which is beneficial for processes that are highly sensitive and need high precision.

TYPES OF OPERATING SYSTEMS

- 5. Real-time OS

Advantages	Disadvantages
They use device and systems maximally, hence more output.	They have a low capacity to run tasks simultaneously.
They allow fast shifting from one task to another.	They use heavy system resources.
The focus is on current tasks, and less focus is put on the queue.	They run on complex algorithms that are not easy to understand.
They can be used in embedded systems.	They're unsuitable for thread priority because of the system's inability to switch tasks.
Real-time systems are meticulously programmed, hence free of errors.	
They allow easy allocation of memory.	

TYPES OF OPERATING SYSTEMS

- 6. Mobile OS
  - Mobile operating systems such as Android OS, Apple and Windows mobile OS run exclusively on small devices such as smartphones, tablets and wearables.
  - The system combines the features of a personal computer with additional features useful for a handheld device.
  - Mobile operating systems start when a device is powered on to provide access to installed applications
  - Mobile operating systems also manage wireless network connectivity

TYPES OF OPERATING SYSTEMS

- 6. Mobile OS

Advantages	Disadvantages
Most systems are easy for users to learn and operate.	Some mobile OS put a heavy drain on a device's battery, requiring frequent recharging.
	Some systems are not user-friendly.

OPERATING SYSTEM STRUCTURES

- It is easier to create an operating system in pieces, much as we break down larger issues into smaller, more manageable sub-problems.

- Every segment is also a part of the operating system.

- Operating system structure can be thought of as the strategy for connecting and incorporating various operating system components within the kernel

OPERATING SYSTEM STRUCTURES

- 1. Simple Structure
- It is the simplest OS Structure & is not well defined;
- It can only be used for small and limited systems.
- In this structure, the interfaces and levels of functionality are well separated; hence programs can access I/O routines which can cause unauthorized access to I/O routines.
- This structure is implemented in MS-DOS operating system which is made up of various layers, each with its own set of functions
- The layers are: Application Program, System Program, MS-DOS device drivers and ROM BIOS device drivers

OPERATING SYSTEM STRUCTURES

- 1. Simple Structure

OPERATING SYSTEM STRUCTURES

- 1. Simple Structure

Advantages of Simple Structure:

- Because there are only a few interfaces and levels, it is simple to develop.
- Because there are fewer layers between the hardware and the applications, it offers superior performance.

Disadvantages of Simple Structure:

