

Operating Systems

- Basic Functions:
 - Schedule programs for execution
 - Coordinate the execution of programs
 - Store and retrieve files

Evolution of Shared Computing

- Batch processing
- Interactive processing
 - Requires real-time processing
- Time-sharing/Multitasking
 - Implemented by Multiprogramming
- Multiprocessor machines

Figure 3.1 Batch processing

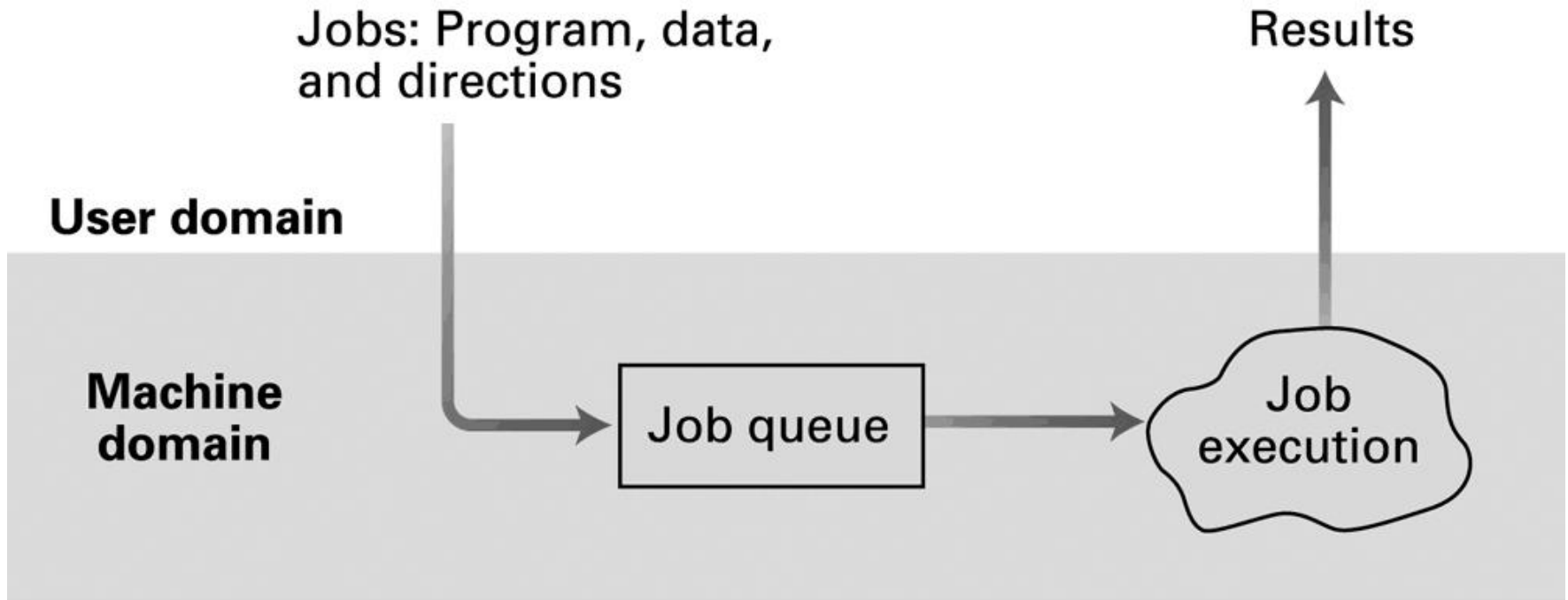
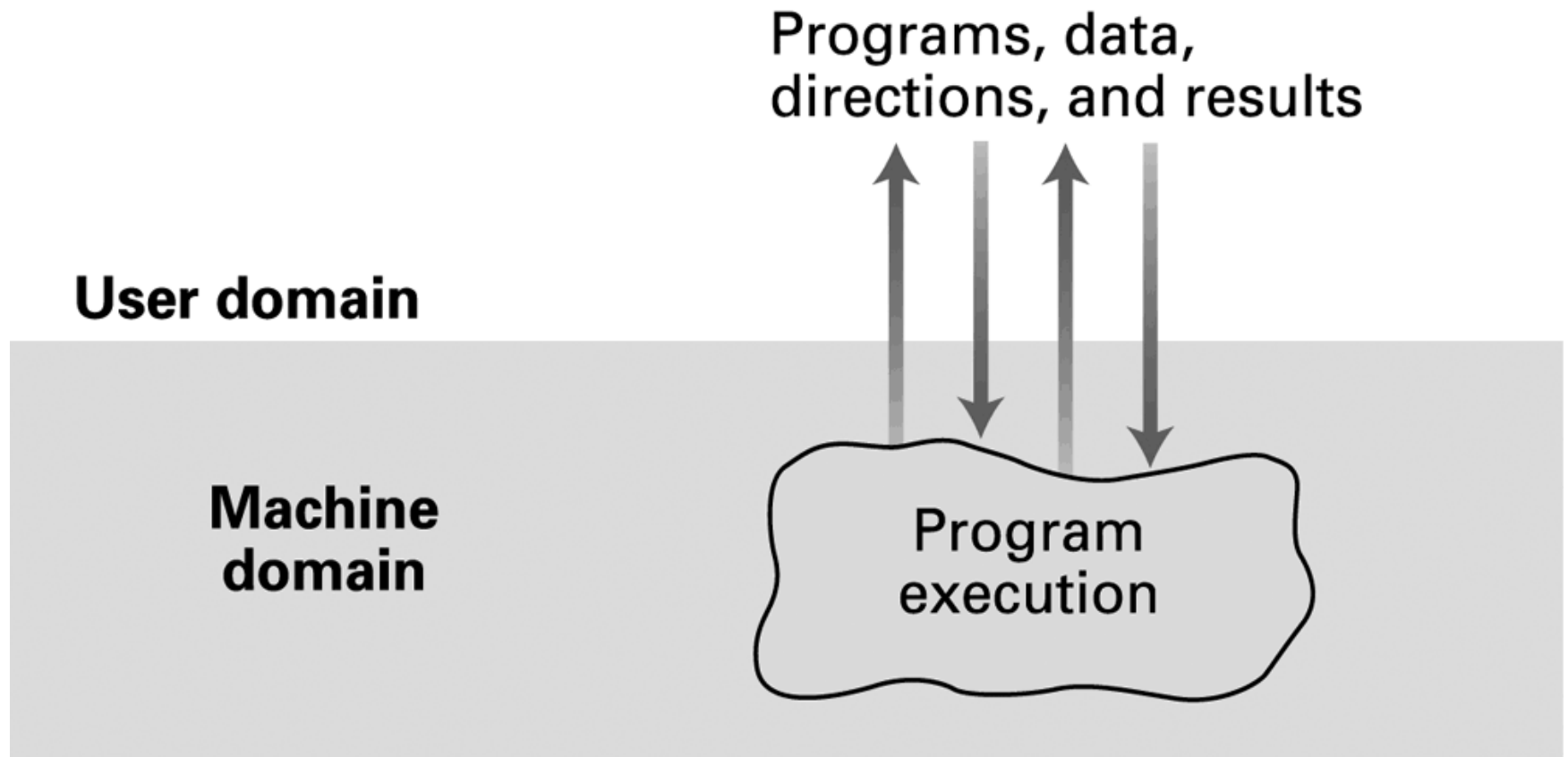


