INTRODUCING OPERATING SYSTEMS

SWE 232

Operating Systems Internals and Design Principle, Eighth Edition

Learning Outcomes

1. Ability to **distinguish** the relationship between OS and hardware with the aid of process, thread, memory, file, device and network management.

   [C4-Analysis]

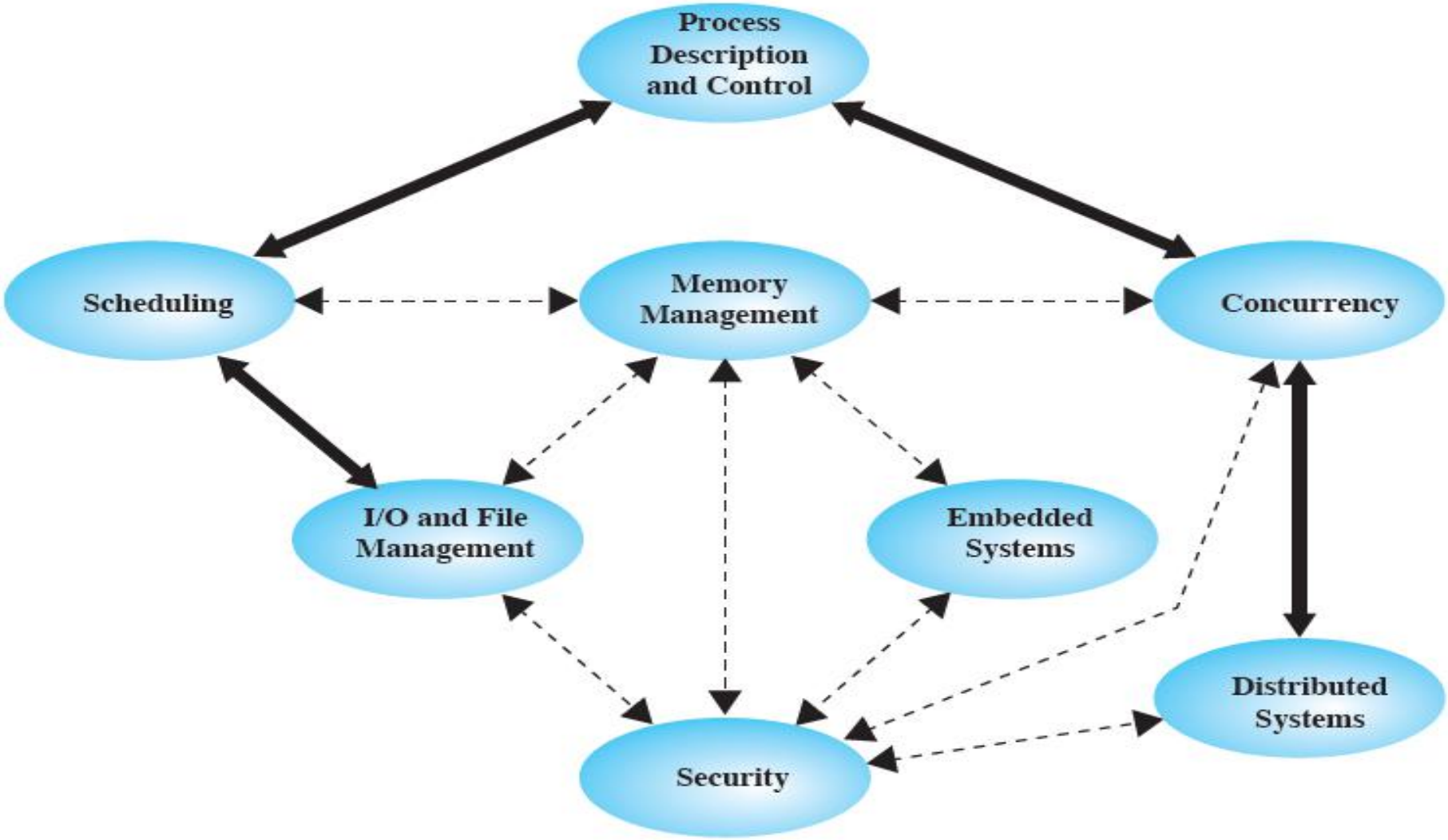
2. Ability to **design** and implement tiny application on GNU/Linux OS platform.

