Introduction

Fan Wu

Department of Computer Science and Engineering

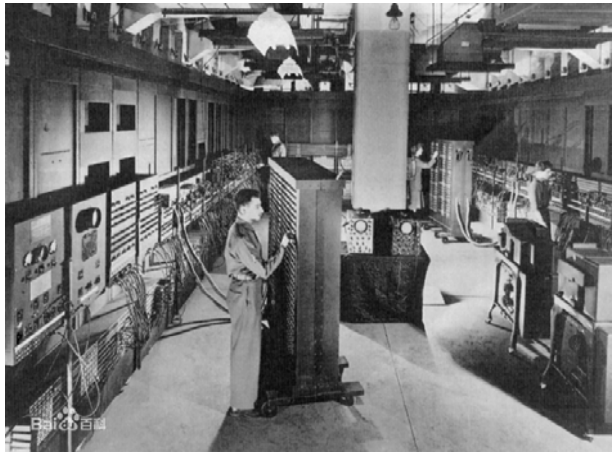
Shanghai Jiao Tong University

Spring 2022

First Generation - 1940's

- Mechanical relays → vacuum tubes
- No operating system
- No programming language
- Wiring up plugboards to control the computer
- Numerical calculations

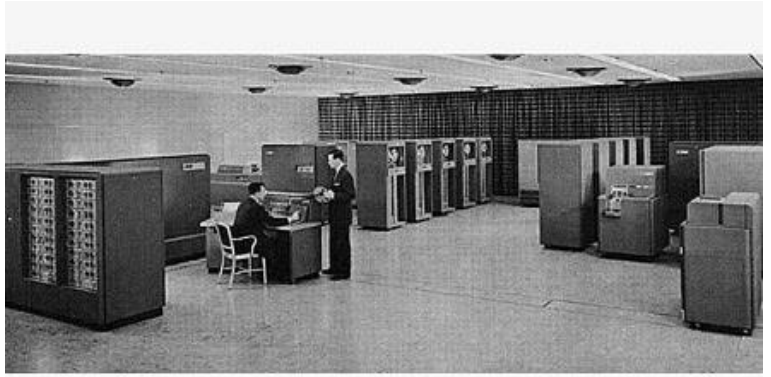
Atanasoff–Berry Computer (ABC) in 1942



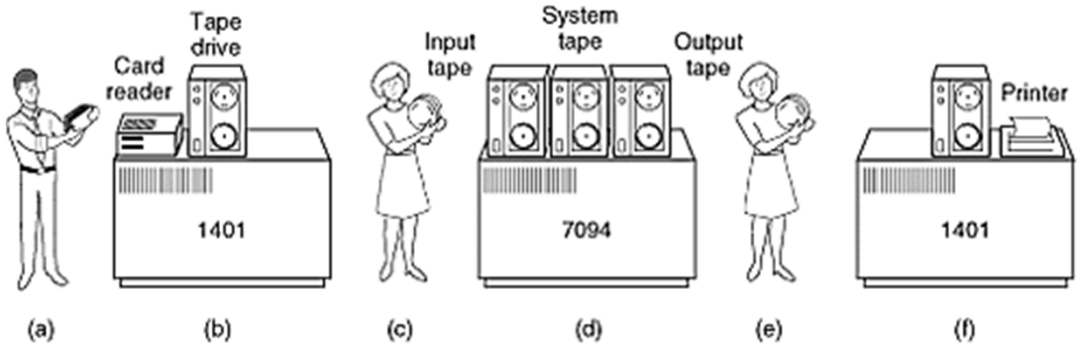
ENIAC in 1946

Second Generation - 1950's

- Transistors
- First operating system: GM-NAA-I/O
  - General Motors research division
  - For IBM mainframe computers
- Punched paper cards
  - FORTRAN or assembler
- Single-stream batch system
  - Programs and data were submitted in groups



IBM 701 in 1952



Third Generation - 1960's

- Integrated Circuits (ICs)
- OS/360
- New features:
  - Multiprogramming 多用户程序
  - Spooling 假脱机
  - Time-sharing 兼容
    - CTSS(Compatible Time Sharing System))
- MULTICS (MULTiplexed Information and Computing Service) x did not isolate users