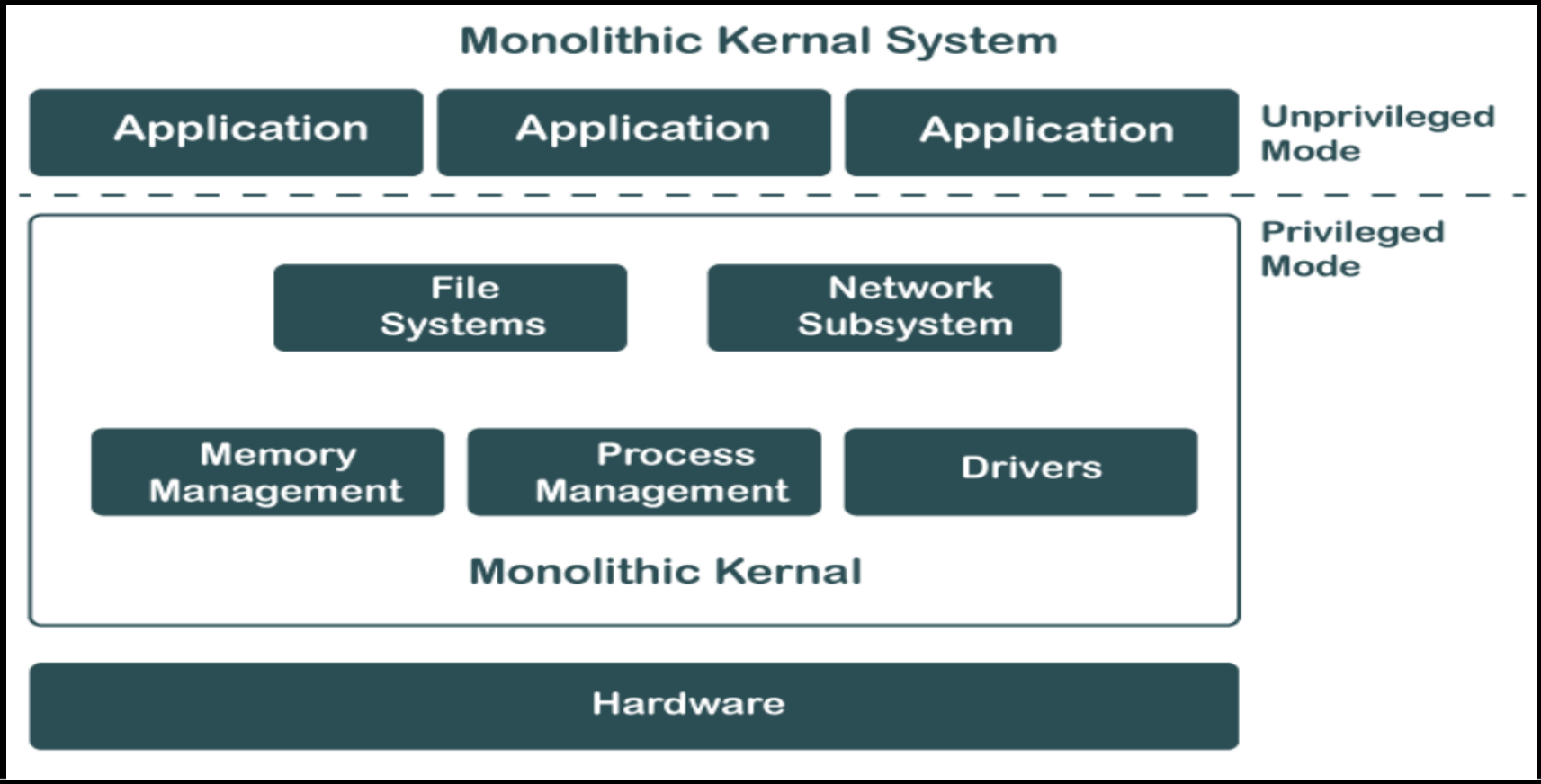
- The entire operating system breaks if just one user program malfunctions.
- Since the layers are interconnected, and in communication with one another, there is no abstraction or data hiding.
- The operating system's operations are accessible to layers, which can result in data tampering and system failure.

OPERATING SYSTEM STRUCTURES

- 2. Monolithic structure
  - The monolithic operating system controls all aspects of the operating system's operation, including file management, memory management, device management, and operational operations.
  - The monolithic operating system is often referred to as the monolithic kernel.
  - This is an old operating system that was used in banks to carry out simple tasks like batch processing and time-sharing, which allows numerous users at different terminals to access the Operating System.

OPERATING SYSTEM STRUCTURES

- 2. Monolithic structure



OPERATING SYSTEM STRUCTURES

- 2. Monolithic structure

Advantages of Monolithic Structure:

- Because layering is unnecessary and the kernel alone is responsible for managing all operations, it is easy to design and execute.
- Due to the fact that functions like memory management, file management, process scheduling, etc., are implemented in the same address area, the monolithic kernel runs rather quickly when compared to other systems. Utilizing the same address speeds up and reduces the time required for address allocation for new processes.

Disadvantages of Monolithic Structure:

- The monolithic kernel's services are interconnected in address space and have an impact on one another, so if any of them malfunctions, the entire system does as well.
- It is not adaptable. Therefore, launching a new service is difficult.

