

System Architecture

History of Operating Systems

Some slides from A. D. Joseph, University of Berkeley

See also:

www.osdata.com/kind/history.htm

www.armory.com/~spectre/tech.html

courses.cs.vt.edu/~cs1104/VirtualMachines/OS.1.html

en.wikipedia.org/wiki/History_of_operating_systems



Moore's Law Drives OS Change

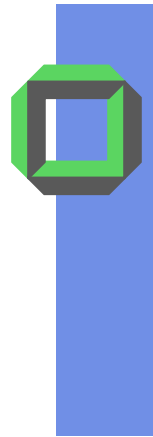
	1981	2006	Factor
CPU MHz, Cycles/inst	10 3—10	3200x4 0.25—0.5	1,280 6—40
DRAM capacity	128KB	4GB	32,768
Disk capacity	10MB	1TB	100,000
Net bandwidth	9600 b/s	1 Gb/s	110,000
# addr bits	16	32	2
#users/machine	10	≤ 1	≤ 0.1
Price	\$25,000	\$4,000	0.2

Typical academic computer 1981 vs 2006

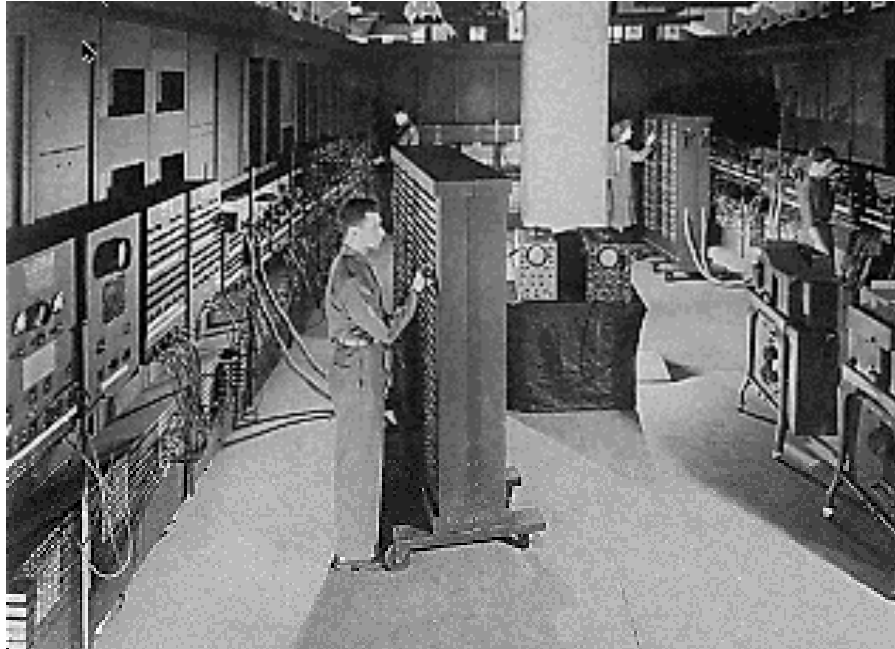


Moore's Law Effects

- Nothing like this in any other area of business
- Transportation in over 200 years:
 - Only 2 orders of magnitude from horseback @10mph to Concorde @1000mph
 - Computers do this every decade
- What does this mean for us?
 - Techniques have to vary over time to adapt to changing tradeoffs
- Let's 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



Dawn of Time ENIAC: (1945-55)

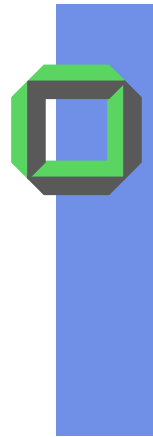


- “The machine designed by Eckert and Mauchly was a monstrosity. When it was finished, the ENIAC filled an entire room, weighed **30 tons**, and consumed **200 kilowatts** of power.”
- <http://ei.cs.vt.edu/~history/ENIAC.Richey.HTML>

