

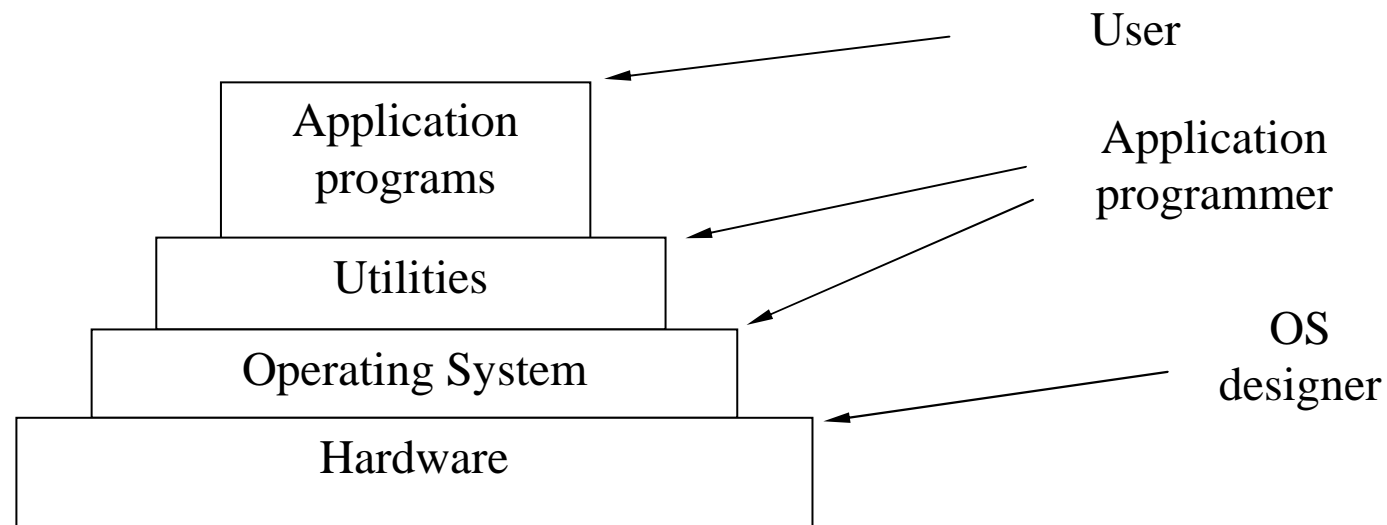
1. Introduction to Operating Systems

1.1 Introduction

- No single definition of what an operating system (OS) is
- The OS is the software that is closest to the computer hardware. Goals:
 - Abstract/hide the hardware complexity
 - ⇒ Interface or virtual machine, easier to use than the hardware
 - Guarantee the correct functioning of the computer
 - ⇒ Balanced management of resources (CPU, memory, I/O devices)
- What is an operating system? Two viewpoints:
 - User/programmer: set of routines that allow using the resources of a computer, ignoring its precise characteristics
 - ⇒ Functional viewpoint (use of the OS): virtual machine
 - Designer: software running on top of the hardware in order to manage the resources in an efficient manner
 - ⇒ This viewpoint corresponds to the implementation of the OS

1.2 Functional viewpoint of operating systems

- Interfaces of an operating system:
 - System calls (OS primitives)
 - Command Interpreter: text oriented (command-line, shell), graphical...
- System call interface (functions of the virtual machine):
 - Interface between applications and the OS (set of routines)
 - Source-code level compatibility among different computers



1.3 Operating system functions

- Program execution. Load instructions and data into main memory, initialize files and I/O devices, prepare other resources...
- Access to Input/Output devices. Hide their complexity, allowing programs to do I/O operations in terms of simple reads and write instructions
- Controlled access to files. Manage file formats and storing. Control concurrent accesses to files
- Access to the system. Control the permission to enter the system. Control the permission to access the resources
- Error detection and recovery. Detect both hardware and software errors. Actions: retry, finish program, warn...
- Accounting. Measure the use of resources. Goals: detect new needs, improve efficiency, correct mistakes, billing...

