Operating Systems basics, security features, and virtualisation

Slides adapted from: CS124 CalTech, CS45 Stanford University,

Mihai Ordean

Summary

- OS components, memory layout/MMU, processes, threads
- Sandboxing as a security feature
- Virtualisation etc.

Operating Systems

Operating systems are used in many different contexts, for fulfilling many different purposes e.g.,:

- Mainframe and server operating systems must maximize utilization of hardware
  - Operating system doesn't require a graphical user interface
  - Rather, must support very efficient handling of I/O, and possibly scheduling of many processes
- Personal computers must be easy to use, and responsive to user input
  - Maximizing hardware utilization is less important – responding to user interaction is top priority!
  - Much more code is devoted to making the computer easy to use
  - Important to provide a simplified, user-friendly user interface

Operating Systems

Mobile device / tablet OSes have several challenging, often conflicting constraints

- Must be responsive and user-friendly, like PC operating systems
- But, must also try to maximize battery life through careful hardware resource management

Smartphones, must support download, installation, execution, and uninstallation of wide range of applications

- But, basic device capabilities (e.g. voice calls, SMS) must also be rock-solid reliable

- Must support intermittent connectivity, particularly when programs are using that connectivity!

Operating Systems

- By far the most common kind of computer now is the embedded computer
  - In your microwave oven, your printer, your WiFi router, your DVD player, controlling your car engine, your point-and-shoot camera
- Embedded OSes tend to have very limited capabilities
  - Systems tend to support a specific, fixed set of tasks
  - Systems aren't designed to run arbitrary programs on them
- Can still include a variety of basic OS capabilities
  - Basic thread-management and scheduling support
  - Basic memory management capabilities
  - Support for software upgrades
  - Support for peripherals like flash cards, USB drives, networking, …

Operating Systems

Real-time operating systems focus on completing tasks by a specific deadline

Most general-purpose operating systems provide soft real-time support, e.g. for media playback

- Not considered a system failure if the OS misses a deadline from time to time (e.g. your media playback just sounds choppy)

Some real-time OSes provide hard real-time guarantees

- If the OS misses a deadline, this is considered a fatal error!

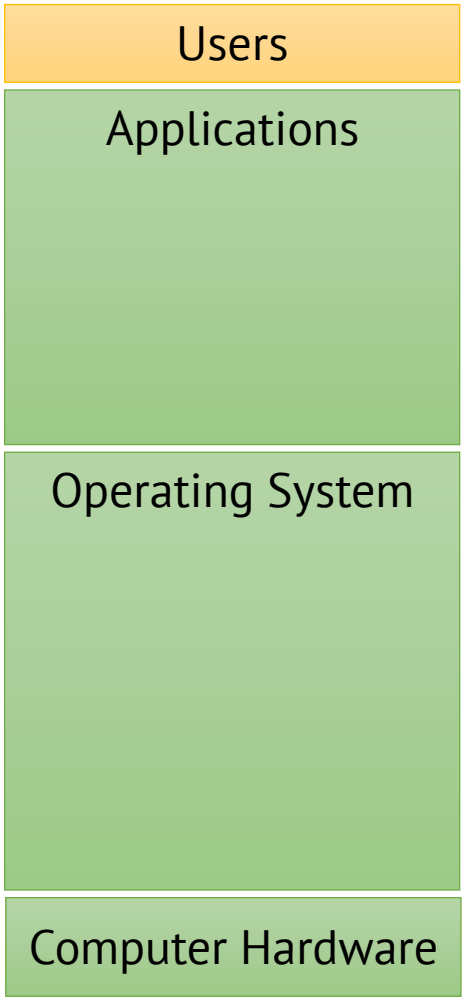
Example: a computer system for running an automobile manufacturing assembly line

- The OS receives inputs from sensors along the assembly line.
- If the OS doesn't satisfy guarantees for processing input data and controlling automated machinery, physical damage will occur
- If OS misses its timing deadlines: Failure! Halt the assembly line!

