

Operating Systems Orientation

Copyright ©: Nahstedt, Angrave, Abdelzaher, Kravets, Gupta

1

Objectives

- Explain the main purpose of operating systems and describe milestones of OS evolution
- Explain fundamental machine concepts
 - Instruction processing
 - Memory hierarchy
 - Interrupts
 - I/O
- Explain fundamental OS concepts
 - System calls
 - Processes
 - Synchronization
 - Files
- Explain the POSIX standard (UNIX specification)

Copyright ©: Nahstedt, Angrave, Abdelzaher, Kravets, Gupta

2

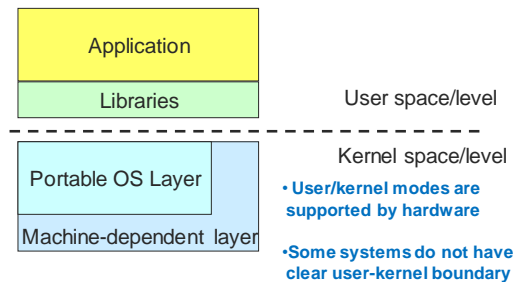
What is an Operating System?

- It is an *extended machine*
 - Hides the messy details that must be performed
 - Presents user with a virtual machine interface, easier to use
- It is a *resource manager*
 - Each program gets time with the resource
 - Each program gets space on the resource

Copyright ©: Nahstedt, Angrave, Abdelzaher, Kravets, Gupta

3

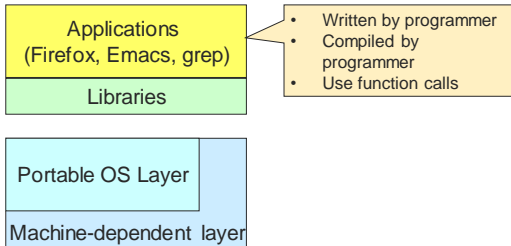
A Peek into Unix



Copyright ©: Nahstedt, Angrave, Abdelzaher, Kravets, Gupta

4

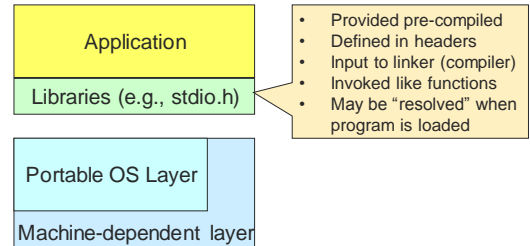
[Application]



Copyright ©: Nahstedt, Angrave, Abdelzaher, Kravets, Gupta



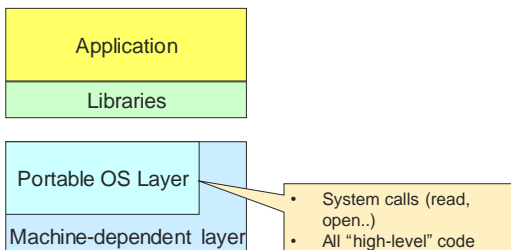
[Unix: Libraries]



Copyright ©: Nahstedt, Angrave, Abdelzaher, Kravets, Gupta



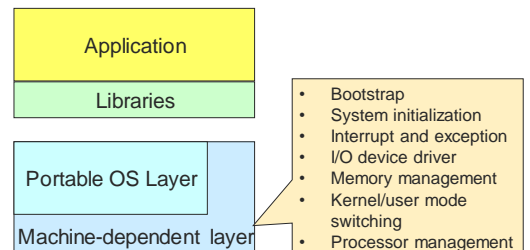
[Typical Unix OS Structure]



Copyright ©: Nahstedt, Angrave, Abdelzaher, Kravets, Gupta



[Typical Unix OS Structure]



Copyright ©: Nahstedt, Angrave, Abdelzaher, Kravets, Gupta



History of Computer Generations

- First generation 1945 – 1955
 - Vacuum tubes, plug boards
- Second generation 1955 - 1965
 - Transistors, batch systems
- Third generation 1965 – 1980
 - ICs and multiprogramming
- Fourth generation 1980 – present
 - Personal computers

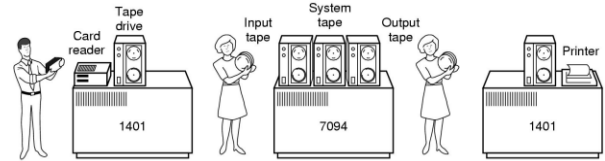
Batch → Multiprogramming → Time sharing → Personal

