Layers of Computer System

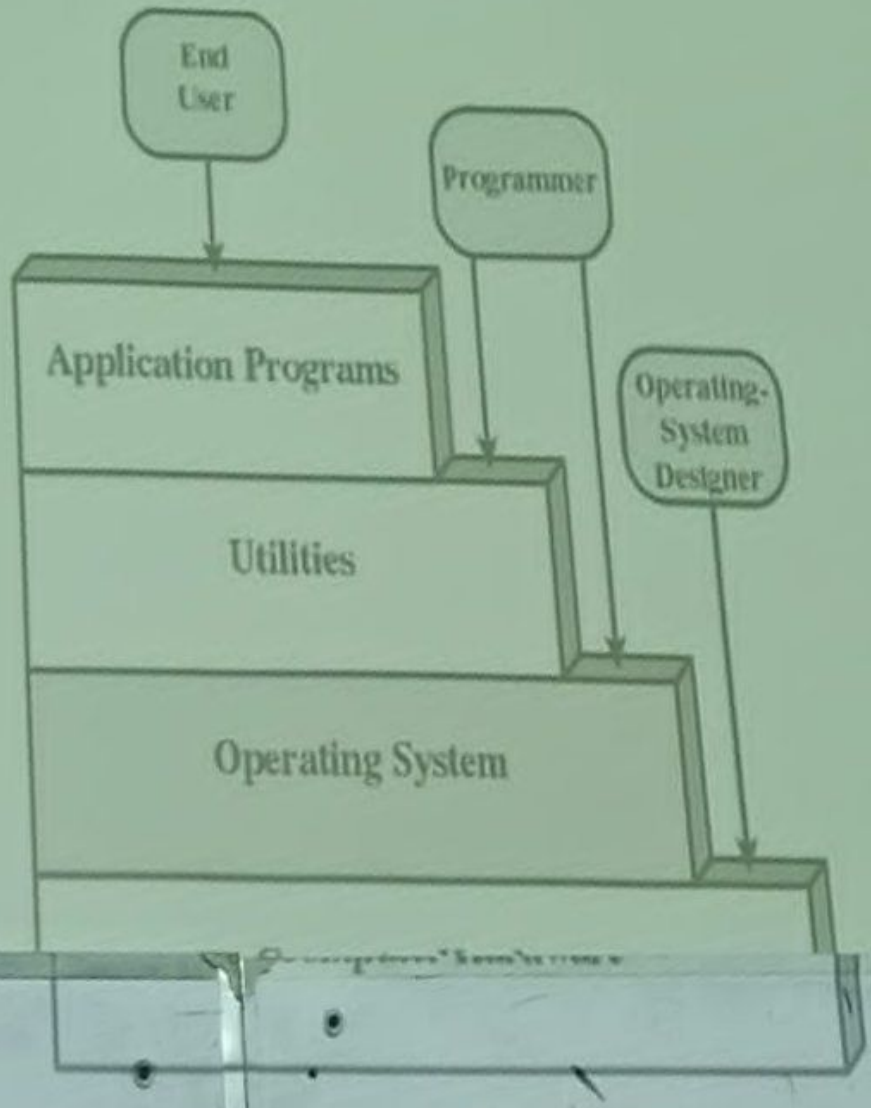


Figure 2.1 Layers and Views of a Computer System

Services Provided by the Operating System

- Program development
  - Editors and debuggers
- Program execution
- Access to I/O devices
- Protected access to files

Services Provided by the Operating System

- Error detection and response
  - internal and external hardware errors
    - memory error
    - device failure
  - software errors
    - arithmetic overflow
    - access forbidden memory locations
  - error recovery

Services Provided by the Operating Syste

- Accounting
  - collect statistics
  - monitor performance
  - used to anticipate future enhancements
  - used for billing users

Kernel

- Portion of operating system that is resident in main memory
- Contains a machine-independent part (code for system calls) and a machine-dependent part (device drivers)
- Maintains the OS state
- Executes in privileged/supervisor mode

Evolution of Operating Systems

- Simple Batch Systems
  - Monitors
    - Software that controls the running programs
    - Batch jobs together
    - Program branches back to monitor when finished
    - Resident monitor is in main memory and available for execution

Job Control Language (JCL)

