

COA

Chapter 08: | OS Support – Scheduling

Prof. Dr. Mostafa Elhosseini

<https://youtube.com/drmelhosseini>

| AGENDA

- Scheduling

| Scheduling

- The key to multiprogramming is **scheduling**.
- In fact, **four types of scheduling** are typically involved
 - Long term
 - Medium term
 - Short term
 - I/O

| Process

- A program in execution
- The “animated spirit” of a program
- That entity to which a processor is assigned

| Long-Term Scheduling - Admission

- Determines which programs are admitted to the system for processing
- Controls the degree of multiprogramming (number of processes in memory)
- Once admitted, a **job or user program becomes a process** and is added to the queue for the short-term scheduler

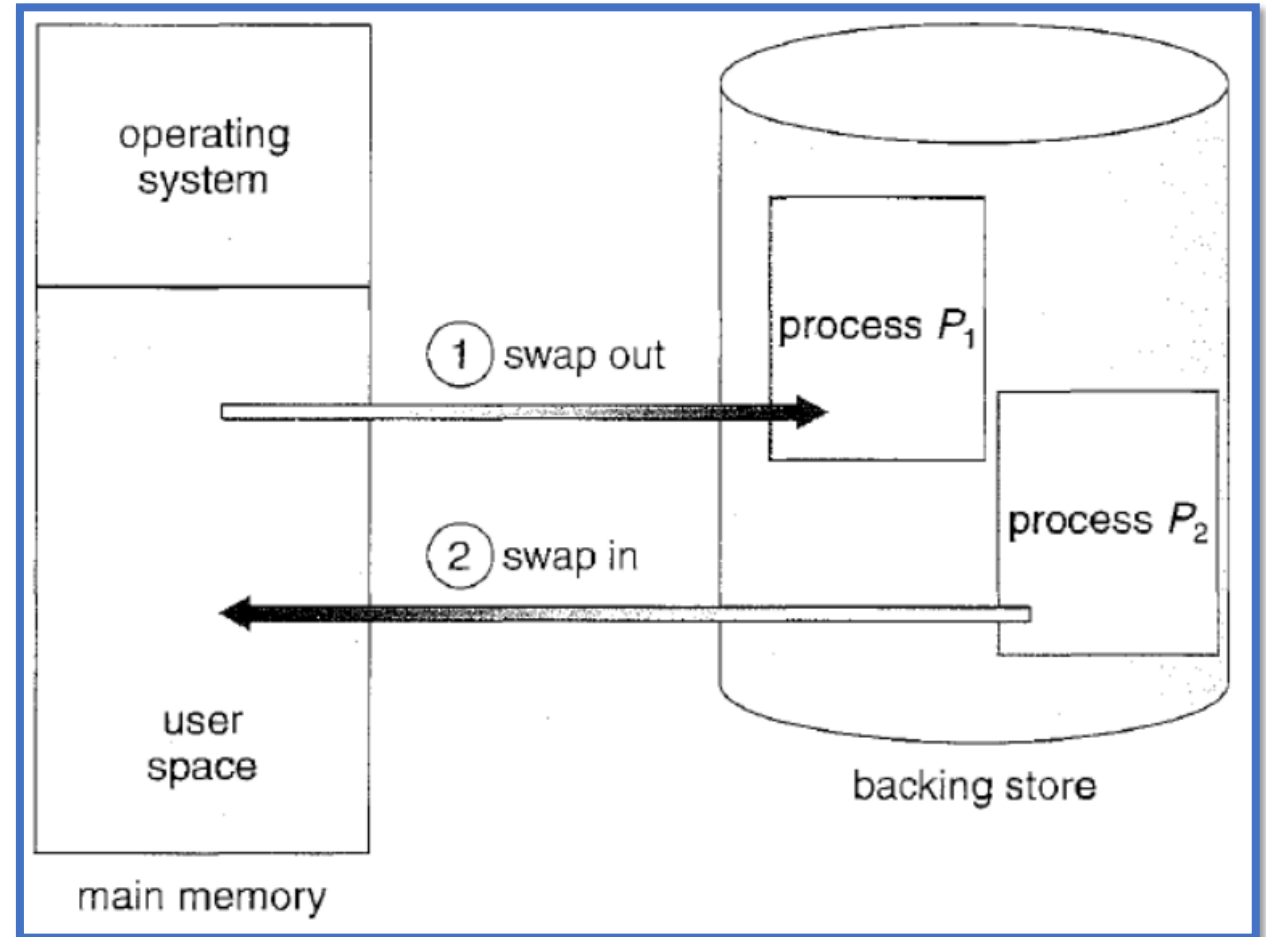
| Medium-Term Scheduling

A **process** can be swapped temporarily out of memory to a backing store and then **brought back** into memory for continued execution