1.4 Evolution of operating systems

- Batch systems:
 - First OS: monitor program, punch cards, operator. The monitor chained programs. The CPU was most of the time idle, waiting for I/O completion. In order to improve CPU usage, multiprogramming was developed
- Time sharing systems:
 - Together with multiprogramming, in the 60s. Interactive systems, dumb terminals (no CPU, no memory, no disk), *quantum*. Goal: response time as short as possible
- Teleprocessing systems:
 - Remote terminals (PCs nowadays), connected to the system via a proprietary protocol: reservation systems, banking transaction systems

1.4 Evolution of operating systems (continued)

- Personal systems (PCs, workstations):
 - 80s, hardware becomes cheaper. They were initially single-user and monoprogrammed (e.g., MS-DOS). Today they are multiprogrammed, with sophisticated graphical user interfaces. User = administrator
- Networked systems:
 - computers are connected via a network protocol
TCP/IP (public) vs. Novell, AppleTalk (proprietary)
 - each computer has its own OS
 - servers ↔ clients
 - types of networks: local (*Ethernet*), wide area (*Internet*)
- Distributed systems:
 - resources/services of various machines are transparently integrated. Goals: resource sharing, improve efficiency, fault-tolerance...

1.5 Classification of operating systems

- Monoprogrammed / Multiprogrammed:

Concurrent program execution \equiv processes

Multitasking \rightarrow programs of a single user

Multiprocessing (many CPUs) \rightarrow real parallel execution

- Single terminal / multiterminal
- Single user / multiuser: distinguish users
- Proprietary / open source: related to a specific brand/company

* Microcomputer OS \rightarrow minicomputer OS \rightarrow mainframe OS (old classification)

* User friendly / not friendly: graphical interfaces

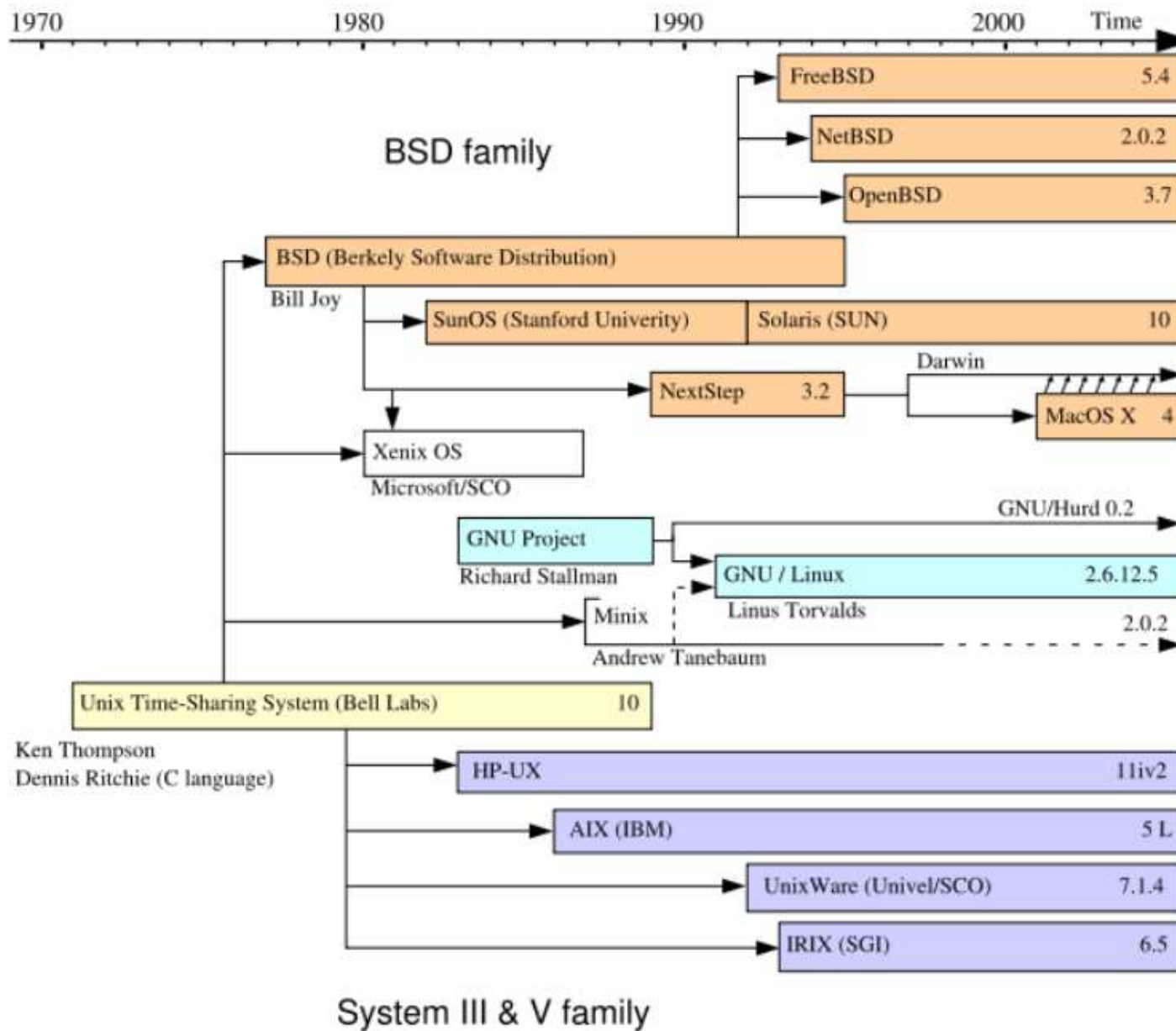
A computer can host more than one OS (e.g., Windows and Linux in a PC). Also, there are OS emulators (e.g., *Win4Lin*, *VMWare*, *VirtualPC*...)