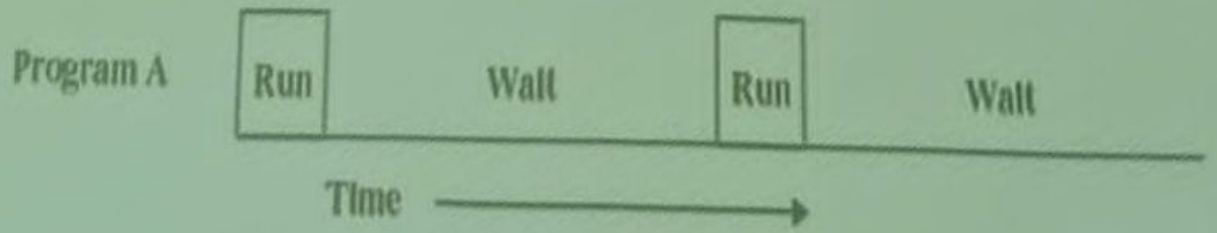
- Special type of programming language
- Provides instruction to the monitor
  - what compiler to use
  - what data to use

Evolution of Operating Systems

- Serial Processing
  - No operating system
  - Machines run from a console with display lights and toggle switches, input device, and printer
  - Schedule tome
  - Setup included loading the compiler, source program, saving compiled program, and loading and linking

Uniprogramming

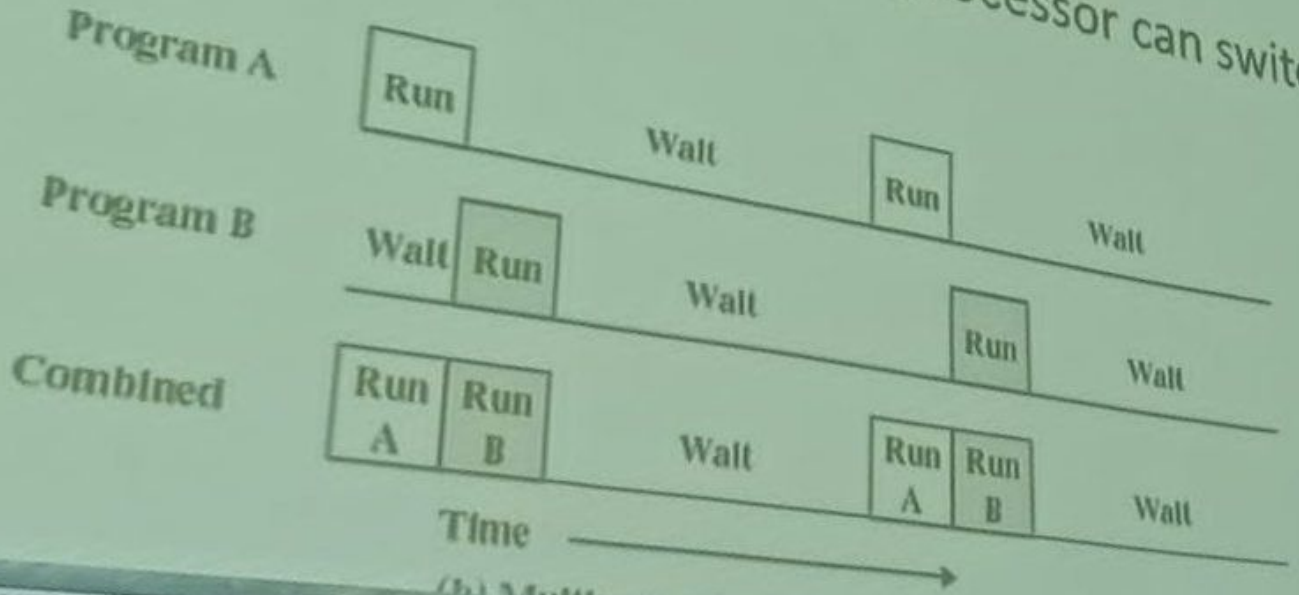
- Processor must wait for I/O instruction to complete before preceding



(a) Uniprogramming

Multiprogramming

- When one job needs to wait for I/O, the processor can switch to th other job



Time Sharing

- Using multiprogramming to handle multiple interactive jobs
- Processor's time is shared among multiple users
- Multiple users simultaneously access the system through terminals

Main OS Concepts

- Processes
- Memory Management
- Information protection and security
- Scheduling and resource management
- System structure

Process

- Consists of three components
  - An executable program
  - Associated data needed by the program
  - Execution context of the program
    - All information the operating system needs to manage the process