Copyright ©: Nahstedt, Angrave, Abdelzaher, Kravets, Gupta

9



History of Operating Systems



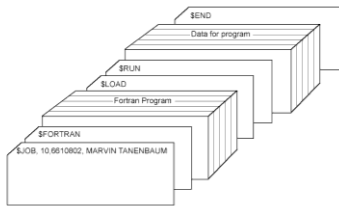
- Early systems
 - bring cards to 1401
 - read cards to tape
 - put tape on 7094 which does computing
 - put tape on 1401 which prints output

Copyright ©: Nahstedt, Angrave, Abdelzaher, Kravets, Gupta

10



History of Operating Systems



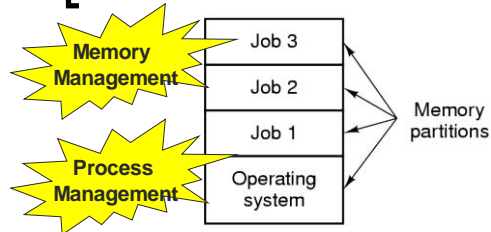
- Structure of a typical job
 - 2nd generation

Copyright ©: Nahstedt, Angrave, Abdelzaher, Kravets, Gupta

11



History of Operating Systems



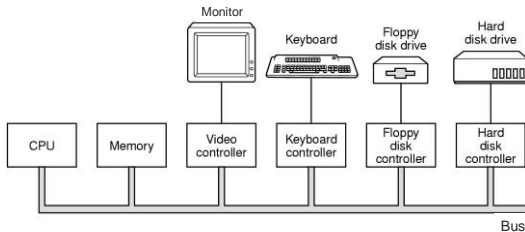
- Multiprogramming/timesharing system
 - Three jobs in memory – 3rd generation

Copyright ©: Nahstedt, Angrave, Abdelzaher, Kravets, Gupta

12



Computer Hardware Review



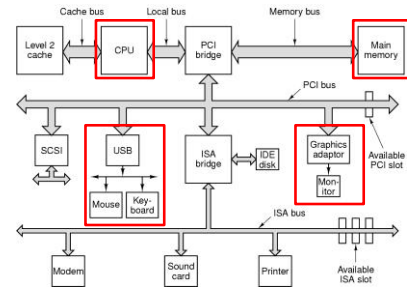
- Components of a simple personal computer

Copyright ©: Nahstedt, Angrave, Abdelzاهر, Kravets, Gupta

13



Early Pentium system

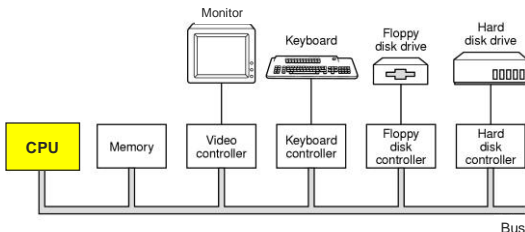


Copyright ©: Nahstedt, Angrave, Abdelzاهر, Kravets, Gupta

14



Computer Hardware Review



- Components of a simple personal computer

Copyright ©: Nahstedt, Angrave, Abdelzاهر, Kravets, Gupta

15



CPU , From CS231

- Fetch instruction from code memory
- Fetch operands from data memory
- Perform operation (and store result)
- (Check interrupt line)
- Go to next instruction
- 'Conventional CPU' (Ignore pipeline, optimization complexities)

Copyright ©: Nahstedt, Angrave, Abdelzاهر, Kravets, Gupta

16



CPU Registers

- Fetch instruction from code memory
- Fetch operands from data memory
- Perform operation (and store result)
- Go to next instruction

- Note: CPU must maintain certain state
 - Current instructions to fetch (program counter)
 - Location of code memory segment
 - Location of data memory segment

Copyright ©: Nahstedt, Angrave, Abdelzaher, Kravets, Gupta

17



CPU Register Examples

- Hold instruction operands
- Point to start of
 - Code segment
 - Data segment
 - Stack segment
- Point to current position of
 - Instruction pointer
 - Stack pointer

Copyright ©: Nahstedt, Angrave, Abdelzaher, Kravets, Gupta

18



CPU Register Examples

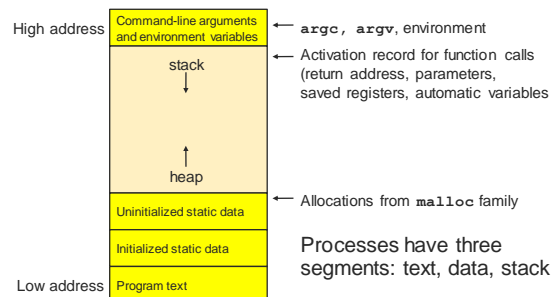
- Hold instruction operands
- Point to start of
 - Code segment
 - Data segment
 - Stack segment
- Point to current position of
 - Instruction pointer
 - Stack pointer
 - Why stack?