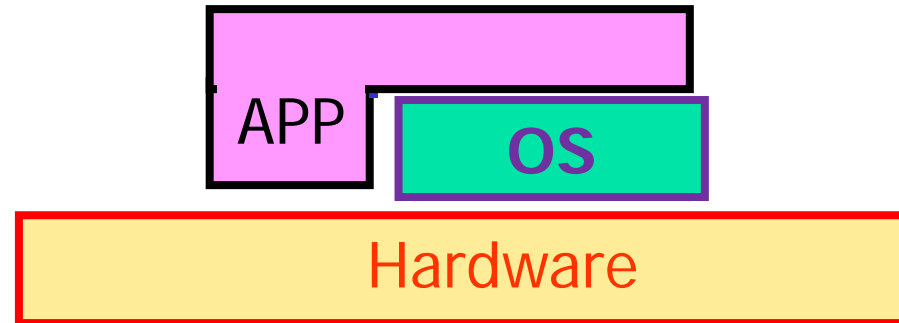


History Phase 1: 1948-70

Expensive Hardware
Cheap Humans



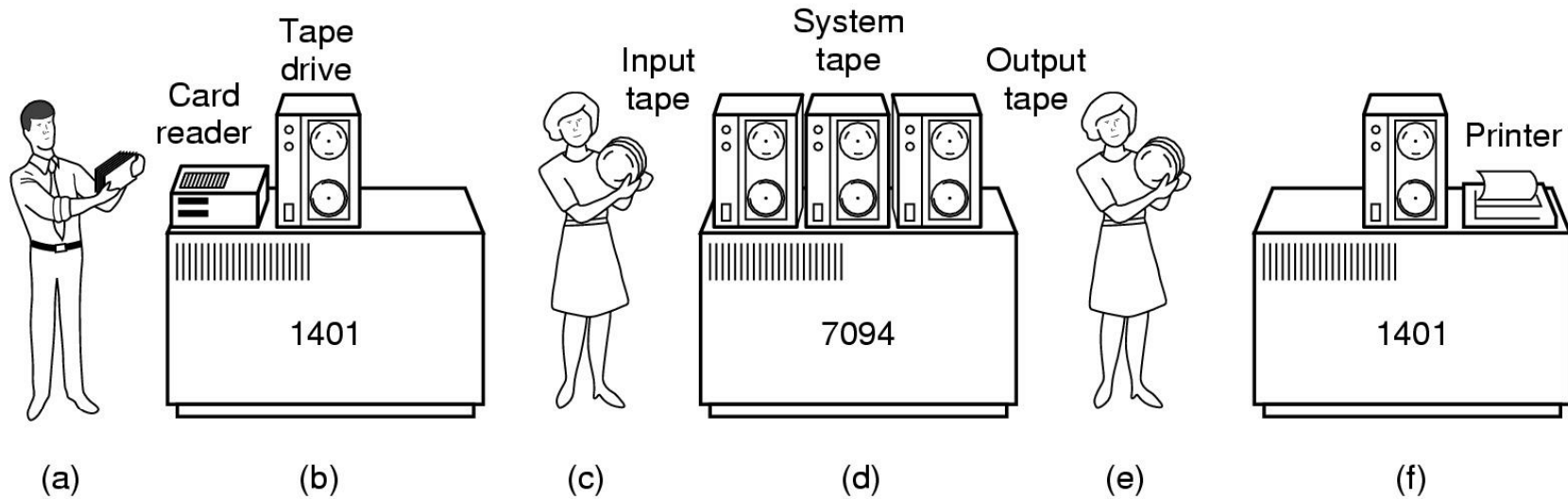
History OS: Evolution Step 0



- Simple OS: One program, one user, one machine:
 - examples: early computers, early PCs,
 - embedded controllers such as Nintendo, cars, elevators
 - OS just a library of standard services, e.g. standard device drivers, interrupt handlers, I/O
- Non-problems: **No malicious people. No bad programs**
⇒ A minimum of complex interactions
- Problem: poor utilization, expensive



History of Systems



- Early batch system
 - bring cards to 1401
 - read cards to tape
 - put tape on 7094 which does computing
 - put tape on 1401 which prints output

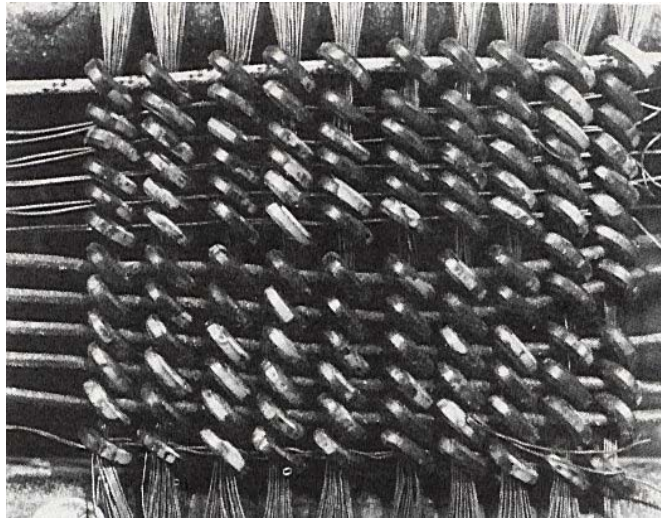


History Phase 1

- 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⇒ very bad
 - Feed computer batches and make users wait
- **No protection**: what if batch program was buggy?



Core Memories (1950s & 60s)



The first magnetic core memory, from the IBM 405 Alphabetical Accounting Machine.

- 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”
 - Rumor that IBM consulted Life Saver company
- <http://www.columbia.edu/acis/history/core.html>



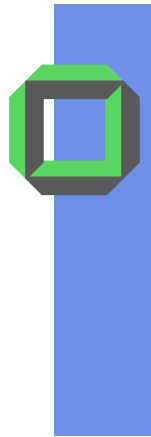
Late 60s - Early 70s

- **Data channels, Interrupts:** overlap I/O and compute
 - DMA – Direct Memory Access for I/O devices
 - I/O can be completed asynchronously
- **Multiprogramming:** $n > 1$ 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



Late 60s - Early 70s

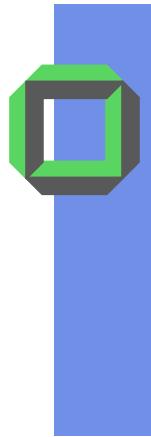
- Complexity starts to get out of hand:
 - Multics: announced in 1963, ran in 1969
 - **1777** people “contributed to MIT’s Multics” (30-40 core dev)
 - Turing award lecture from Fernando Corbató (key researcher): “On building systems that will fail”
 - OS 360: released with **1000** known bugs
 - “Anomalous Program Activity Report”
- OS finally becomes an important science:
 - *How to deal with complexity?*
 - Result: UNIX based on Multics, but vastly simplified



