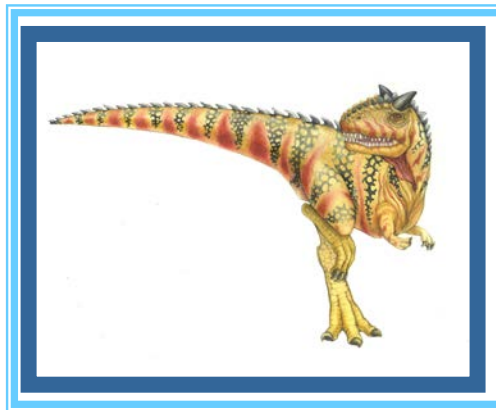# Chapter 3: Process-Concept

Process Concept

- An operating system executes a variety of programs:
  - Batch system – jobs
  - Time-shared systems – user programs or tasks
- Textbook uses the terms *job* and *process* almost interchangeably
- Process – a program in execution; process execution must progress in sequential fashion
- A process includes:
  - program counter
  - stack
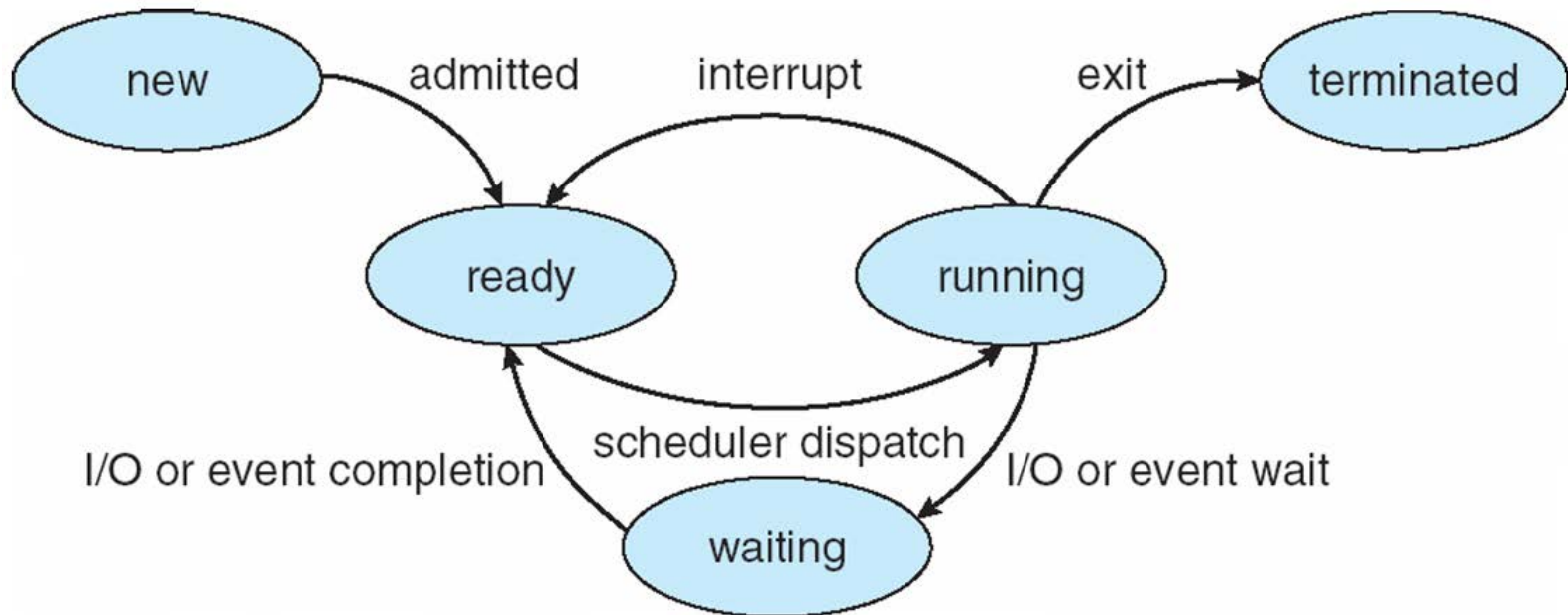 - data section