OPERATING SYSTEM STRUCTURES

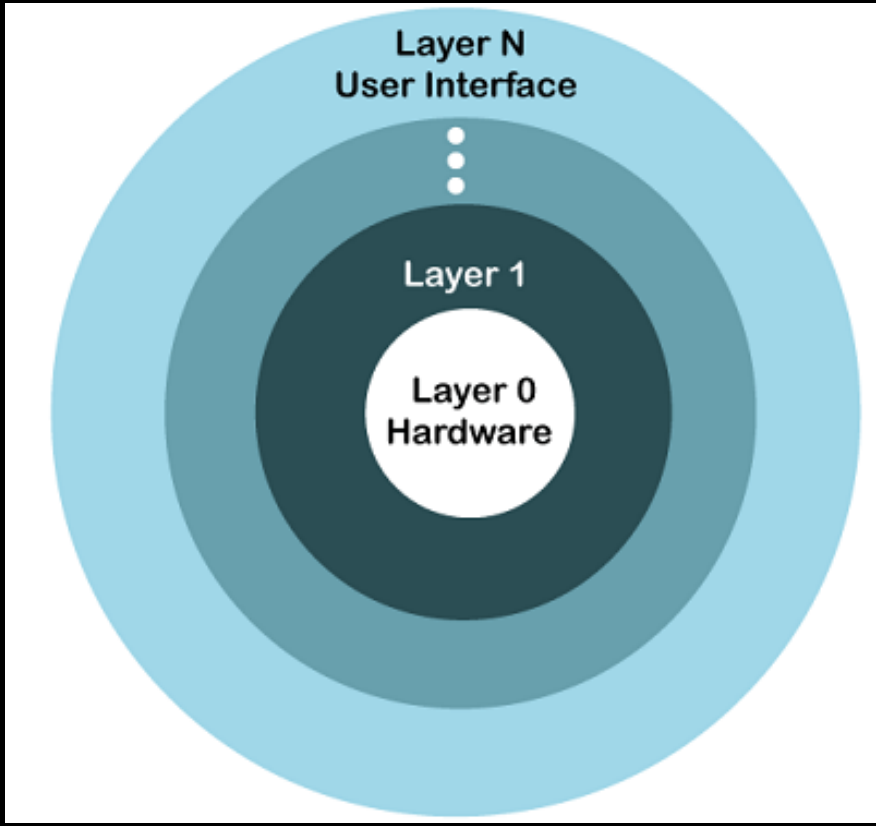
- 3. Layered structure
  - The OS is separated into layers or levels in this kind of arrangement. Layer 0 (the lowest layer) contains the hardware, and layer 1 (the highest layer) contains the user interface (layer N).
  - These layers are organized hierarchically, with the top-level layers making use of the capabilities of the lower-level ones.

WEEK 1 : LECTURE 1 - INTRODUCTION

Bachelor's Degree in Information Technology
BITOS4111, OPERATING SYSTEMS
Trimester: 05
From **July 2024** to **October 2024**

OPERATING SYSTEM STRUCTURES

- 3. Layered structure



OPERATING SYSTEM STRUCTURES

- 3. Layered structure

Advantages of Layered Structure:

- Work duties are separated since each layer has its own functionality, and there is some amount of abstraction.
- Debugging is simpler because the lower layers are examined first, followed by the top layers.

Disadvantages of Layered Structure:

- Performance is compromised in layered structures due to layering.
- Construction of the layers requires careful design because upper layers only make use of lower layers' capabilities.

