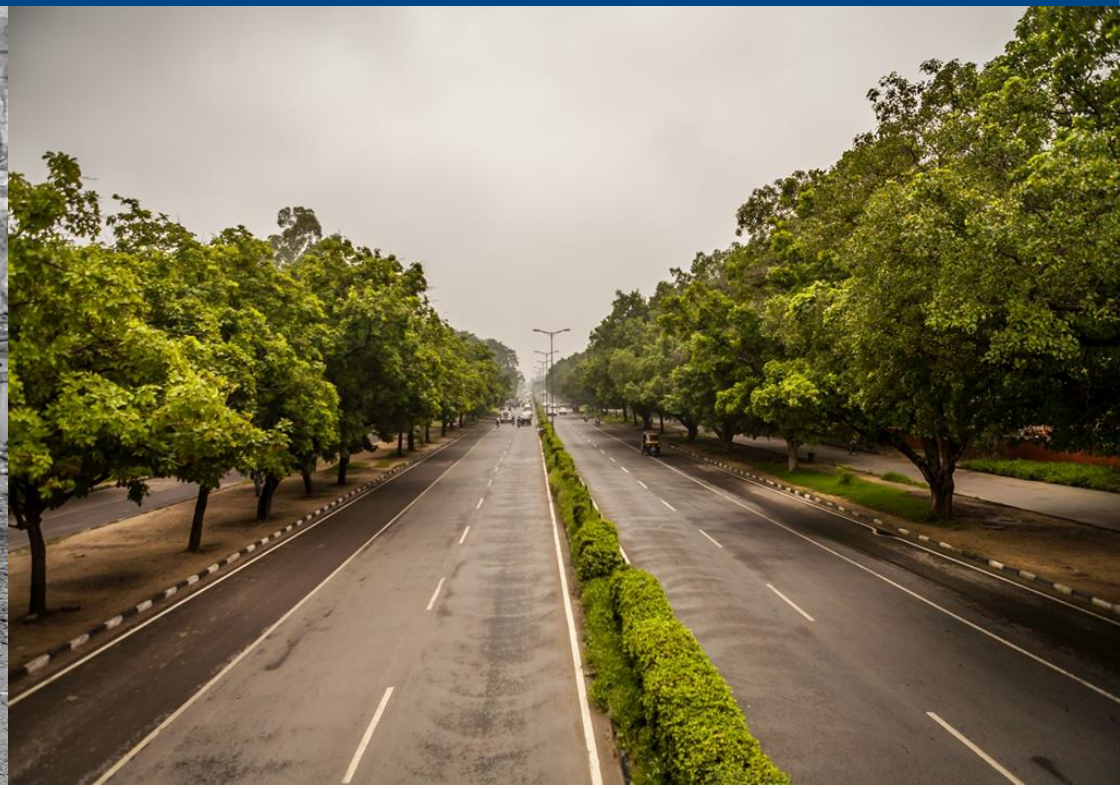
Operating Systems



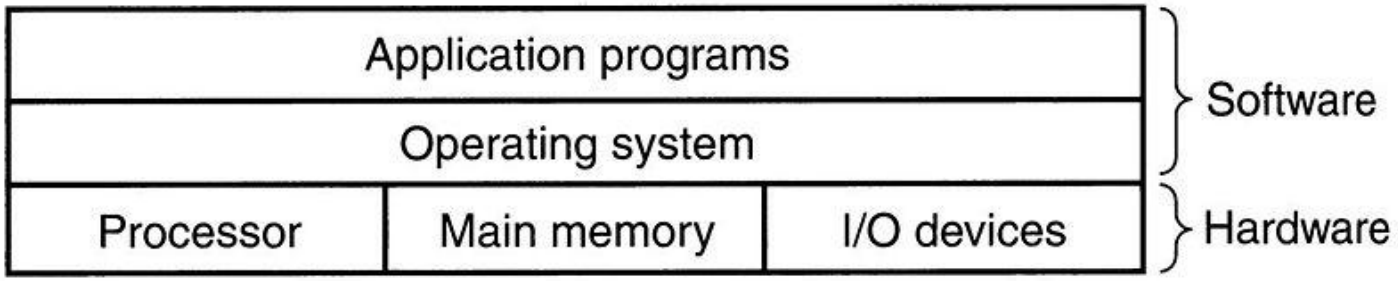
Introduction (Lecture-1)