1.6 Examples and families

1.6.1. Unix: families and versions

- 1970 Bell-Labs AT&T. C programming language. 1974 public version available
- Unix families:

Ultrix	–	Digital	}	System V	ATT	
Xenix	–	Microsoft		BSD	Univ. Berkeley:	Sun (Solaris)
AIX	–	IBM				
PCs	–	SCO, SantaCruz				
	–	Linux (RedHat, Debian, SuSe, Ubuntu...):			www.linux.org	
- Multiprogrammed, multiuser, multiterminal. Text oriented command interpreters (shell, C-shell, K-shell...), also graphical (Openwin, Motif, KDE, Gnome...). Some versions support multiprocessing



1.6.2. Microsoft: MS-DOS, Windows, Windows95/98/ME, WindowsNT/2000, WindowsXP...

- 1980: IBM. Contract with Microsoft (1981) → MS-DOS
- MS-DOS 1.0 (12 KB): Compatible with CP/M, better disk management, faster

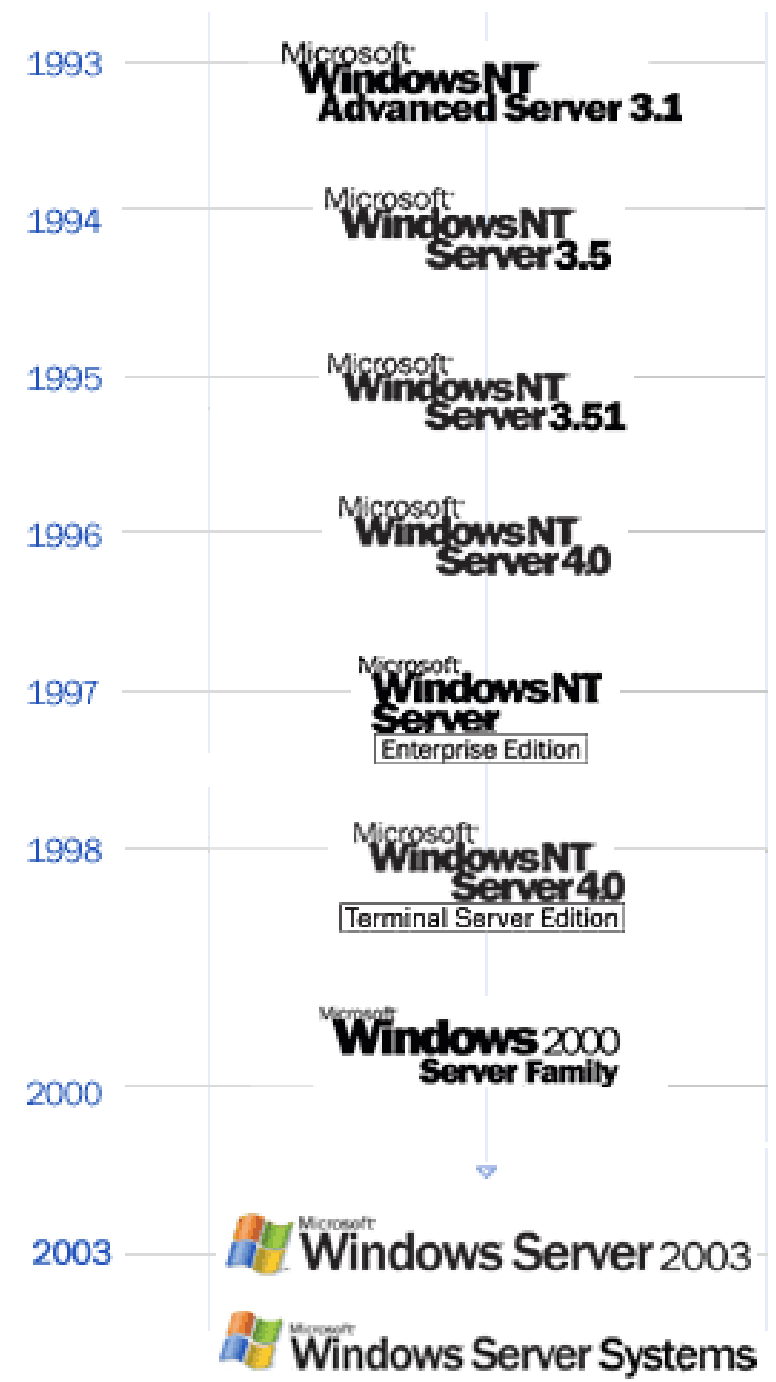
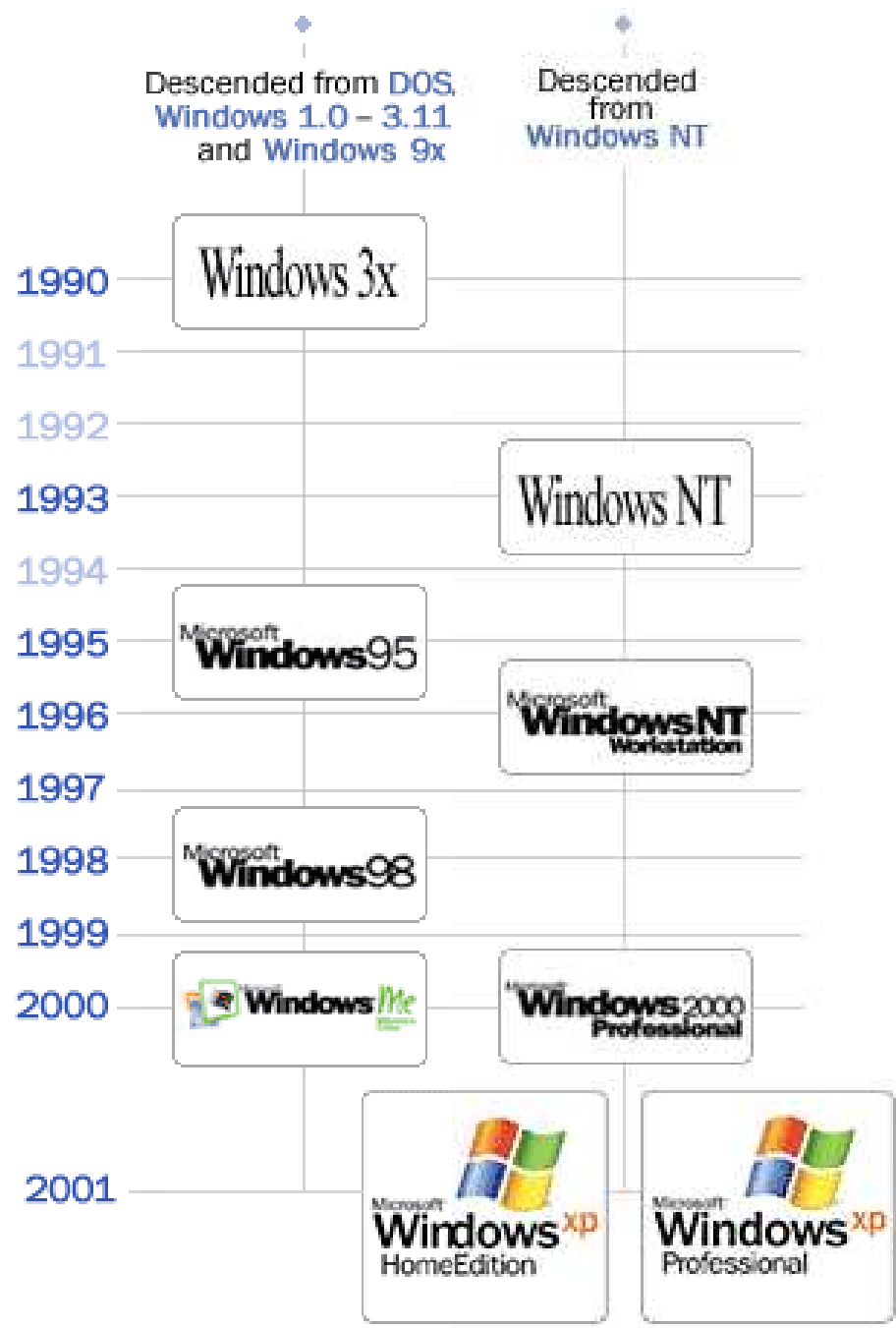
Single directory per floppy disk, limited to 64 files