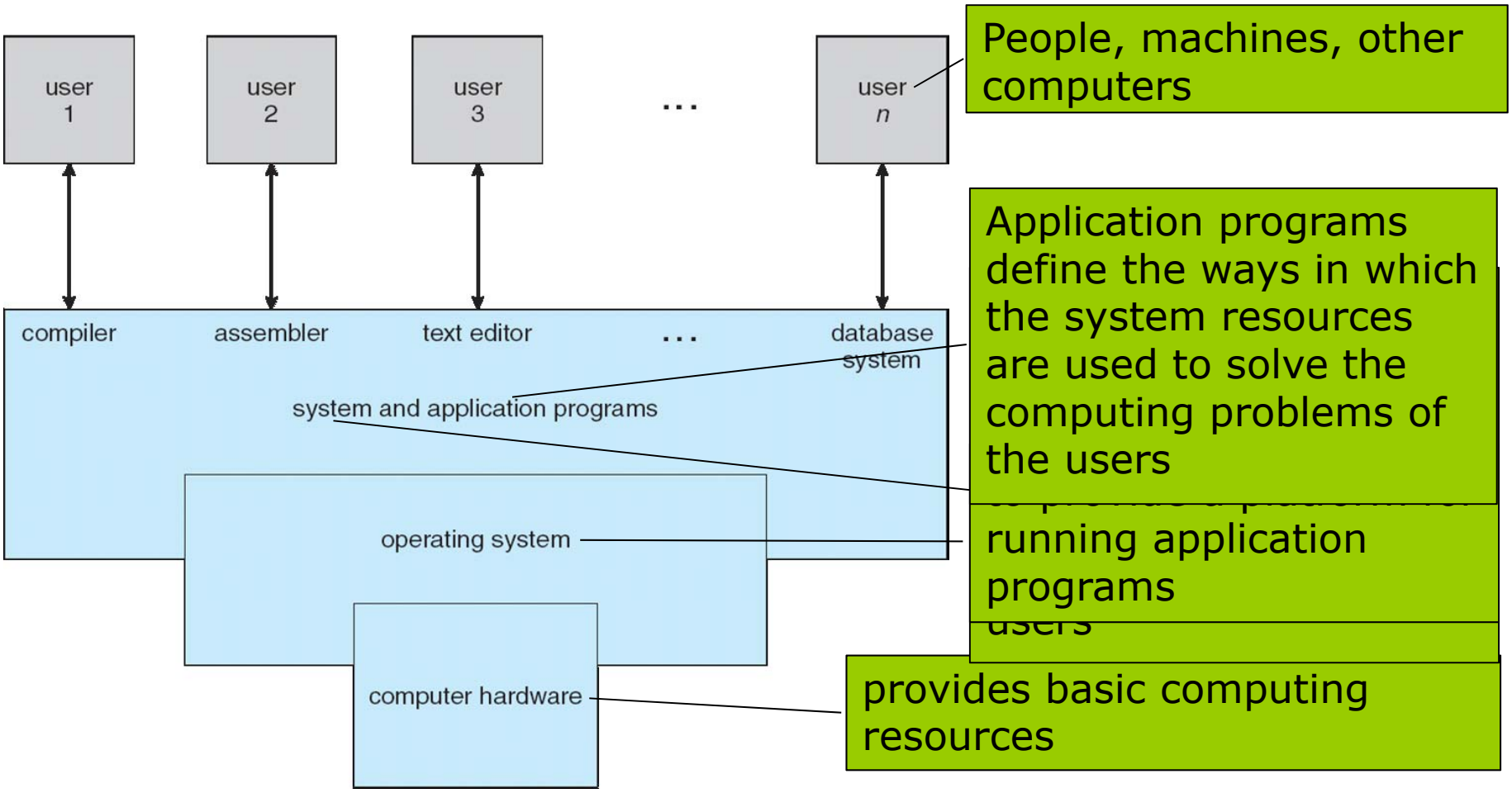
IBM System/360 in 1964

Fourth Generation – 1970's~

Operating Systems

- UNIX-family: BSD(Berkeley Software Distribution), System-V, GNU/Linux, MINIX, Nachos, OS X, iOS the largest family.
- BSD-family: FreeBSD, NetBSD, OpenBSD
- System-V-family: AIX, HP-UX, IRIX, Solaris
- Linux-family: Red Hat, Debian, Ubuntu, Fedora, openSUSE, Linux Mint, Google's Android, WebOS, Meego
- Microsoft-family: MS-DOS, Microsoft Windows, Windows Mobile, Win-CE, WP
- AmigaOS
- Symbian, MeeGo
- Google Chrome OS
- OS/2
- XrossMediaBar(XMB) for PS3, Orbis OS for PS4
- Input Output System for Wii
- Tiny-OS, LynxOS, QNX, VxWorks

Four Components of a Computer System



Computer System Structure

- **Hardware** – provides basic computing resources
  - CPU, memory, I/O devices
- **Operating system** – Controls and coordinates use of hardware among various applications and users
- **System programs** – are computer software designed to operate the computer hardware and to provide a platform for running application programs
  - BIOS and device drivers
- **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

What is an Operating System?

- An operating system is a program that manages the computer hardware
- A program that acts as an intermediary between the computer user 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

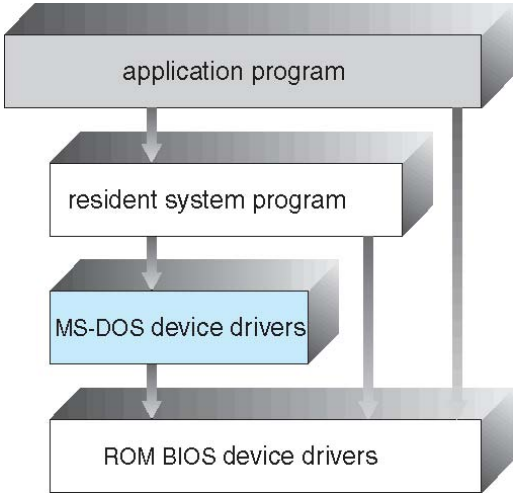
Operating System Definition

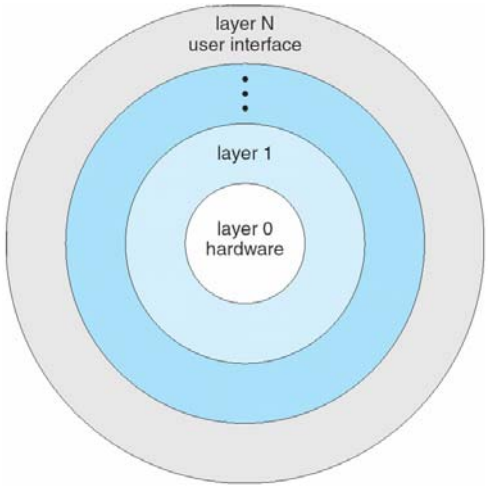
- OS is a **resource allocator**
  - Manages all resources
  - Decides between conflicting requests for efficient and fair resource use
- OS is a **control program**
  - Controls execution of programs to prevent errors and improper use of the computer

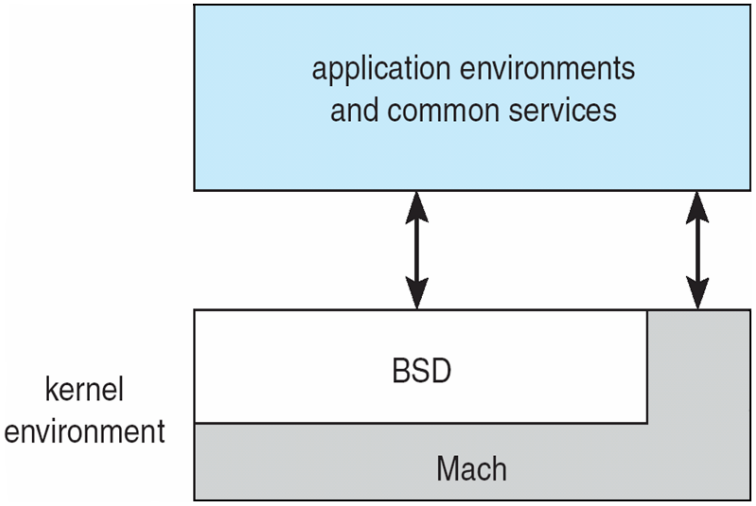
Operating System Definition (Cont.)

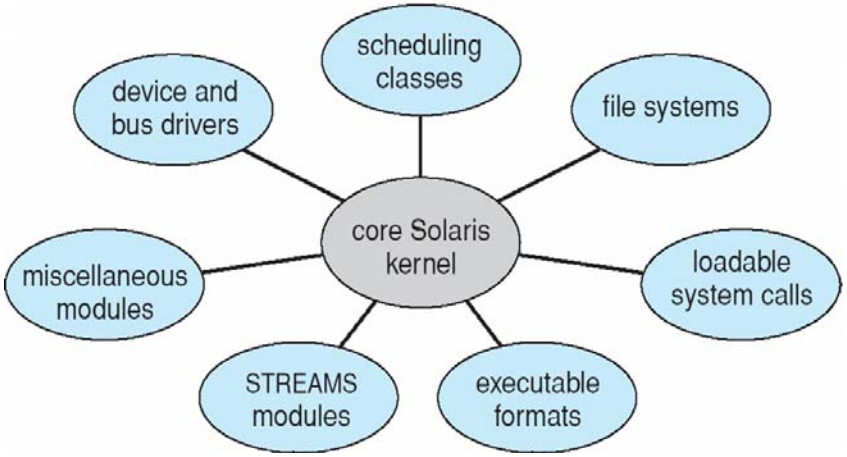
- No universally accepted definition
- “Everything a vendor ships when you order an operating system” is good approximation
  - But varies wildly
- “The one program running at all times on the computer” is the **kernel**. Everything else is either a system program (ships with the operating system) or an application program.
- “An operating system (OS) is system **software**, consisting of programs and data, that **runs on computers, manages computer hardware and software resources, and provides common services for execution of various application software**.” --- From Wikipedia