Process in Memory
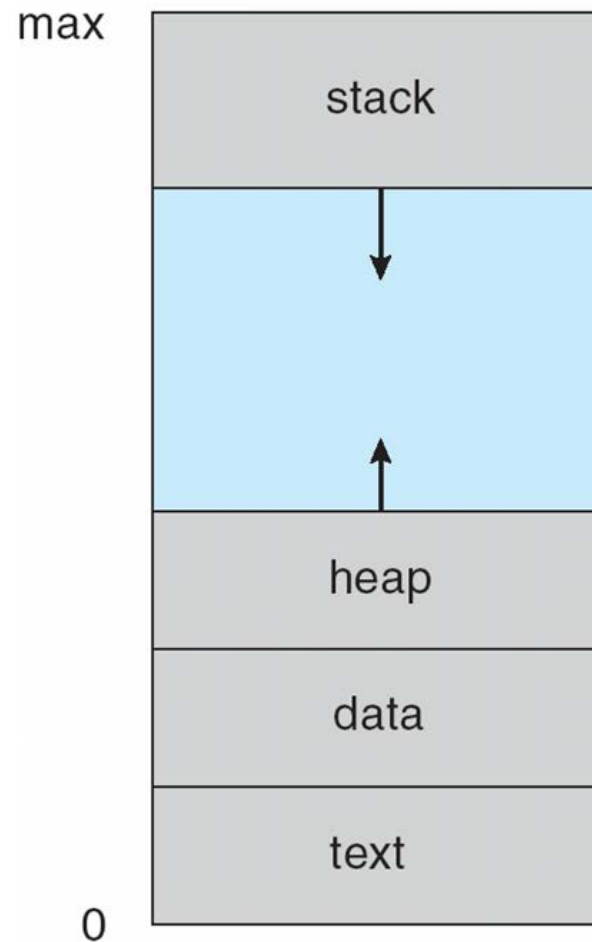
Process State

- As a process executes, it changes *state*
    - **new**: The process is being created
    - **running**: Instructions are being executed
    - **waiting**: The process is waiting for some event to occur
    - **ready**: The process is waiting to be assigned to a processor
    - **terminated**: The process has finished execution

Diagram of Process State

Process Control Block (PCB)

Information associated with each process

- Process state

- Program counter

- CPU registers

- CPU scheduling information

- Memory-management information

- Accounting information

- I/O status information

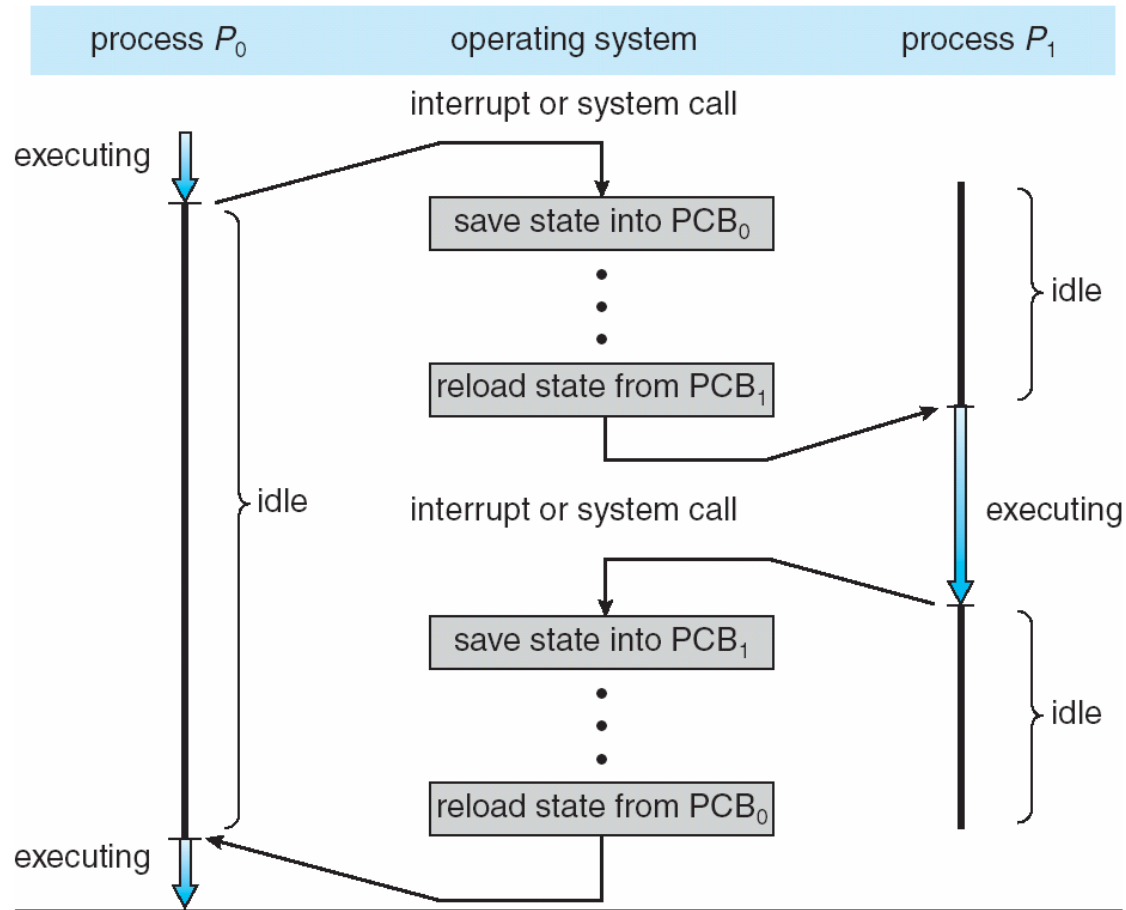# Process Control Block (PCB)