   [C5-Synthesis]

3. Investigate and analyze the current OSs issues using theoratical concepts of OS.

   [C4-Analysis]

Operating System Topics



Lesson Learning Objectives

After completing this chapter, you should be able to describe:

- Innovations in operating systems development
- The basic role of an operating system
- The major operating system software subsystem managers and their functions
- The types of machine hardware on which operating systems run

Lesson Learning Objectives (cont'd.)

- The differences among batch, interactive, real-time, hybrid, and embedded operating systems
- Multiprocessing and its impact on the evolution of operating system software
- Virtualization and core architecture trends in new operating systems

What is an Operating System?

- **Computer System**
  - Software (programs)
  - Hardware (physical machine and electronic components)
- **Operating System**
  - Part of computer system (software)
  - Manages all hardware and software
    - Manages every file, device, section of main memory and nanosecond of processing time
    - Controls *who* can use the system
    - Controls *how* system is used

What is an Operating System?

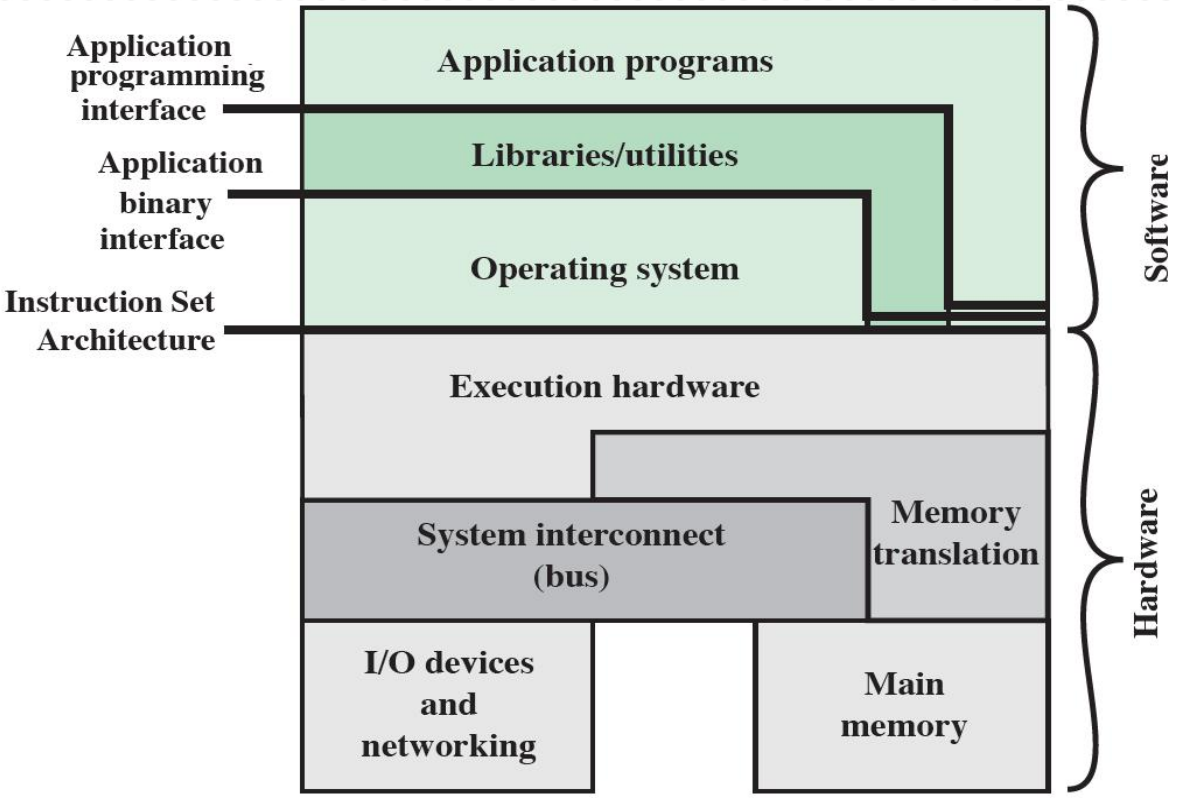


Figure: Computer Hardware and Software Structure

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

What is an Operating System?