Process

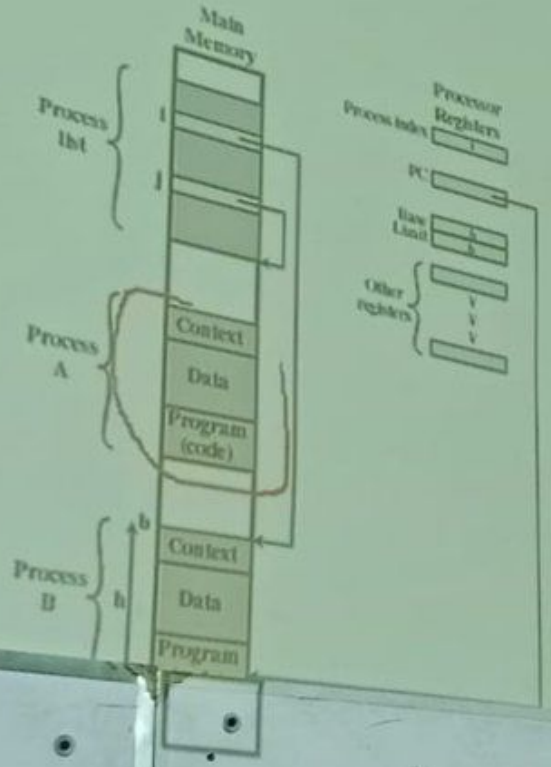


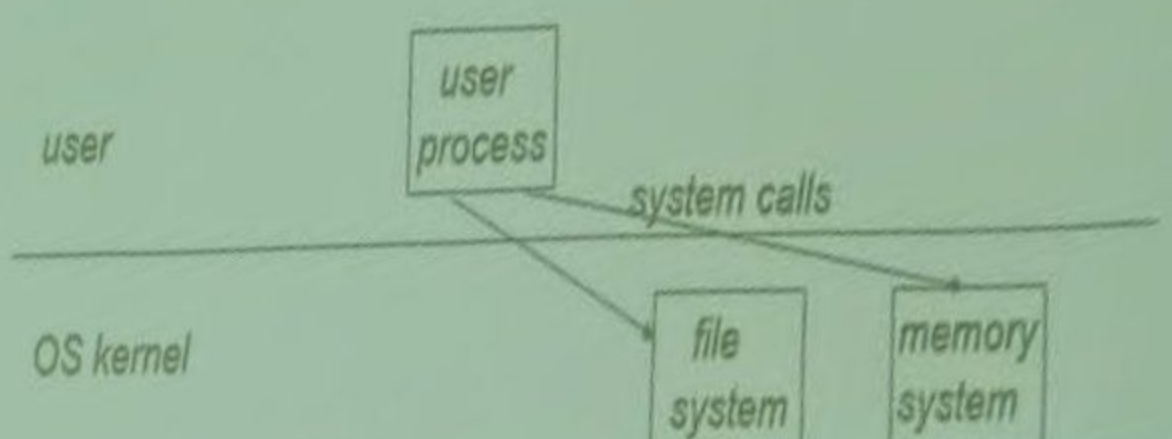
Figure 2.8 Typical Process Implementation

Memory Management

- Process isolation
- Automatic allocation and management
- Support for modular programming
- Protection and access control
- Long-term storage

Traditional OS structure

- monolithic/layered systems
  - one/N layers all executed in "kernel-mode"
  - good performance but rigid



Characteristics of Modern Operating Syste

- Multithreading
  - process is divided into threads that can run simultaneously
- Thread
  - dispatchable unit of work
  - executes sequentially and is interruptable
- Process is a collection of one or more threads and associate resources

Characteristics of Modern Operating Systems

- Distributed operating systems
  - provides the illusion of a single main memory and single secondary memory space
  - used for distributed file system