process state
process number
program counter
registers
memory limits
list of open files
• • •

process P_0 operating system   process P_1

interrupt or system call

executing

save state into PCB_0

⋮

reload state from PCB_1

idle

idle

interrupt or system call

executing

save state into PCB_1

⋮

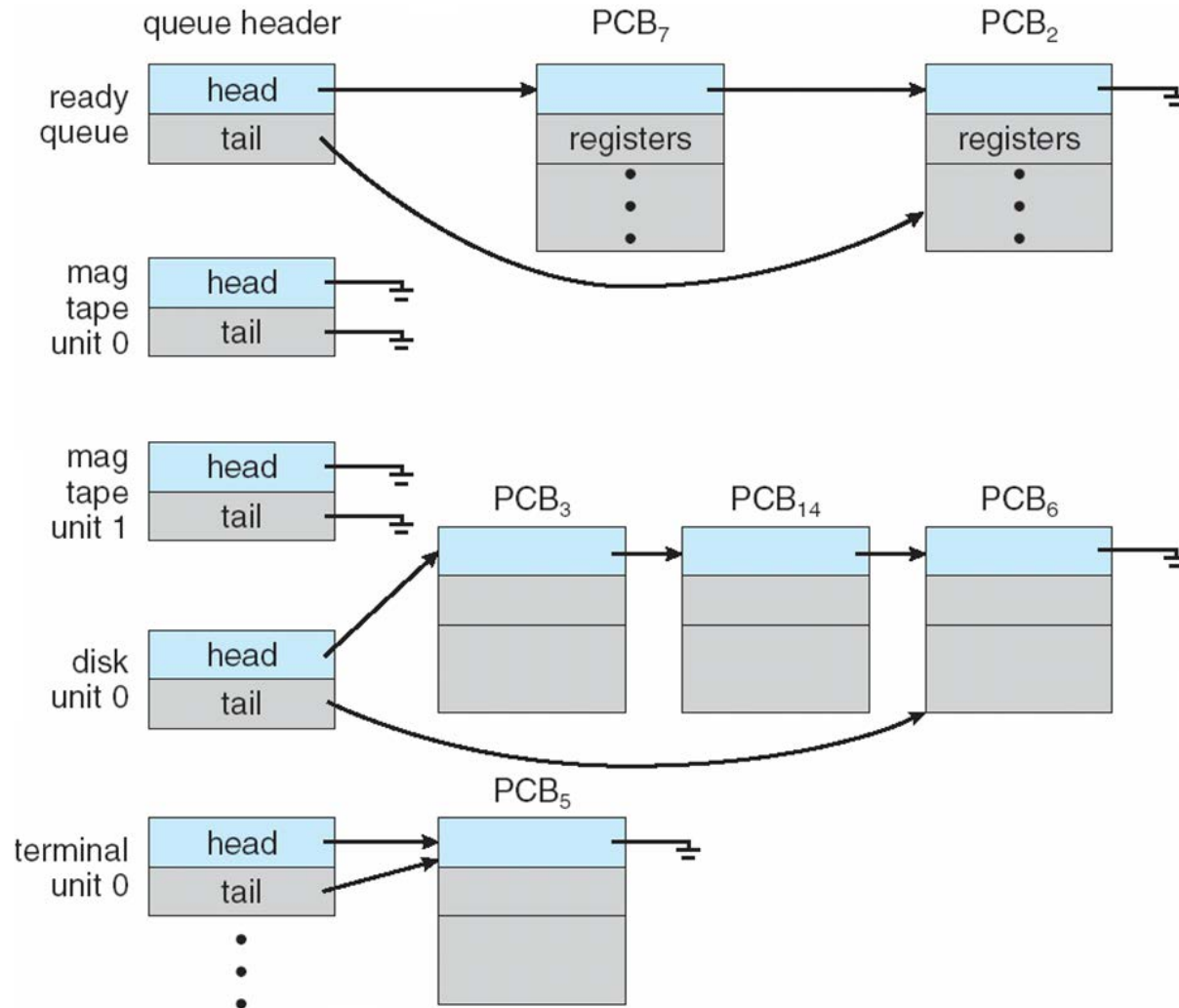reload state from PCB_0

idle

executing

Process Scheduling Queues

- **Job queue** – set of all processes in the system

- **Ready queue** – set of all processes residing in main memory, ready and waiting to execute