OS Components

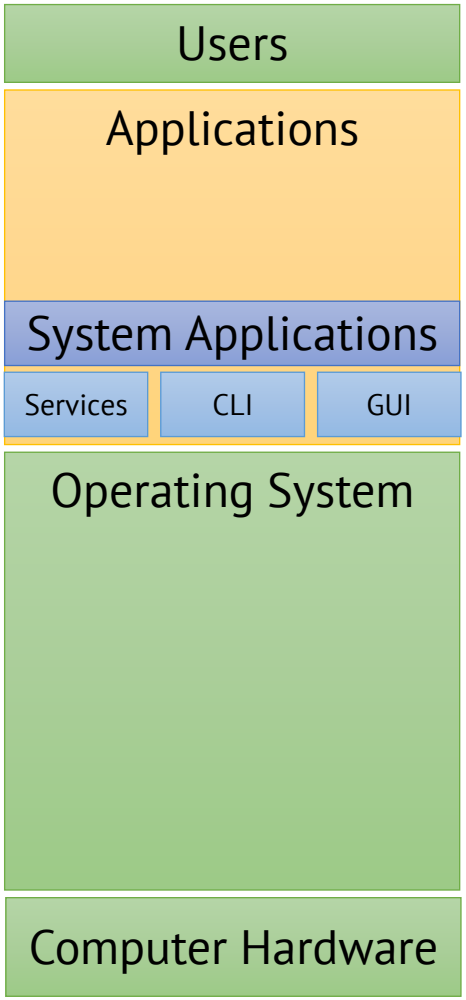
Common components of operating systems:

- Users:
  - Want to solve problems by using computer hardware
  - OS may support only one user at a time, or many concurrent users, depending on system requirements
  - Some systems usually have no users, so they have an extremely minimal UI
    - e.g. automobile engine computers



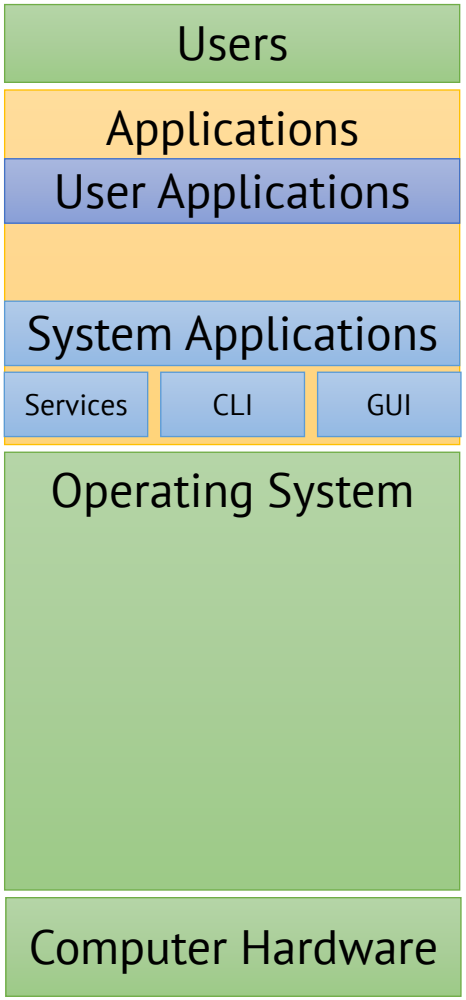
OS Components

- Common components of operating systems:
- Applications allow users to solve problems with the computer's resources
  - Applications rely on the OS to manage those resources
- Some applications are provided by the operating system
  - Services for providing and managing system resources
  - Command shells (e.g., bash, zsh, sh)
  - GUI programs (X-server, system config tools, etc.)