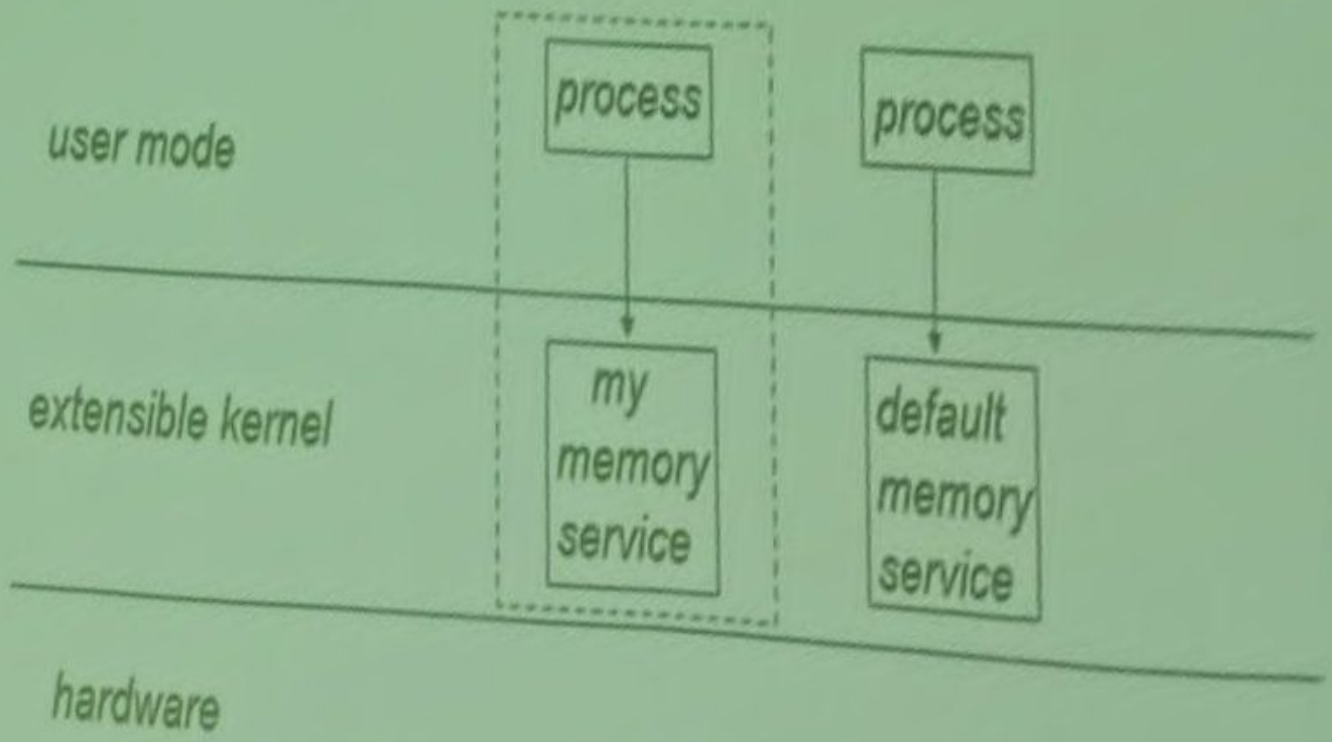
Characteristics of Modern Operating Systems

- Symmetric multiprocessing
  - there are multiple processors
  - these processors share same main memory and I/O facilities
  - All processors can perform the same functions

Characteristics of Modern Operating Systems

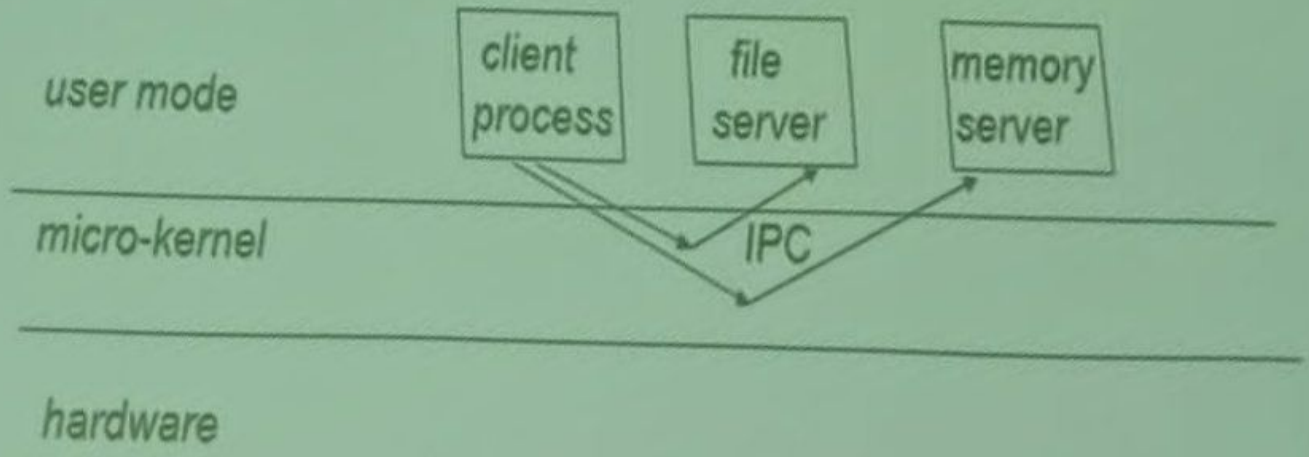
- Microkernel architecture
  - assigns only a few essential functions to the kernel
    - address space
    - interprocess communication (IPC)
    - basic scheduling