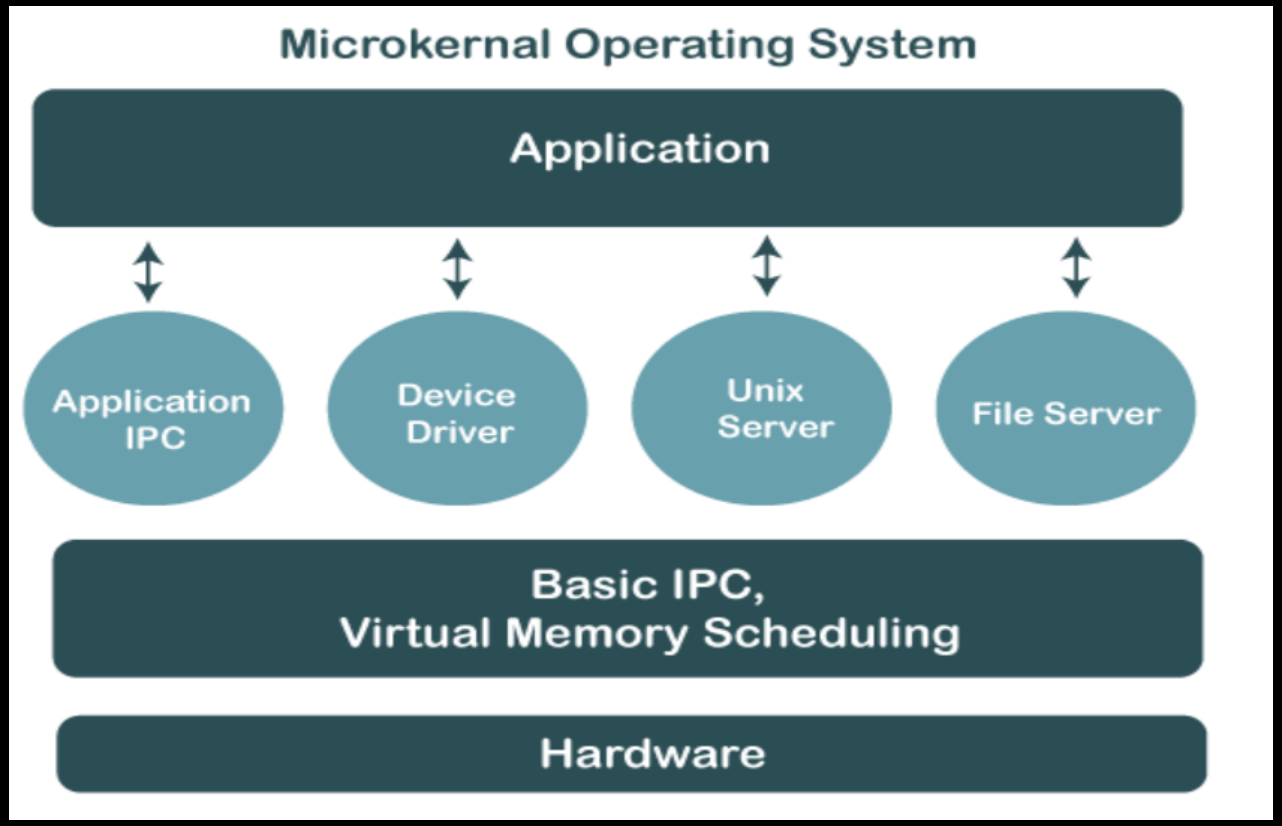
OPERATING SYSTEM STRUCTURES

- 4. Micro-kernel structure
  - The operating system is created using a micro-kernel framework that strips the kernel of any unnecessary parts.
  - Systems and user applications are used to implement these optional kernel components.
  - So, Micro-Kernels is the name given to these systems that have been developed.

WEEK 1 : LECTURE 1 - INTRODUCTION

OPERATING SYSTEM STRUCTURES

- 4. Micro-kernel structure



OPERATING SYSTEM STRUCTURES

- 4. Micro-kernel structure

Advantages of Micro-Kernel Structure:

- It enables portability of the operating system across platforms.
- Due to the isolation of each Micro-Kernel, it is reliable and secure.
- The reduced size of Micro-Kernels allows for successful testing.
- The remaining operating system remains unaffected and keeps running properly even if a component or Micro-Kernel fails.

Disadvantages of Micro-Kernel Structure:

- The performance of the system is decreased by increased inter-module communication.
- The construction of a system is complicated.

OPERATING SYSTEM STRUCTURES

- 5. Exo-Kernel Structure
  - An exokernel is a type of operating system kernel that provides only the most basic services, such as memory management and communication.
  - All other services are provided by user-level processes.
  - This makes exokernels highly modular and flexible, but also very complex to implement.

  - In an exokernel, the kernel provides a minimal interface to applications and exposes hardware resources directly, allowing applications to manage these resources.

OPERATING SYSTEM STRUCTURES

- 5. ExoKernel Structure

Advantages of Exokernel Structure:

- Application performance is enhanced by it.
- Accurate resource allocation and revocation enable more effective utilisation of hardware resources.
- New operating systems can be tested and developed more easily.
- Every user-space program is permitted to utilise its own customised memory management.

Disadvantages of Exokernel Structure:

- A decline in consistency
- Exokernel interfaces have a complex architecture.

OPERATING SYSTEM STRUCTURES

- 6. Modular Structure
  - In a modular operating system structure, the operating system is divided into a set of independent modules.
  - Each module is responsible for a specific task, such as memory management, process scheduling, or device drivers.
  - Modules can be loaded and unloaded dynamically, as needed.

OPERATING SYSTEM STRUCTURES

- 6. Modular Structure

Advantages of Modular Structure

- A modular structure is highly modular, meaning that each module is independent of the others. This makes it easier to understand, develop, and maintain the operating system.
- A modular structure is very flexible. New modules can be added easily, and existing modules can be modified or removed without affecting the rest of the operating system.

Disadvantages of Modular Structure

- There can be some performance overhead associated with the communication between modules. This is because modules must communicate with each other through well-defined interfaces.
- A modular structure can be more complex than other types of operating system structures. This is because the modules must be carefully designed to ensure that they interact correctly.