- **Computer System**
  - S_____
  - H_____
- **Operating System**
  - S_____
  - M_____

A program that

_____

Operating System Software

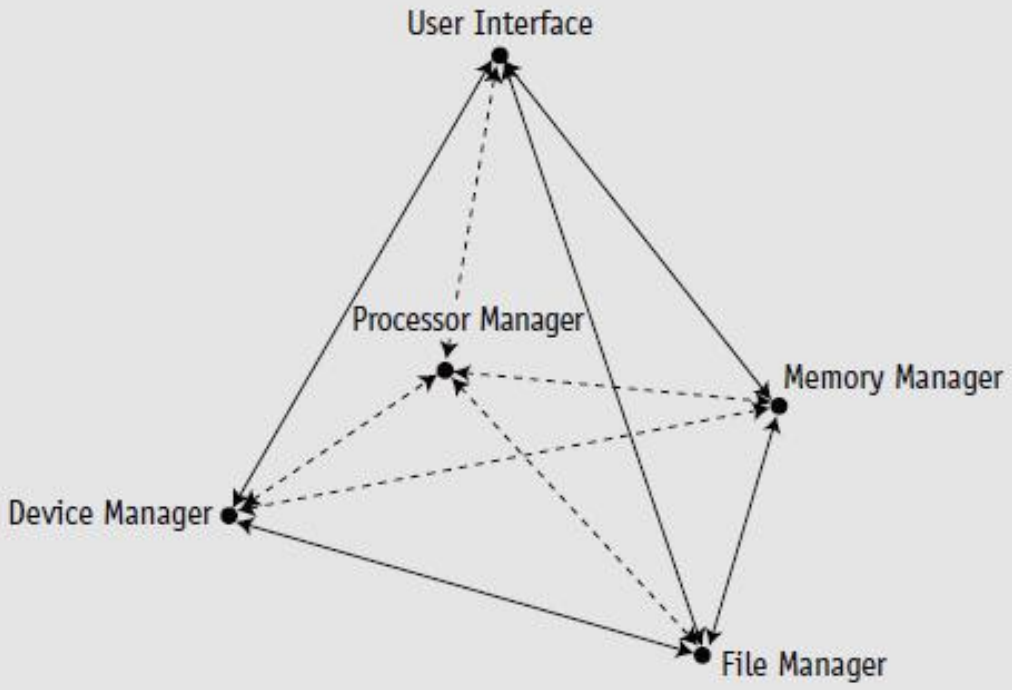
- Includes four essential subsystem managers
  - Memory Manager
  - Processor Manager
  - Device Manager
  - File Manager

  Network Manager (all modern OS)

Operating System Software (cont'd.)

(figure 1.1)

This model of a non-networked operating system shows four subsystem managers supporting the User Interface.



Operating System Software

- Includes four essential subsystem managers
  - M_____ Manager
  - P_____ Manager
  - D_____ Manager
  - F_____Manager
  - N_____Manager

Operating System Software (cont'd.)

- Each manager:
  - Works closely with other managers
  - Performs a unique role
- Manager tasks
  - Monitor its resources continuously
  - Enforce policies determining:
    - Who gets what, when, and how much
  - Allocate the resource (when appropriate)
  - Deallocate the resource (when appropriate)

Operating System Software (cont'd.)

