Operating System Overview

3431

Agenda

- Assignment 1 is out
- Today's lecture
  - Continue systems overview
  - OS jargon and computing environment
  - OS overview
    - Key abstractions
    - Structure
    - Operation
  - Textbook reading: 1.4-1.8, 2.3-2.10

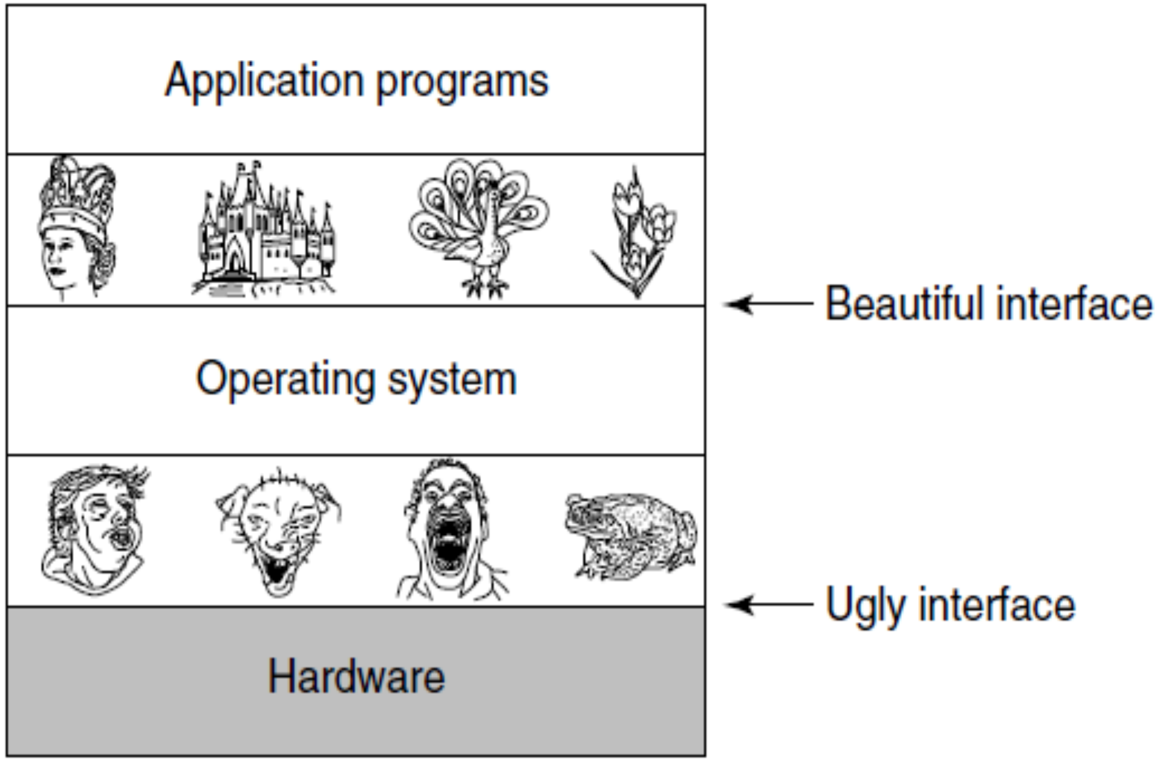
OS Goals

- Provide an environment for applications to run
  - Allow users to use applications

  → **Convenience**
- Provide the resources needed
  - Allow applications to use computer resources

  → **Efficiency**
- Support interactivity
  - Allow users and application to interact

  → **Ability to evolve**



Question: How to achieve these goals?

OS Roles

- OS is a **resource allocator**
  - Provide abstractions through which computer resources are accessed
  - Manages all resources (Processor, Memory, I/O devices, etc…)
  - 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
  - Protects hardware, software, and users from each other

The Three Key Abstractions

- **Process** → CPU
  - Instance of a program
  - Associated with a space
  - May have 0 or more files open for access

- **Memory (Address) Space** →RAM
  - The range of memory locations that a process can access
  - Each location is uniquely identified by an address
  - Each location may be read from, written to, or executed

- **File** → Secondary storage, Network, and Peripheral Devices
  - A stream of data 1001001001101010110110101010110101001
  - Identified by a name in the OS’s file system or a file descriptor