- Part of the swapping function
- Swapping-in decision is based on the need to manage the degree of multiprogramming
- Swapping-in decision will consider the memory requirements of the swapped-out processes

| Medium-Term Scheduling

- Multiprogramming environment with a **round-robin CPU-scheduling** algorithm.
- Swapping requires a backing store. The backing store is commonly a fast disk

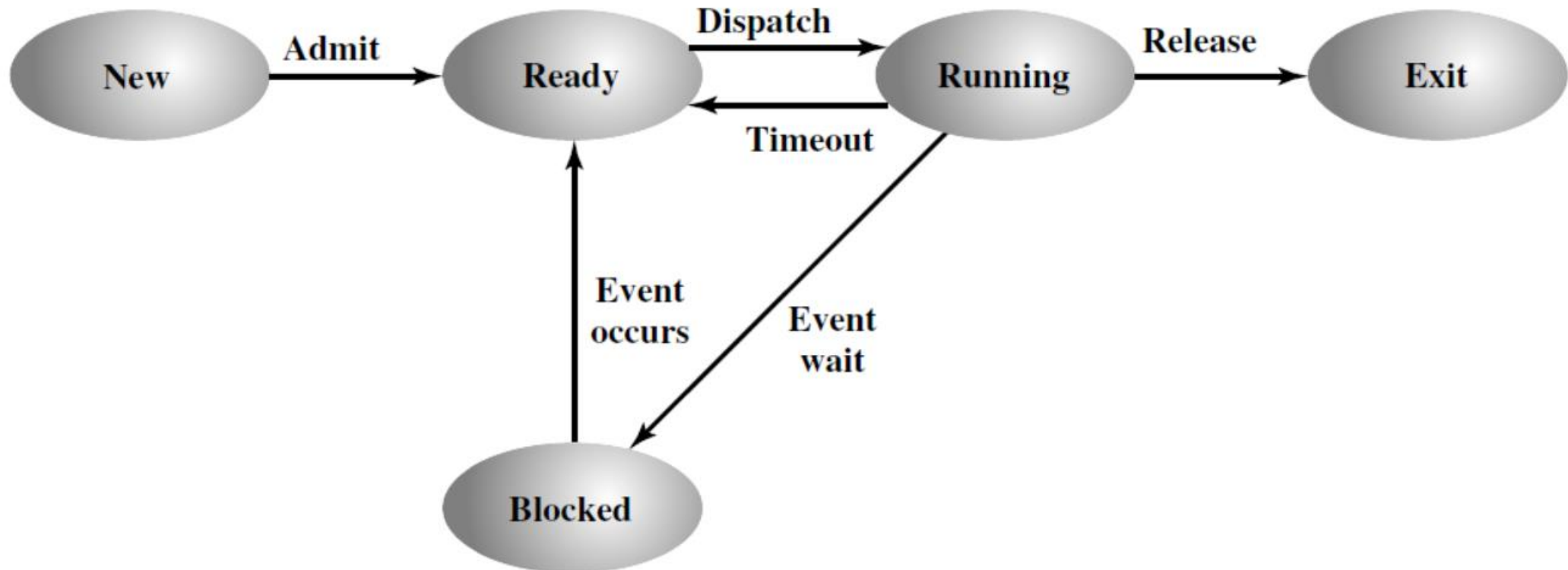


| Short-Term Scheduling

- The **long-term scheduler** executes relatively **infrequently**
 - Makes the decision of **whether or not to take on a new process**, and which one to take
- The **short-term scheduler**, also known as the **dispatcher**, executes **frequently** and makes the decision of **which job to execute next**

| Process states

- During the **lifetime** of a process, its **status will change** a number of times. Its status at any point in time is referred to as a **state**



| Process states

- **New:** A program is **admitted** by the high-level scheduler but is **not yet ready** to execute. The OS will initialize the process, moving it to the ready state.
- **Ready:** The process is ready to execute and is awaiting access to the processor.
- **Running:** The process is being executed by the processor.
- **Waiting:** The process is **suspended** from execution waiting for some system resource, such as I/O.
- **Halted:** The process has **terminated** and will be destroyed by the OS.