- Resources include:
  - Hardware (CPUs, memory areas, printers, tape drives, modems, and disk drives)
  - Software (compilers, application programs, and data files)

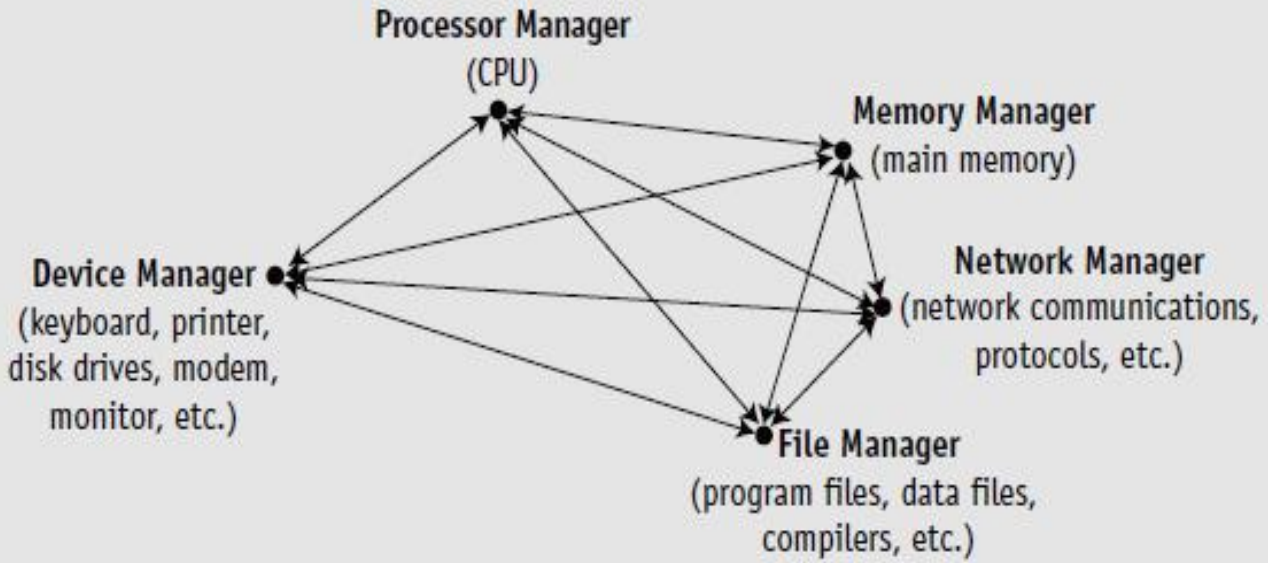
Operating System Software (cont'd.)

- Resources include:
  - H_____ (_____)
  - S_____ (_____)

Operating System Software (cont'd.)

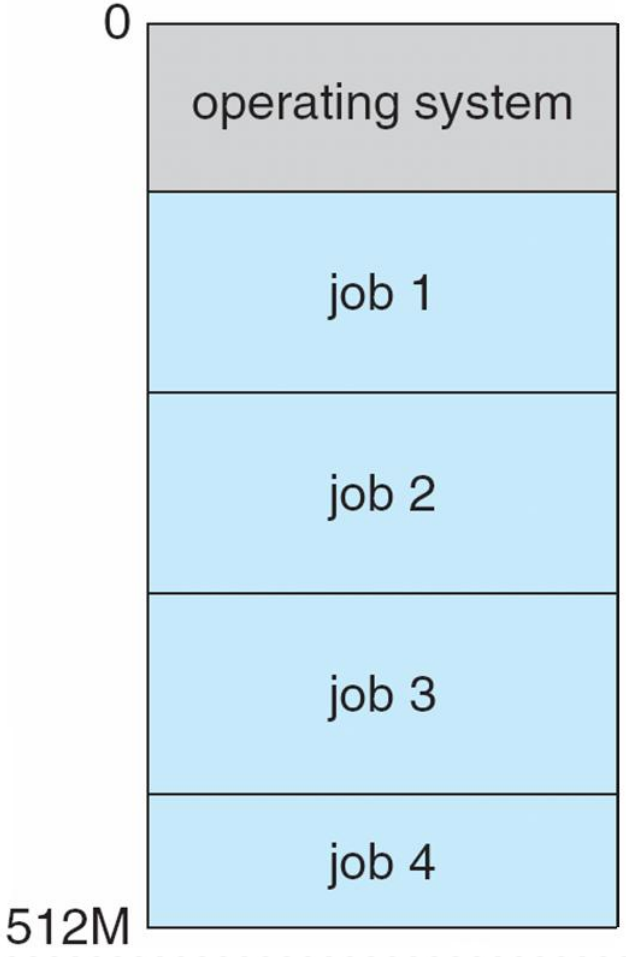
(figure 1.2)

Networked systems have a Network Manager that assumes responsibility for networking tasks while working harmoniously with every other manager.