Operating System Structures

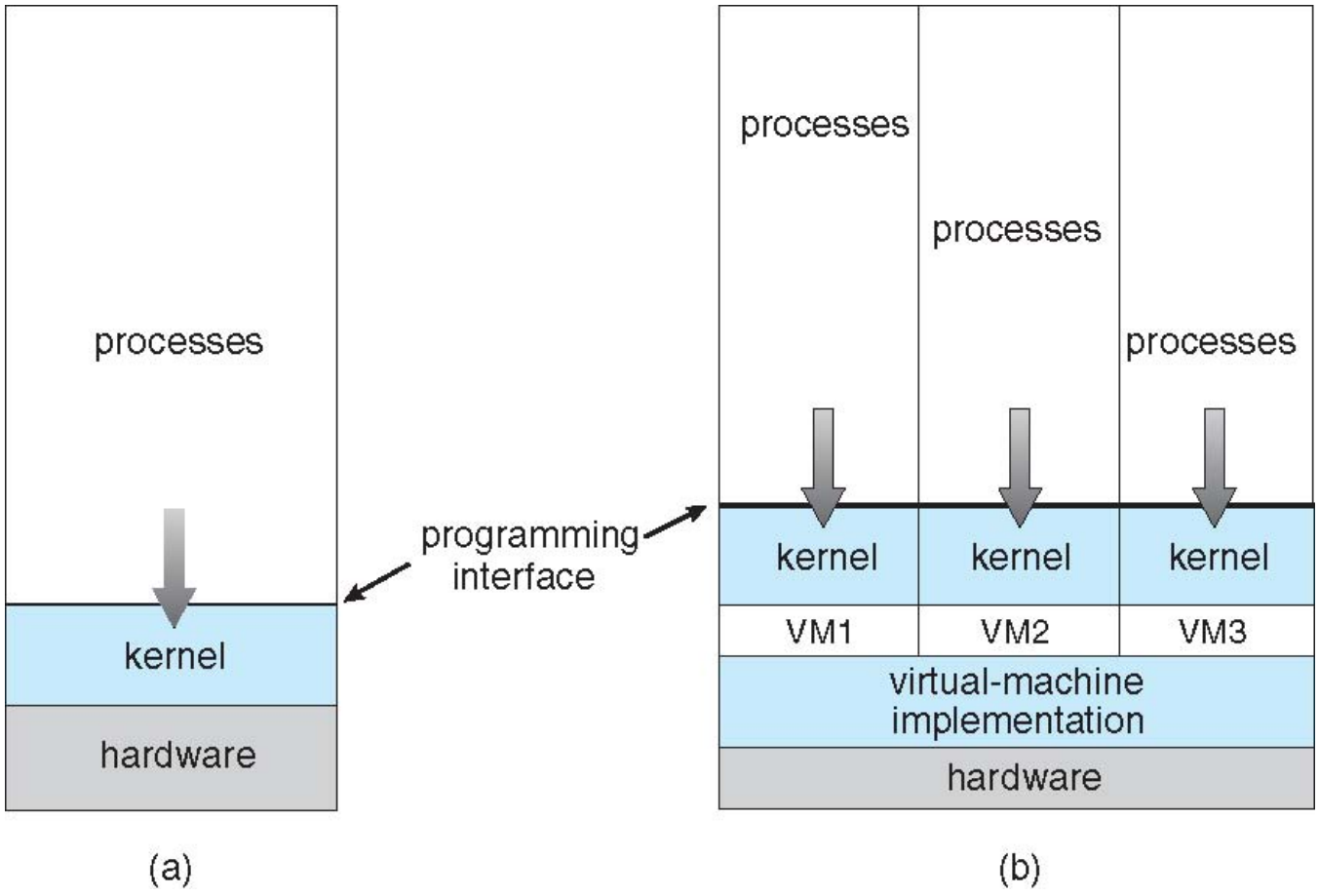






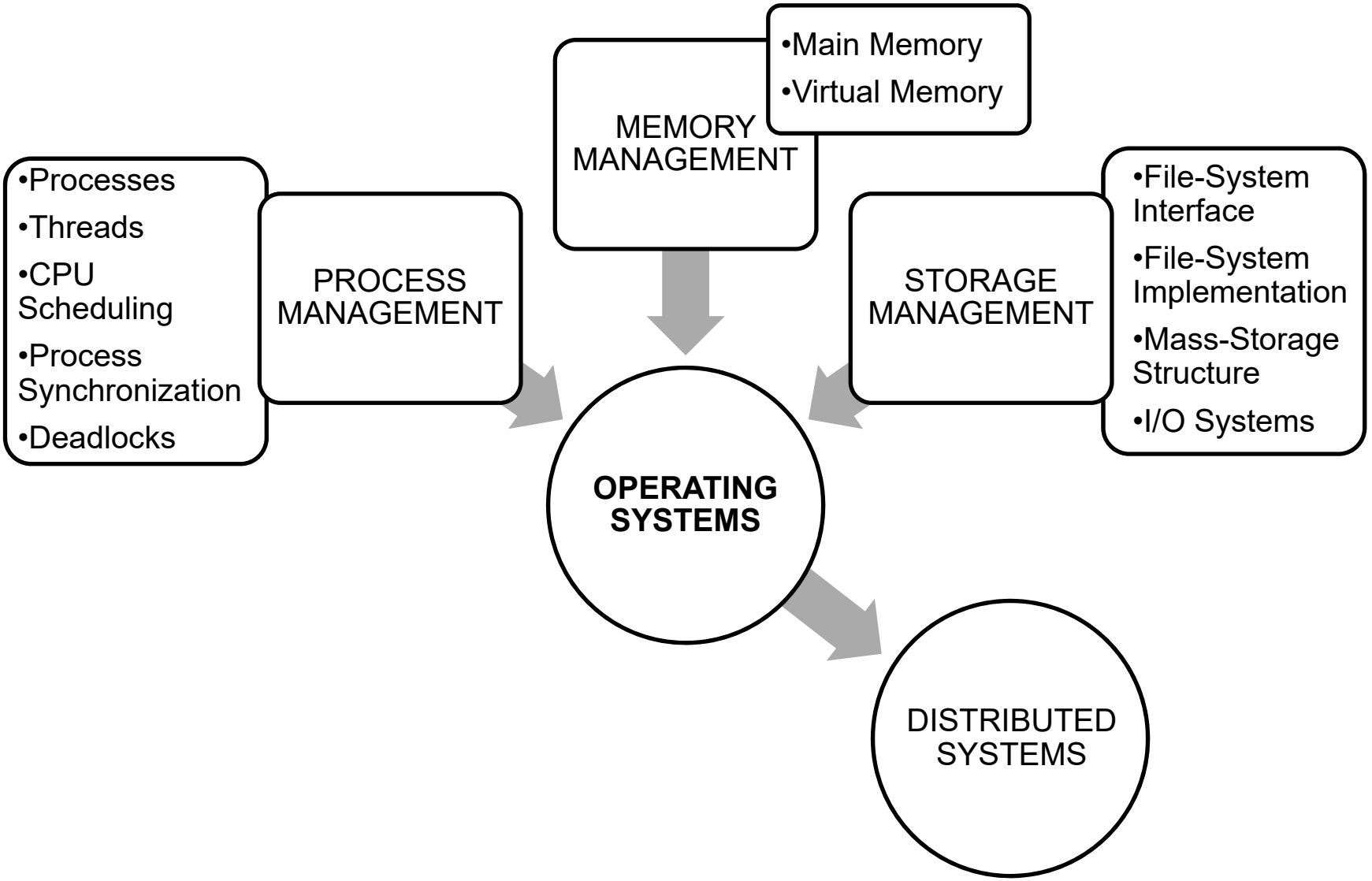


Virtual Machines

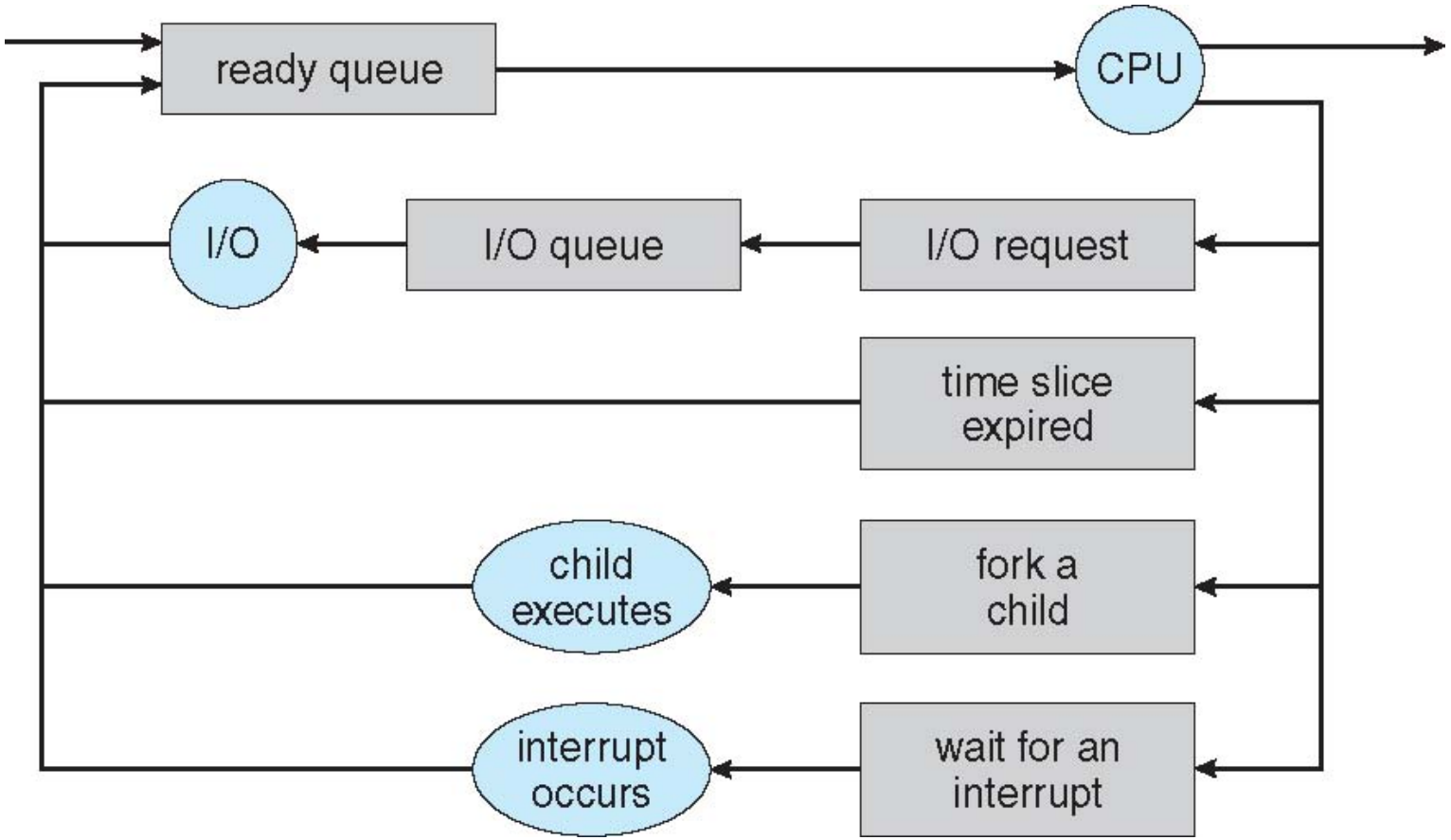


(a) Bare Machine (b) Virtual Machine

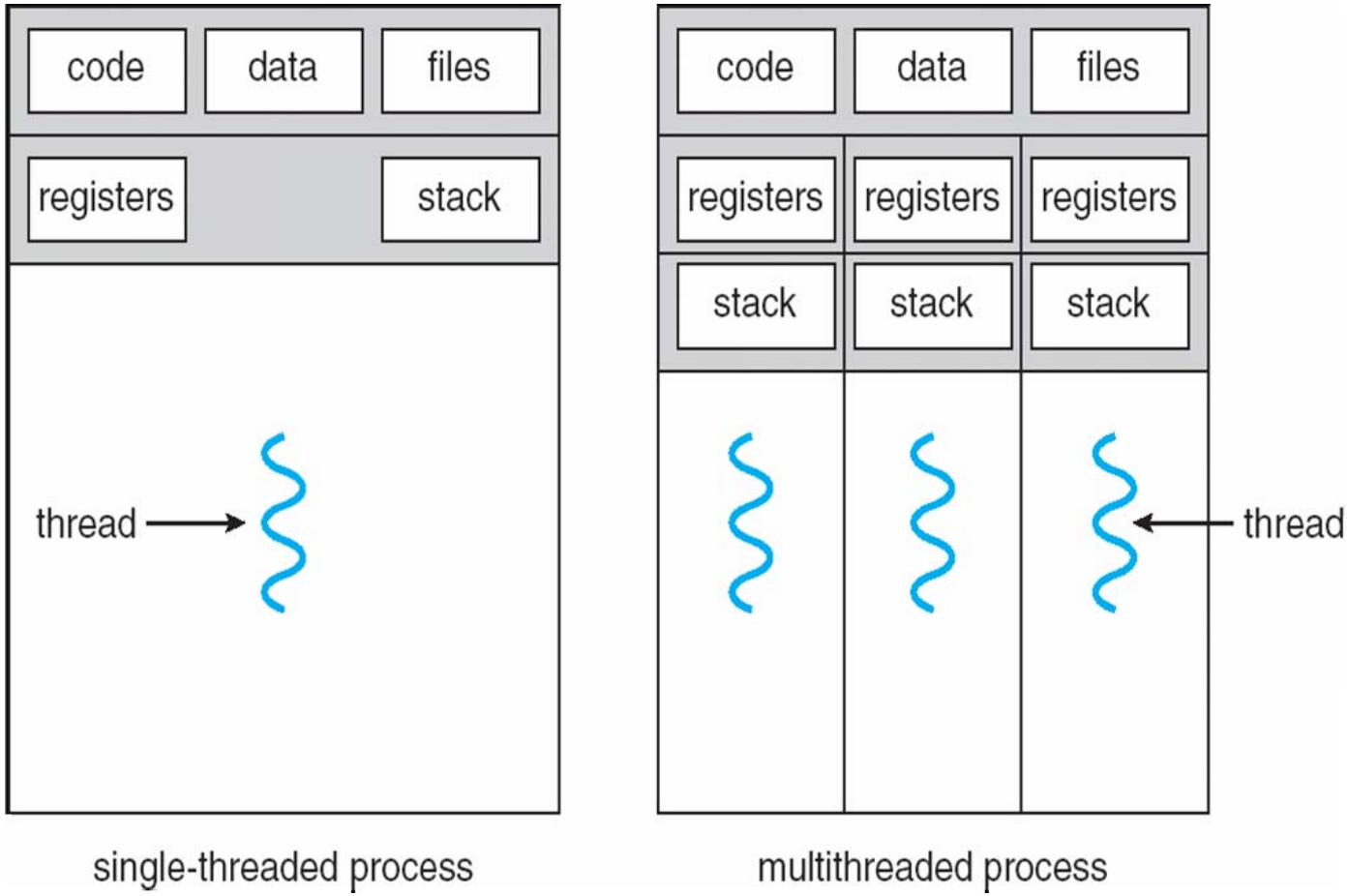
Course Outline



Process Scheduling

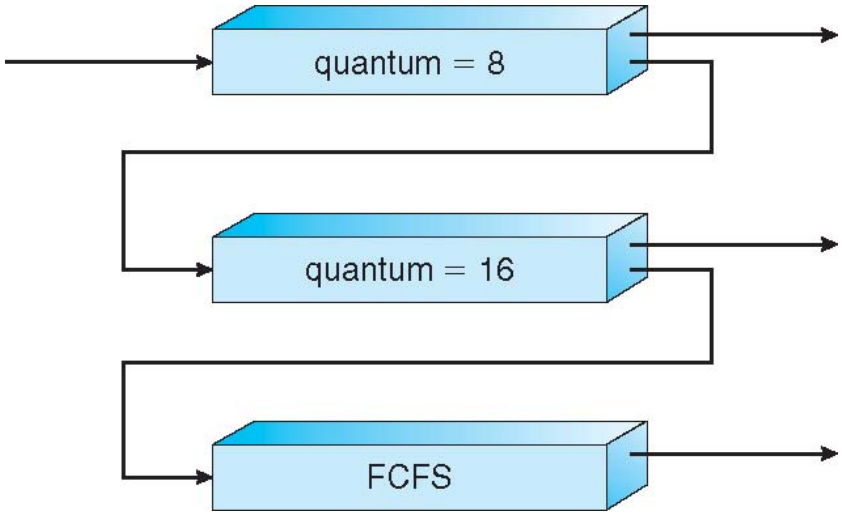


Single and Multithreaded Processes



CPU Scheduling