Monsoon 2014, IIIT-H

Suresh Purini

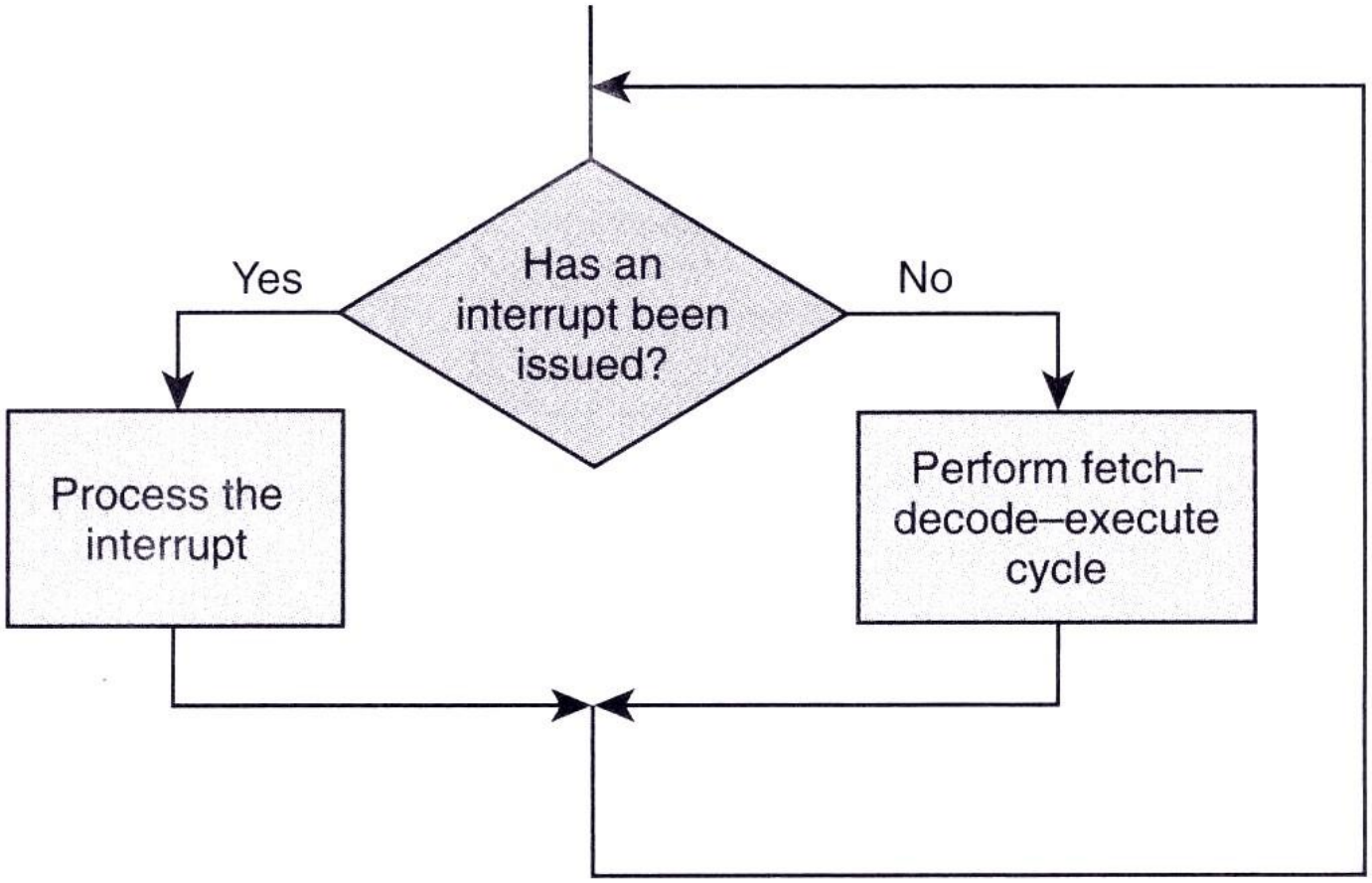
Bootstrapping (the course)

- What is an Operating System?
  - A program that acts as an intermediary between a user of a computer and the computer hardware.



  - Software to manage a computer's resources for its users and applications
- What purpose(s) does it serve?
- Examples: .....
- Trick Question: Are Fedora and Ubuntu different operating systems?