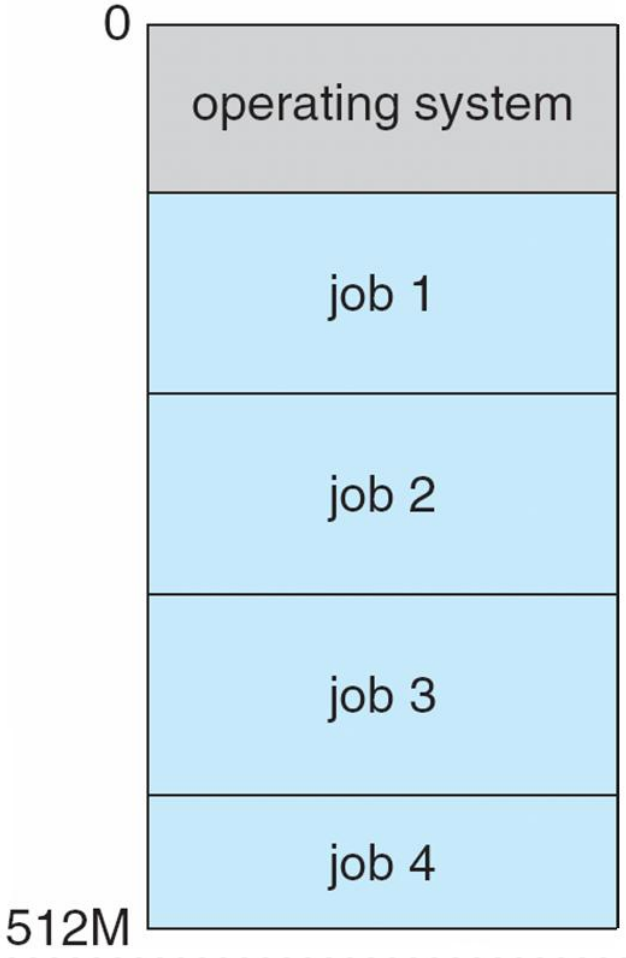
Main Memory Management

- In charge of main memory
  - Random Access Memory (RAM)
- Responsibilities include:
  - Preserving space in main memory occupied by operating system
  - Checking validity and legality of memory space request
  - Setting up memory tracking table
    - Tracks usage of memory by sections
    - Needed in multiuser environment
  - Deallocating memory to reclaim it



Main Memory Management

- In charge of main memory
  - R___ A___ M___ (RAM)
- Responsibilities include:
  - P_____ s_____
  - C____ v____ and l_____ of memory space request
  - S_____ t_____ t_____
  - D_____m_____



Processor Management

- In charge of allocating **Central Processing Unit** (CPU)
- Tracks **process** status
  - An instance of program execution
- Two levels of responsibility:
  - Handle jobs as they enter the system
    - Handled by Job Scheduler
  - Manage each process within those jobs
    - Handled by Process Scheduler

Processor Management

- allocating **C**_____**P**_____ **U**____
- T_____ **p**_____ s_____
- Two levels of responsibility:
  - H_____ jobs
    - Handled by Job Scheduler
  - M_____ process
    - Handled by Process Scheduler

Device Management

- In charge of monitoring all resources
  - Devices, channels, and control units
- Responsibilities include:
  - Choosing most efficient resource allocation method
    - Printers, ports, disk drives, etc.
    - Based on scheduling policy
  - Allocating the device
  - Starting device operation
  - Deallocating the device

Device Management

- M_____ all resources
- Responsibilities include:
  - C_____ most efficient resource allocation method
  - A_____ the device
  - S_____ device operation
  - D_____ the device