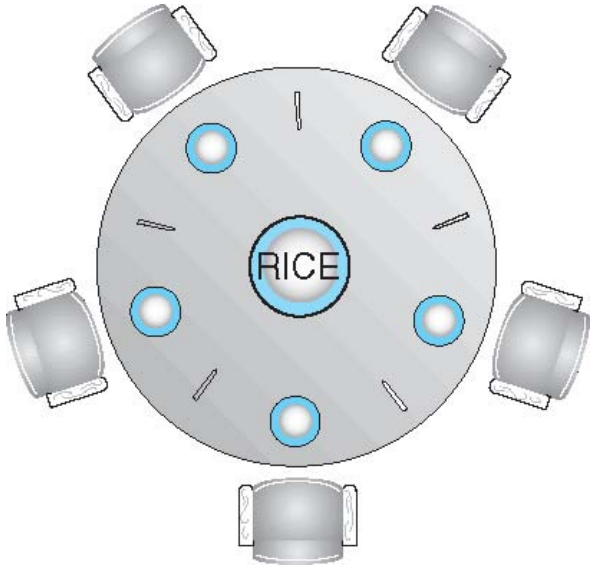
- First-Come, First-Served (FCFS) Scheduling
- Shortest-Job-First (SJF) Scheduling
- Priority Scheduling
- Round-Robin Scheduling
- Multilevel Queue Scheduling
- Multilevel Feedback Queue Scheduling



Process Synchronization

Dining-Philosophers Problem

- Philosophers spend their lives thinking and eating
- Don't interact with their neighbors, occasionally try to pick up 2 chopsticks (one at a time) to eat from bowl