The Multics System (~ 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.”
- <http://www.multicians.org/multics-stories.html>



Ritchie & Thompson at PDP 11



Unix needed 16 KB*

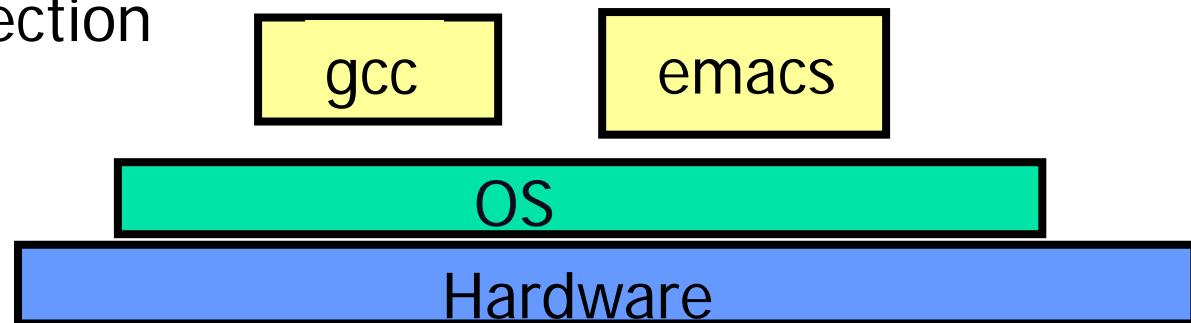
users could only get 8 KB
for their application

*at that time a mini-OS



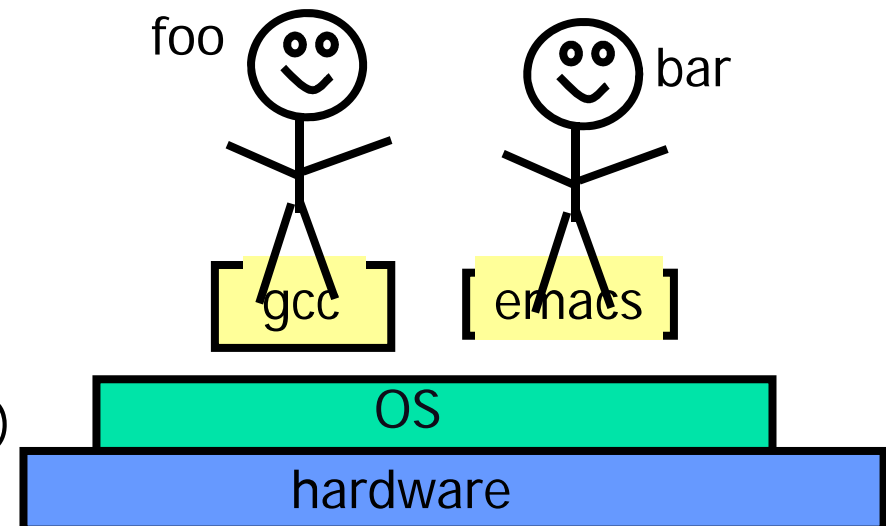
History OS: Evolution Step 1

- Simple OS is inefficient:
 - a waiting process blocks everything else on the machine
- (Seemingly) Simple hack:
 - run more than one process at once
 - when one process blocks, switch to another
- A couple of problems: *what if a program*
 - *does infinite loops or*
 - *starts randomly scribbling on memory?*
- OS adds protection
 - Interposition
 - Preemption
 - Privilege



