Principles of Operating Systems

CS 446/646

1. Introduction to Operating Systems

René Doursat

Department of Computer Science & Engineering
University of Nevada, Reno

Spring 2006

Principles of Operating Systems

CS 446/646

0. Course Presentation
1. Introduction to Operating Systems
2. Processes
3. Memory Management
4. CPU Scheduling
5. Input/Output
6. File System
7. Case Studies

Principles of Operating Systems

CS 446/646

1. Introduction to Operating Systems

a. Role of an O/S

b. O/S History and Features

c. Types of O/S

d. Major O/S Components

e. System Calls

f. O/S Software Architecture

g. Examples of O/S

Principles of Operating Systems

CS 446/646

1. Introduction to Operating Systems

a. Role of an O/S

- ✓ One layer in a computer system architecture
- ✓ Top-down view: convenient user interface
- ✓ Bottom-up view: efficient resource manager
- ✓ Software view: a program that can evolve

b. O/S History and Features

c. Types of O/S

d. Major O/S Components

e. System Calls

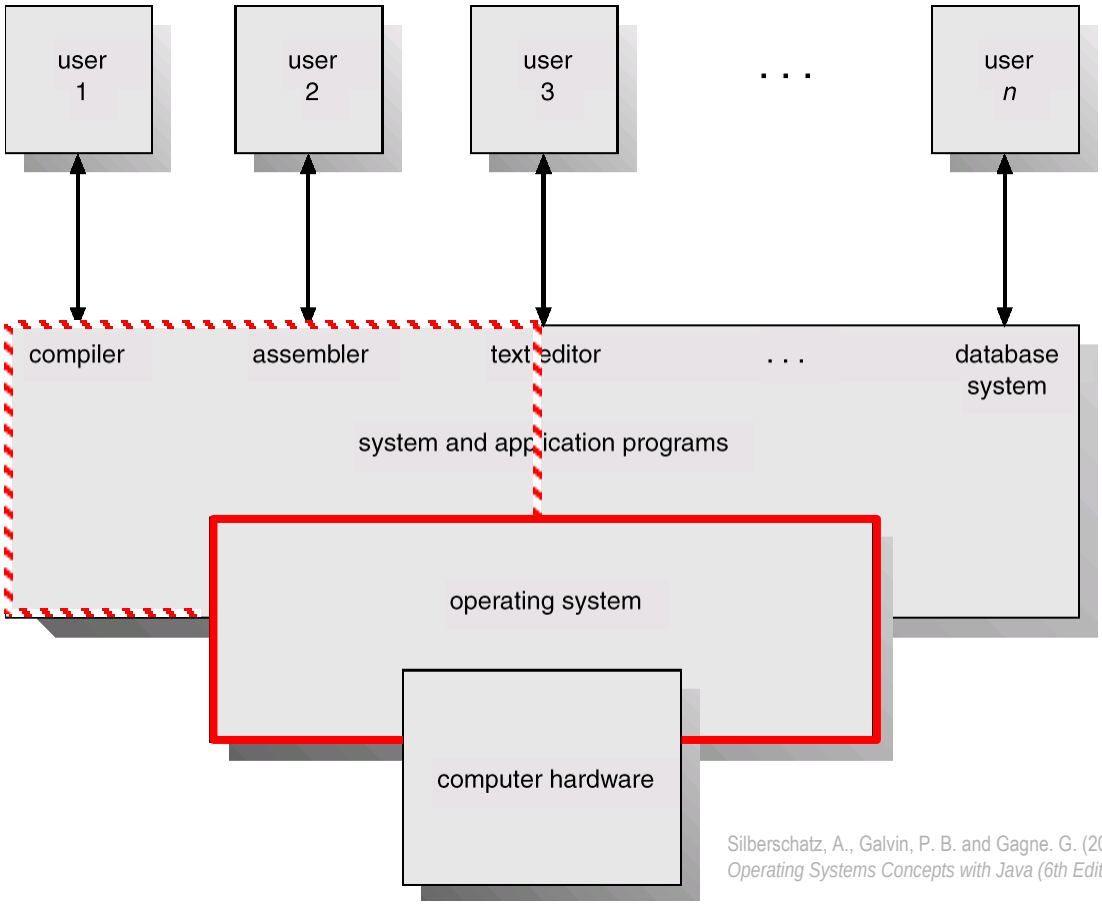
f. O/S Software Architecture

g. Examples of O/S

1.a Role of an Operating System

Software layer in a computer system architecture

- The Silberschatz "pyramid" view



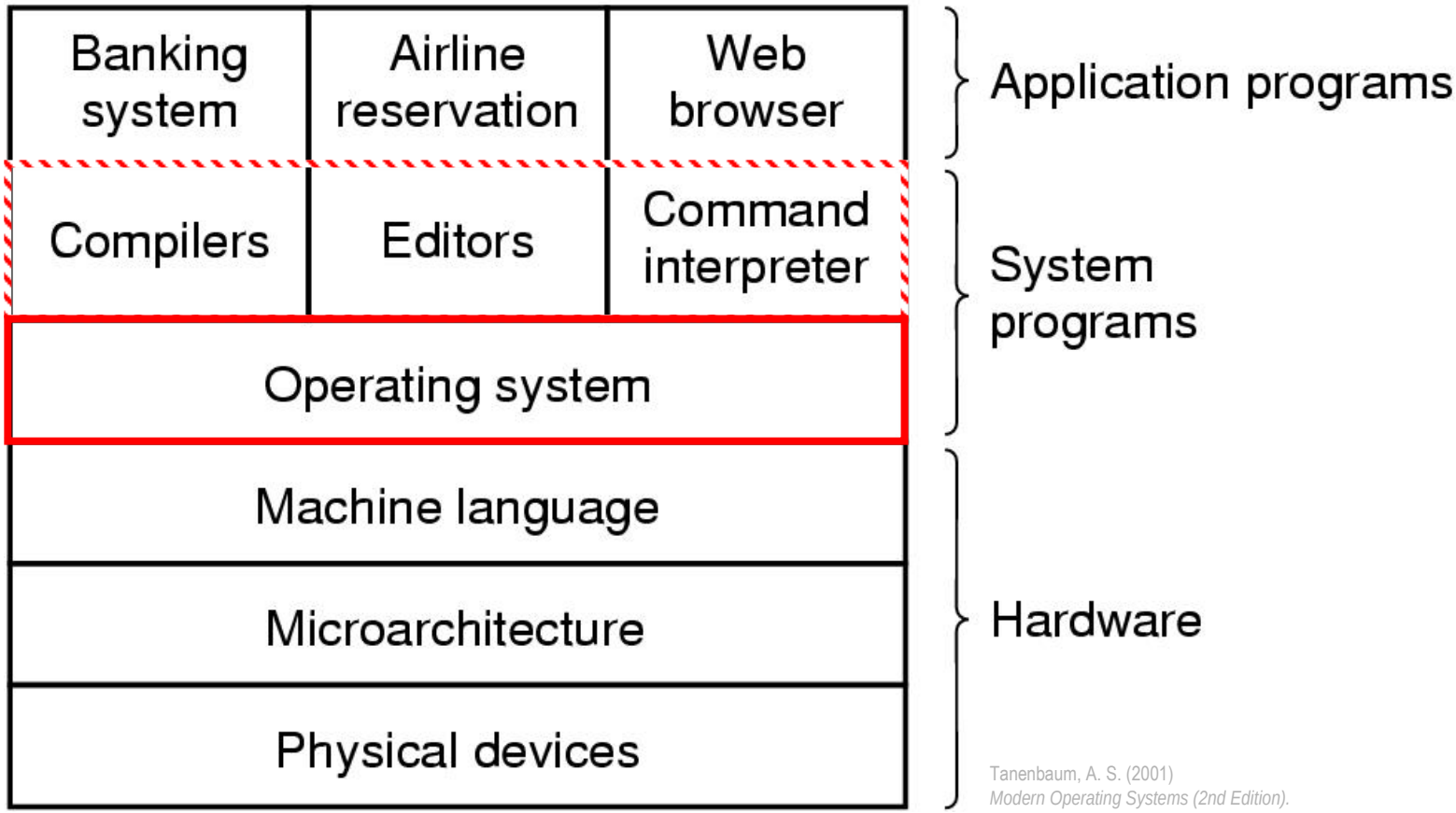
Silberschatz, A., Galvin, P. B. and Gagne. G. (2003) *Operating Systems Concepts with Java (6th Edition).*

