Goals for Today

- Operating Systems Structure
- History of Operating Systems
  - Really a history of resource-driven choices
- Operating Systems Organizations
- Abstractions and layering

Note: Some slides and/or pictures in the following are adapted from slides ©2005 Silberschatz, Galvin, and Gagne. Many slides generated from lecture notes by Joseph.

What if we didn’t have an Operating System?

- Source Code → Compiler → Object Code → Hardware
- How do you get object code onto the hardware?
- How do you print out the answer?
- Once upon a time, had to Toggle in program in binary and read out answer from LED’s!

Altair 8080

Simple OS: What if only one application?

- Examples:
  - Very early computers
  - Early PCs
  - Embedded controllers (elevators, cars, etc)
- OS becomes just a library of standard services
  - Standard device drivers
  - Interrupt handlers
  - Math libraries
More thoughts on Simple OS

- What about Cell-phones, Xboxes, etc?
  - Is this organization enough?
- Can OS be encoded in ROM/Flash ROM?
- Does OS have to be software?
  - Can it be Hardware?
  - Custom Chip with predefined behavior
  - Are these even OSs?

More complex OS: Multiple Apps

- Full Coordination and Protection
  - Manage interactions between different users
  - Multiple programs running simultaneously
  - Multiplex and protect Hardware Resources
    » CPU, Memory, I/O devices like disks, printers, etc
- Facilitator
  - Still provides Standard libraries, facilities

- Would this complexity make sense if there were only one application that you cared about?

Example: Protecting Processes from Each Other

- Problem: Run multiple applications in such a way that they are protected from one another
- Goal:
  - Keep User Programs from Crashing OS
  - Keep User Programs from Crashing each other
  - [Keep Parts of OS from crashing other parts?]
- (Some of the required) Mechanisms:
  - Address Translation
  - Dual Mode Operation
- Simple Policy:
  - Programs are not allowed to read/write memory of other Programs or of Operating System
**Address Translation**

- **Address Space**
  - A group of memory addresses usable by something
  - Each program (process) and kernel has potentially different address spaces.
- **Address Translation**
  - Translate from Virtual Addresses (emitted by CPU) into Physical Addresses (of memory)
  - Mapping often performed in Hardware by Memory Management Unit (MMU)

**Dual Mode Operation**

- **Hardware** provides at least two modes:
  - “Kernel” mode (or “supervisor” or “protected”)
  - “User” mode: Normal programs executed
- Some instructions/ops prohibited in user mode:
  - Example: cannot modify page tables in user mode
    - Attempt to modify => Exception generated
- Transitions from user mode to kernel mode:
  - System Calls, Interrupts, Other exceptions

**UNIX System Structure**

- **User Mode**
  - Applications
  - Standard Libs
    - Compilers and interpreters
    - System libraries

- **Kernel Mode**
  - Kernel
    - System-call interface to the kernel
    - Signals
    - Terminal handling
    - Character I/O system
    - Terminal drivers
  - File system
    - Swapping
    - Block I/O system
    - Disk and tape drivers
    - CPU scheduling
  - Memory management
    - Demand paging

- **Hardware**
  - Terminal controllers
  - Device controllers
  - Memory controllers
Moore’s Law Change Drives OS Change

<table>
<thead>
<tr>
<th></th>
<th>1981</th>
<th>2009</th>
<th>Factor</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU MHz, Cycles/inst</td>
<td>10</td>
<td>Quad 3.2G</td>
<td>1,280</td>
</tr>
<tr>
<td></td>
<td>3–10</td>
<td>0.25–0.5</td>
<td>6–40</td>
</tr>
<tr>
<td>DRAM capacity</td>
<td>128KB</td>
<td>66B</td>
<td>49,152</td>
</tr>
<tr>
<td>Disk capacity</td>
<td>10MB</td>
<td>1.5TB</td>
<td>150,000</td>
</tr>
<tr>
<td>Net bandwidth</td>
<td>9600 b/s</td>
<td>1 GB/s</td>
<td>110,000</td>
</tr>
<tr>
<td># addr bits</td>
<td>16</td>
<td>64</td>
<td>4</td>
</tr>
<tr>
<td>#users/machine</td>
<td>10s</td>
<td>≤ 1</td>
<td>≤ 0.1</td>
</tr>
<tr>
<td>Price</td>
<td>$25,000</td>
<td>$3,500</td>
<td>0.2</td>
</tr>
</tbody>
</table>