Process

- A process can be defined as:
  - A program in execution.
  - It is a unit of work within the system.
  - Program is a passive entity, process is an active entity.
- Process needs resources to accomplish its task
  - CPU, memory, I/O, files
  - Initialization data
- Process termination requires reclaim of any reusable resources

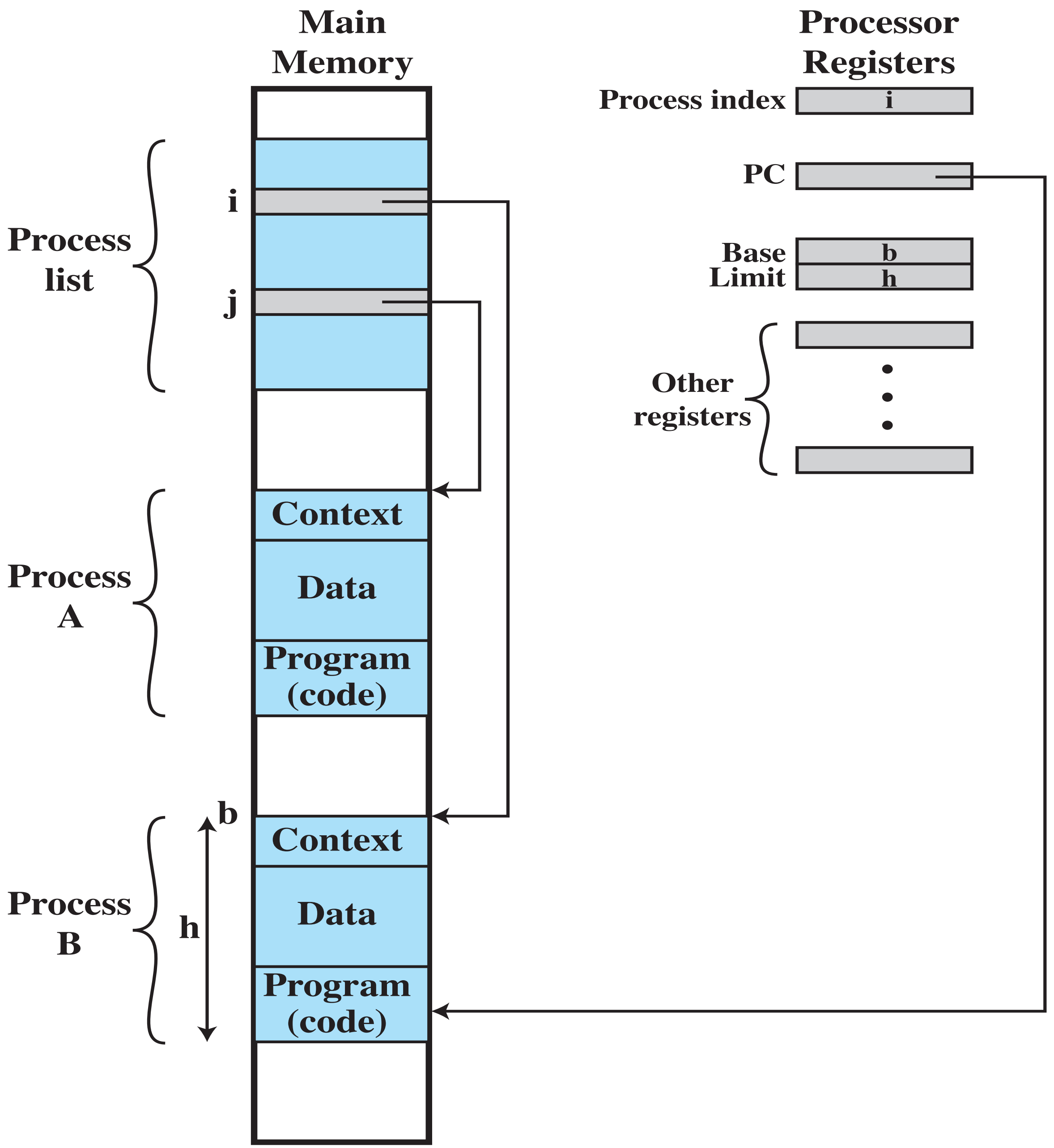
Components of a Process

- Three components
  - An **executable** program
  - The associated **data** needed by the program (variables, work space, buffers, etc.)
  - The process **state**
    - Execution context of the program