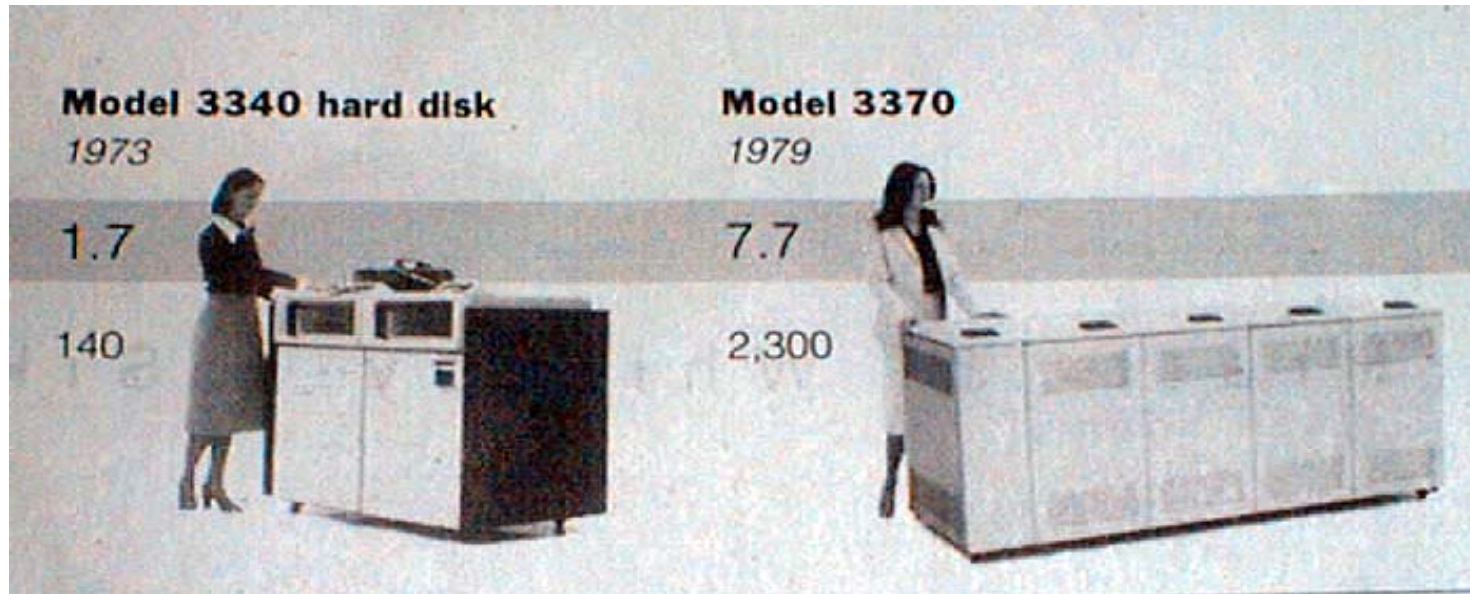
History OS: Evolution Step 2

- Simple OS is too expensive:
 - one user = one computer \Rightarrow
- (Seemingly) simple hack:
 - Allow more than one user at once
 - *Does machine now run n times slower?* Usually not
 - Key observation: users are active in bursts
 - If idle, give resources to others
- Problems: *what if*
 - *users are greedy*
 - *evil*
 - *or just too numerous?*
- OS adds protection
 - (notice: as we try to utilize resources, complexity grows)





Early Disk History



1973:
1.7 Mbit/sq. in
140 MBytes

1979:
7.7 Mbit/sq. in
2300 MBytes

source: New York Times, 2/23/98, page C3,
"Makers of disk drives crowd even more data into even smaller spaces"



History Phase 2: 70 - 85

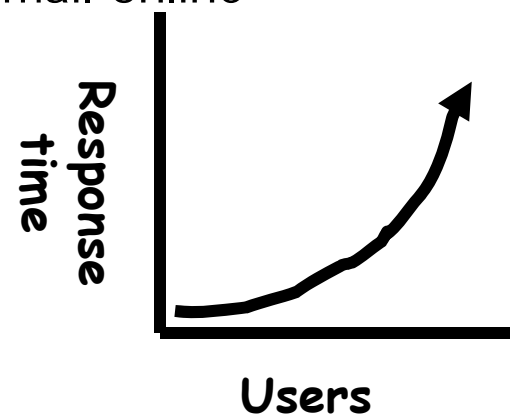
Cheaper HW

More Expensive Humans



History Phase 2

- 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
 - Inefficient queuing





History Phase 3: 81 - 89

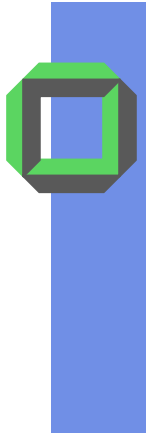
Very Cheap HW

Very Expensive Humans



History Phase 3 (1981—)

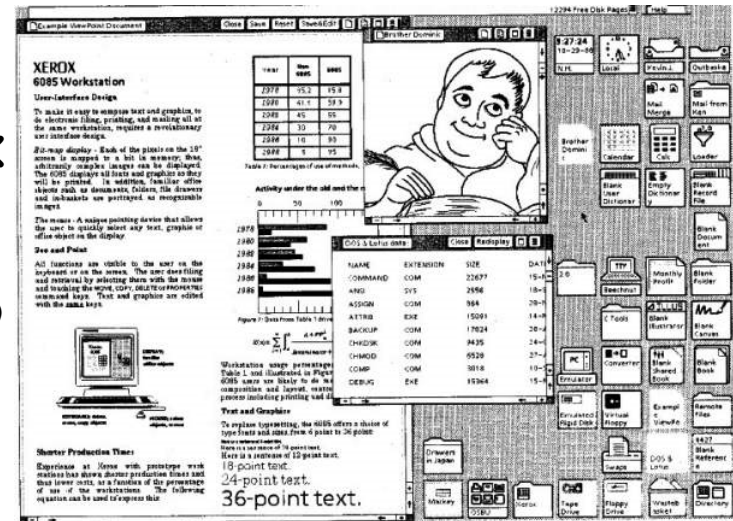
- Computer costs \$1K, Programmer costs \$100K/year
 - If you can make someone 1% more efficient by giving them a computer, it's worth it
 - Use computers to make people more efficient
- Personal computing:
 - Computers cheap, so give everyone a PC
- Limited Hardware Resources Initially:
 - OS becomes 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 HW gets cheaper, does the need for OS research go away?*