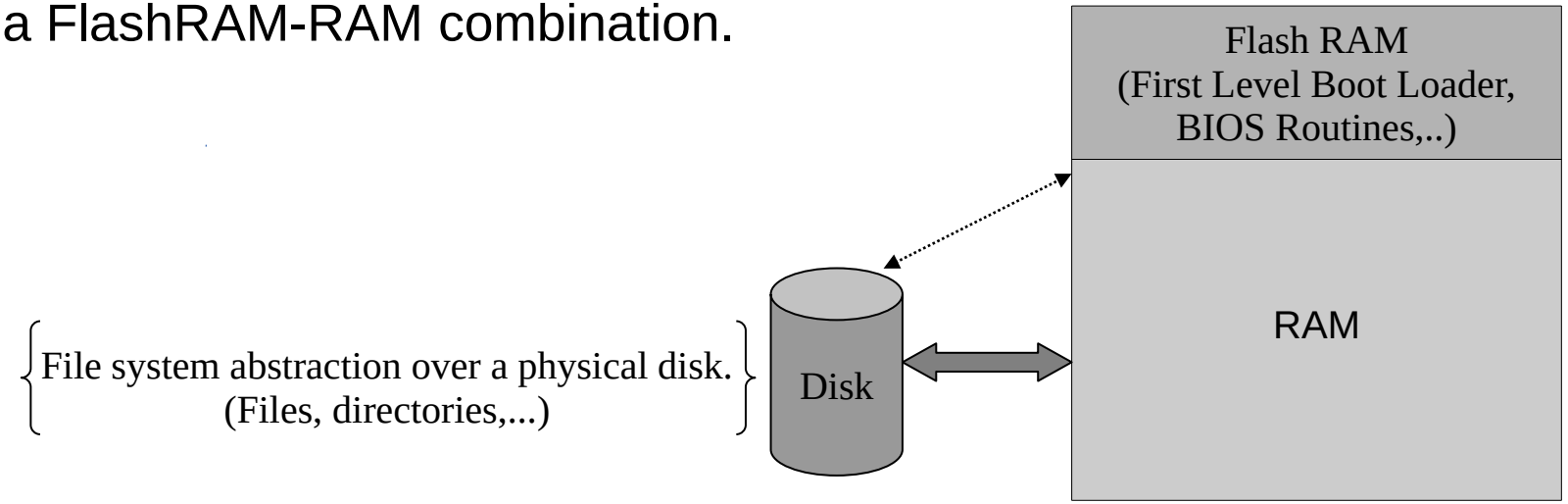
Basic Processor Model



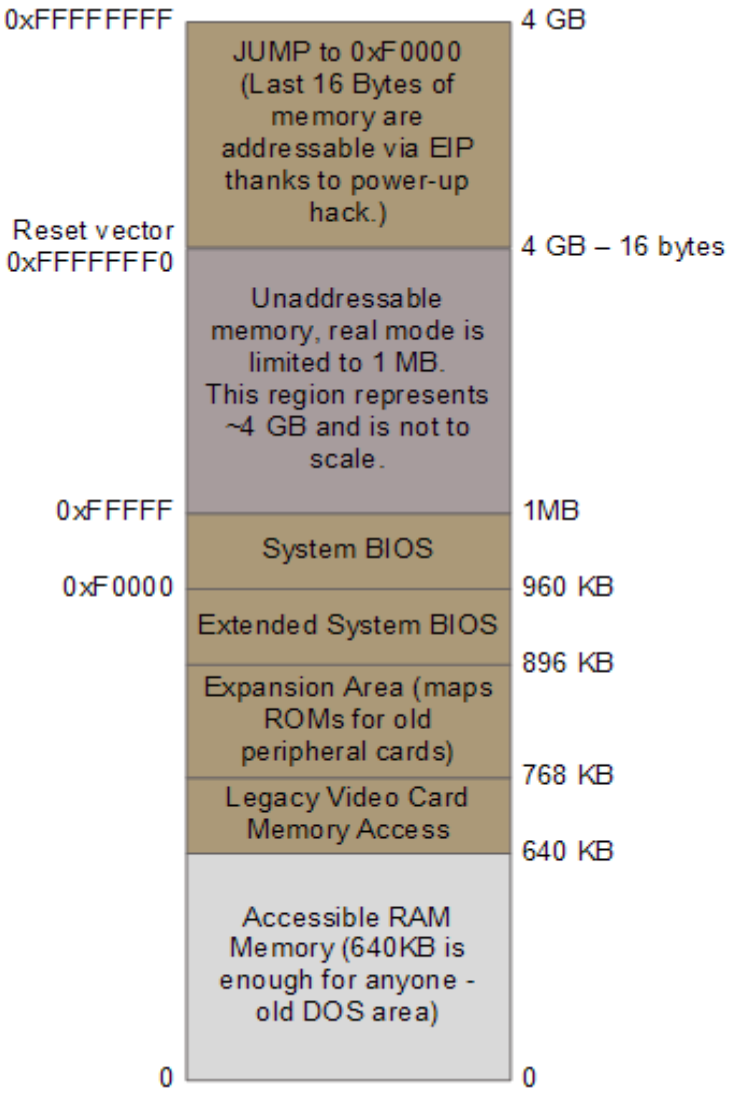
- Why is this an abstraction of the actual processor?
- Key Question: Given this processor abstraction, how do we build multi-user, multi-tasking and “whatever” operating systems such as Linux, Android, iOS.

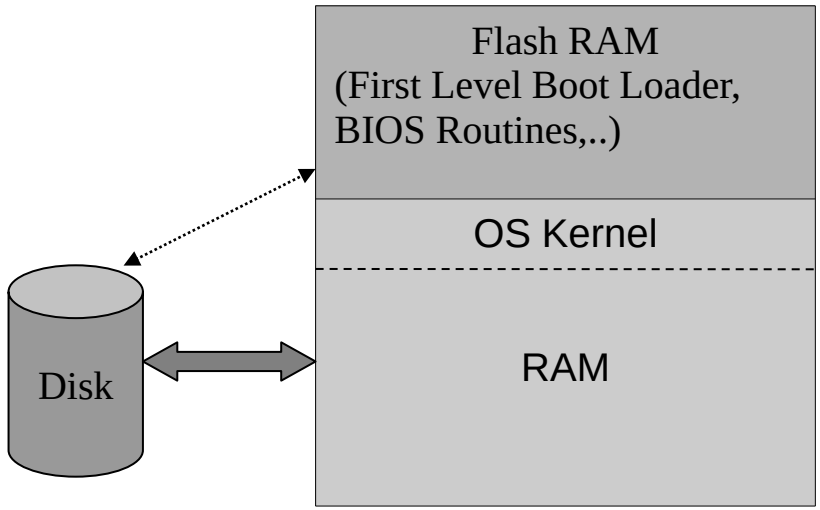
Bootstrapping Problem

- When we switch-on a processor
  1. PC value gets initialized to XXX (say zero).
  2. FE/DE/EXE cycle starts with the instruction at the address XXX.
- Issue: Who is going to place the desired instruction (rather the bootstrap program) at the XXX memory location?
- Idea 1: Use ROM. What's the problem?
- Idea 2: Use Flash RAM. What's the problem?
- Idea 3: Use a FlashRAM-RAM combination.