Typical academic computer 1981 vs 2009

Moore’s law effects

- Nothing like this in any other area of business
- Transportation in over 200 years:
  - 2 orders of magnitude from horseback @10mph to Concorde @1000mph
  - Computers do this every decade (at least until 2002)
- What does this mean for us?
  - Techniques have to vary over time to adapt to changing tradeoffs
- Place a lot more emphasis on principles
  - The key concepts underlying computer systems
  - Less emphasis on facts that are likely to change over the next few years...
- Let’s examine the way changes in $/MIP has radically changed how OS’s work

Administrivia

- Waitlist: Everyone has been let into the class
- Cs162-xx accounts:
  - Make sure you got an account form
  - We have more forms for those of you who didn’t get one
- If you haven’t logged in yet, you need to do so
- Nachos readers:
  - Will include lectures and printouts of all of the code
- Video “Screencast” archives available off lectures page
  - Just click on the title of a lecture for webcast
  - Only works for lectures that I have already given!

Administrivia: Time for Project Signup

- 4–5 members to a group
- All members of a group should be in same discussion section
- If you want to change your discussion section, please send a request to: cs162-staff@lists.berkeley.edu
  - Request are not guaranteed to be accommodated, as we need to balance the section enrollment
  - Send your request by Monday, 1/25, 11:59
- Watch “Group/Section Assignment Link” for final assignments by Tuesday, 1/26
- Next, you’ll pick your group (we’ll tell you how to do next Tuesday)
  - Due Friday 1/28 by 11:59pm

<table>
<thead>
<tr>
<th>Section</th>
<th>Time</th>
<th>Location</th>
<th>TA</th>
</tr>
</thead>
<tbody>
<tr>
<td>101</td>
<td>W 10:00A–11:00A</td>
<td>2 Evans</td>
<td>Matei Zaharia</td>
</tr>
<tr>
<td>102</td>
<td>W 2:00P–3:00P</td>
<td>75 Evans</td>
<td>Andy Konwinski</td>
</tr>
<tr>
<td>103</td>
<td>W 3:00P–4:00P</td>
<td>2 Evans</td>
<td>Ben Hindman</td>
</tr>
</tbody>
</table>
Academic Dishonesty Policy

- Copying all or part of another person's work, or using reference material not specifically allowed, are forms of cheating and will not be tolerated. A student involved in an incident of cheating will be notified by the instructor and the following policy will apply:
  
  http://www.eecs.berkeley.edu/Policies/acad.dis.shtml

- The instructor may take actions such as:
  - require repetition of the subject work,
  - assign an F grade or a 'zero' grade to the subject work,
  - for serious offenses, assign an F grade for the course.

- The instructor must inform the student and the Department Chair in writing of the incident, the action taken, if any, and the student's right to appeal to the Chair of the Department Grievance Committee or to the Director of the Office of Student Conduct.

- The Office of Student Conduct may choose to conduct a formal hearing on the incident and to assess a penalty for misconduct.

- The Department will recommend that students involved in a second incident of cheating be dismissed from the University.

Babbage's Mechanical Computer (1822–)

- Problem: compute numerical tables

- Never completed, but first modern architecture:
  - Separated data and program memory;
  - Instruction based operations
  - Supported conditional jumps
  - Separated I/O unit

- Difference engine:
  - Implement finite differences
  - 25,000 parts, 15tons

Dawn of time

ENIAC: (1945–1955)

- Problem: accurately compute the trajectory for shells
  - Speed: 0.05 MIPS (50,000 instructions/sec)
  - "The machine designed by Drs. Eckert and Mauchly was a monstrosity. When it was finished, the ENIAC filled an entire room, weighed thirty tons, and consumed two hundred kilowatts of power."

- http://ei.cs.vt.edu/~history/ENIAC.Richey.HTML

History Phase 1 (1948–1970)

Hardware Expensive, Humans Cheap

- When computers cost millions of $'s, optimize for more efficient use of the hardware!
  - Lack of interaction between user and computer

- User at console: one user at a time
- Batch monitor: load program, run, print

- Optimize to better use hardware
  - When user thinking at console, computer idle⇒BAD!
  - Feed computer batches and make users wait
  - Autograder for this course is similar

- No protection: what if batch program has bug?
Punch Cards (1940s & 60s)

- Type program on punch cards (e.g., one card one line of code)
- Submit box with punch cards (make sure cards are not mixed!)
- Get results after a few hours or days

Core Memories (1950s & 60s)

- Core Memory stored data as magnetization in iron rings
  - Iron "cores" woven into a 2-dimensional mesh of wires
  - Origin of the term "Dump Core"
- See: http://www.columbia.edu/acis/history/core.html

History Phase 1½ (late 60s/early 70s)