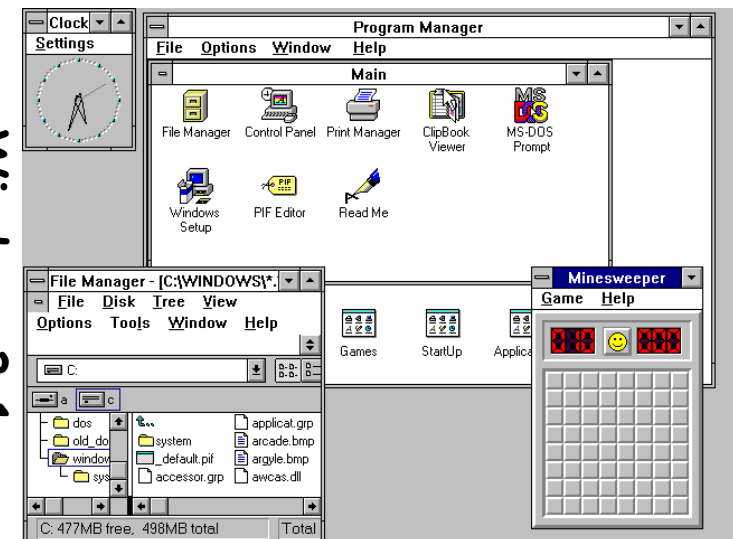
Graphical User Interfaces

- CS160 ⇒ All about GUIs
- Xerox Star: 1981
 - Originally a research project (Alto)
 - First "mice", "windows"
- Apple Lisa/Machintosh: 1984
 - "Look and Feel" suit 1988
- Microsoft Windows:
 - Win 1.0 (1985) } Single Level
 - Win 3.1 (1990) } Single Level
 - Win 95 (1995) } Single Level
 - Win NT (1993) } HAL/Protection
 - Win 2000 (2000) } No HAL/Full Prot
 - Win XP (2001) } No HAL/Full Prot

Xerox Star



Windows 3.1





History Phase 4: 89 - 95

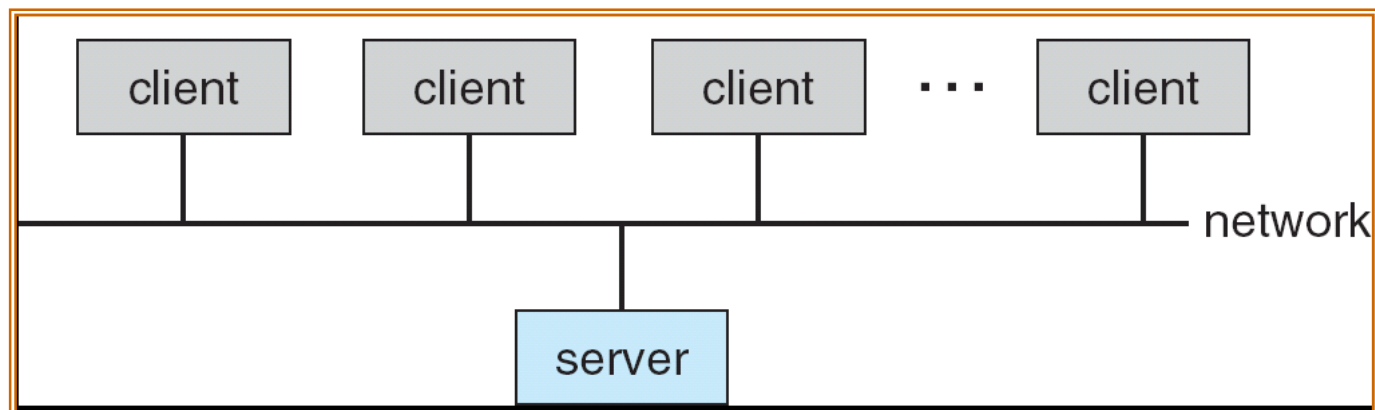
Distributed Systems

Not in this course



History Phase 4

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





History Phase 5: 95 - ...

Mobile Systems

Not in this course



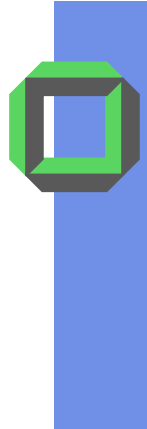
History Phase 5:

- Ubiquitous Mobile Devices
 - Laptops, PDAs, phones
 - Small, portable, and inexpensive
 - Recently twice as many smart phones as PDAs
 - Many computers/person
 - Limited capabilities (memory, CPU, power, etc...)



History Phase 5:

- 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 (P2P)
 - Many devices with equal responsibilities work together
 - Components of "OS" spread across globe



CITRIS's Model: A Societal Scale Information System

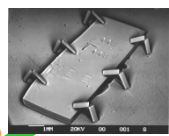
- Center for Information Technology Research in the Interest of Society
- The Network is the OS
 - Functionality spread throughout network