OPERATING SYSTEM STRUCTURES

- 7. Virtual Machines
  - Virtual Machines (VMs) are a form of virtualization technology that allows multiple operating systems to run on a single physical machine simultaneously.
  - Each virtual machine acts as an independent, isolated system with its own OS and applications.

OPERATING SYSTEM STRUCTURES

- 7. Virtual Machines (VM)

Advantages of Virtual Machines

- VMs provide a high degree of isolation between guest operating systems. This makes it difficult for malware or other problems in one guest to affect other guests or the host system.
- VMs can be used to create secure environments for running untrusted code. For example, a VM can be used to run a web browser without risking the entire system to malware infection.
- VMs can be easily moved from one physical machine to another. This makes it easy to deploy and manage applications across multiple servers.

Disadvantages of Virtual Machines

- VMs typically have some performance overhead compared to running software directly on the hardware. This is because the VM must emulate the hardware for the guest operating system.
- VMs can be complex to manage. This is because they require additional software to be installed and configured.
- VMs can consume a significant amount of system resources. This is because they must run a guest operating system in addition to the host operating system.

OPERATING SYSTEM SERVICES

- An operating system provides the environment within which programs are executed.

- Internally, operating systems vary greatly, since they are organized along many different lines.
  - However, there are many commonalities.

OPERATING SYSTEM SERVICES

- For a computer to start running—for instance, when it is powered up or rebooted—it needs to have an initial program to run.
- This initial program, or bootstrap program, tends to be simple.
  - Typically, it is stored within the computer hardware in firmware
- It initializes all aspects of the system, from CPU registers to device controllers to memory contents.
- The bootstrap program must know how to load the operating system and how to start executing that system.

OPERATING SYSTEM SERVICES

- If there are no processes to execute, no I/O devices to service, and no users to whom to respond, an operating system will sit quietly, waiting for something to happen.
- Events are almost always signaled by the occurrence of an interrupt

- Another form of interrupt is a trap (or an exception), a software-generated interrupt caused either by an error (e.g., division by 0) or by a specific request from a user program that an OS service be performed by executing a special operation called a system call.

OPERATING SYSTEM SERVICES

Multiprogramming and Multitasking

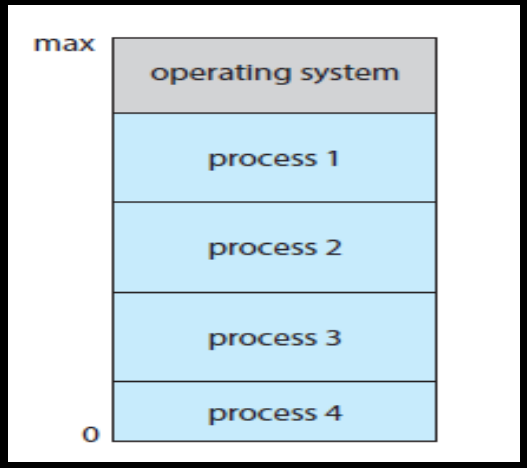
- One of the most important aspects of operating systems is the ability to run multiple programs, as a single program cannot keep either the CPU or the I/O devices busy at all times.

- Also, users typically want to run more than one program at a time as well.

OPERATING SYSTEM SERVICES

Multiprogramming and Multitasking

- Multiprogramming increases CPU utilization, as well as keeping users satisfied, by organizing programs so that the CPU always has one to execute.
- In multi-programmed system, a program in execution is termed a process.



OPERATING SYSTEM SERVICES

Multiprogramming and Multitasking

- The idea is as follows:
  - The operating system keeps several processes in memory simultaneously.
  - The operating system picks and begins to execute one of these processes.
  - Eventually, the process may have to wait for some task, such as an I/O operation, to complete.
    - In a non-multiprogrammed system, the CPU would sit idle.

OPERATING SYSTEM SERVICES

Multiprogramming and Multitasking

- Multitasking is a logical extension of multiprogramming.

- In multitasking systems, the CPU executes multiple processes by switching among them, but the switches occur frequently, providing the user with a fast response time.

OPERATING SYSTEM SERVICES

Multiprogramming and Multitasking

- Having several processes in memory at the same time requires some form of memory management.