Abstract view of the components of a computer system

1.a Role of an Operating System

Software layer in a computer system architecture

- The Tanenbaum "layered" view



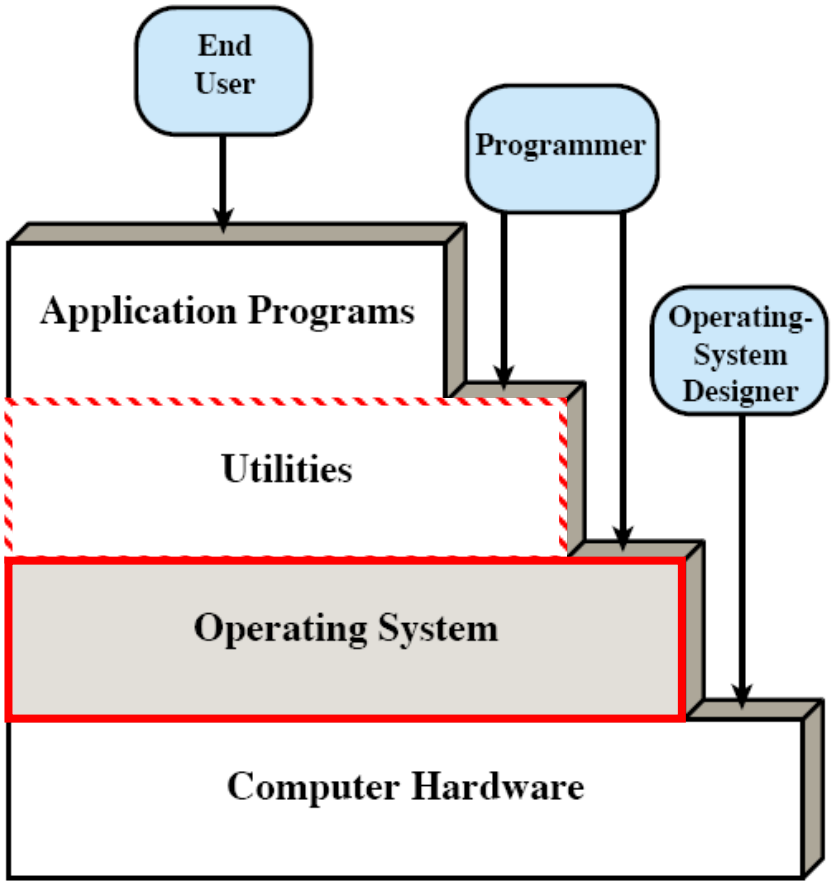
Tanenbaum, A. S. (2001)
Modern Operating Systems (2nd Edition).

A computer system consists of hardware, system programs and application programs

1.a Role of an Operating System

Software layer in a computer system architecture

- The Stallings "stairs" view



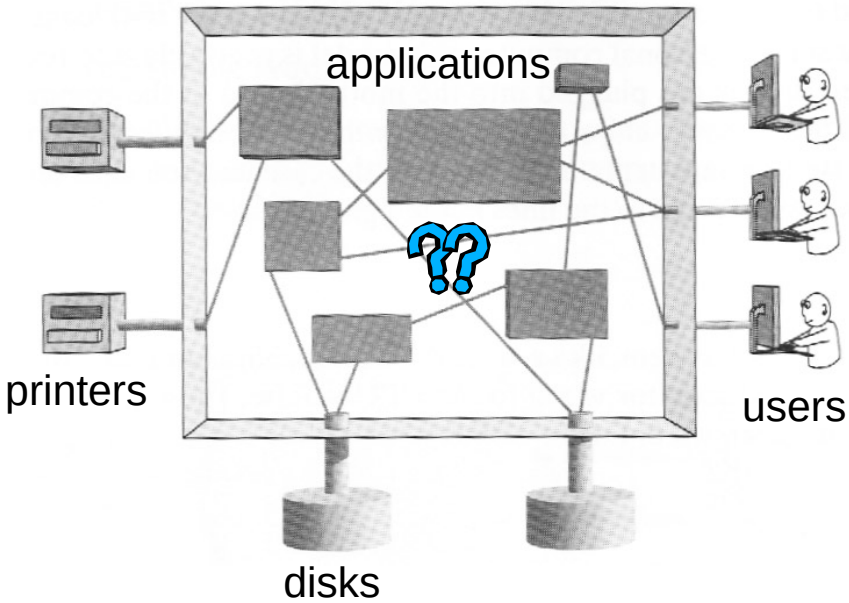
Stallings, W. (2004) *Operating Systems: Internals and Design Principles (5th Edition).*