OS Components

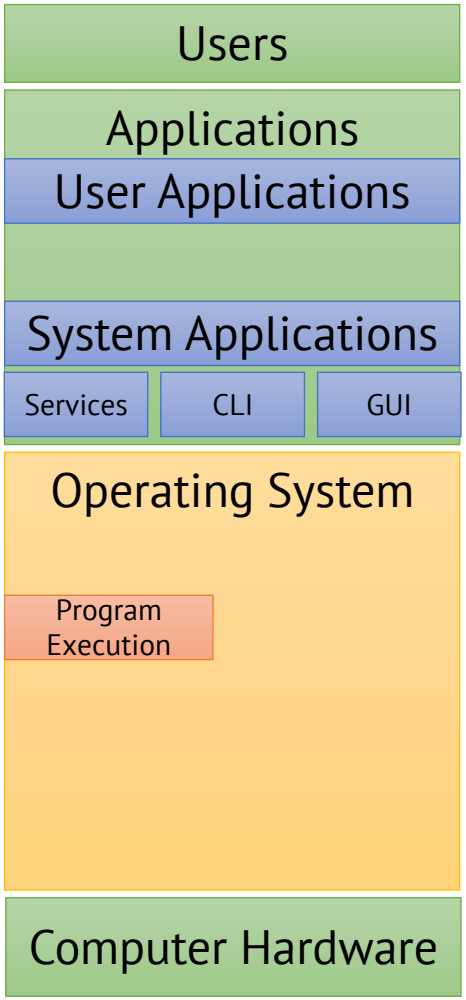
- Common components of operating systems:
- Applications allow users to solve problems with the computer's resources
  - Applications rely on the OS to manage those resources
- User applications are designed to solve specific problems
  - e.g. text editors, compilers, web servers
  - e.g. web browsers, word processors, spreadsheets



OS Components

Common components of operating systems:

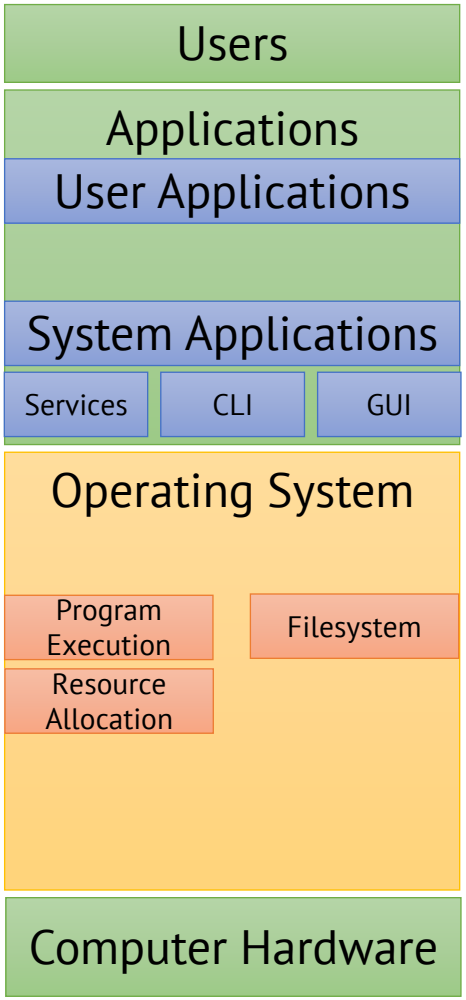
- The OS itself can provide many different facilities
  - Not every OS provides all of these facilities
- Most obvious facility: program execution
  - Load and run programs
  - Perform runtime linking of shared libraries
  - Handle program termination (possibly with errors!)
  - Pass along signals, etc.



OS Components

Common components of operating systems:

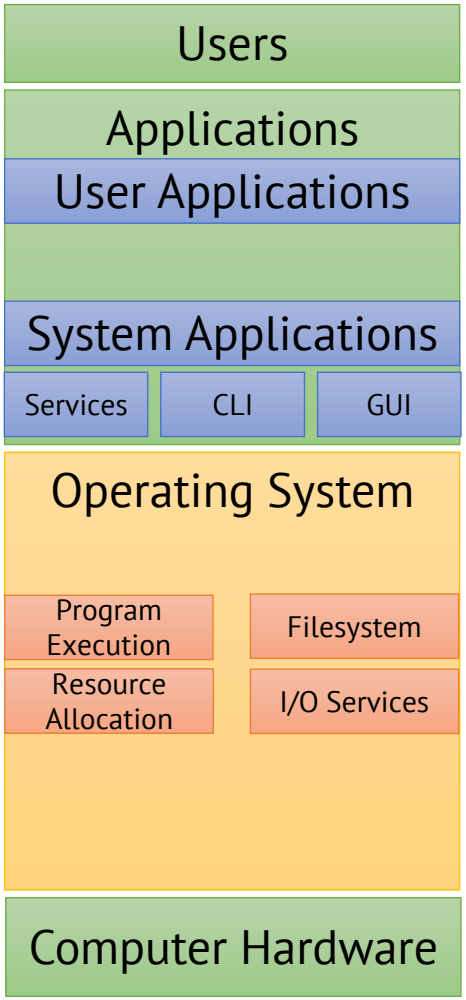
- Another obvious facility: resource allocation
- Resources to manage:
  - Processor(s) – especially if OS supports multitasking
  - Main memory
  - Filesystem/external storage
  - Other devices/peripherals
- Filesystems:
  - OS usually supports several different filesystems
  - May also require periodic maintenance