File Management

- In charge of tracking every file in the system
  - Data files, program files, compilers, application programs
- Responsibilities include:
  - Enforcing user/program resource access restrictions
    - Uses predetermined access policies
  - Controlling user/program modification restrictions
    - Read-only, read-write, create, delete
  - Allocating resource
    - Opening the file
    - Deallocating file (by closing it)

File Management

- T_____ every file in the system
- Responsibilities include:
  - E_____
  - C_____
    - Read-only, read-write, create, delete
  - A_____resource
    - O_____ the file
    - D_____ file (by closing it)

Cooperation Issues

- Essential manager
  - Perform individual tasks *and*
  - Harmoniously interact with other managers
    - Requires incredible precision
  - No single manager performs tasks in isolation
  - Network manager
    - Convenient way to share resources
    - Controls user access

A Brief History of Machine Hardware

- **Initial Hardware:** physical machine and electronic components
  - **Main memory (RAM)**
    - Data/Instruction storage and execution
  - **Input/Output devices (I/O devices)**
    - All peripheral devices in system
    - Printers, disk drives, CD/DVD drives, flash memory, and keyboards
  - **Central processing unit (CPU)**
    - Controls interpretation and execution of instructions
    - Controls operation of computer system

A Brief History of Machine Hardware (cont'd.)

- **Advances in computer technology**
  - Dramatic changes
    - Physical size, cost, and memory capacity
  - Networking
    - Integral part of modern computer systems
  - Mobile society information delivery
    - Creating strong market for handheld devices
  - New classification
    - By processor capacity, not memory capacity
  - Moore's Law
    - Computing power rises exponentially

A Brief History of Machine Hardware (cont'd.)

(table 1.1)

A brief list of platforms and sample operating systems listed in alphabetical order.

Platform	Operating System
Microcomputers	Linux, UNIX (includes Mac), Windows
Mainframe computers	IBM z/390, Linux, UNIX
Supercomputers	IRIX, Linux, UNICOS
Workstations, servers	Linux, UNIX, Windows
Networks	Linux, NetWare, UNIX, Windows
Personal digital assistants	BlackBerry, Linux, Palm OS, Windows Mobile

Types of Operating Systems

- Five categories
  - Batch
  - Interactive
  - Real-time
  - Hybrid
  - Embedded
- Two distinguishing features
  - Response time
  - How data enters into the system

Types of Operating Systems

- Five categories
  - B_____
  - I_____
  - R_____
  - H_____
  - E_____
- Two distinguishing features
  - Response time
  - How data enters into the system

Types of Operating Systems (cont'd.)

- **Batch Systems**
  - Input relied on punched cards or tape
  - Efficiency measured in throughput
- **Interactive Systems**
  - Faster turnaround than batch systems
  - Slower than real-time systems
  - Introduced to provide fast turnaround when debugging programs
  - Time-sharing software developed for operating system

Types of Operating Systems (cont'd.)

- **Real-time systems**
  - Reliability is key
  - Fast and time limit sensitive
  - Used in time-critical environments
    - Space flights, airport traffic control, high-speed aircraft
    - Industrial processes
    - Sophisticated medical equipment
    - Distribution of electricity
    - Telephone switching
  - Must be 100% responsive, 100% of the time

Types of Operating Systems (cont'd.)

- **Hybrid systems**
  - Combination of batch and interactive
  - Accept and run batch programs in the background
    - Interactive load is light
- **Embedded systems**
  - Computers placed inside other products
  - Adds features and capabilities
  - Operating system requirements
    - Perform specific set of programs
    - Not interchangeable among systems
    - Small kernel and flexible function capabilities

Brief History of Operating Systems Development

- **1940s: first generation**
  - Computers based on vacuum tube technology
  - No standard operating system software
  - Typical program included every instruction needed by the computer to perform the tasks requested
  - Poor machine utilization
    - CPU processed data and performed calculations for fraction of available time
  - Early programs
    - Designed to use the resources conservatively
    - Understandability is not a priority

Brief History of Operating Systems Development (cont'd.)