Layers and views of a computer system

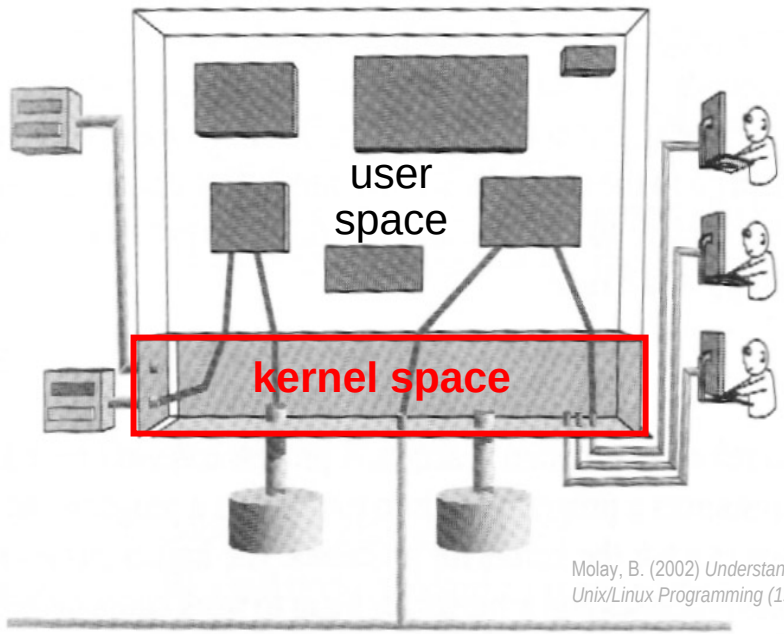
1.a Role of an Operating System

Software layer in a computer system architecture

- The Molay "aquarium" view
  - ✓ everything must transit through the O/S or "kernel"



How are they all connected?



Molay, B. (2002) *Understanding Unix/Linux Programming (1st Edition).*

The kernel manages all connections

1.a Role of an Operating System

Software layer in a computer system architecture

- A computer system consists of (bottom-up):
  1. hardware
  2. firmware (BIOS)
  3. operating system
  4. system programs
  5. application programs
  6. users

1.a Role of an Operating System

Software layer in a computer system architecture

1. Hardware
   - ✓ provides basic computing resources
   - ✓ CPU, memory, disk, other I/O devices
2. Firmware (BIOS)
   - ✓ software permanently stored on chip (but upgradable)
   - ✓ loads the operating system during boot
3. Operating system
   - ✓ controls and coordinates the use of the hardware among the various application programs for the various users

1.a Role of an Operating System

Software layer in a computer system architecture

4. System programs
   - ✓ basic development tools (shells, compilers, editors, etc.)
   - ✓ not strictly part of the core of the operating system
5. Application programs
   - ✓ define the logic in which the system resources are used to solve the computing problems of the users
   - ✓ compilers, database systems, video games, business programs, etc.
6. Users
   - ✓ people, other computers, machines, etc.

1.a Role of an Operating System

Software layer in a computer system architecture

Key notion: An operating system is a program that acts as an **intermediary** between a user of a computer and the computer hardware.