OS Components

Common components of operating systems:

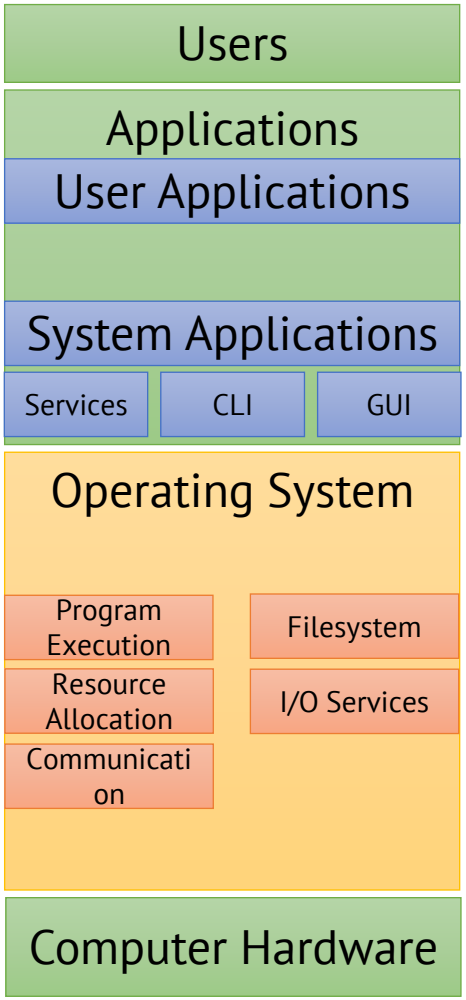
- Disks and other peripheral devices require specific interactions to function properly
  - I/O subsystem provides facilities to control computer hardware devices
  - Often interact via I/O ports
  - **Do not** want apps to do this!
- Usually modularized by using a device-driver abstraction
  - Present a clean abstraction for the rest of the OS to use
  - Encapsulate gory details of talking to hardware



OS Components

Common components of operating systems:

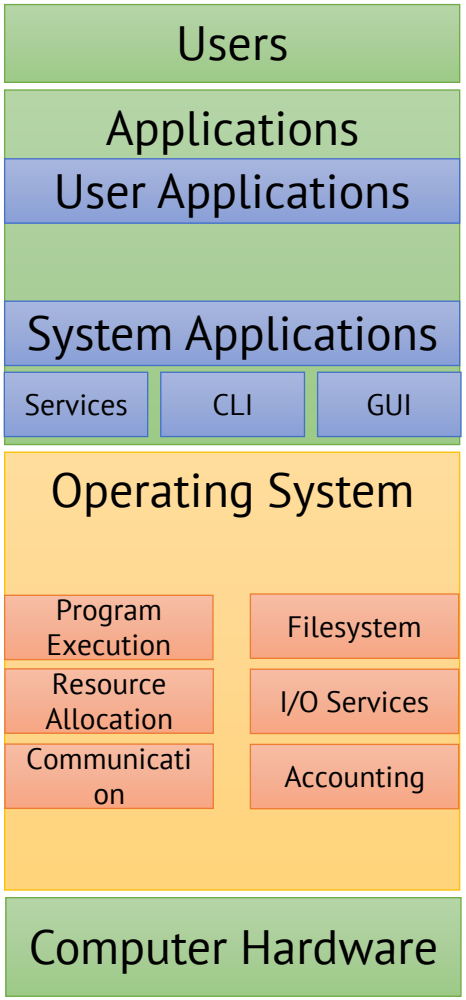
- OSs require internal and external methods of communication:
- Collaborating processes need to share information
  - Called **Inter-Process Communication** (IPC)
  - Many mechanisms: pipes, shared memory, message-passing, localhost, Unix sockets, d-bus, etc.
- Some processes need to communicate with other computer systems
  - Many kinds of networking



