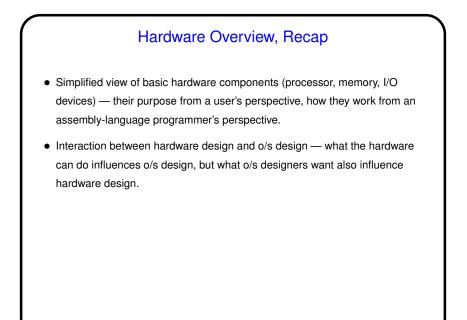


Slide 2

Minute Essay From Last Lecture Question: I once had a learning experience about "how DOS is different from a real o/s". Summary version: A program using pointers (possibly uninitialized) caused the whole machine to lock up and need to be power-cycled. What do you think went wrong? Answer: The program changed memory at the addresses pointed to by the uninitialized pointers — and this memory was being used by the o/s, possibly to store something related to interrupt handling. A "real" o/s wouldn't allow this!



Operating System Services, Again Process management. Memory management. I/O subsystem. File systems. Security. Shell.

Process Management

• "Process" abstraction to represent one of a collection of "things happening at the same time".

A working definition — "program in execution" (program code plus associated variables, sequence of states tracking progress through code and changes in variables).

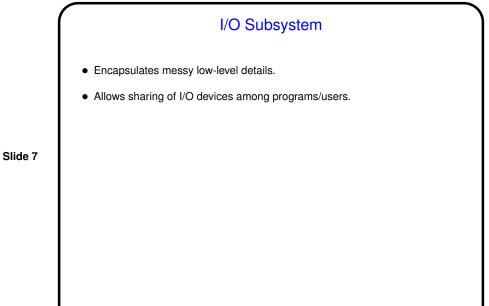
Slide 5

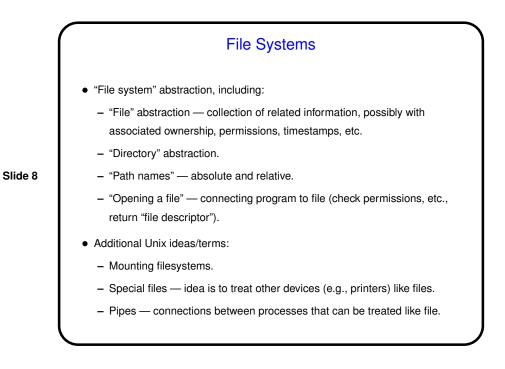
- "Concurrent" execution via interleaving of actions.
 In effect, each process has a "virtual CPU", with the actual CPU repeatedly suspending one process to work on another ("context switch").
- O/s must provide a way to manage this, including ways to create processes, allow/force them to terminate, communicate among them (e.g., to coordinate/synchronize).



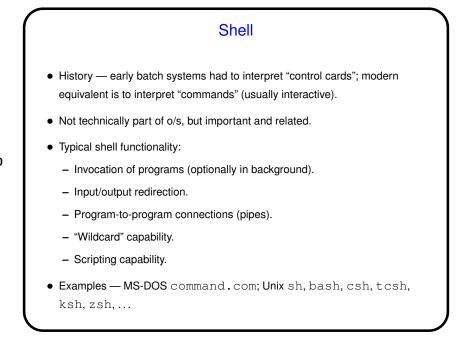
- Managing physical memory:
 - How to divide it up among processes/programs/users each has an "address space" of memory it can access.
 - How to protect each process's memory from other processes (requires h/w support, but managed by o/s).

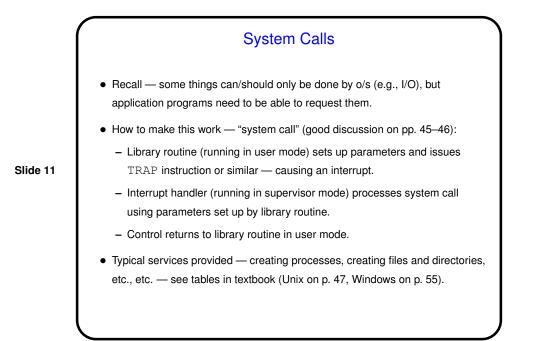
- Managing address spaces (virtual memory):
 - Originally, address space limited by size of physical memory.
 - "Virtual memory" allows bigger address spaces, by shuffling data between disk and physical memory.

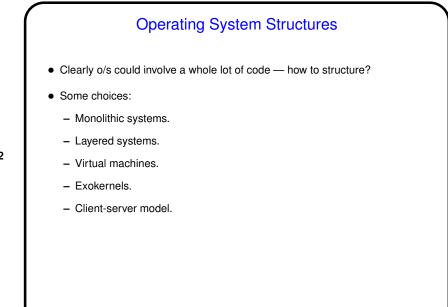


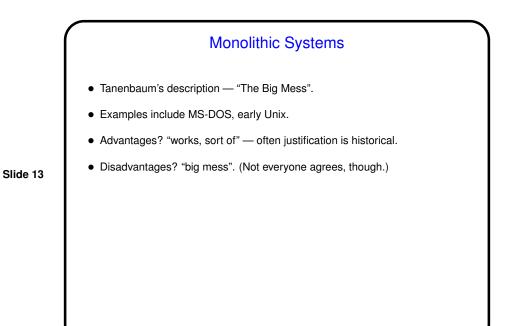




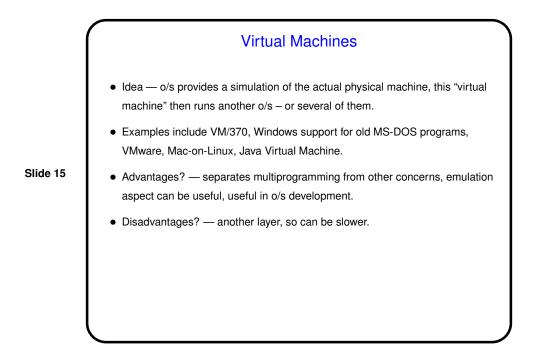


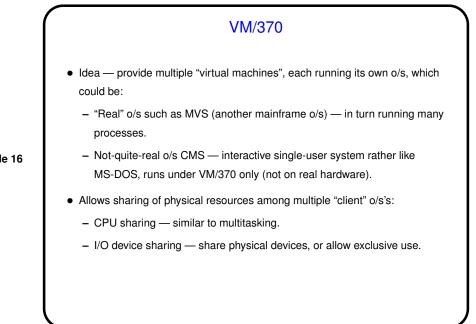


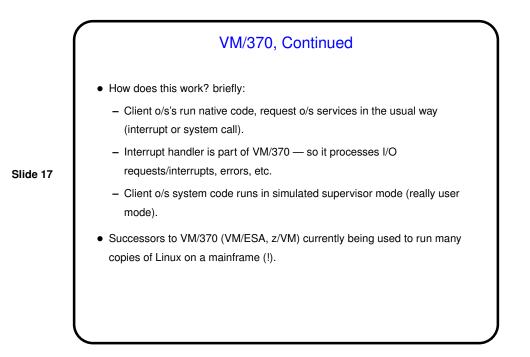




Layered Systems
Idea — use layers of abstraction, just as one structures application programs.
Examples include THE, MULTICS, OS/2, Windows NT (more so in early releases).
Advantages? — nice separation of concerns, modularity.
Disadvantages? — tricky to plan layers, performance can be slow.







 Minute Essay

 • This wraps up lectures on chapter 1; is there anything that was particularly unclear or you want to know more about?

 Slide 18