Bootstrapping in x86 Processors





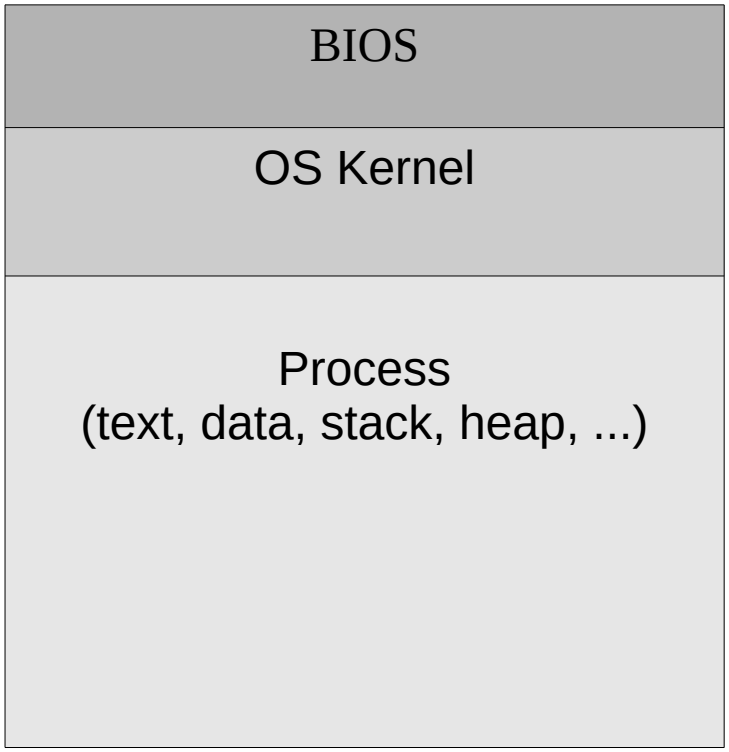
Reading Assignment

1. http://duartes.org/gustavo/blog/post/kernel-boot-process/
2. http://arjunsreedharan.org/post/82710718100/kernel-101-lets-write-a-kernel

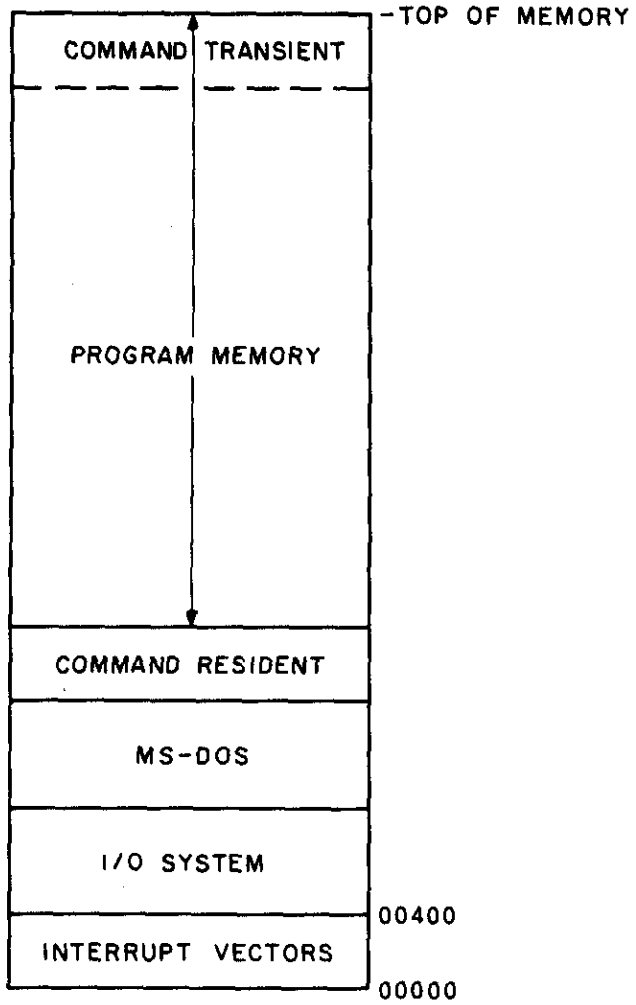
Single Tasking Operating System (MS-DOS)

- Definition: A program in execution is called a process (or a task).
- Single Tasking OS: Only one active process (residing in memory) at any time.

Memory Map



MS-DOS Memory Map



Issues with MS-DOS kind of Operating Systems

- Resource under utilization
  - CPU: ....
  - Memory: ...
- Lack of robustness due to poor or no protection mechanisms
  - How to isolate the kernel?
- What if the program is buggy and gets into an infinite loop?
- Are there any advantages?
  - You cannot use Facebook as you do your assignments.
  - Almost zero overhead (gaming!)
  - Hey, I want to execute single application anyways!