  - Need both to eat, then release both when done
- In the case of 5 philosophers
  - Shared data
    - Bowl of rice (data set)
    - Semaphore chopstick [5] initialized to 1



Deadlock Avoidance

Example of Banker's Algorithm

- 5 processes P_0 through P_4;

  3 resource types:

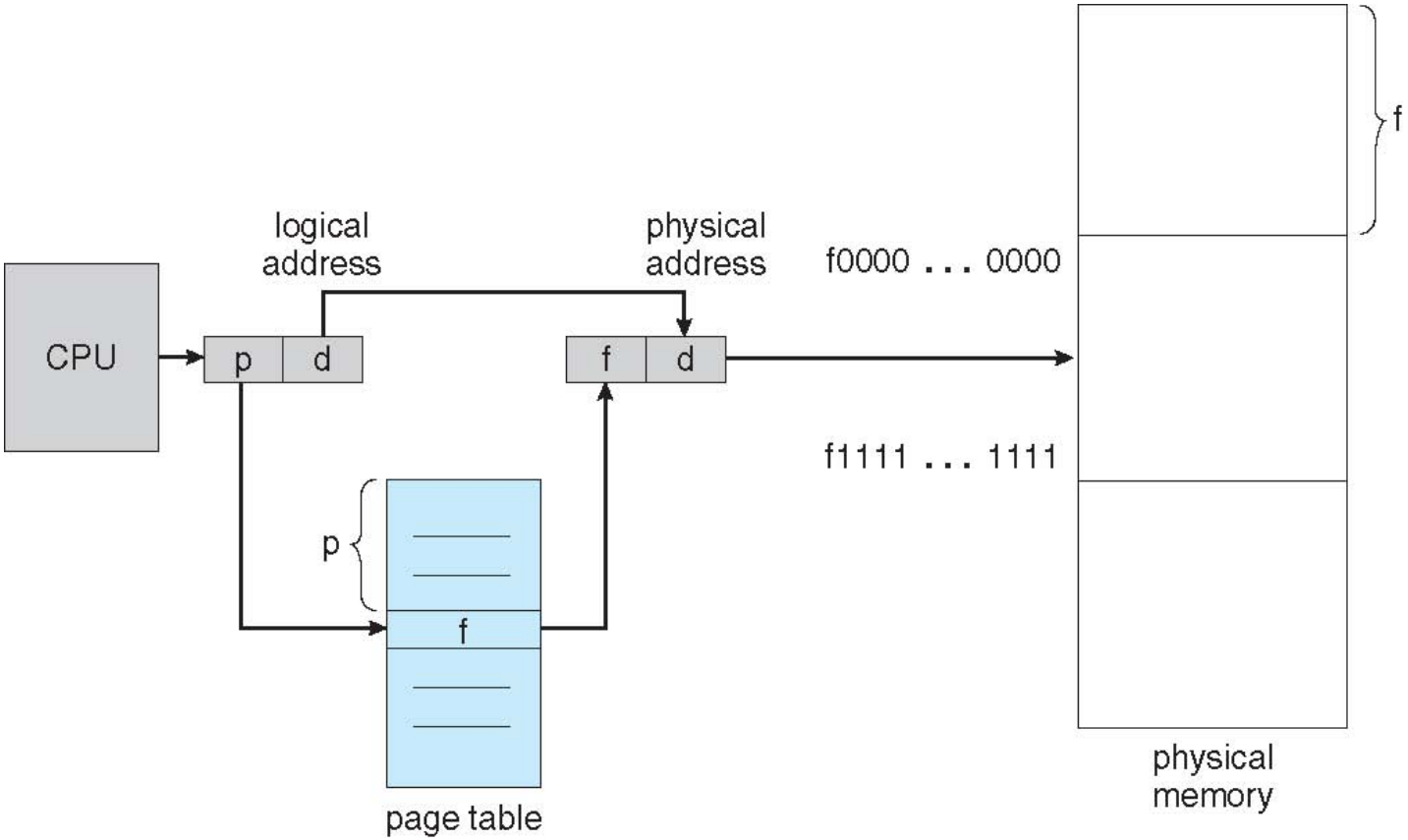
  A (10 instances), B (5 instances), and C (7 instances)

