What is an Operating System?

- Examples?

- What does it do?

What is an Operating System?

- **United States v. Microsoft**
  
  - **Dave Farber [DOJ witness]:**
    
    "An Operating System is software that controls the execution of programs on computer systems and may provide low-level services such as resource allocation, scheduling and input-output control in a form which is sufficiently simple and general so that these services are broadly useful to software developers."
  
  - **Ed Felten [DOJ Witness]:**
    
    "An operating system is software that provides services relating to booting the machine and starting programs, interfacing with the hardware, managing and scheduling use of hardware resources, and basic security services."

What is an Operating System?

- **Jim Allchin [Microsoft Witness]**
  
  "A computer operating system, like any other type of software, is a set of instructions that causes a computer to carry out specified functions. Although no clear line of demarcation exists between the functions performed by operating system software and other types of software, operating systems generally serve, at a bare minimum, as the computer's 'central nervous system,' scheduling the execution of tasks by the central processing unit and controlling the flow of information within the computer and between the computer and any peripheral devices that may be attached to it. It is important to bear in mind that all software is a series of instructions to a computer, and terms that have evolved to categorize such software are merely descriptive of general categories of functionality. There is no widely accepted definition of an operating system; it is a concept that has evolved over time based on what is technically possible and what customer[s] have said they want."

What is an Operating System?

  
  "An 'operating system' is a software program that controls the allocation and use of computer resources (such as central processing unit time, main memory space, disk space, and input/output channels). The operating system supports the functions of software programs, called 'applications,' that perform specific user-oriented tasks."

- **So what did this definition mean for this case?**

- **What eventually happened?**
## What is an Operating System?

- Modern operating systems support:
  - Software tools for creating programs
    - libraries, compilers
  - Running multiple programs.
    - multiprogramming
  - Saving/accessing data.
    - files, virtual memory
  - User interaction.
    - window system
  - Interaction with other systems.
    - networking
  - Core applications programs.
    - client-server

- Execution Control.
  - OS keeps track of state of CPU, devices.

- External Devices.
  - Display, keyboard, mouse, disks, CD, network.

- Virtual machines
  - OS presents machine for each person/program
  - OS implements abstract devices

## Multiprogramming

Operating system "kernel" keeps track of several programs.
- CPU does 1 thing at a time.
- Goal: illusion of multiple machines.

**INTERRUPT:**
- Part of hardware of real machines
  - stop
  - save Program Counter (PC) somewhere "special"
  - change PC
- Necessary to manage input-output devices.
  - mouse click, keyboard

OS allows several programs/processes to "share" CPU by keeping table of "current" PC's for programs setting clock to interrupt periodically.

- Context switching
- round-robin or user priorities

**What about multiprocessing?**

## Virtual Memory

**Problem 1:** several programs need to share same memory.
- Direct solution: apportion up the memory.

**Problem 2:** program needs more memory than machine has.
- Direct solution: "overlays."
- program shuffles its own data in and out of memory to disk

"Better" solution: VIRTUAL MEMORY (1960's).
- All programs assume access to all memory.
- Each program actually uses a small portion.

- **How many bits is enough to represent address in virtual memory?**

**Memory hierarchy**
- local: fast, small, expensive
- remote: slow, huge, cheap

**Tradeoff speed for cost.**
*Cache* recently accessed information.

## Key OS developments

- **Window manager**
  - PARC (Alto), Macintosh, Windows, Web browser?
- Problems with a virtual view into the system?

- **Client/server**
  - Client: request service.
  - Server: do the work.
  - Examples?

- **Networking & Distributed systems**
  - Multiple systems working together over the Internet

- **File system**
  - Goal: provide simple abstraction (sequence of bytes) for user programs.