- Data channels, Interrupts: overlap I/O and compute
  - DMA – Direct Memory Access for I/O devices
  - I/O can be completed asynchronously
- Multiprogramming: several programs run simultaneously
  - Small jobs not delayed by large jobs
  - More overlap between I/O and CPU
  - Need memory protection between programs and/or OS
- Complexity gets out of hand:
  - Multics: announced in 1963, ran in 1969
    » 1777 people "contributed to Multics" (30-40 core dev)
    » Turing award lecture from Fernando Corbató (key researcher): "On building systems that will fail"
  - OS 360: released with 1000 known bugs (APARs)
    » "Anomalous Program Activity Report"
- OS finally becomes an important science:
  - How to deal with complexity???
  - UNIX based on Multics, but vastly simplified

A Multics System (Circa 1976)

- The 6180 at MIT IPC, skin doors open, circa 1976:
  - "We usually ran the machine with doors open so the operators could see the AQ register display, which gave you an idea of the machine load, and for convenient access to the EXECUTE button, which the operator would push to enter BOS if the machine crashed."
Early Disk History

1973:
- 1.7 Mbit/sq. in
- 140 MBytes

1979:
- 7.7 Mbit/sq. in
- 2,300 MBytes

Contrast: Seagate 2TB,
- 400 Gbit/SQ in,
- 3½ in disk
- 4 platters

History Phase 2 (1970 – 1985)

- Hardware cheaper, humans expensive
- Computers available for tens of thousands of dollars instead of millions
- OS technology maturing/stabilizing
- Interactive timesharing:
  - Use cheap terminals (~$1000) to let multiple users interact with the system at the same time
  - Sacrifice CPU time to get better response time
  - Users do debugging, editing, and email online
- Problem: Thrashing
  - Performance very non-linear response with load
  - Thrashing caused by many factors including
    - Swapping, queueing

The ARPANet (1968–1970’s)

- Paul Baran
  - RAND Corp, early 1960s
  - Communications networks that would survive a major enemy attack
- ARPAnet: Research vehicle for “Resource Sharing Computer Networks”
  - 2 September 1969: UCLA first node on the ARPAnet
  - December 1969: 4 nodes connected by 56 kbps phone lines
  - 1971: First Email
  - 1970’s: <100 computers

ARPANET GEOGRAPHIC MAP, OCTOBER 1980

BBN team that implemented the interface message processor
ARPANet Evolves into Internet

- First E-mail SPAM message: 1 May 1978 12:33 EDT
- 80–83: TCP/IP, DNS; ARPANET and MILNET split
- 85–86: NSF builds NSFNET as backbone, links 6 Supercomputer centers, 1.5 Mbps, 10,000 computers
- 87–90: link regional networks, NSI (NASA), ESNet (DOE), DARTnet, TWBNet (DARPA), 100,000 computers

<table>
<thead>
<tr>
<th>ARPANet</th>
<th>SATNet</th>
<th>NSNet</th>
<th>PRNet</th>
<th>IPv4</th>
<th>IPv6</th>
</tr>
</thead>
</table>

SATNet: Satellite network
PRNet: Radio Network

Types of Networks

- Geographical distance
  - Local Area Networks (LAN): Ethernet, Token ring, FDDI
  - Metropolitan Area Networks (MAN): DQDB, SMDS
  - Wide Area Networks (WAN): X.25, ATM, frame relay
- Caveat: LAN, MAN, WAN may mean different things
  - Service, network technology, networks
- Information type
  - Data networks vs. telecommunication networks
- Application type
  - Special purpose networks: airline reservation network, banking network, credit card network, telephony
  - General purpose network: Internet

Network Components (Examples)

<table>
<thead>
<tr>
<th>Links</th>
<th>Interfaces</th>
<th>Switches/routers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fibers</td>
<td>Ethernet card</td>
<td>Large router</td>
</tr>
<tr>
<td>Coaxial Cable</td>
<td>Wireless card</td>
<td>Telephone switch</td>
</tr>
</tbody>
</table>

History Phase 3 (1981—)

- Computer costs $1K, Programmer costs $100K/year
  - If you can make someone 1% more efficient by giving them a computer, it's worth it!
  - Use computers to make people more efficient
- Personal computing:
  - Computers cheap, so give everyone a PC
- Limited Hardware Resources Initially:
  - OS becomes a subroutine library
  - One application at a time (MSDOS, CP/M, …)
- Eventually PCs become powerful:
  - OS regains all the complexity of a "big" OS
  - Multiprogramming, memory protection, etc (NT, OS/2)
- Question: As hardware gets cheaper does need for OS go away?
History Phase 3 (con’t)

Graphical User Interfaces

- CS160 ⇒ All about GUIs
- Xerox Star: 1981
  - Originally a research project (Alto)
  - First “mice”, “windows”
- Apple Lisa/Macintosh: 1984
  - “Look and Feel” suit 1988
- Microsoft Windows:
  - Win 1.0 (1985)
  - Win 3.1 (1990)
  - Win 95 (1995)
  - Win NT (1993)
  - Win XP (2001)
  - Win Vista (2007)