1.a Role of an Operating System

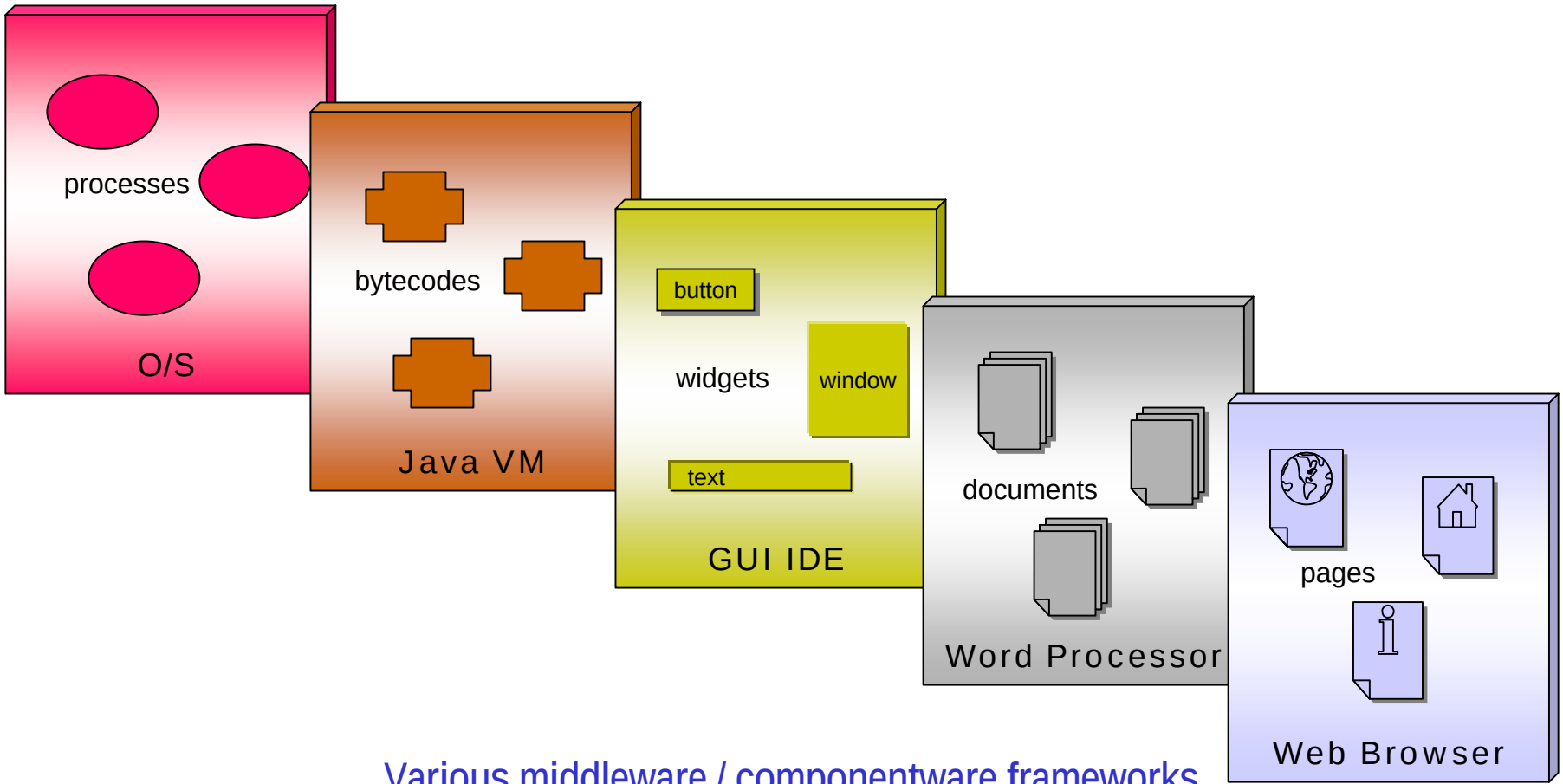
Software layer in a computer system architecture

- Top-down (user) view
  - ✓ the O/S is a convenient application interface
    - the O/S hides the messy details which must be performed
    - the O/S presents user with a virtual machine easier to use
- Bottom-up (hardware) view
  - ✓ the O/S performs efficient resource usage and management
    - time multiplexing: each program gets to use a resource
    - space multiplexing: each program gets part of a resource
- Software layer view
  - ✓ the O/S is an evolvable and scalable software
    - the O/S permits effective development and introduction of new system functions without interfering with service