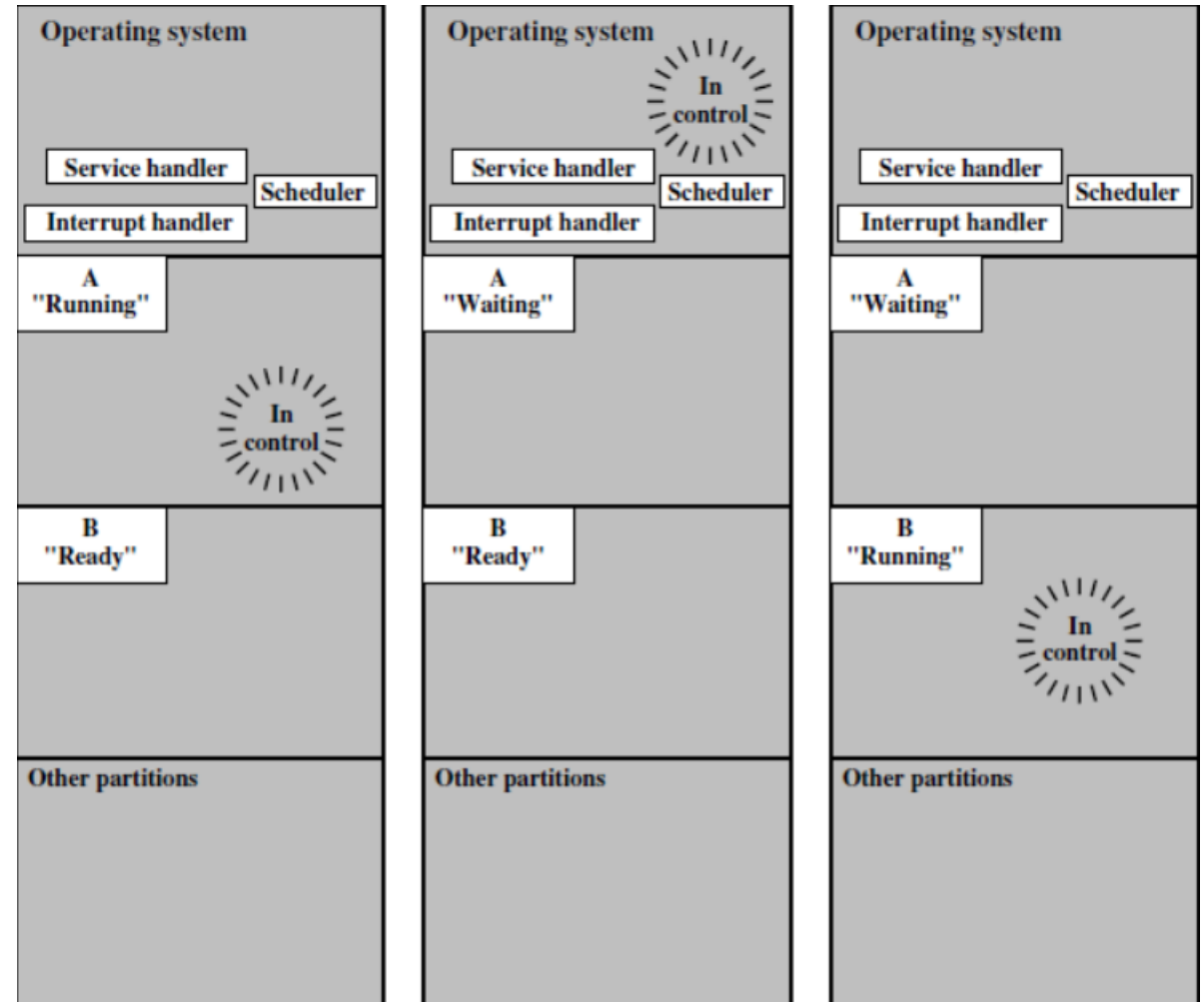
| Process Control Block

For each process in the system, the OS must maintain information indicating the state of the process and other information necessary for process execution

- Identifier – State – Priority - Program counter -
- **Memory pointers:** The starting and ending locations of the process in memory
- **Context data:** Data that are present in registers in the processor while the process is executing
- I/O status
- **Accounting information:** include the amount of processor time and clock time used, time limits, account numbers

| Scheduling Techniques

- The processor **ceases** to execute instructions in A and **begins** executing instructions in the **OS area**.
- This will happen for one of three reasons