Scalable, Reliable, Secure Services



Frigidaire online refrigerator



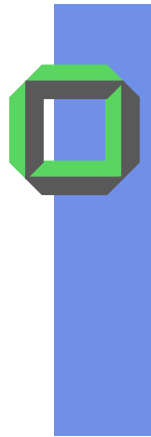
MEMS for Sensor Nets

Mobile, Ubiquitous Systems

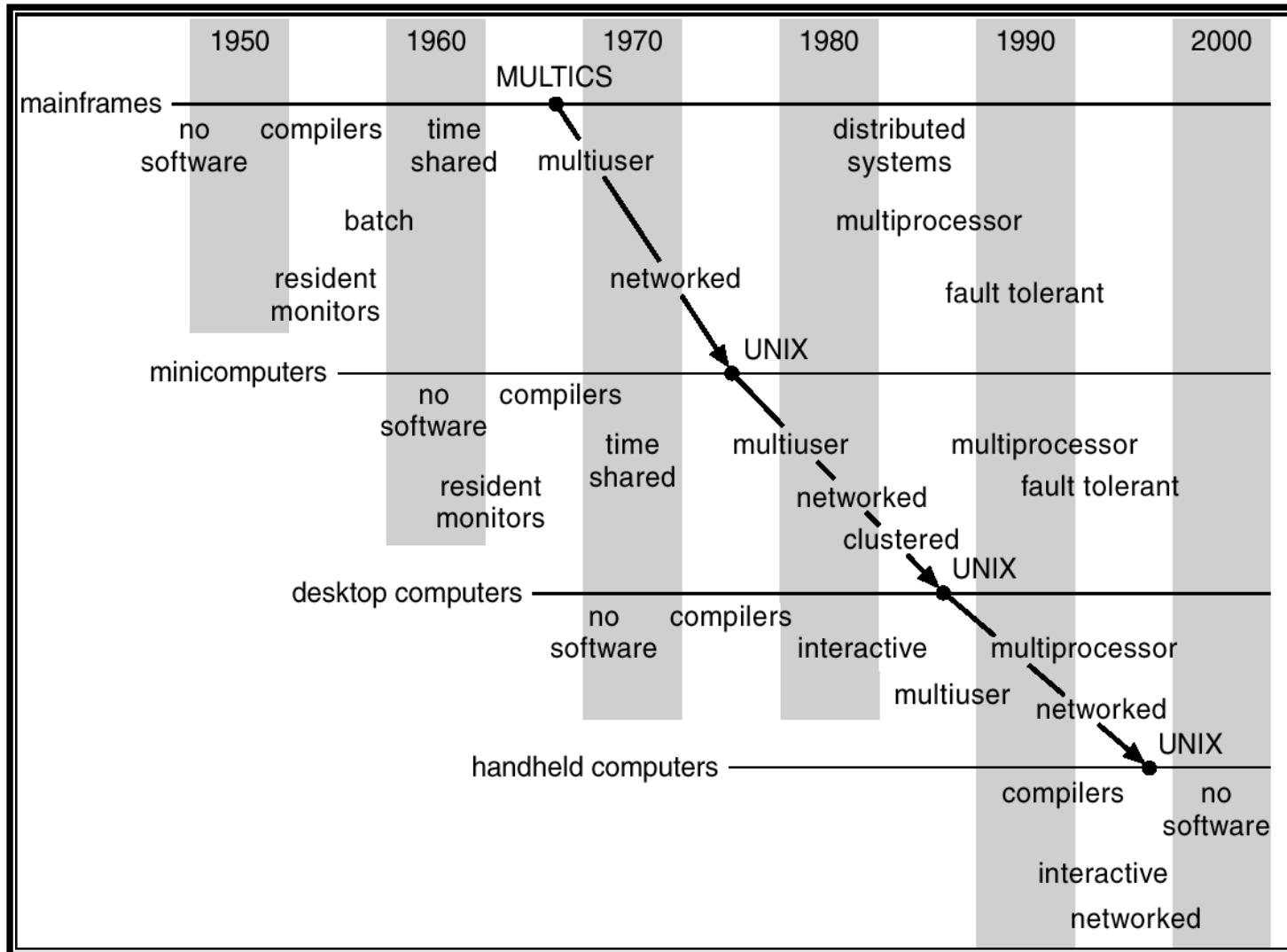


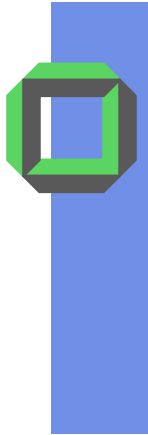
Moore's Law (2): Modern Laptop

	1981	2005	2006 Ultralight Laptop
CPU MHz, Cycles/inst	10 3—10	3200x4 0.25—0.5	1830 0.25—0.5
DRAM capacity	128KB	4GB	2GB
Disk capacity	10MB	1TB	100GB
Net bandwidth	9600 b/s	1 Gb/s	1 Gb/s (wired) 54 Mb/s (wireless) 2 Mb/s (wide-area)
# addr bits	16	32	32
#users/machine	10s	≤ 1	$\leq \frac{1}{4}$
Price	\$25,000	\$4,000	\$2500