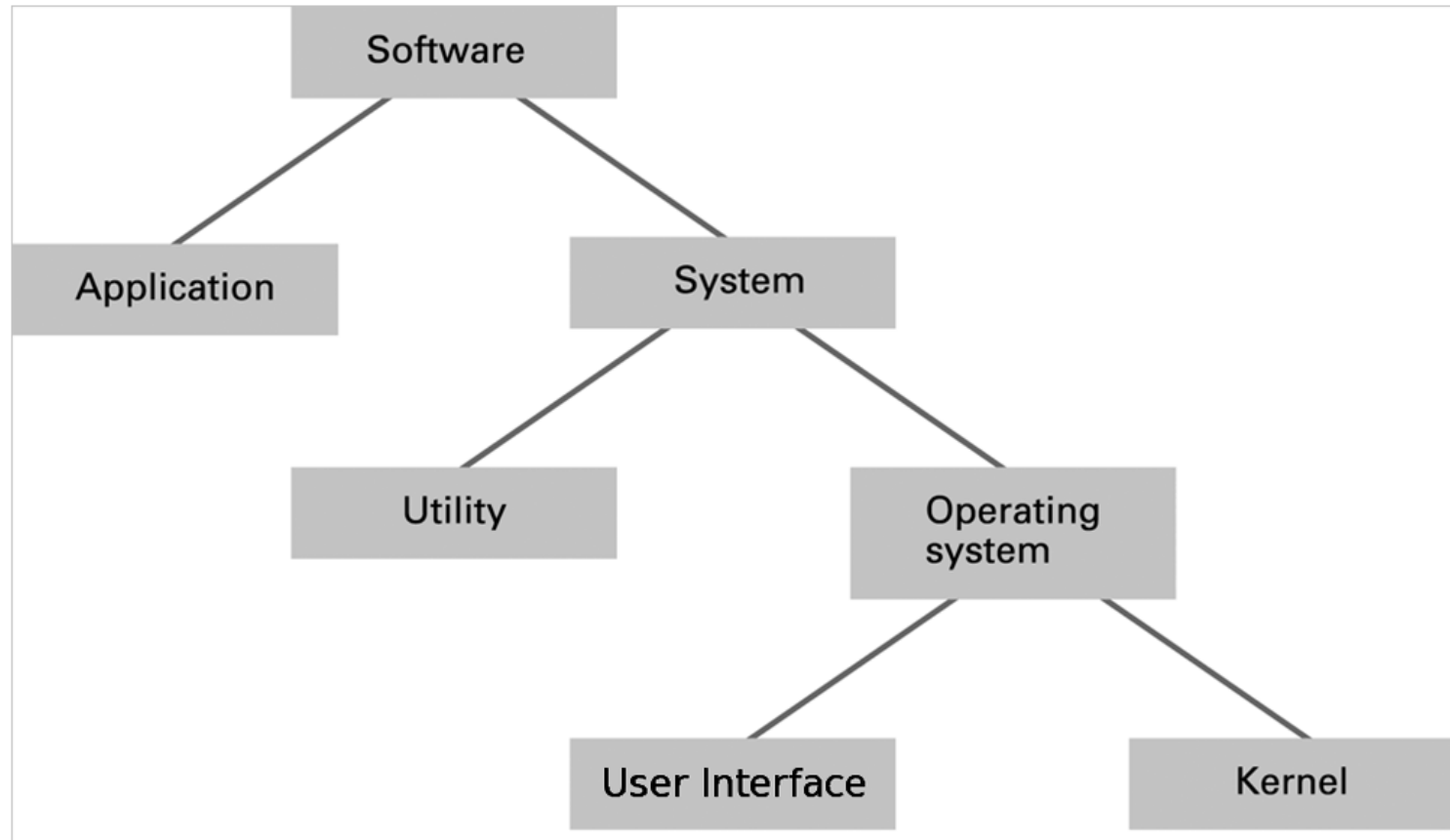
Figure 3.2 Interactive processing



Types of Software

- Application software
 - Performs specific tasks for users
- System software
 - Provides infrastructure for application software
 - Consists of operating system and utility software