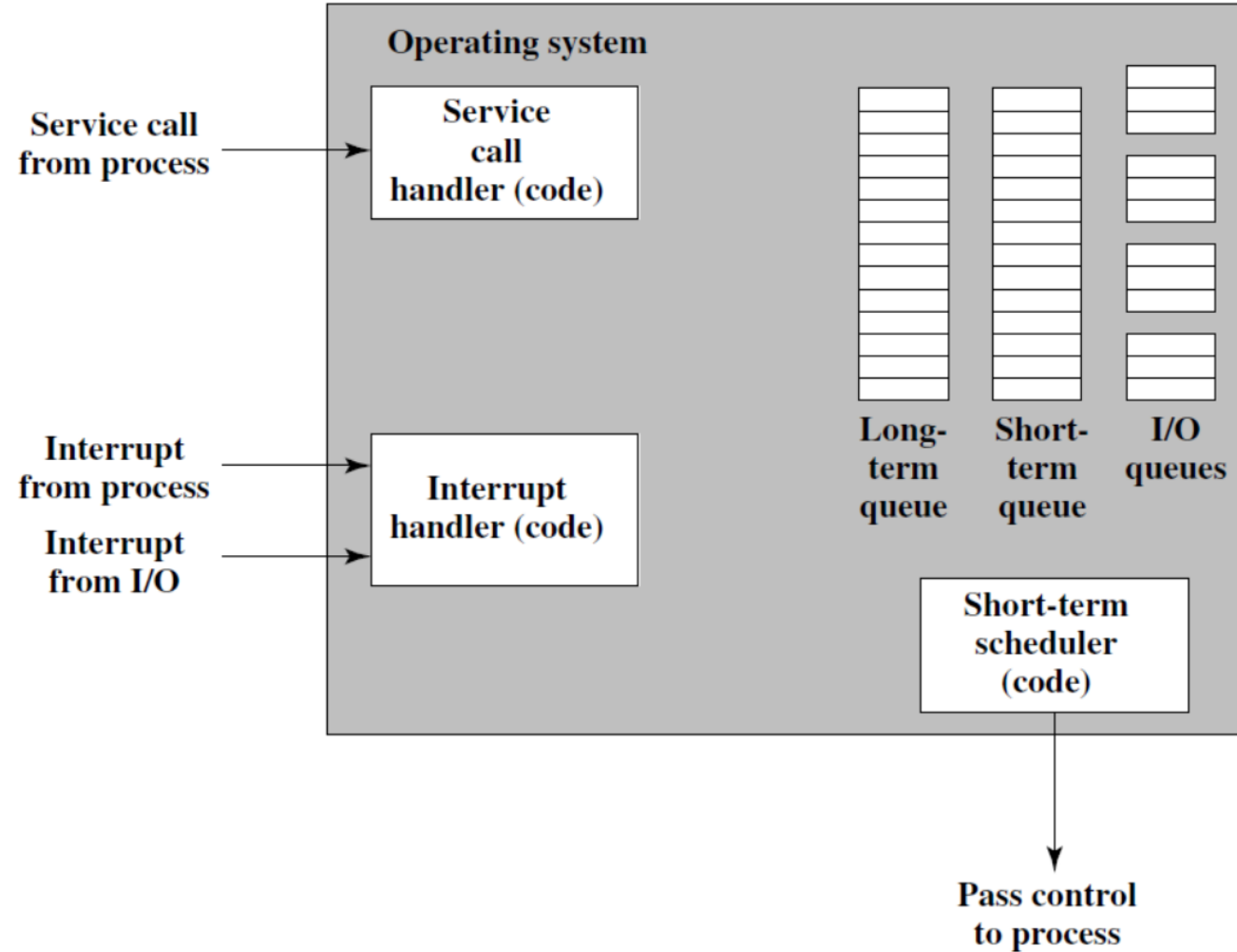


| Scheduling Techniques

This will happen for one of three reasons:

- Process A issues a **service call** (e.g., an I/O request) to the OS
- Process A causes **an interrupt**.
 - An interrupt is a hardware-generated signal to the processor.
 - When this signal is detected, the processor ceases to execute A and transfers to the interrupt handler in the OS
 - Ex. attempting to execute a privileged instruction.
 - Ex. Timeout
- Some event unrelated to process A that requires attention causes an **interrupt**
 - completion of an I/O operation

| Key elements of OS



| Process Scheduling