- The execution context
  - Is the internal data by which the OS is able to supervise and control the process
  - Includes the contents of the various process registers
  - Includes information such as the priority of the process and whether the process is waiting for the completion of a particular I/O event

Process Management

- The entire state of the process at any instant is contained in its context
- New features can be designed and incorporated into the OS by expanding the context to include any new information needed to support the feature



Process Management

- Single-threaded process has one **program counter** specifying location of next instruction to execute
  - Process executes instructions sequentially, one at a time, until completion
- Multi-threaded process has one program counter per thread
- Typically system has many processes, some user, some operating system running concurrently on one or more CPUs
  - Concurrency by multiplexing the CPUs among the processes / threads

Process Management Activities

The operating system is responsible for the following activities in connection with process management:

- Creating and deleting both user and system processes
- Suspending and resuming processes
- Providing mechanisms for process synchronization
- Providing mechanisms for process communication
- Providing mechanisms for deadlock handling

Memory Management

- Principle storage management responsibilities of the OS
  - Process Isolation
  - Automatic allocation and management
  - Protection and access control
  - Long-term storage

Virtual Memory

- A facility that allows programs to address memory from a logical point of view, without regard to the amount of main memory physically available
- Conceived to meet the requirement of having multiple user jobs reside in main memory concurrently

Paging

- Allows processes to be comprised of a number of fixed-size blocks, called pages
- Program references a word by means of a *virtual address,* consisting of a page number and an offset within the page
- Each page of a process may be located anywhere in main memory
- The paging system provides for a dynamic mapping between the virtual address used in the program and a *real address* (or physical address) in main memory

Scheduling and Resource Management

- Key responsibility of the OS:
  - Manage the various resources available to it
  - Schedule their use by the various active processes
- Any resource allocation and scheduling policy must consider three factors: Fairness, Efficiency, and Responsiveness

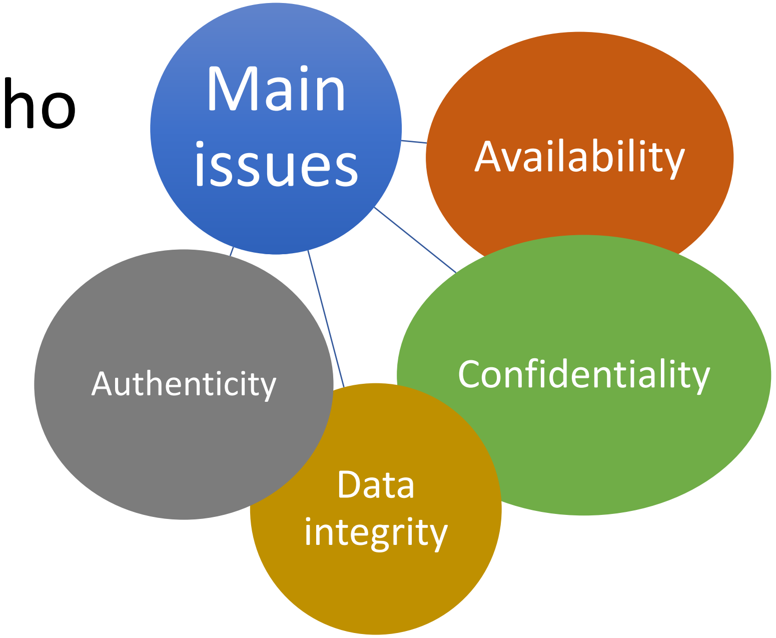
Efficiency
Fairness
Differential responsiveness

Files

- OS provides uniform, logical view of information storage
  - Abstracts physical properties to logical storage unit - **file**
  - Each medium is controlled by device (i.e., disk drive, tape drive)
    - Varying properties include access speed, capacity, data-transfer rate, access method (sequential or random)

File-System Management

- Files usually organized into directories
- Access control on most systems to determine who can access what
  - Information protection and security
- OS activities include
  - Creating and deleting files and directories
  - Primitives to manipulate files and dirs
  - Mapping files onto secondary storage
  - Backup files onto stable (non-volatile) storage media