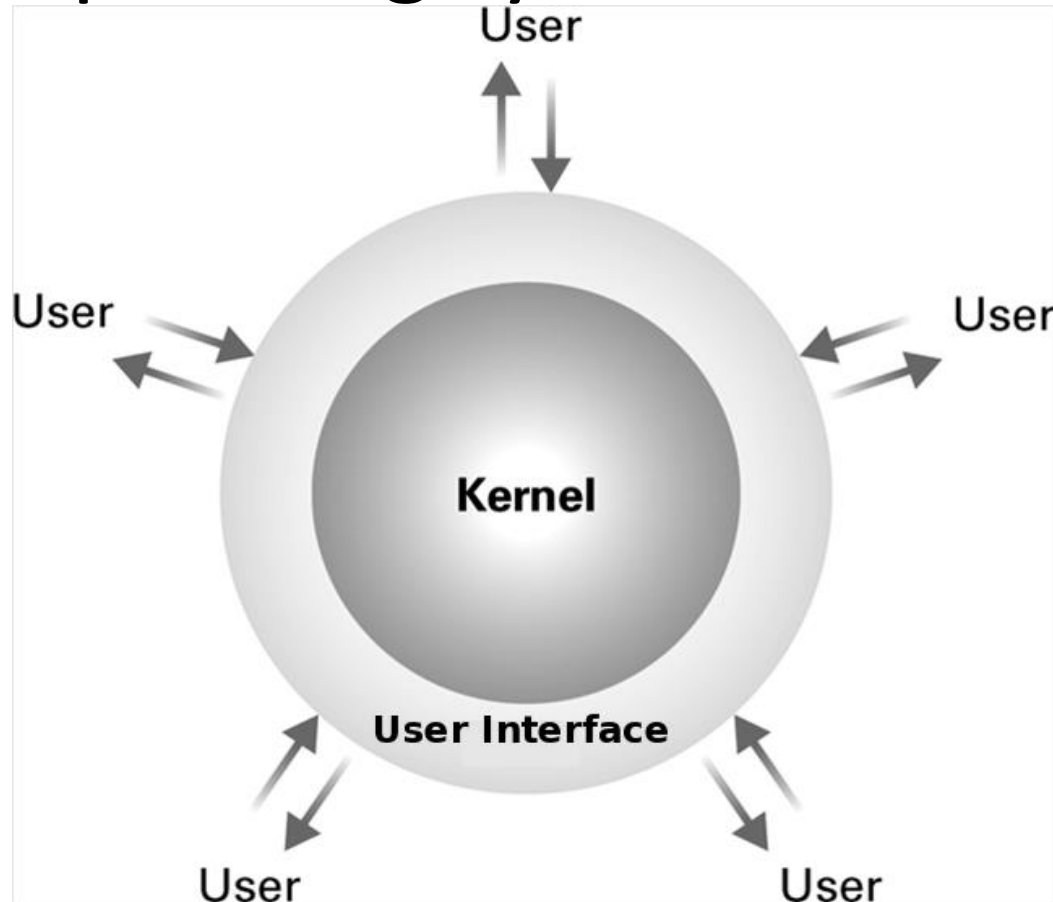
Figure 3.3 Software classification



Operating System Components

- **User Interface:** Communicates with users
 - Text based (Shell)
 - Graphical user interface (GUI)
- **Kernel:** Performs basic required functions
 - File manager
 - Device drivers
 - Memory manager
 - Scheduler and dispatcher

Figure 3.4 The user interface act as an intermediary between users and the operating system kernel



File Manager

- **Directory (or Folder):** A user-created bundle of files and other directories (subdirectories)
- **Directory Path:** A sequence of directories within directories