- In addition, if several processes are ready to run at the same time, the system must choose which process will run next.
  - Making this decision is CPU scheduling

- In a multitasking system, the operating system must ensure reasonable response time.

OPERATING SYSTEM SERVICES

Multiprogramming and Multitasking

- A common method for doing so is virtual memory, a technique that allows the execution of a process that is not completely in memory

- The main advantage of this scheme is that it enables users to run programs that are larger than actual physical memory.
- Further, it abstracts main memory into a large, uniform array of storage, separating logical memory as viewed by the user from physical memory

OPERATING SYSTEM SERVICES

Dual-Mode and Multimode Operation

- Since the operating system and its users share the hardware and software resources of the computer system, a properly designed operating system must ensure that an incorrect (or malicious) program cannot cause other programs—or the operating system itself—to execute incorrectly.

- In order to ensure the proper execution of the system, we must be able to distinguish between the execution of operating-system code and user-defined code.

OPERATING SYSTEM SERVICES

Dual-Mode and Multimode Operation

- At the very least, we need two separate modes of operation:
  - user mode
  - kernel mode (also called supervisor mode, system mode, or privileged mode).

- A bit, called the mode bit, is added to the hardware of the computer to indicate the current mode: kernel (0) or user (1).

OPERATING SYSTEM SERVICES

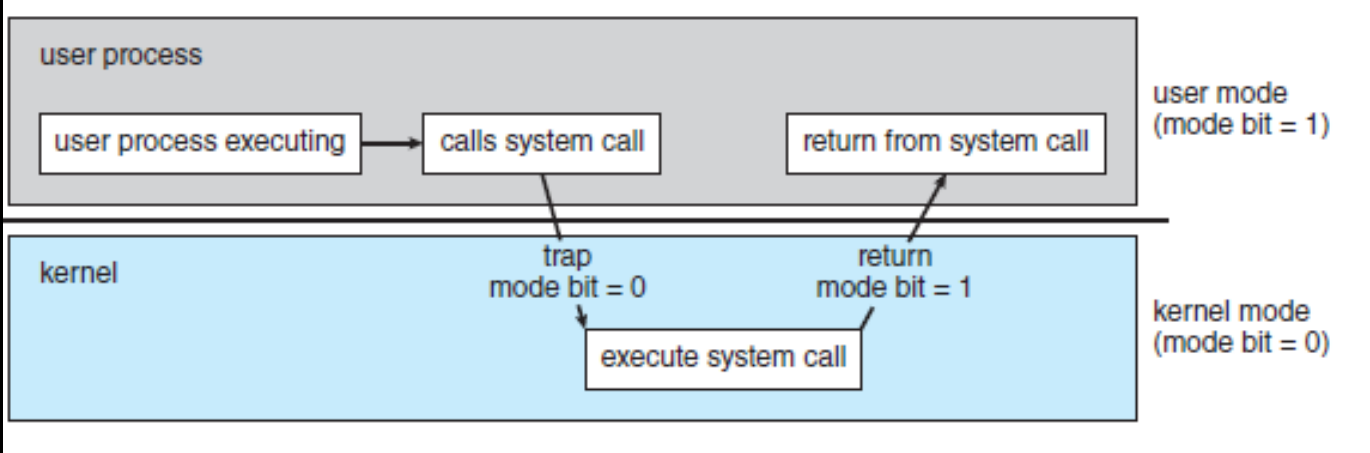
Dual-Mode and Multimode Operation

- With the mode bit, we can distinguish between a task that is executed on behalf of the operating system and one that is executed on behalf of the user.

- When the computer system is executing on behalf of a user application, the system is in user mode.
- However, when a user application requests a service from the operating system (via a system call), the system must transition from user to kernel mode to fulfill the request.

OPERATING SYSTEM SERVICES

Dual-Mode and Multimode Operation



OPERATING SYSTEM SERVICES

Dual-Mode and Multimode Operation

- The dual mode of operation provides us with the means for protecting the operating system from errant users—and errant users from one another.
-
- We accomplish this protection by designating some of the machine instructions that may cause harm as privileged instructions.

- The hardware allows privileged instructions to be executed only in kernel mode.

OPERATING SYSTEM SERVICES

Timer

- We must ensure that the operating system maintains control over the CPU.

- We cannot allow a user program to get stuck in an infinite loop or to fail to call system services and never return control to the operating system.
  - To accomplish this goal, we can use a timer.