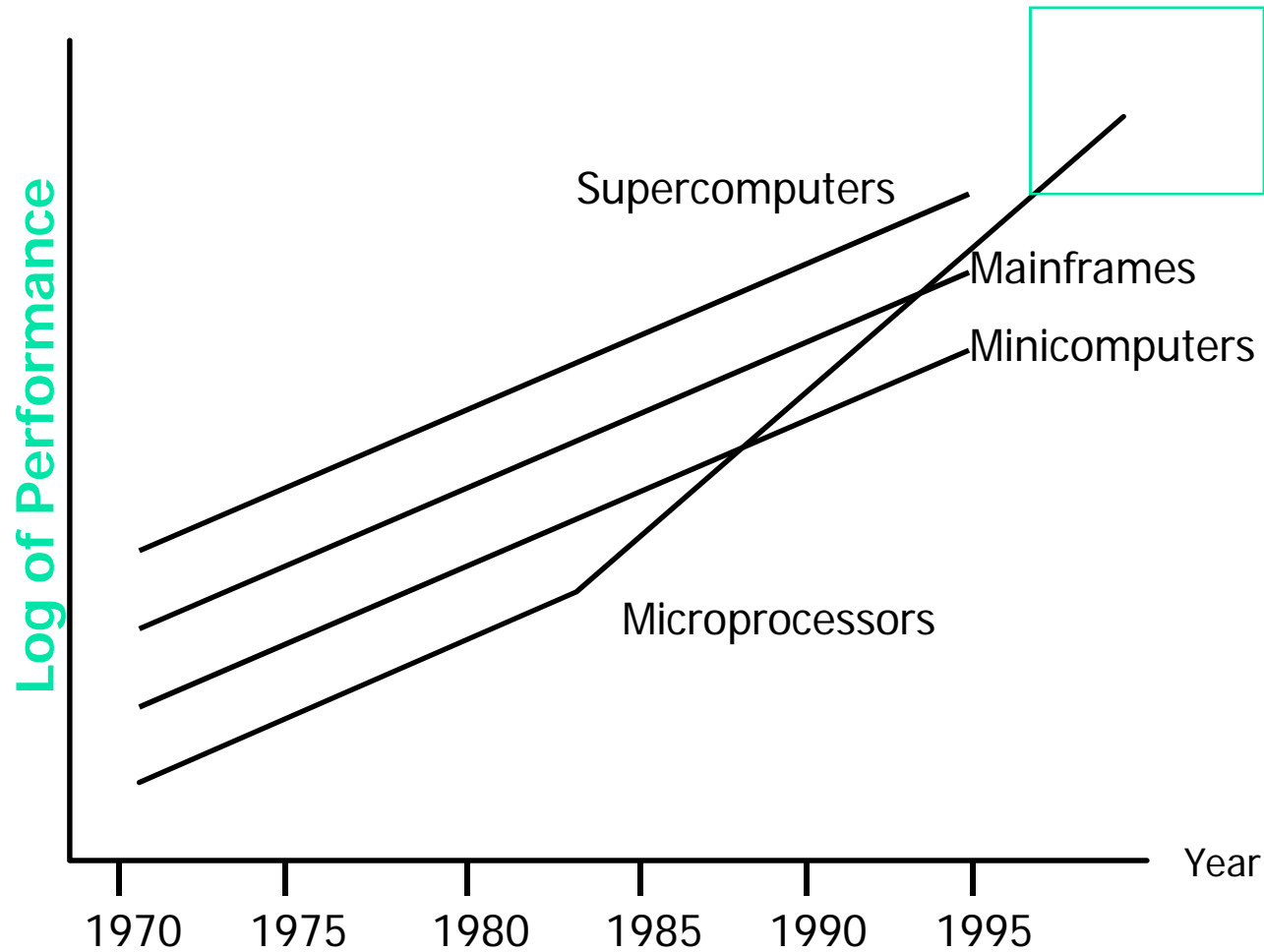


Migration of OS Concepts/Features





Compare: Performance Trends

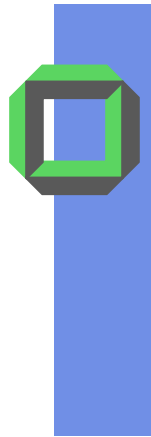




Timeline of OS

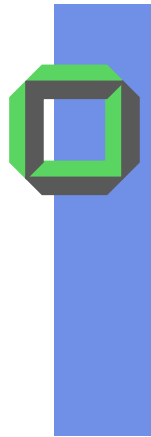
See wikipedia

See also List of OS in wikipedia



OS Timeline (50s & early 60s)

- 1956
 - GM-NAA I/O
- 1959
 - SHARE Operating System
- 1960
 - IBSYS
- 1961
 - CTSS
 - MCP
- 1962
 - GCOS
- 1964
 - EXEC 8
 - OS/360 (announced)
 - TOPS-10
- 1965
 - Multics (announced)
 - OS/360 (shipped)
 - Tape Operating System (TOS)



OS Timeline (60s & 70s)

1966

- DOS/360 (IBM)
- MS/8

1967

- ACP (IBM)
- CP/CMS
- ITS
- WAITS

1969

- TENEX
- Unix

1970s

1970

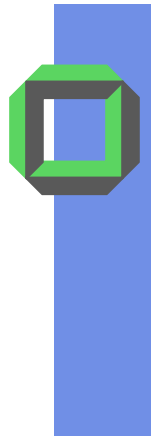
- DOS/BATCH 11 (PDP-11)

1971

- OS/8

1972

- MFT (Operating System)
- MVT
- RDOS
- SVS
- VM/CMS



OS Timeline (late 70s)

1973

- [Alto OS](#)
- [RSX-11D](#)
- [RT-11](#)
- [VME](#)

1974

- [MVS](#) (MVS/XA)

1975

- [BS2000](#)

1976

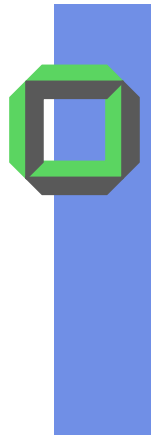
- [CP/M](#)
- [TOPS-20](#)

1978

- [Apple DOS 3.1](#) (*first Apple OS*)
- [TripOS](#)
- [VMS](#)
- [Lisp Machine](#) (CADR)

1979

- [POS](#)
- [NLTSS](#)



OS Timeline (1980 ...