- MS-DOS 2.0: (1983, PC/XT) Hard disk, hierarchical directory system, Input/Output redirection (imported from Unix)
- 1984 (PC/AT): Intel 80286 (extended address space, memory protection mechanisms). MS-DOS 3.0 (same as 2.0, 36 KB)
3.1 version: network support

- 1990: Windows (not a real OS): graphical interface on top of MS-DOS, offering multitasking

- 1995: Windows95, multitasking, memory protection, and 32 bit addressing.
Windows98 and WindowsME as evolutions

Windows NT (New Technology) → Windows 2000 → Windows XP → ...



1.6.3. IBM: MVS, AS/400, VM

Mainframes. Proprietary OS

- MVS: ↑ complex, ↑ big; teleprocessing applications, fast access to huge databases, security mechanisms
- VM: virtual machine operating system
- AS/400: similar to MVS, but for minicomputers
- OS/2 (Warp): microcomputers (PCs)

1.6.4. Others: Macintosh, Digital, Google....

- MacOS: first graphical user interface. Proprietary system: only in Apple computers ↓
- Digital VMS: used heavily in academia (minicomputers)
- Mach, Amoeba, Chorus: distributed operating systems
- Real-time OS: Windows XP Embedded, QNX, RTLinux...
- PDA, smartphone, tablet: iOS, Windows Phone, Palm OS, Symbian, Android