Snapshot at time T_0:

	Max	Allocation	Need	Available
	A B C	*A B C*	*A B C*	*A B C*
P_0	7 5 3	0 1 0	7 4 3	3 3 2
P_1	3 2 2	2 0 0	1 2 2	
P_2	9 0 2	3 0 2	6 0 0	
P_3	2 2 2	2 1 1	0 1 1	
P_4	4 3 3	0 0 2	4 3 1	

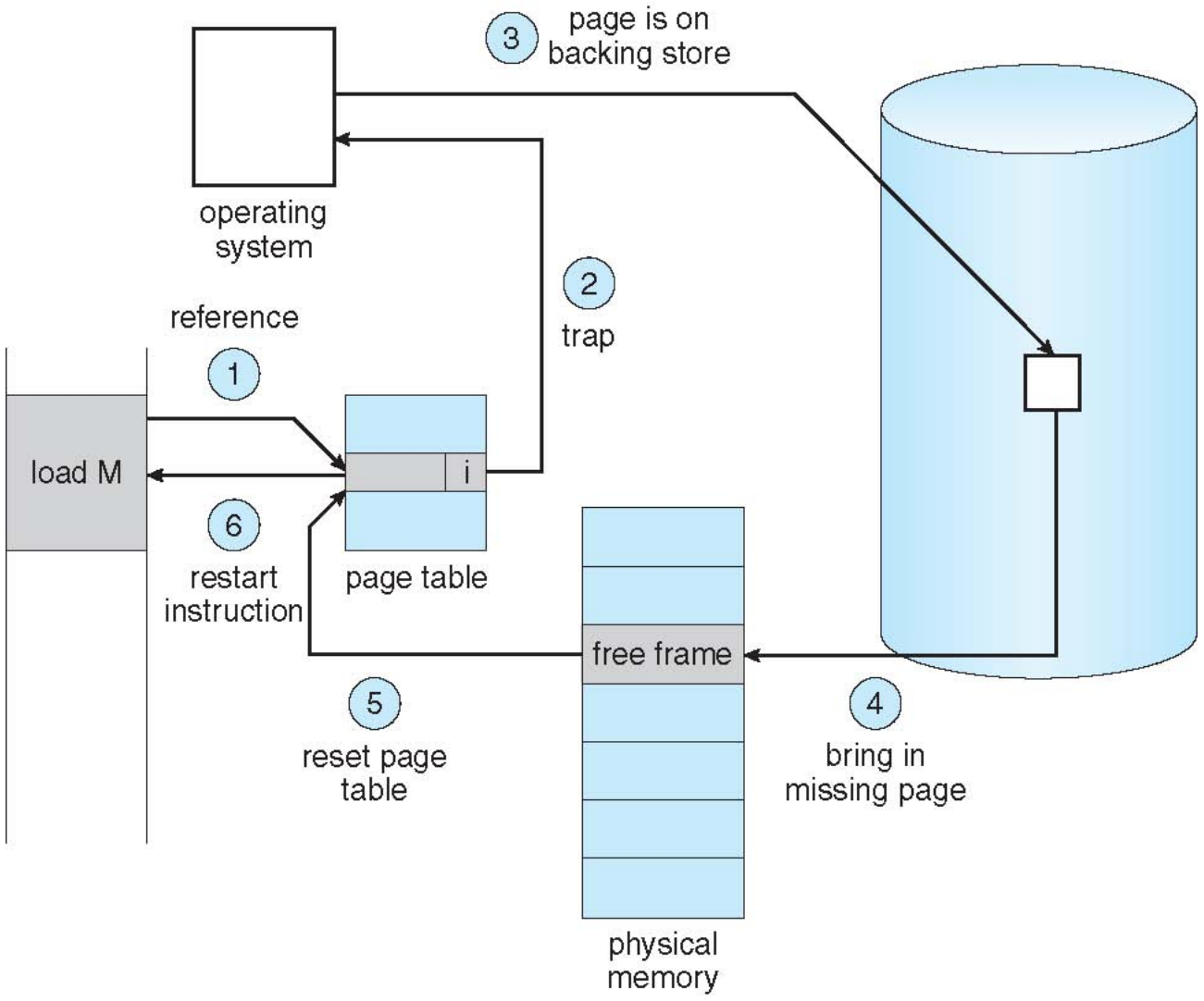
- The system is in a safe state since the sequence < P_1, P_3, P_0, P_2, P_4> satisfies safety criteria