1.a Role of an Operating System

Top-down view: convenient user interface

- The middleware / componentware paradigm



Various middleware / componentware frameworks

1.a Role of an Operating System

Top-down view: convenient user interface

- The middleware / componentware paradigm
  - ✓ in this paradigm, the O/S is a “container” or platform
  - ✓ the user applications are the “components” that live in the container
  - ✓ the O/S takes care of the low-level “plumbing” environment around the applications, so that the applications can focus on their internal logic and purpose
  - ✓ the O/S provides **services** to the applications, not the least execute them
  - ✓ same paradigm in other middleware frameworks: Java virtual machine, GUI environment, Web browser, etc.

1.a Role of an Operating System

Top-down view: convenient user interface

- Services provided by the O/S to user applications
  - ✓ program execution
  - ✓ access to I/O devices
  - ✓ controlled access to files
  - ✓ communications
  - ✓ error detection and response

1.a Role of an Operating System

Top-down view: convenient user interface

- Program execution
  - ✓ the O/S loads programs and data into memory, initializes I/O devices and files, schedules the execution of programs
- Access to I/O devices
  - ✓ the O/S hides I/O device details from applications (direct I/O access is forbidden) and offers a simplified I/O interface
- Controlled access to files
  - ✓ the O/S organizes data into files, controls access to the files (create, delete, read, write) and preserves their integrity

1.a Role of an Operating System

Top-down view: convenient user interface

- Communications
  - ✓ the O/S allows exchange of information between processes, which are possibly executing on different computers
- Error detection and response
  - ✓ the O/S properly handles hardware failures and software errors with the least impact to running applications (ending, retrying or reporting)

1.a Role of an Operating System

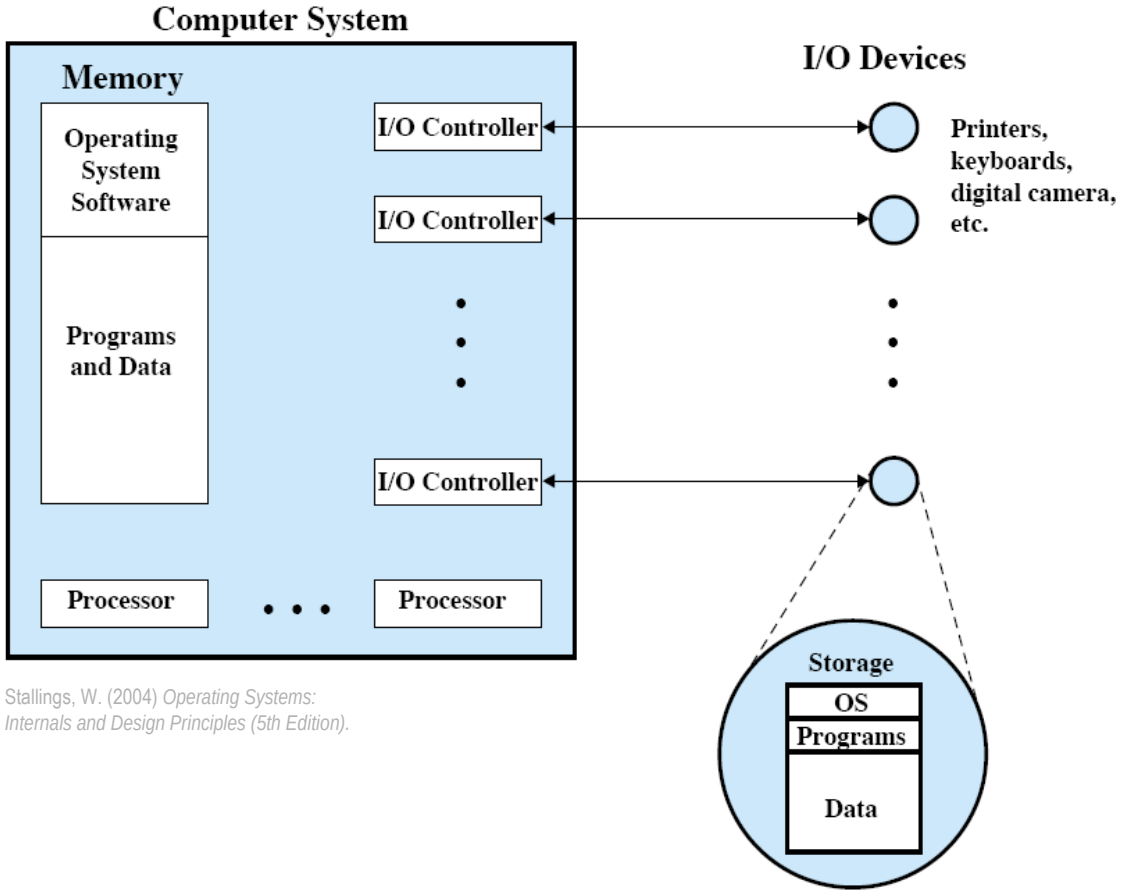
Top-down view: convenient user interface