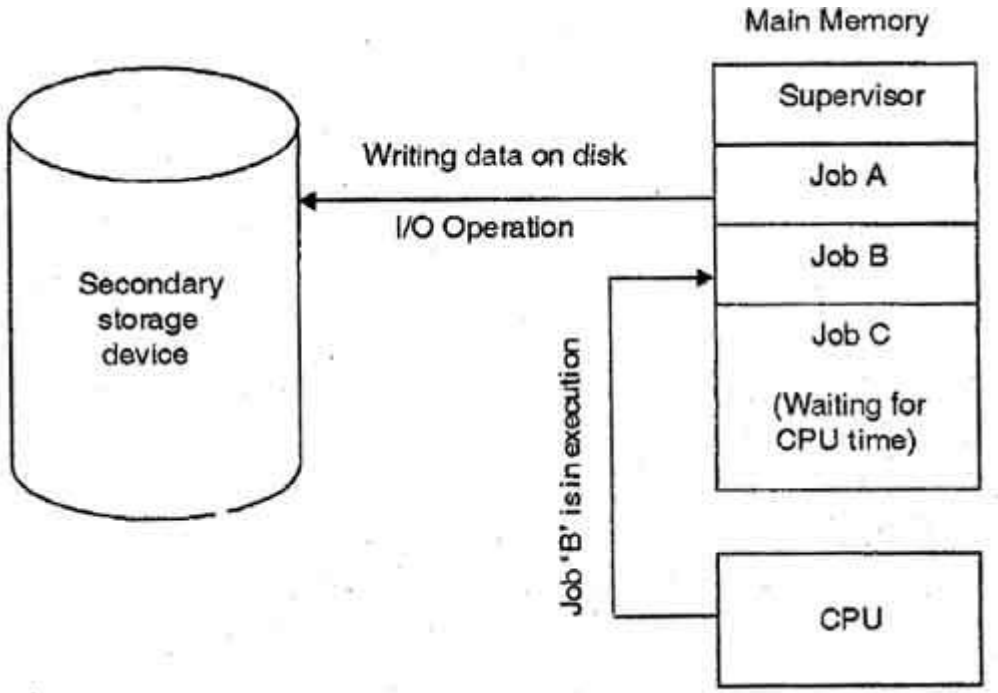
  [Check OSv (http://osv.io)]

So You Want to Fix MS-DOS!

- Kernel isolation (protection)
- Ability to terminate a program running forever.
- Better resource utilization
  - Do something else while
    - An user is typing his input on keyboard
    - Disk controller is reading a block of requested data
    - .....