Copyright ©: Nahstedt, Angrave, Abdelzaher, Kravets, Gupta

19



Sample Layout for program image in main memory

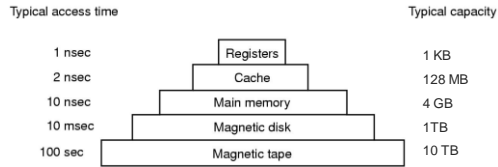


Copyright ©: Nahstedt, Angrave, Abdelzaher, Kravets, Gupta

20



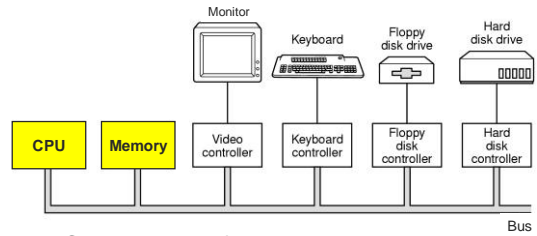
Memory Hierarchy



Copyright ©: Nahstedt, Angrave, Abdelzaher, Kravets, Gupta

21

Computer Hardware Review

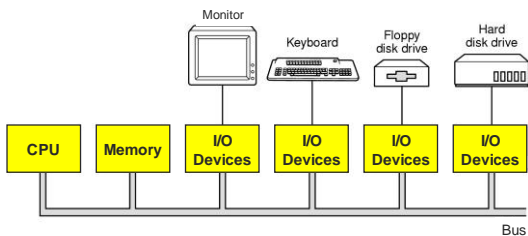


Components of a simple personal computer

Copyright ©: Nahstedt, Angrave, Abdelzaher, Kravets, Gupta

22

Computer Hardware Review

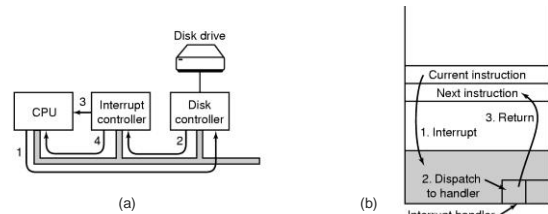


Components of a simple personal computer

Copyright ©: Nahstedt, Angrave, Abdelzaher, Kravets, Gupta

23

I/O Interrupt Mechanism



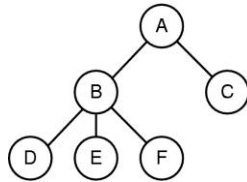
Steps in starting an I/O device and getting interrupt
How the CPU is interrupted

Copyright ©: Nahstedt, Angrave, Abdelzaher, Kravets, Gupta

24

Operating System Concepts

- Process
 - An executable instance of a program
 - Only one process can use the CPU at a time
- A process tree
 - A created two child processes, B and C
 - B created three child processes, D, E, and F



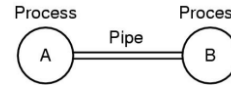
Copyright ©: Nahstedt, Angrave, Abdelzaher, Kravets, Gupta

25



Operating System Concepts

- Context Switching
 - How would you switch CPU execution from one process to another?
- Semaphores
 - Control access to resources
- Inter-process Communication
 - Two processes connected by a pipe

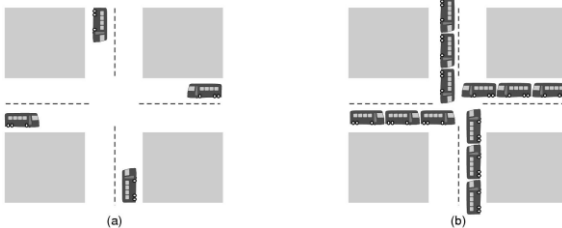


Copyright ©: Nahstedt, Angrave, Abdelzaher, Kravets, Gupta

26



Shared Resources, Conflicts, and Deadlocks



(a) A potential deadlock

(b) An actual deadlock

Copyright ©: Nahstedt, Angrave, Abdelzaher, Kravets, Gupta

27



System Times

Item	Time	Scaled Time in Human Terms (2 billion times slower)
Processor cycle	0.5 ns (2 GHz)	1 s
Cache access	1 ns (1 GHz)	2 s
Memory access	15 ns	30 s
Context switch	5,000 ns (5 micros)	167 m
Disk access	7,000,000 ns (7 ms)	162 days
System quanta	100,000,000 (100 ms)	6.3 years

Copyright ©: Nahstedt, Angrave, Abdelzaher, Kravets, Gupta

28



[Summary]

- Resource Manager
- Hardware independence
- Virtual Machine Interface
- POSIX
- Concurrency & Deadlock