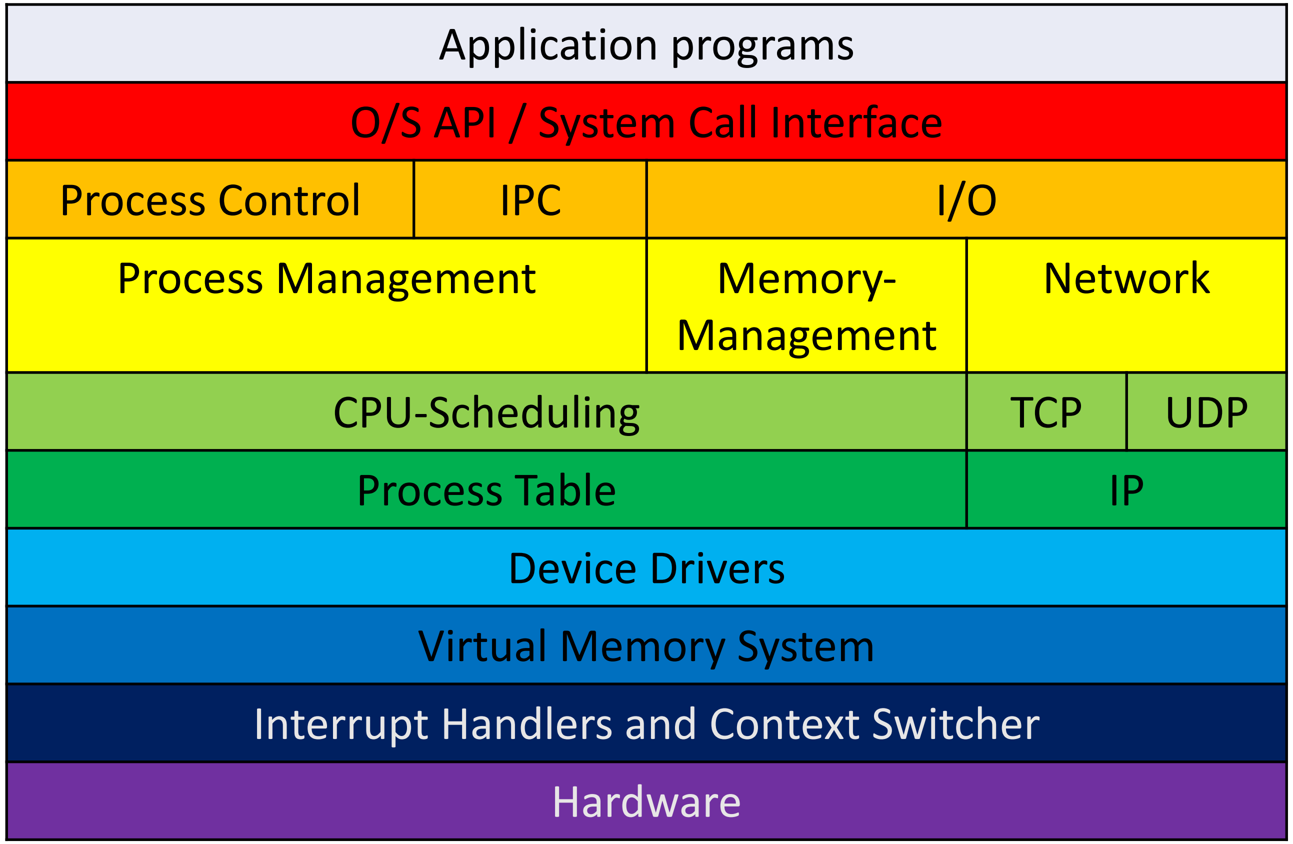


OS Structure

- OS are amongst the most complex pieces of software ever developed
- Design and implementation of OS not “solved”, but some approaches have proven successful
  - Start by defining goals and specifications
  - Affected by choice of hardware and type of system
- User goals and System goals
  - User goals
    - convenient to use, easy to learn, reliable, safe and fast
  - System goals
    - easy to design, implement and maintain, flexible, reliable, error-free, and efficient

The Layered Approach

- With modularity, layers are selected such that each uses functions (operations) and services of only lower-level layers
- Allows separation between
  - **Policy**
    - What needs to be done
  - **Mechanism**
    - How it is done



Application programs

O/S API / System Call Interface

Process Control | IPC | I/O

Process Management | Memory-Management | Network

CPU-Scheduling | TCP | UDP

Process Table | IP

Device Drivers

Virtual Memory System

Interrupt Handlers and Context Switcher

Hardware