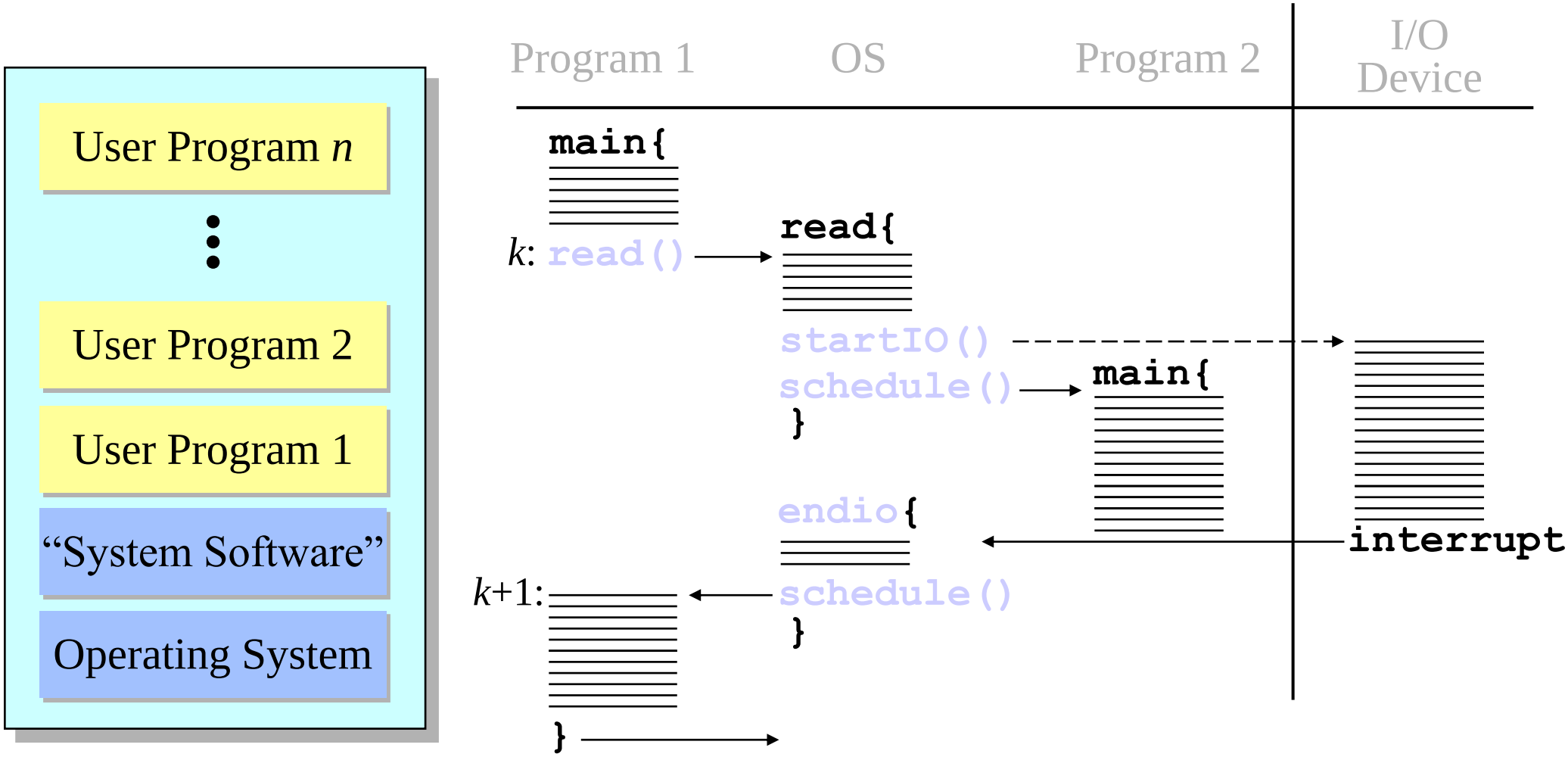
Multiprogrammed Operating Systems

- Idea: Switch between multiple jobs based on I/O requests.



Multiprogramming (‘65-‘80)

Keep several jobs in memory and multiplex CPU between jobs



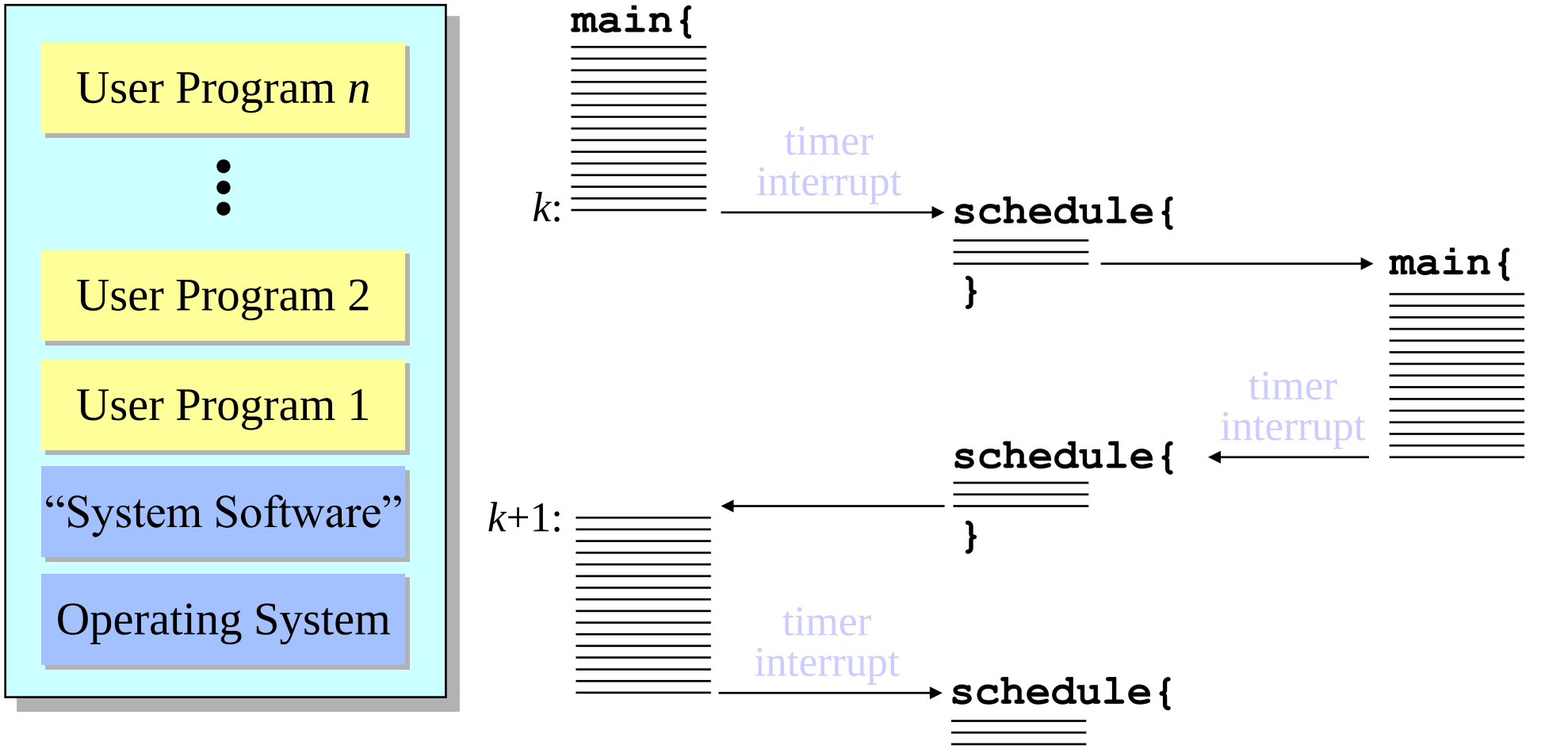
Multiprogramming

Issues

- Kernel isolation
- Program isolation
- Scheduling: Which job to pick next?
- Any issues with respect to memory?
- Question: Can you do FB and program simultaneously?