- Summary: viewed from the user side, the operating system is like a "mandatory middleman":
  - ✓ on the one hand it forbids you to directly access hardware resources, you must talk to it first
  - ✓ on the other hand it provides you with excellent service and simplifies your life, so you wouldn't want it any differently anyway

1.a Role of an Operating System

Bottom-up view: efficient resource manager

- Managing the hardware "plumbing"



Stallings, W. (2004) *Operating Systems: Internals and Design Principles (5th Edition).*

The operating system as resource manager

1.a Role of an Operating System

Bottom-up view: efficient resource manager

- Managing the hardware "plumbing"
  - ✓ in order to offer services to the user programs, the O/S "turns around" and manages the inner workings of the hardware
  - ✓ it needs to ensure efficient operation of the computer system
  - ✓ it controls the movement, storage and processing of data
  - ✓ the O/S is a peculiar form of control, though:
    - it is not a distinct and separate part of what it controls
    - it is just a program like any other program executed by the processor
    - therefore, it must frequently relinquish control and depend on the processor to allow it to regain control

1.a Role of an Operating System

Bottom-up view: efficient resource manager

- Duties performed by the O/S as a resource manager
  - ✓ resource allocator
  - ✓ operation control program
  - ✓ system access
  - ✓ accounting and statistics

1.a Role of an Operating System

Bottom-up view: efficient resource manager

- Resource allocator
  - ✓ the O/S is a program that allocates resources to multiple users and multiple jobs running at the same time (“pecking order”)
- Operation control program
  - ✓ the O/S is a program that controls the execution of user programs (“supervisor”) and operations of I/O devices (“driver”)
- System access
  - ✓ the O/S ensures that all access to resources is protected, including authorization, conflict resolution, etc. (“guardian”)
- Accounting and usage statistics
  - ✓ the O/S keeps performance monitoring data (“auditor”)

1.a Role of an Operating System

Note: There is no clear-cut separation between the services of an operating system as an application interface and its duties as a hardware manager.

The distinction is rather between **what** an O/S has to offer and **how** it does it.

For example, resolving concurrency is both a user service and a resource management technique.

1.a Role of an Operating System

Software view: a program that can evolve

- The O/S is a program like any other program
  - ✓ functions in the same way as ordinary computer software
  - ✓ is a set of instructions that are executed by the processor
  - ✓ relinquishes control of the processor, then recovers it again: the CPU alternates between O/S and programs
- As a critical program, the O/S must be able to evolve
  - ✓ hardware upgrades plus new types of hardware
  - ✓ new services
  - ✓ fixes
  - → a modular software architecture is the most appropriate (see 1.f)

Principles of Operating Systems

CS 446/646

1. Introduction to Operating Systems

a. Role of an O/S

- ✓ One layer in a computer system architecture
- ✓ Top-down view: convenient user interface
- ✓ Bottom-up view: efficient resource manager
- ✓ Software view: a program that can evolve

b. O/S History and Features

c. Types of O/S

d. Major O/S Components

e. System Calls

f. O/S Software Architecture

g. Examples of O/S