History Phase 4 (1988—): Distributed Systems

- Networking (Local Area Networking)
  - Different machines share resources
  - Printers, File Servers, Web Servers
  - Client – Server Model
- Services
  - Computing
  - File Storage

History Phase 4 (1988—): Internet

- Developed by the research community
  - Based on open standard: Internet Protocol
  - Internet Engineering Task Force (IETF)
- Technical basis for many other types of networks
  - Intranet: enterprise IP network
- Services Provided by the Internet
  - Shared access to computing resources: telnet (1970’s)
  - Shared access to data/files: FTP, NFS, AFS (1980’s)
  - Communication medium over which people interact
    - email (1980’s), on-line chat rooms, instant messaging (1990’s)
    - audio, video (1990’s, early 00’s)
    - Medium for information dissemination
      - USENET (1980’s)
      - WWW (1990’s)
      - Audio, video (late 90’s, early 00’s) – replacing radio, TV?
      - File sharing (late 90’s, early 00’s)

Network “Cloud”
Regional Nets + Backbone

Backbones + NAPs + ISPs

LAN: Local Area Network

ISP: Internet Service Provider
NAP: Network Access Point

The Morris Internet Worm (1988)

- Internet worm (Self-reproducing)
  - Author Robert Morris, a first-year Cornell grad student
  - Launched close of Workday on November 2, 1988
  - Within a few hours of release, it consumed resources to the point of bringing down infected machines

- Techniques
  - Exploited UNIX networking features (remote access)
  - Bugs in finger (buffer overflow) and sendmail programs (debug mode allowed remote login)
  - Dictionary lookup-based password cracking
  - Grappling hook program uploaded main worm program
LoveLetter Virus (May 2000)

- E-mail message with VBScript (simplified Visual Basic)
- Relies on Windows Scripting Host
  - Enabled by default in Win98/2000
- User clicks on attachment \( \Rightarrow \) infected!
  - E-mails itself to everyone in Outlook address book
  - Replaces some files with a copy of itself
  - Searches all drives
  - Downloads password cracking program
- 60-80% of US companies infected and 100K European servers

History Phase 5 (1995—): Mobile Systems

- Ubiquitous Mobile Devices
  - Laptops, PDAs, phones
  - Small, portable, and inexpensive
    - Many computers/person!
  - Limited capabilities (memory, CPU, power, etc...)
- Wireless/Wide Area Networking
  - Leveraging the infrastructure
  - Huge distributed pool of resources extend devices
  - Traditional computers split into pieces. Wireless keyboards/mice, CPU distributed, storage remote
- Peer-to-peer systems
  - Many devices with equal responsibilities work together
  - Components of “Operating System” spread across globe

Datacenter is the Computer

- (From Luiz Barroso’s talk at RAD Lab 12/11)
- Google program == Web search, Gmail,…
- Google computer ==
  - Thousands of computers, networking, storage
  - Warehouse-sized facilities and workloads may be unusual today but are likely to be more common in the next few years

Migration of Operating-System Concepts and Features
History of OS: Summary

- Change is continuous and OSs should adapt
  - Not: look how stupid batch processing was
  - But: Made sense at the time
- Situation today is much like the late 60s
  - Small OS: 100K lines
  - Large OS: 10M lines (5M for the browser!)
    » 100-1000 people-years
- Complexity still reigns
  - NT developed (early to late 90's): Never worked well
  - Windows 2000/XP: Very successful
  - Windows Vista (aka "Longhorn") delayed many times
    » Finally released in January 2007
    » Slow adoption rate
  - Windows 7: very successful
- CS162: understand OSs to simplify them

Implementation Issues
(How is the OS implemented?)

- Policy vs. Mechanism
  - Policy: What do you want to do?
  - Mechanism: How are you going to do it?
  - Should be separated, since both change
- Algorithms used
  - Linear, Tree-based, Log Structured, etc...
- Event models used
  - threads vs event loops
- Backward compatibility issues
  - Very important for Windows 2000/XP
- System generation/configuration
  - How to make generic OS fit on specific hardware

Conclusion

- Rapid Change in Hardware Leads to changing OS
  - Batch ⇒ Multiprogramming ⇒ Timesharing ⇒ Graphical UI ⇒ Ubiquitous Devices ⇒ Cyberspace/Metaverse/??
- OS features migrated from mainframes ⇒ PCs
- Policy vs Mechanism
  - Crucial division: not always properly separated!
- Complexity is always out of control
  - However, “Resistance is NOT Useless!”