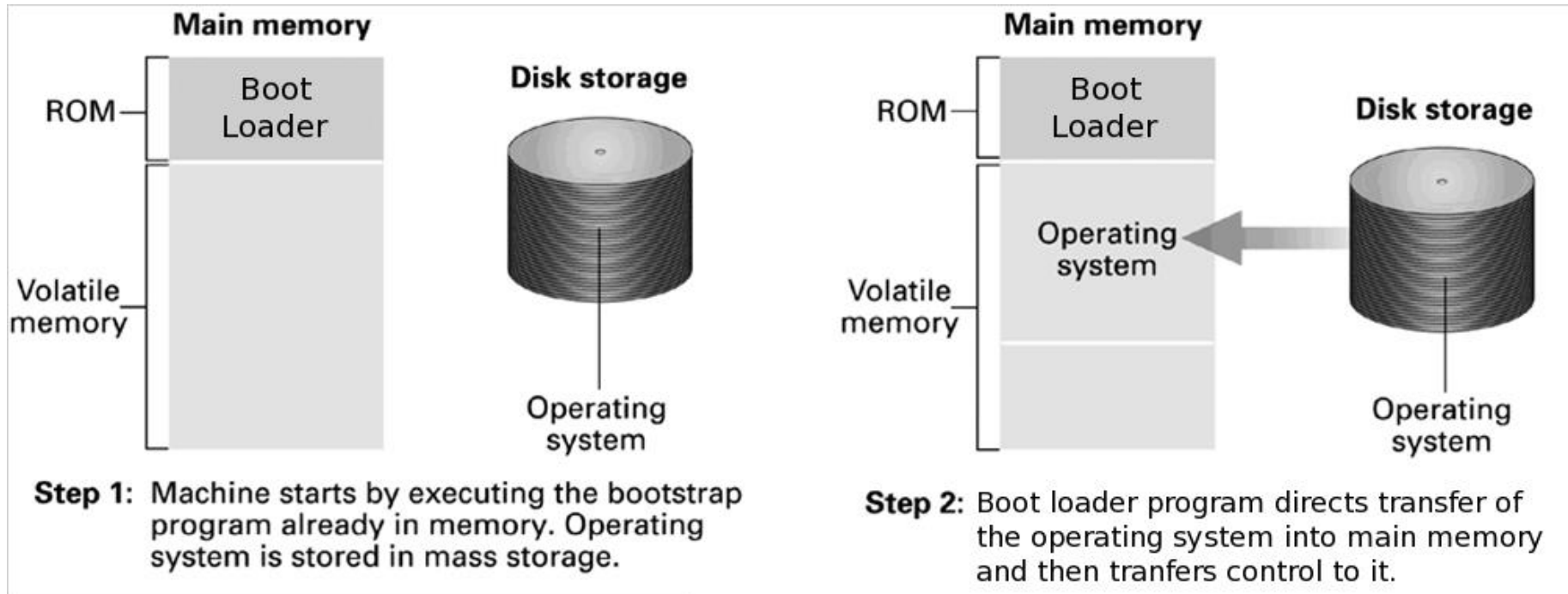
Memory Manager

- Allocates space in main memory
- May create the illusion that the machine has more memory than it actually does
 - (**virtual memory**) in which blocks of data (**pages**) are shifted back and forth between main memory and mass storage

Getting it Started (Bootstrapping)

- **Boot loader:** Program in ROM (example of firmware)
 - Run by the CPU when power is turned on
 - Transfers operating system from mass storage to main memory
 - Executes jump to operating system

Figure 3.5 The booting process



Processes

- **Process:** The activity of executing a program
- **Process State:** Current status of the activity
 - Program counter
 - General purpose registers
 - Related portion of main memory