OPERATING SYSTEM SERVICES

Timer

- A timer can be set to interrupt the computer after a specified period.
- The period may be fixed (for example, 1/60 second) or variable (for example, from 1 millisecond to 1 second).
- A variable timer is generally implemented by a fixed-rate clock and a counter.

OPERATING SYSTEM SERVICES

Timer

- Before turning over control to the user, the operating system ensures that the timer is set to interrupt.

- If the timer interrupts, control transfers automatically to the operating system, which may treat the interrupt as a fatal error or may give the program more time.

- Clearly, instructions that modify the content of the timer are privileged.

OPERATING SYSTEM SERVICES

Resource Management

- As we have seen, an operating system is a resource manager.

- The system's CPU, memory space, file-storage space, and I/O devices are among the resources that the operating system must manage.

OPERATING SYSTEM SERVICES

Resource Management

- Generally the OS does the following
  - Process Management
  - Memory Management
  - File-System Management
  - Mass-Storage Management
  - Cache Management
  - I/O System Management
  - Security and Protection Management
  - Virtualization

VIRTUAL MACHINES

- Virtualization is a technology that allows us to abstract the hardware of a single computer (the CPU, memory, disk drives, network interface cards, and so forth) into several different execution environments, thereby creating the illusion that each separate environment is running on its own private computer.

- These environments can be viewed as different individual operating systems (for example, Windows and UNIX) that may be running at the same time and may interact with each other.

VIRTUAL MACHINES

- A user of a virtual machine can switch among the various operating systems in the same way a user can switch among the various processes running concurrently in a single operating system

- Virtualization allows operating systems to run as applications within other operating systems.

- Broadly speaking, virtualization software is one member of a class that also includes emulation.

VIRTUAL MACHINES

- Emulation, which involves simulating computer hardware in software, is typically used when the source CPU type is different from the target CPU type

- E.g., when Apple switched from the IBM Power CPU to the Intel x86 CPU in their machines, it included an emulation facility called "Rosetta," which allowed applications compiled for the IBM CPU to run on the Intel CPU.
- That same concept can be extended to allow an entire operating system written for one platform to run on another

VIRTUAL MACHINES

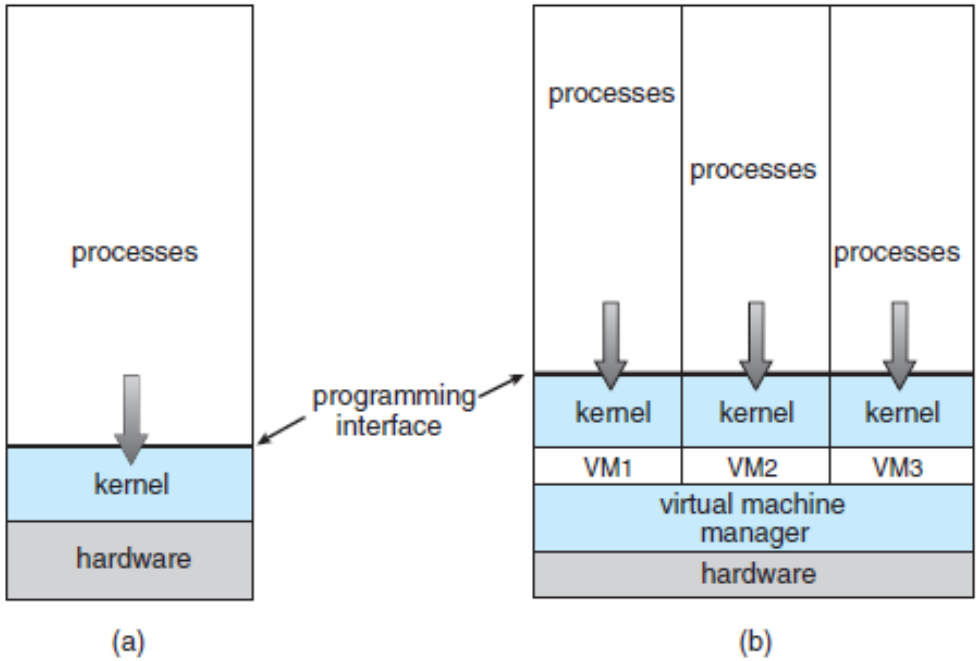


Figure 1.16 A computer running (a) a single operating system and (b) three virtual machines.

THE END