1980

- [OS-9](#)
- [QDOS](#)
- [SOS](#)
- [XDE \(Tajo\)](#) (*Xerox Development Environment*)
- [Xenix](#)

1981

- [MS-DOS](#)

1982

- [Commodore DOS](#)
- [SunOS](#) (1.0)
- [Ulrix](#)

1983

- [Lisa OS](#)
- [Coherent](#)
- [Novell Netware](#)
- [ProDOS](#)

1984

- [Macintosh OS](#) (*System 1.0*)
- [MSX-DOS](#)
- [QNX](#)
- [UniCOS](#)



OS Timeline (Late 80s)

1985

- [AmigaOS](#)
- [Atari TOS](#)
- [MIPS OS](#)
- [Microsoft Windows 1.0](#) (*First Windows*)

1986

- [AIX](#)
- [GS-OS](#)
- [HP-UX](#)

1987

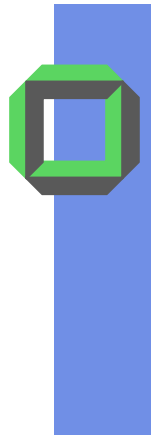
- [Arthur](#)
- [IRIX](#) (*3.0 is first SGI version*)
- [Minix](#)
- [OS/2](#) (1.0)
- [Microsoft Windows 2.0](#)

1988

- [A/UX](#) (Apple Computer)
- [LynxOS](#)
- [MVS/ESA](#)
- [OS/400](#)

1989

- [NeXTSTEP](#) (1.0)
- [RISC OS](#)
- [SCO Unix](#) (*release 3*)



OS Timeline (90s)

1990

- [Amiga OS 2.0](#)
- [BeOS \(v1\)](#)
- [OSF/1](#)
- [Windows 3.0](#)

1991

- [Linux](#)

1992

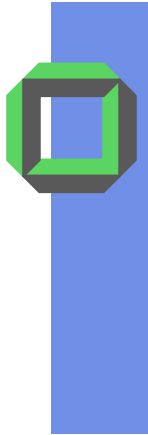
- [386BSD 0.1](#)
- [Amiga OS 3.0](#)
- [Solaris](#) (*2.0 is first not called SunOS*)
- [Windows 3.1](#)

1993

- [Plan 9](#) (First Edition)
- [FreeBSD](#)
- [NetBSD](#)
- [Windows NT 3.1](#) (*First version of NT*)

1995

- [Digital UNIX](#) (*aka [Tru64](#)*)
- [OpenBSD](#)
- [OS/390](#)
- [Windows 95](#)



OS Timeline

1996

- [Windows NT 4.0](#)

1997

- [Inferno](#)
- [Mac OS 7.6](#) (*first officially-named Mac OS*)
- [SkyOS](#)

1998

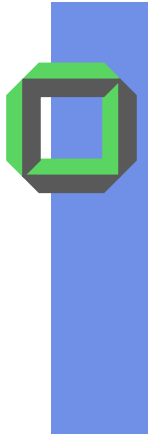
- [Windows 98](#)

1999

- [AROS](#) (Boot for the first time in Stand Alone version)
- [Mac OS 8](#)

2000

- [AtheOS](#)
- [Mac OS 9](#)
- [MorphOS](#)
- [Windows 2000](#)
- [Windows Me](#)



OS Timeline

2001

- [Amiga OS 4.0 \(May 2001\)](#)
- [Mac OS X 10.1](#)
- [Windows XP](#)
- [z/OS](#)

2002

- [Syllable](#)
- [Mac OS X 10.2](#)

2003

- [Windows Server 2003](#)
- [Mac OS X 10.3](#)

2005

- [Mac OS X 10.4](#)

Complete for yourselves



Classification of OS

See categorization in wikipedia



Categorization of OS

- 1 Early, and historically important
- 2 Proprietary
- 3 Nonproprietary Unix-like
- 4 Nonproprietary Non-Unix-like
- 5 Disk Operating System
- 6 Network Operating Systems
- 7 Generic/commodity, Non-Unix, and other
- 8 Operating Systems for Soviet Personal Computer
- 9 Hobby OS
- 10 Embedded
- 11 Interpreted
- 12 Fictional Operating Systems
- 13 LEGO Mindstorms



3 Non Proprietary OS

3 Nonproprietary Unix-like

[3.1 Research Unix-like and other POSIX-compliant systems](#)

[3.2 Open source Unix-like](#)

4 Nonproprietary non-Unix-like

[4.1 Research non-Unix-like](#)

[4.2 Open source non-Unix-like](#)



1 + 2 Early & Proprietary OS

1 Early, and historically important

- 1.1 Early, proprietary microcomputer OS

2 Proprietary

- 2.1 Acorn
- 2.2 Amiga
- 2.3 Apple/Macintosh
- 2.4 Array Networks
- 2.5 Atari ST
- 2.6 Burroughs (later Unisys)
- 2.7 Convergent Technologies
- 2.8 Be Incorporated
- 2.9 Digital/Tandem Computers/Compaq/HP
- 2.10 Honeywell
- 2.11 IBM
- 2.12 Microsoft
- 2.13 Non-Standard Language
- 2.14 Other
 - 2.14.1 Other proprietary Unix-like and POSIX-compliant systems
- 2.15 UNIVAC (later Unisys)



10 Embedded Systems

10.1 Personal digital assistants (PDAs)

10.2 Smartphones

10.3 Router

10.4 Microcontroller, Real-time OS



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 under development from early 90's to late 90's
 - Never worked very well
 - Jury still out on Windows 2000/XP
 - Windows Vista (aka "Longhorn") delayed many times
 - Latest release date of 2005, 2006, 2007+
 - Promised by removing some of the intended technology



Further Links to History of OS

<http://www.armory.com/~spectre/tech.html>

www.computinghistorymuseum.org/teaching/papers/research/history_of_operating_system_Moumina.pdf

<http://www.math-cs.gordon.edu/courses/cs322/lectures/history.html>

<http://www.personal.kent.edu/~rmuhamma/OpSystems/Myos/osHistory.htm>