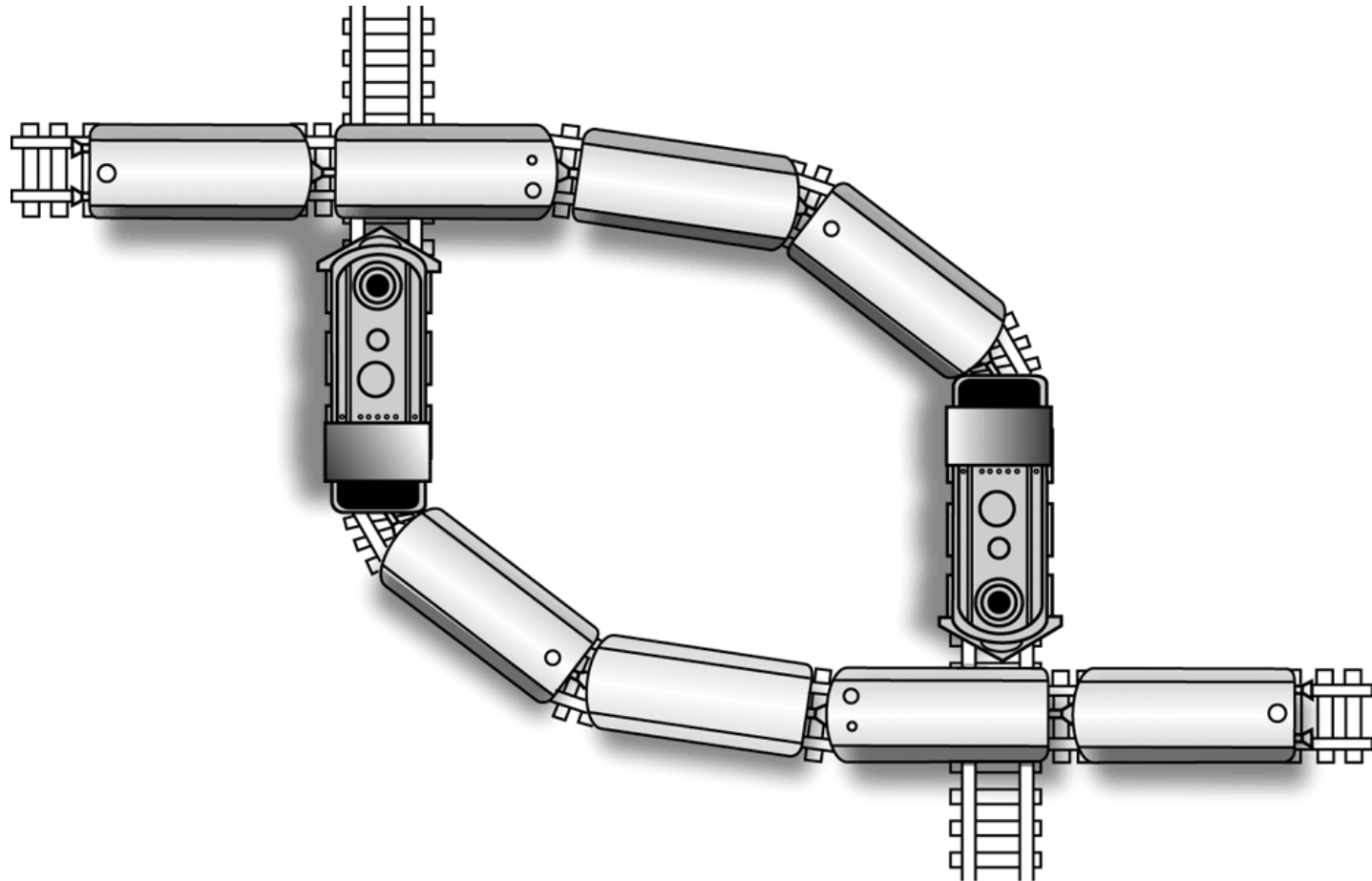
Process Administration

- **Scheduler:** Adds new processes to the process table and removes completed processes from the process table
- **Dispatcher:** Controls the allocation of time slices to the processes in the process table
 - The end of a time slice is signaled by an interrupt.

Deadlock

- Processes block each other from continuing
- Conditions required for deadlock
 1. Competition for non-sharable resources
 2. Resources requested on a partial basis
 3. An allocated resource can not be forcibly retrieved

Figure 3.7 A deadlock resulting from competition for nonshareable railroad intersections



Handling Competition for Resources

- **Semaphore:** A “control flag”
- **Critical Region:** A group of instructions that should be executed by only one process at a time
- **Mutual exclusion:** Requirement for proper implementation of a critical region