Extensible OS kernel



- user processes can load customized OS services into the kernel
- good performance but protection and scalability become problems

Micro-kernel OS



- client-server model, IPC between clients and servers
- the micro-kernel provides protected communication
- OS functions implemented as user-level servers
- flexible but efficiency is the problem
- easy to extend for distributed systems

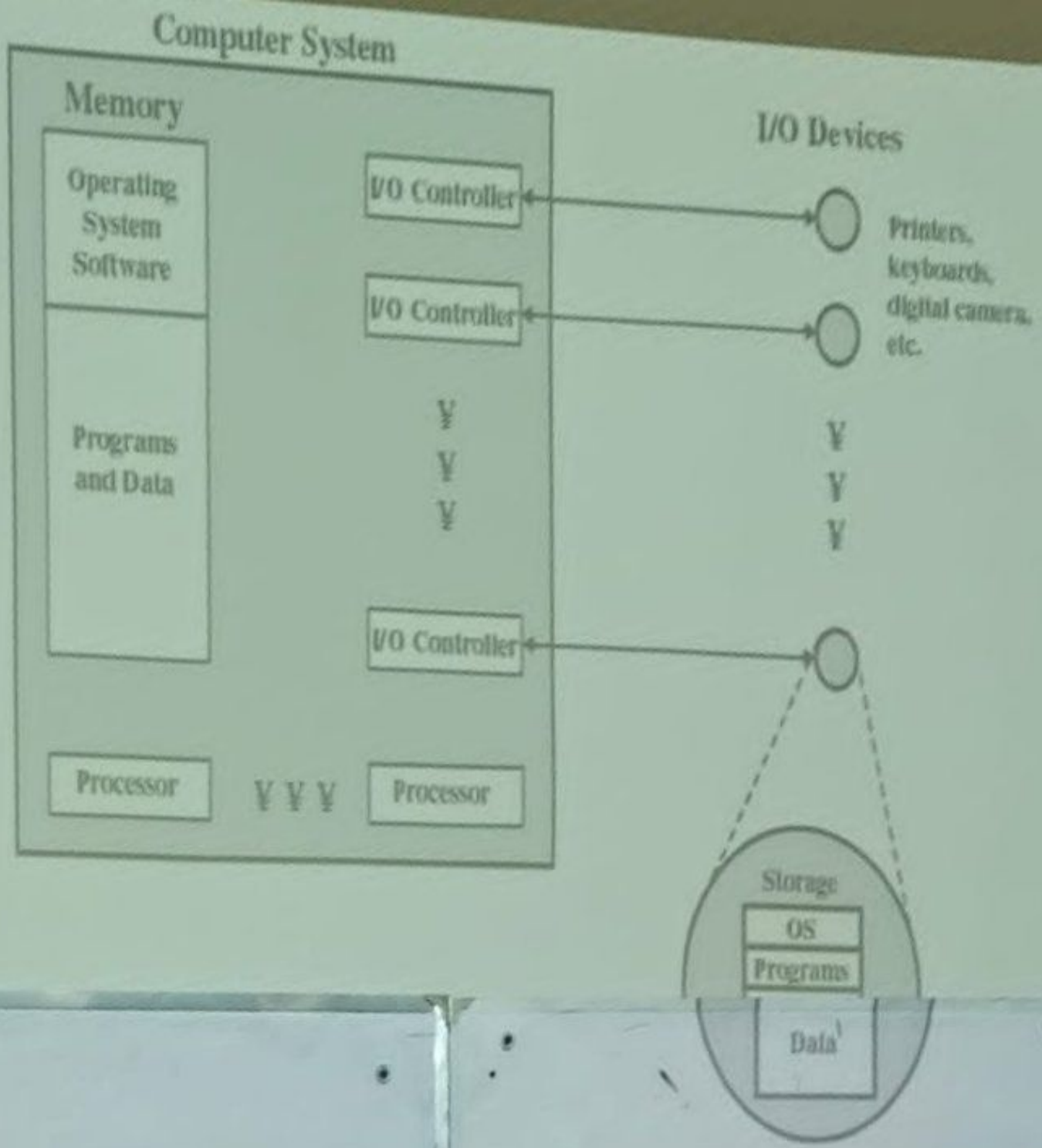


Figure 2.2 The Operating System as Resource Manager

Required Texts and Readings

1. Operating Systems Concepts, 9th edition by Abraham Silberschatz
2. Modern Operating Systems, 4th edition by Andrew S. Tanenbaum
3. Operating Systems, Internals and Design Principles, 9th edition by William Stallings