Scheduling

Content of This Lecture

- Why CPU scheduling?
- Basic scheduling algorithms
  - FIFO (FCFS)
  - Shortest job first
  - Round Robin
  - Priority Scheduling
- Goals:
  - Understand how your program is executed on the machine together with other programs

Copyright ©: Nahrstedt, Angrave, Abdelzaher

Review: State Process Model

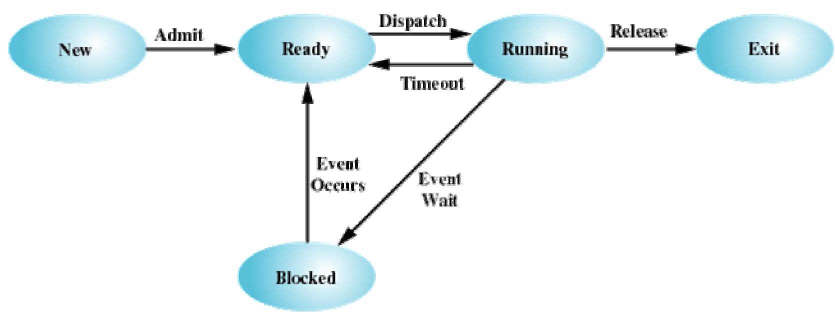


Figure 3.6 Five-State Process Model

3

Copyright ©: Nahrstedt, Angrave, Abdelzaher

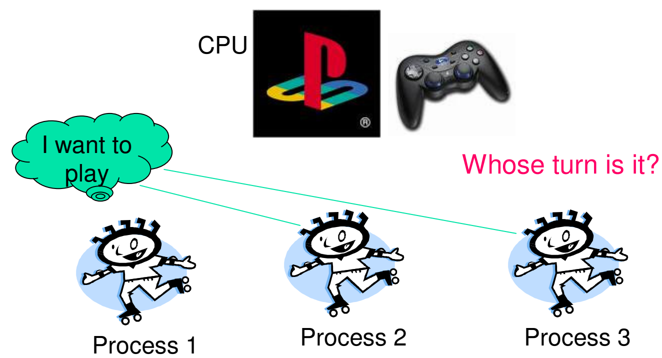
OS Representation of Process via Process Control Block (PCB)

Process management	Memory management	File management
Registers Program counter Program status word Stack pointer Process state Priority Scheduling parameters Process ID Parent process Process group Signals Time when process started CPU time used Children's CPU time Time of next alarm	Pointer to text segment Pointer to data segment Pointer to stack segment	Root directory Working directory File descriptors User ID Group ID

4

Process Scheduling

- Deciding which process/thread should occupy the resource (CPU, disk, etc)



5

Context Switch

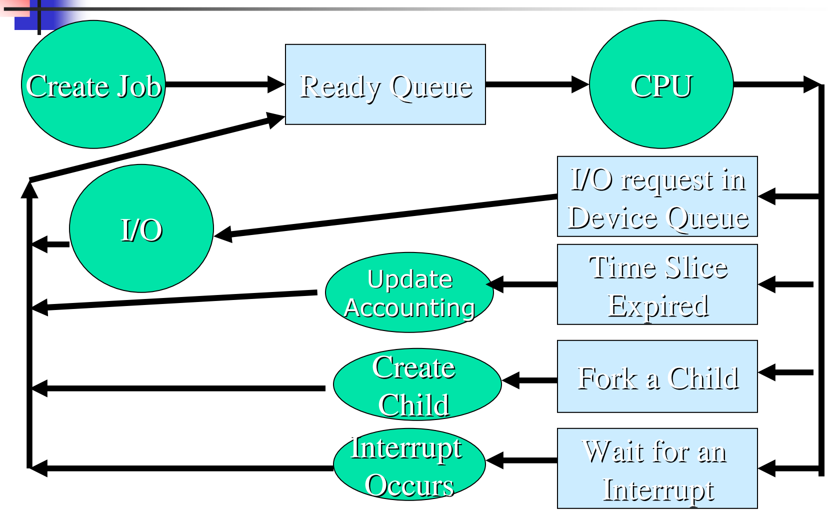
- Switch CPU from one process to another
- Performed by scheduler
- It includes:
  - save PCB state of the old process;
  - load PCB state of the new process;
  - Flush memory cache;
  - Change memory mapping (TLB);
- Context switch is expensive (1-1000 microseconds)
  - No useful work is done (pure overhead)
  - Can become a bottleneck
- Need hardware support

6

When to schedule?

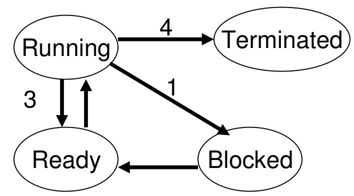
- A new process starts
- The running process exits
- The running process is blocked
- I/O interrupt (some processes will be ready)
- Clock interrupt (every 10 milliseconds)

Queuing Diagram for Processes



Preemptive vs. Non-preemptive scheduling

- **Non-preemptive scheduling:**
  - The running process keeps the CPU until it voluntarily gives up the CPU
    - process exits
    - switches to blocked state
    - 1 and 4 only (no 3)



- **Preemptive scheduling**:
  - The running process can be interrupted and must release the CPU (can be forced to give up CPU)

Scheduling Objectives

- Fairness (equitable shares of CPU)
- Priority (most important first)
- Efficiency (make best use of equipment)
- Encouraging good behavior (can't take advantage of the system)
- Support for heavy loads (degrade gracefully)
- Adapting to different environments (interactive, real-time, multi-media)

Performance Criteria

- Fairness
- Efficiency: keep resources as busy as possible
- Throughput: # of processes that completes in unit time
- Turnaround Time (also called elapse time)
  - amount of time to execute a particular process from the time its entered
- Waiting Time
  - amount of time process has been waiting in ready queue
- Response Time
  - amount of time from when a request was first submitted until first response is produced.
  - predictability and variance
- Proportionality:
  - meet users' expectation
- Meeting Deadlines: avoid losing data

Process Profiles

- I/O – Bound
  - Does too much I/O to keep CPU busy
- CPU – Bound
  - Does too much computation to keep I/O busy
- Process Mix
  - Scheduling should load balance between I/O bound and CPU-bound processes
  - Ideal would be to run all equipment at 100% utilization but that would not necessarily be good for response time

Simple Processor Scheduling Algorithms

- Batch systems
  - First Come First Serve (FCFS)
  - Shortest Job First
- Interactive Systems
  - Round Robin
  - Priority Scheduling
  - …

First Come First Serve (FCFS)

- Process that requests the CPU FIRST is allocated the CPU FIRST.
- Also called FIFO
- Preemptive or Non-preemptive?
- Used in Batch Systems
- Implementation
  - FIFO queues
  - A new process enters the tail of the queue
  - The schedule selects from the head of the queue.

FCFS Example

Process	Duration	Order	Arrival Time
P1	24	1	0
P2	3	2	0
P3	4	3	0

The final schedule:

P1 waiting time: 0
P2 waiting time: 24
P3 waiting time: 27

The average waiting time:
(0+24+27)/3 = 17

What if P1 arrives at time 2

Problems with FCFS

- Non-preemptive
- Not optimal AWT
- Cannot utilize resources in parallel:
  - Assume 1 process CPU bounded and many I/O bounded processes
  - result: Convoy effect, low CPU and I/O Device utilization
  - Why?

Summary

- Why Scheduling?
- Scheduling objectives
- Scheduling Algorithms
  - FCFS (FIFO)