OS Components

Common components of operating systems:

- Some OSes record resource usage data:
  - Accounting facility
- Purpose: systems that enforce per-user restrictions based on CPU usage, storage, network, etc.
  - Very common to bill customers for storage and network use
  - Also, with hypervisors, very easy to bill per-VM for CPU use

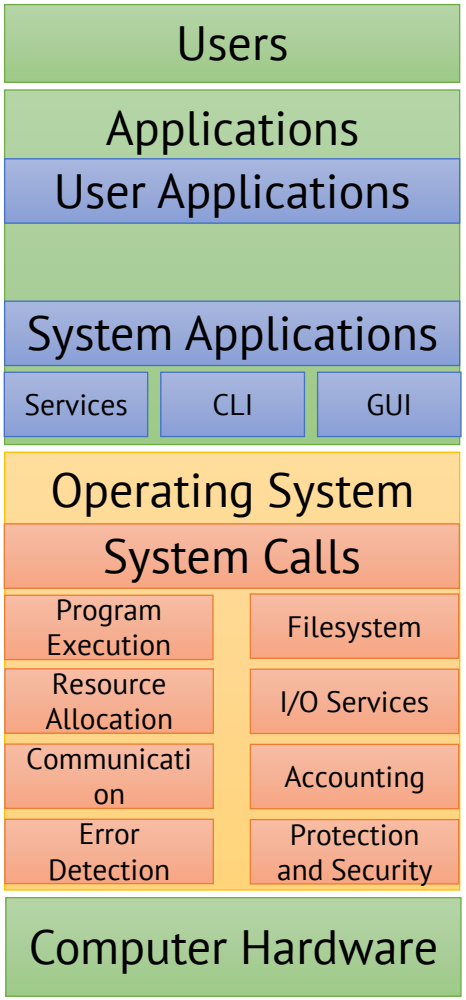


OS Components

Common components of operating systems:

- OSes must handle various errors that occur

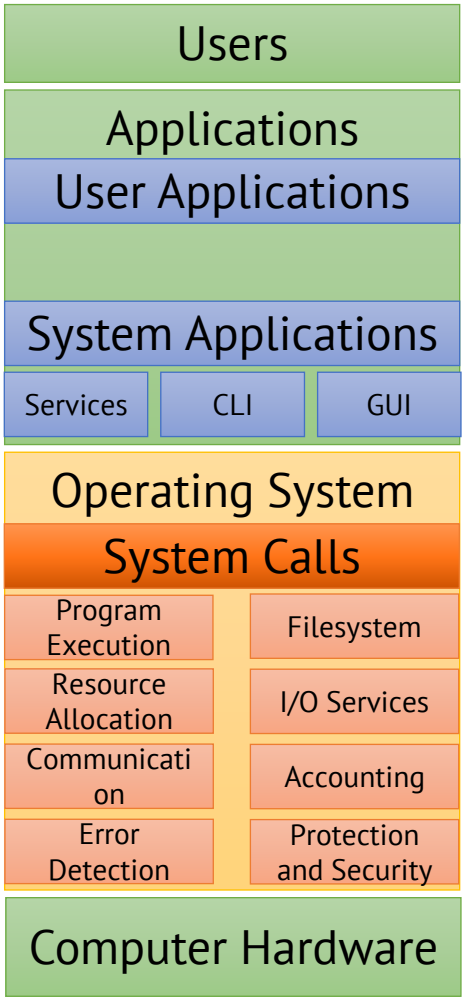
- Varies widely, depending on what the hardware can detect
  - Common errors:
    - Hard disk is full, or broken
    - Filesystem is corrupt
  - Memory errors
    - A program behaves in an invalid way
    - Printer has no paper or ink
  - Less common errors:
    - Processor failure, etc.



OS Components

Common components of operating systems:

- OSes must prevent different kinds of abuses
- OS must be able to protect itself from malicious programs
- Applications are not allowed to directly access operating system code or data
  - Computer hardware must provide this capability!
- All application-interactions with OS are performed via **system calls**



OS Components

Common components of operating systems:

- Operating system must also protect processes from each other
  - A process should not be allowed to access another process' data, unless this is specifically allowed by the process,
    - i.e., the process has built in communication capability
- Again, this requires specific support from the computer hardware