Slide 1

Administrivia

- Homework 1 was due Monday. Deadline extended to Wednesday.
- (A semi-random comment, possibly of interest: Section of chapter 1 about C is new with this edition of Tanenbaum's book. Apparently not all schools make the same choices we do about what language(s) to use in courses!)

Slide 2

Operating System Structures

- Clearly o/s could involve a whole lot of code (e.g., second edition of textbook says 29M lines of code for Windows 2000). How to structure?
- Choices include:
  - Monolithic systems.
  - Layered systems.
  - Microkernels.
  - Client-server model.
  - Virtual machines.
  - Exokernels.

Slide 3

Monolithic Systems

- Tanenbaum's description in the previous edition of the textbook — "The Big Mess". Maybe an exaggeration, since there can be *some* structure.
- Examples include MS-DOS, early UNIX.
- Arguments for/against?

Slide 4

Monolithic Systems, Continued

- Arguments for this approach — "works, sort of"?
- Arguments against — easier for one malfunctioning component to crash others.

Slide 5

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).
- Arguments for/against?

Slide 6

Layered Systems, Continued

- Arguments for — it's an extra layer of abstraction.
- Arguments against — it's an extra layer of abstraction.
- (Or in other words — nice separation of concerns, modularity, but tricky to plan layers, performance can be slow.)

Slide 7

Microkernel Systems

- Idea — make kernel itself as small as possible, package other services separately, as independent processes.
- Examples include MINIX (written by Tanenbaum).
- Arguments for/against?

Slide 8

Microkernel Systems, Continued

- Arguments for — modularity, reliability.
- Arguments against — tricky to plan layers, performance might be reduced.

Slide 9

Virtual Machines

- Idea — o/s provides a simulation of the actual physical machine, this "virtual machine" then runs another o/s – or several of them.
- Examples include VM/370, Windows support for old MS-DOS programs, VMware, Mac-on-Linux, Java Virtual Machine.
- (Notice how this is an idea that fell out of favor for a while, then came back.)
- Arguments for/against?

Slide 10

Virtual Machines, Continued

- Arguments for — separates multiprogramming from other concerns, emulation aspect can be useful, useful in o/s development.
- Arguments against — another layer, so can be slower. Also, may not be possible for some hardware — e.g., if privileged instructions executed in user mode are simply ignored.

Slide 11

VM/370

- Idea — provide multiple "virtual machines", each running its own o/s, which could be:
  - "Real" o/s such as MVS (another mainframe o/s) — in turn running many processes.
  - Not-quite-real o/s CMS — interactive single-user system rather like MS-DOS, runs under VM/370 only (not on real hardware).
- Allows sharing of physical resources among multiple "client" o/s's:
  - CPU sharing — similar to multitasking.
  - I/O device sharing — share physical devices, or allow exclusive use.

Slide 12

VM/370, Continued

- How does this work? briefly:
  - Client o/s's run native code, request o/s services in the usual way (interrupt or system call).
  - Interrupt handler is part of VM/370 — so it processes I/O requests/interrupts, errors, etc.
  - Client o/s system code runs in simulated supervisor mode (really user mode).
- Successors to VM/370 (VM/ESA, z/VM) currently being used to run many copies of Linux on a mainframe (!).

Slide 13

Words of Wisdom?

- A very smart person I know once said the only interesting part of an o/s course was concurrent algorithms (to be covered soon), and the rest is "just details".

  A student a few years ago said "a lot of this just seems like common sense" (once you understand the basic ideas).

  Both sort of right . . .
- Goal of this course is to learn/retain basic ideas. Details may help with that — and can be interesting in themselves — but should not be the focus.
- (Both things to keep in mind as you continue reading and we continue discussing . . .)

Slide 14

Minute Essay

- Tell me something you learned from reading chapter 1 of the textbook.