(figure 1.8)

Dr. Grace Hopper's research journal from her work on Harvard's Mark I computer in 1945 included the remains of the first computer "bug," a moth that had become trapped in the computer's relays causing the system to crash. Today's use of the term "bug" stems from that first moth.

Brief History of Operating Systems Development (cont'd.)

- **1950s: second generation**
  - Focused on cost effectiveness
  - Computers were expensive
    - IBM 7094: $200,000
  - Two widely adopted improvements
    - Computer operators: humans hired to facilitate machine operation
    - Concept of job scheduling: group together programs with similar requirements
  - Expensive time lags between CPU and I/O devices

Brief History of Operating Systems Development (cont'd.)

- **1950s: second generation (cont'd.)**
  - I/O device speed gradually became faster
    - Tape drives, disks, and drums
  - Records blocked *before* retrieval or storage
  - Access methods developed
    - Added to object code by linkage editor
  - Buffer between I/O and CPU introduced
    - Reduced speed discrepancy
  - Timer interrupts developed
    - Allowed job-sharing

Brief History of Operating Systems Development (cont'd.)

- **1960s: third generation**
  - Faster CPUs
  - Speed caused problems with slower I/O devices
  - Multiprogramming
    - Allowed loading many programs at one time
  - Program scheduling
    - Initiated with second-generation systems
    - Continues today
  - Few advances in data management
  - Total operating system customization
    - Suit user's needs

Brief History of Operating Systems Development (cont'd.)

- **1970s**
  - Faster CPUs
  - Speed caused problems with slower I/O devices
  - Main memory physical capacity limitations
    - Multiprogramming schemes used to increase CPU
    - Virtual memory developed to solve physical limitation
  - Database management software
    - Became a popular tool
  - A number of query systems introduced
  - Programs started using English-like words, modular structures, and standard operations

Brief History of Operating Systems Development (cont'd.)

- **1980s**
  - **Cost/performance ratio** improvement of computer components
  - More flexible hardware (firmware)
  - **Multiprocessing**
    - Allowed parallel program execution
  - Evolution of personal computers
  - Evolution of high-speed communications
  - **Distributed processing** and **networked systems** introduced

Brief History of Operating Systems Development (cont'd.)

- **1990s**
  - Demand for Internet capability
    - Sparked proliferation of networking capability
    - Increased networking
    - Increased tighter security demands to protect hardware and software
  - Multimedia applications
    - Demanding additional power, flexibility, and device compatibility for most operating systems

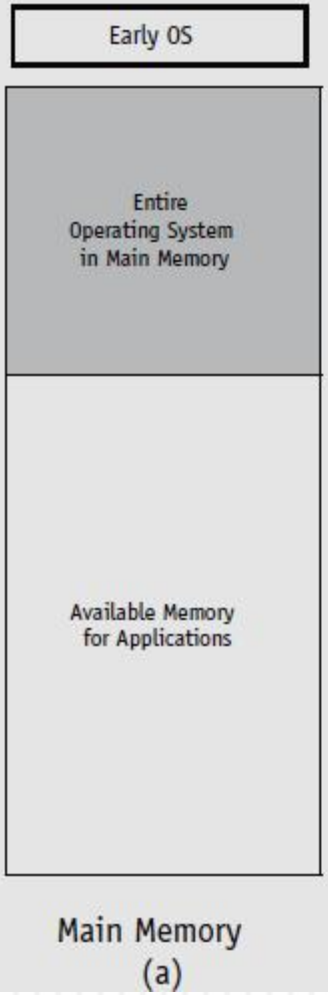
Brief History of Operating Systems Development (cont'd.)