Memory Management

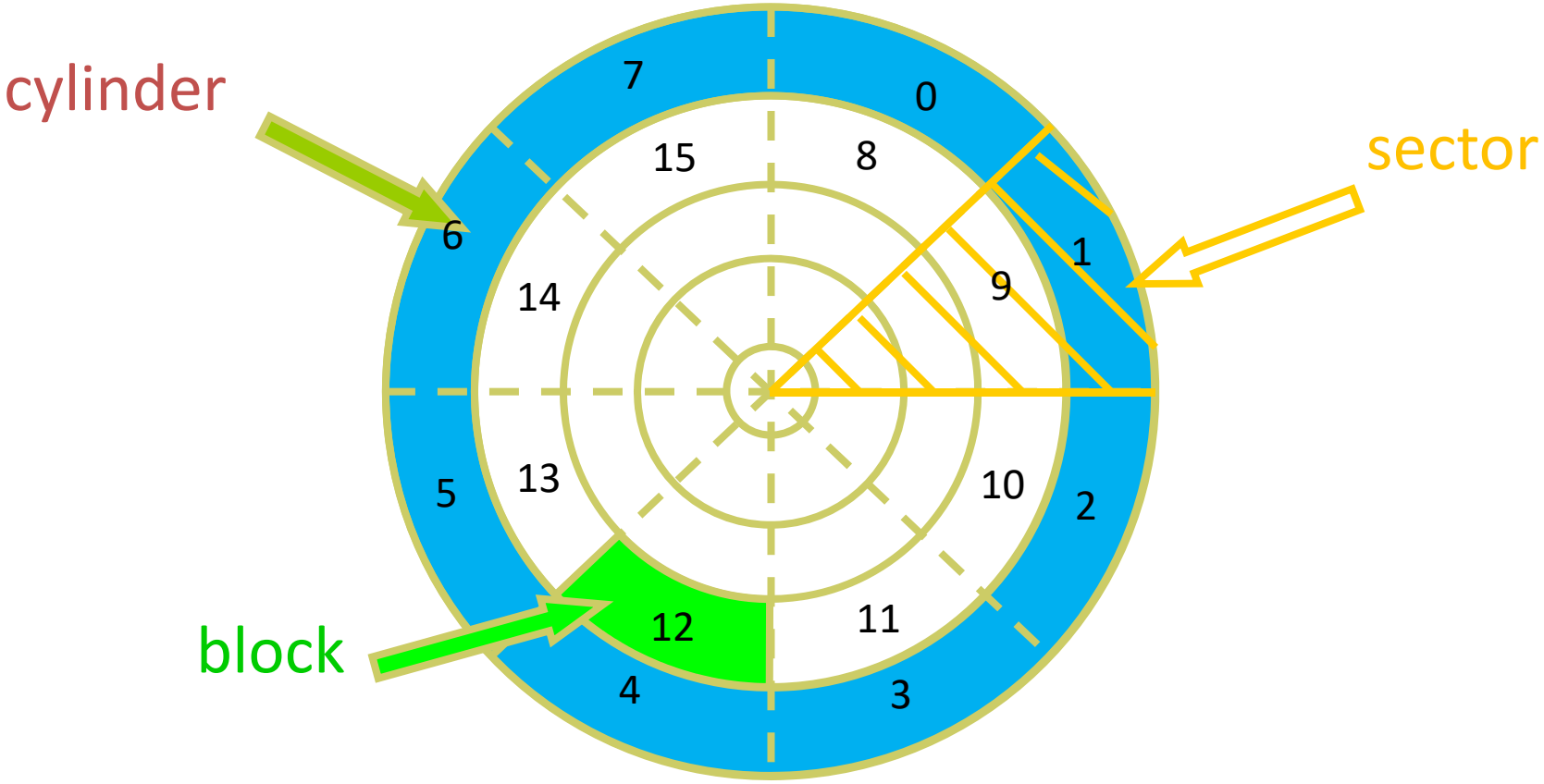


Paging Hardware

Virtual Memory Management

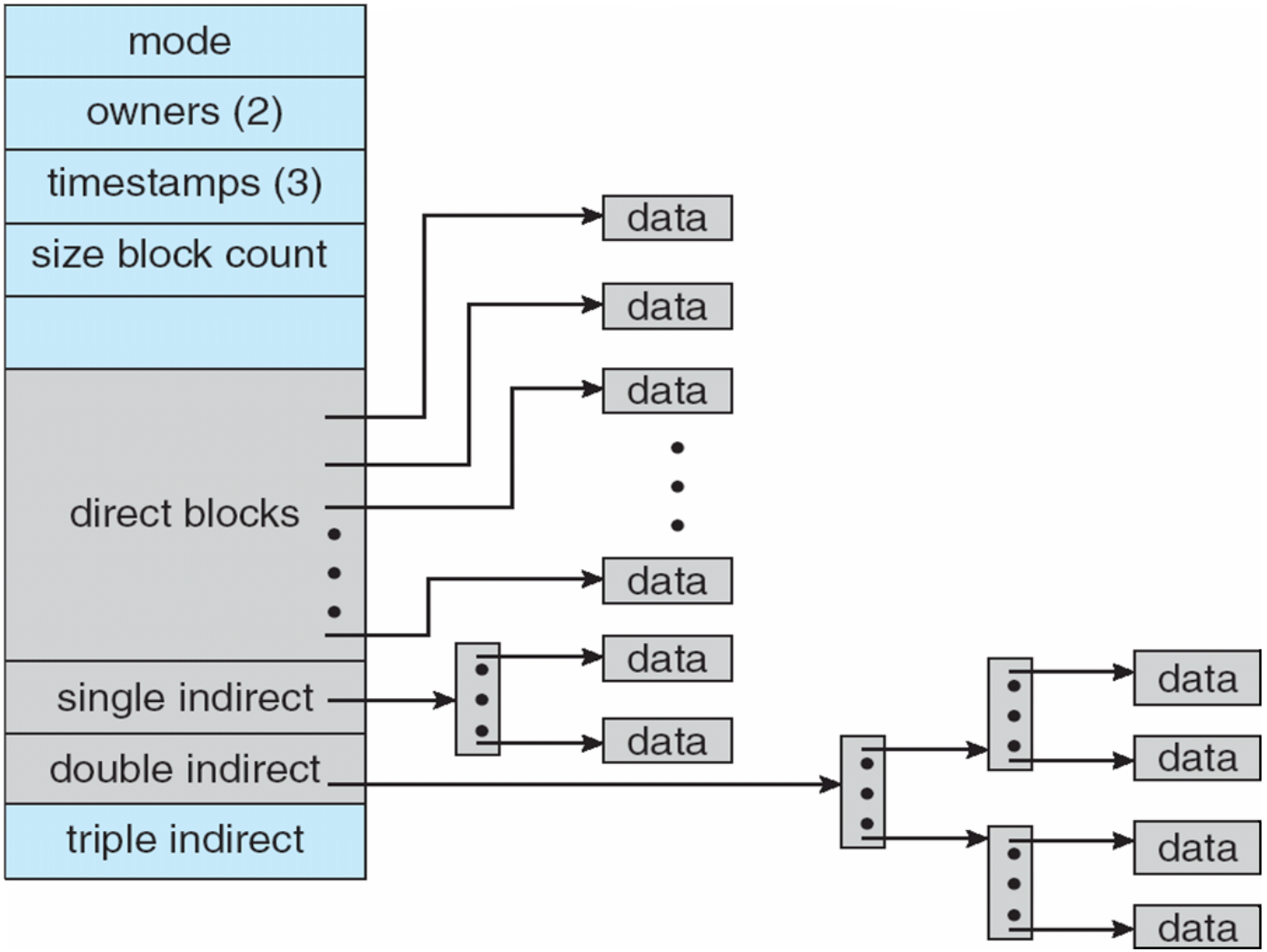


Mass-Storage Systems



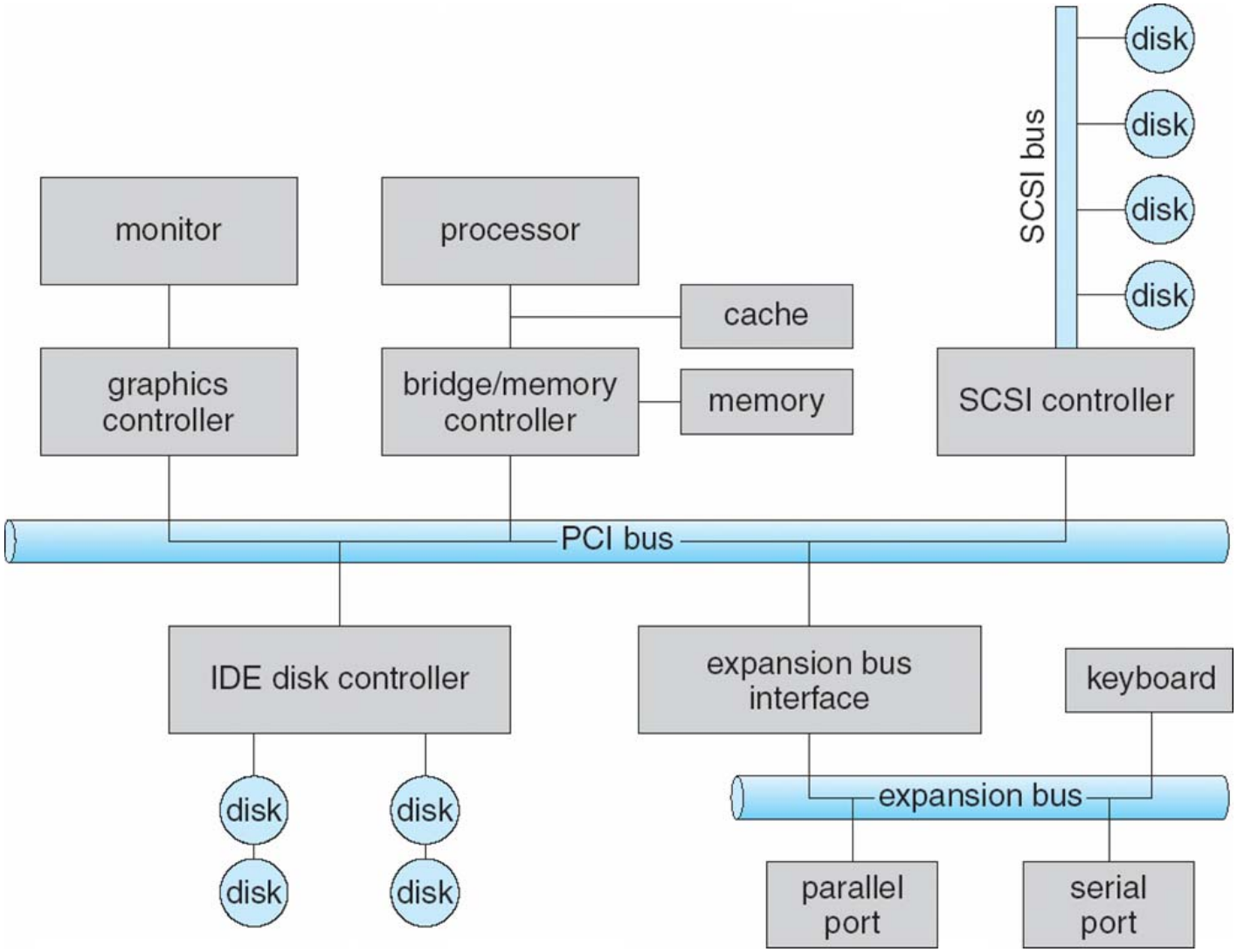