Timesharing Operating Systems (‘70-)

A timer interrupt is used to multiplex CPU among jobs



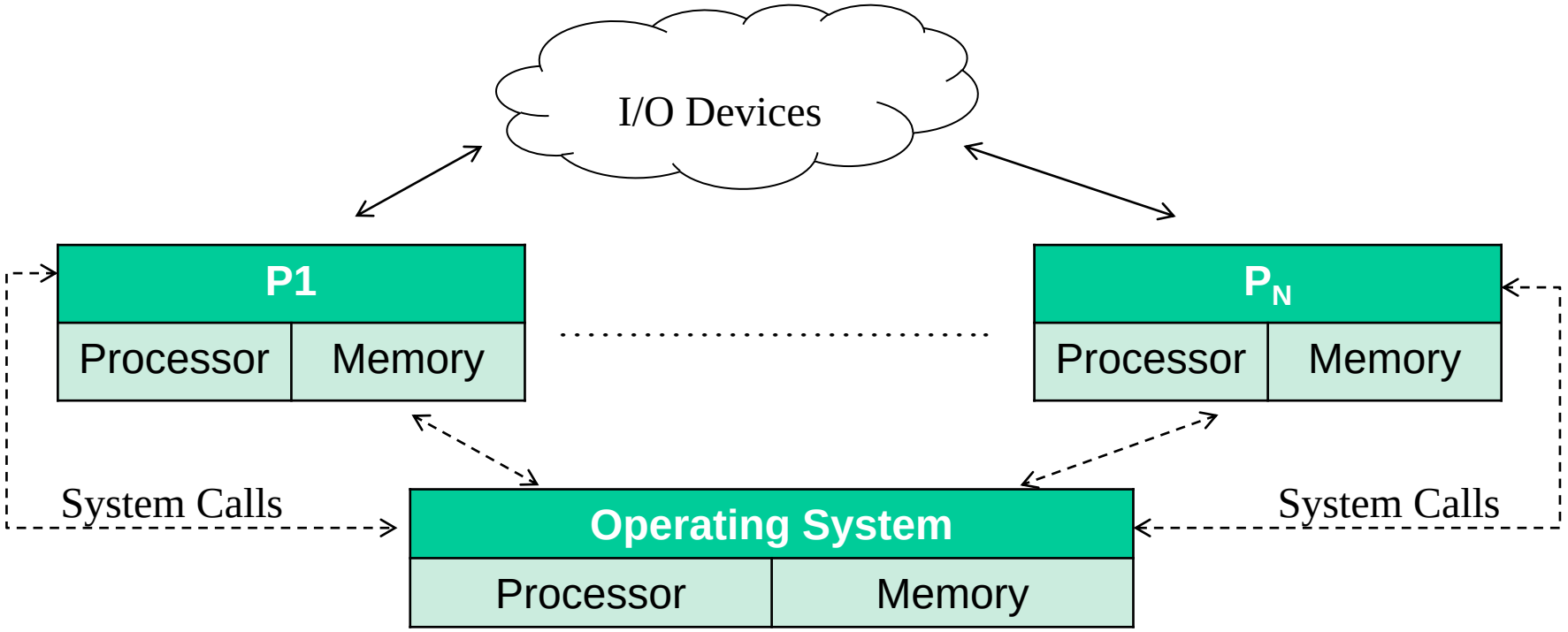
Time Sharing

Issues

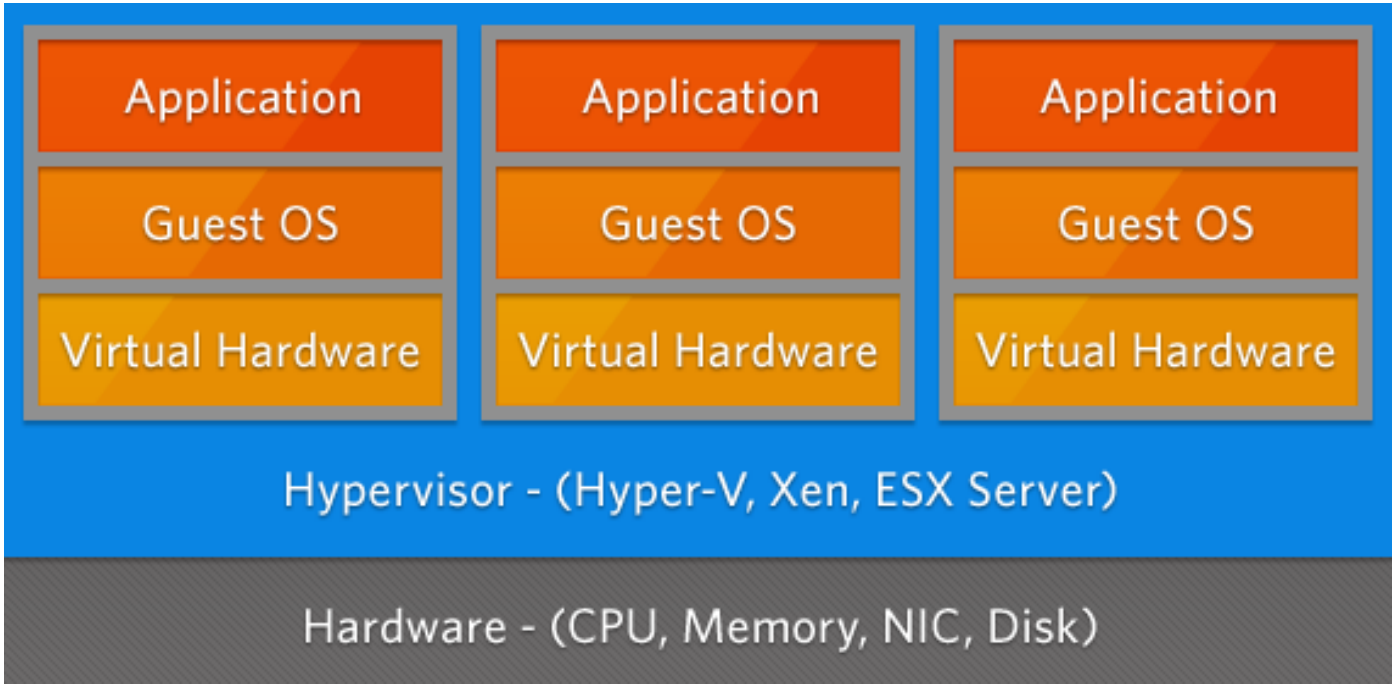
- Kernel isolation
- Program isolation
- Scheduling: Which job to pick next?
- Any issues with respect to memory?
- Context switching overheads
- Question: Can you do FB and program simultaneously?

Virtualization

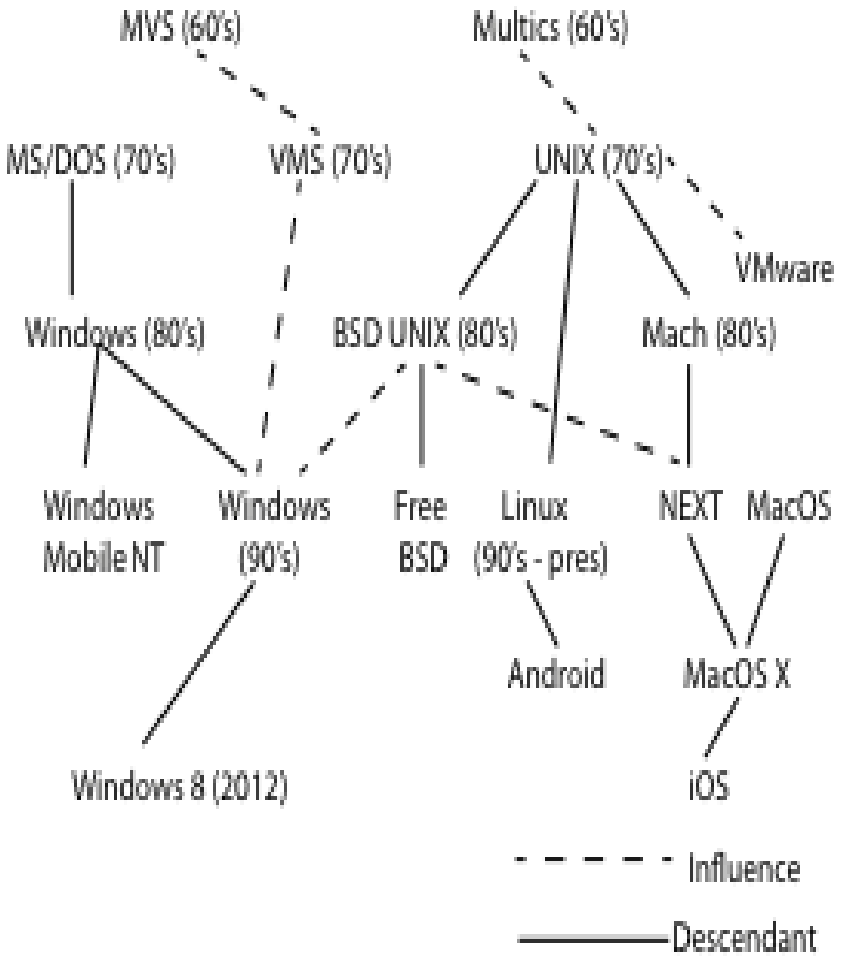
- Goal: Give every program a virtual PC
- Every process
  - Gets an illusion that it has an entire processor for itself (via multiplexing CPU over time)
  - Complete memory address space available (via Virtual Memory)
- However: Indirect interference between processes is possible



Virtualization



OS History


MVS (60's)
Multics (60's)
MS/DOS (70's)
VMS (70's)
UNIX (70's)
VMware
Windows (80's)
BSD UNIX (80's)
Mach (80's)
Windows Mobile NT
Windows (90's)
Free BSD
Linux (90's - pres)
NEXT
MacOS
Android
MacOS X
Windows 8 (2012)
iOS
Influence
Descendant