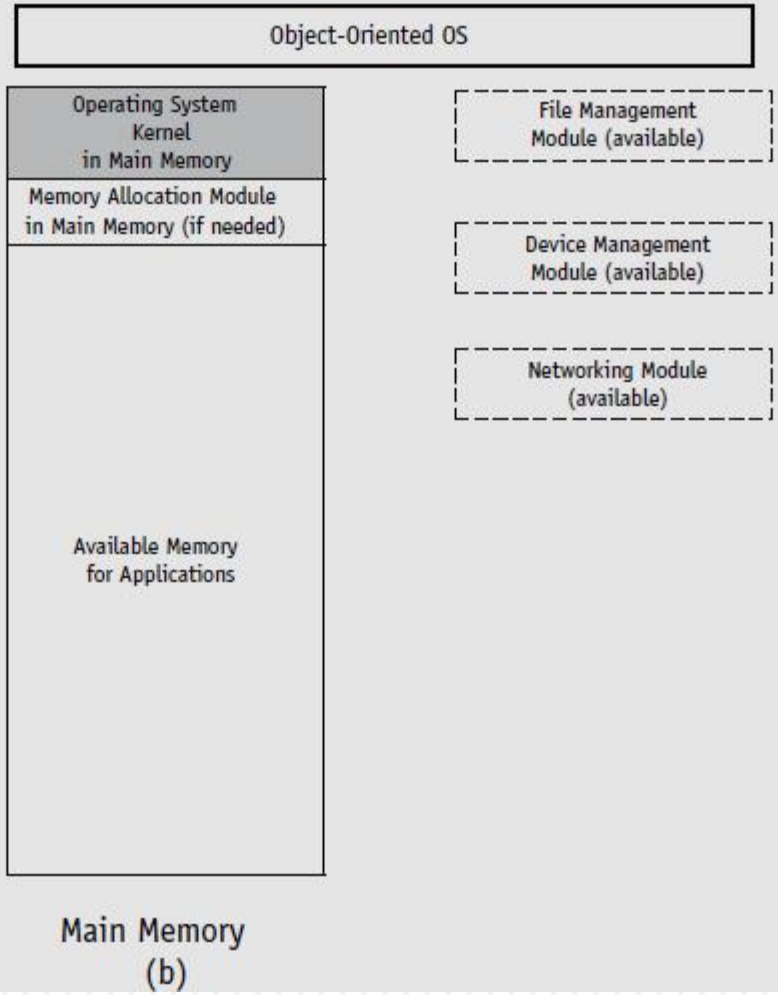
- **2000s**
  - Primary design features support:
    - Multimedia applications
    - Internet and Web access
    - Client/server computing
  - Computer systems requirements
    - Increased CPU speed
    - High-speed network attachments
    - Increased number and variety of storage devices
  - Virtualization
    - Single server supports different operating systems

Object-Oriented Design

- Driving force in system architecture improvements
  - **Kernel** (operating system nucleus)
    - Resides in memory at all times, performs essential tasks, and protected by hardware
  - **Kernel reorganization**
    - Memory resident: process scheduling and memory allocation
    - Modules: all other functions
  - Advantages
    - Modification and customization without disrupting integrity of the remainder of the system
    - Software development more productive

Object-Oriented Design (cont'd.)





(figure 1.15)

Early operating systems (a) loaded in their entirety into main memory. Object-oriented operating systems (b) load only the critical elements into main memory and call other objects as needed.

Principle of Locality

- Find in the book (Appendix 1A: Performance Characteristics of Two-Level Memories): Page 69

Summary

- Operating system overview
- Functions of OS
  - Manages computer system
    - Hardware and software
  - Four essential managers
    - Work closely with the other managers and perform unique role
  - Network Manager
    - Operating systems with networking capability
  - Essential hardware components
    - Memory chips, I/O, storage devices, and CPU

Summary (cont'd.)

- Evolution of OSs
  - Run increasingly complex computers
  - Run increasingly complex computer systems
  - Prior to mid-1970s
    - Computers classified by capacity and price
  - Dramatic changes over time
    - Moore's Law: computing power rises exponentially
    - Physical size, cost, and memory capacity
- Mobile society information delivery
  - Creates strong market for handheld devices
  - Integral in modern computer systems

Summary (cont'd.)

- Five categories of operating systems
  - Batch, interactive, real-time, hybrid, and embedded
- Use of object-oriented design improves the system architecture
- Several ways to perform OS tasks
- Designer determines policies to match system's environment