- **Device queues** – set of processes waiting for an I/O device
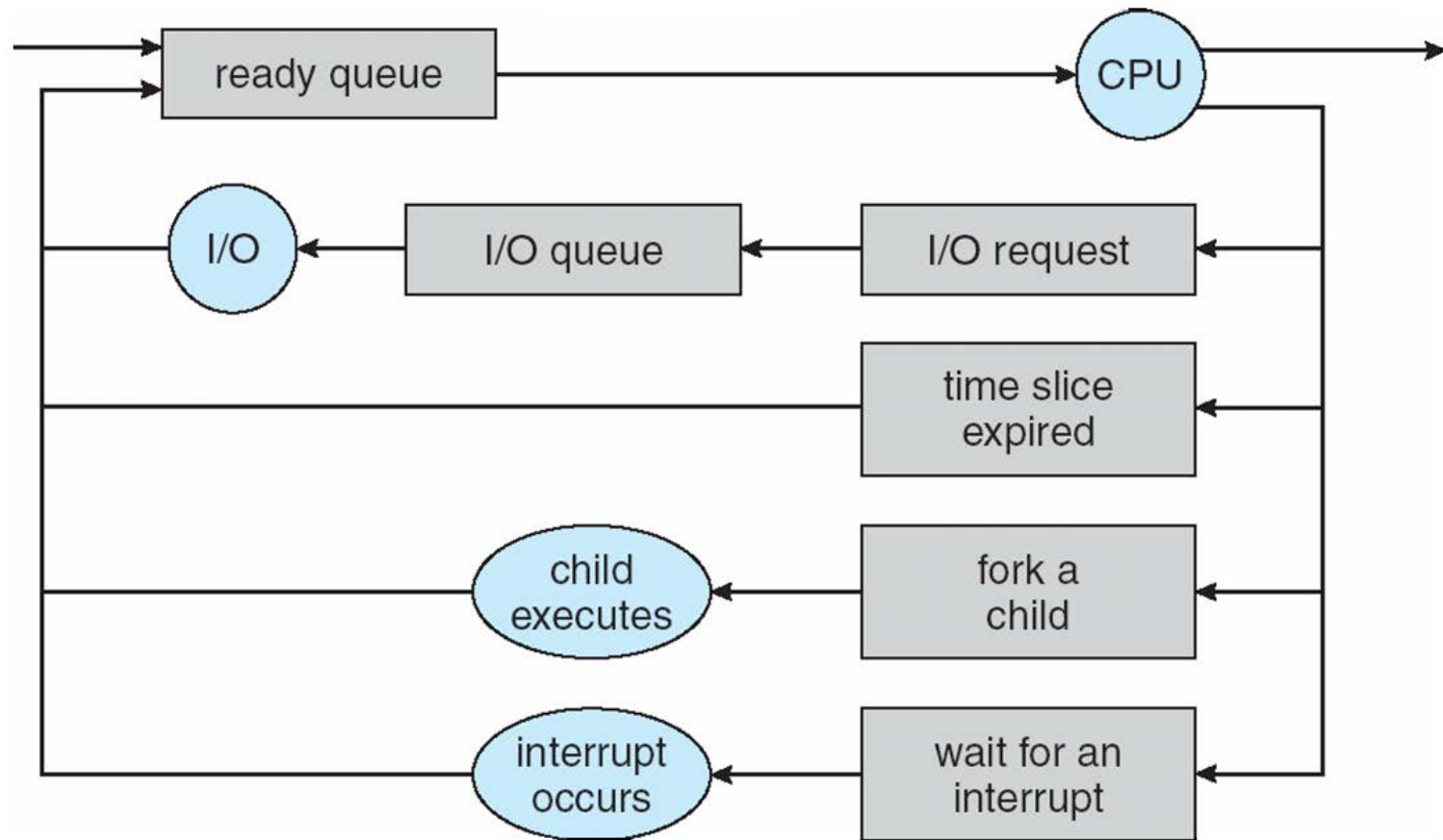
- Processes migrate among the various queues

Ready Queue And Various I/O Device Queues

Representation of Process Scheduling

Schedulers

- **Long-term scheduler** (or job scheduler) – selects which processes should be brought into the ready queue

- **Short-term scheduler** (or CPU scheduler) – selects which process should be executed next and allocates CPU

Schedulers (Cont)

- Short-term scheduler is invoked very frequently (milliseconds) $\Rightarrow$ (must be fast)

- Long-term scheduler is invoked very infrequently (seconds, minutes) $\Rightarrow$ (may be slow)

- The long-term scheduler controls the *degree of multiprogramming*

- Processes can be described as either:

  - **I/O-bound process** – spends more time doing I/O than computations, many short CPU bursts

  - **CPU-bound process** – spends more time doing computations; few very long CPU bursts

Context Switch

- When CPU switches to another process, the system must save the state of the old process and load the saved state for the new process via a context switch

- Context of a process represented in the PCB

- Context-switch time is overhead; the system does no useful work while switching

- Time dependent on hardware support