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	.....

File-System

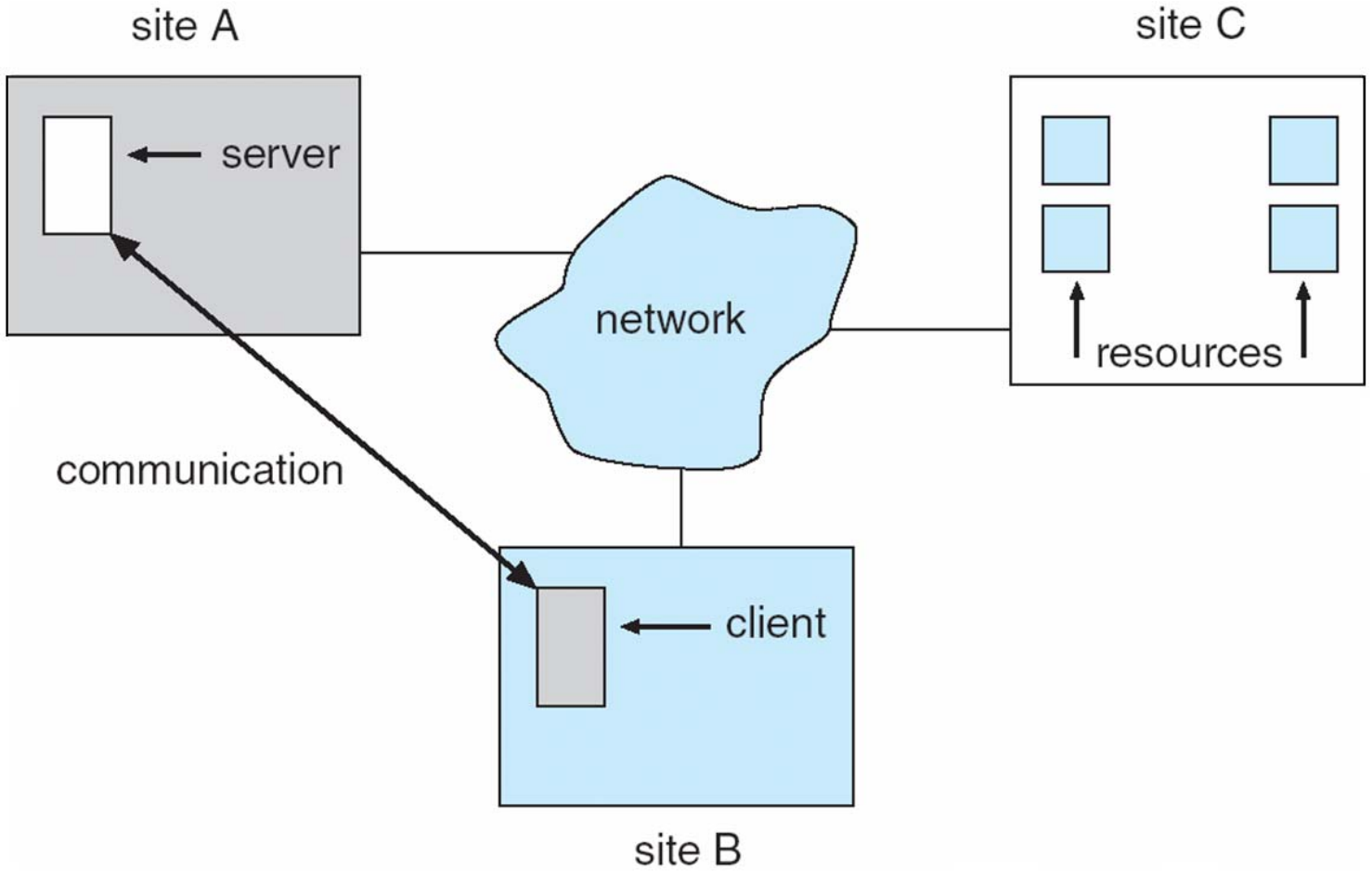


Combined Scheme with **UNIX I-node**

I/O Systems



Distributed System Structure



Homework

- Reading
  - Chapter 1: Introduction