Chapter 12: Secondary-Storage Structures

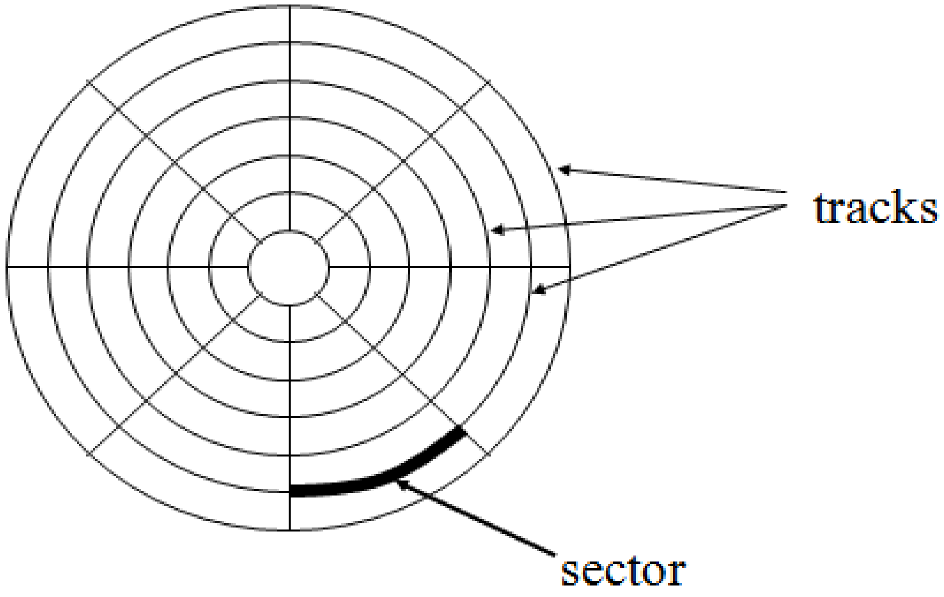
Overview of Mass Storage Structure

- Magnetic disks provide bulk of secondary storage of modern computers
  - Drives rotate at 60 to 200 times per second
  - **Transfer rate** is rate at which data flow between drive and computer
  - **Positioning time** (**random-access time**) is time to move disk arm to desired cylinder (**seek time**) and time for desired sector to rotate under the disk head (**rotational latency**)
  - **Head crash** results from disk head making contact with the disk surface
- Disks can be removable
- Drive attached to computer via **I/O bus**

Magnetic Disks

- Bits of data (0's and 1's) are stored on circular magnetic platters called disks.
- A disk rotates rapidly (60 to 200 times per second).
- A disk head reads and writes bits of data as they pass under the head.
- Often, several platters are organized into a disk pack (or disk drive).

Looking at a surface



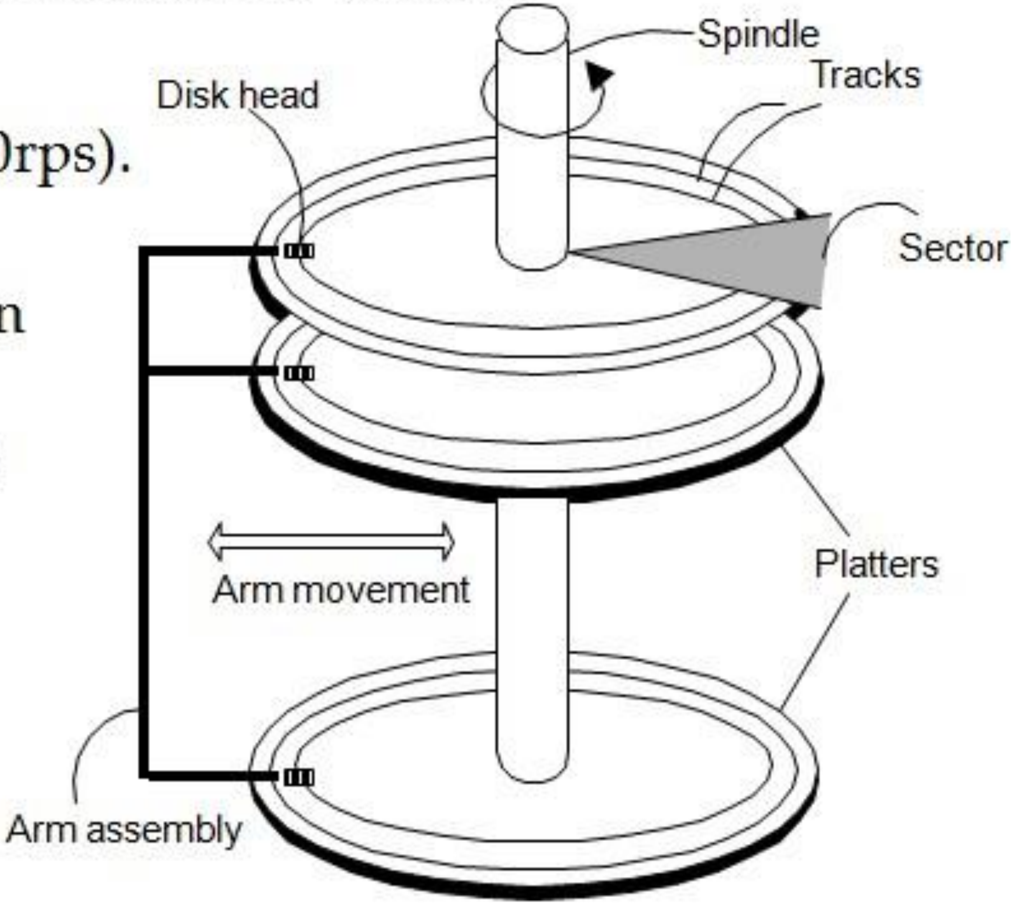
Surface of disk showing tracks and sectors

Organization of Disks

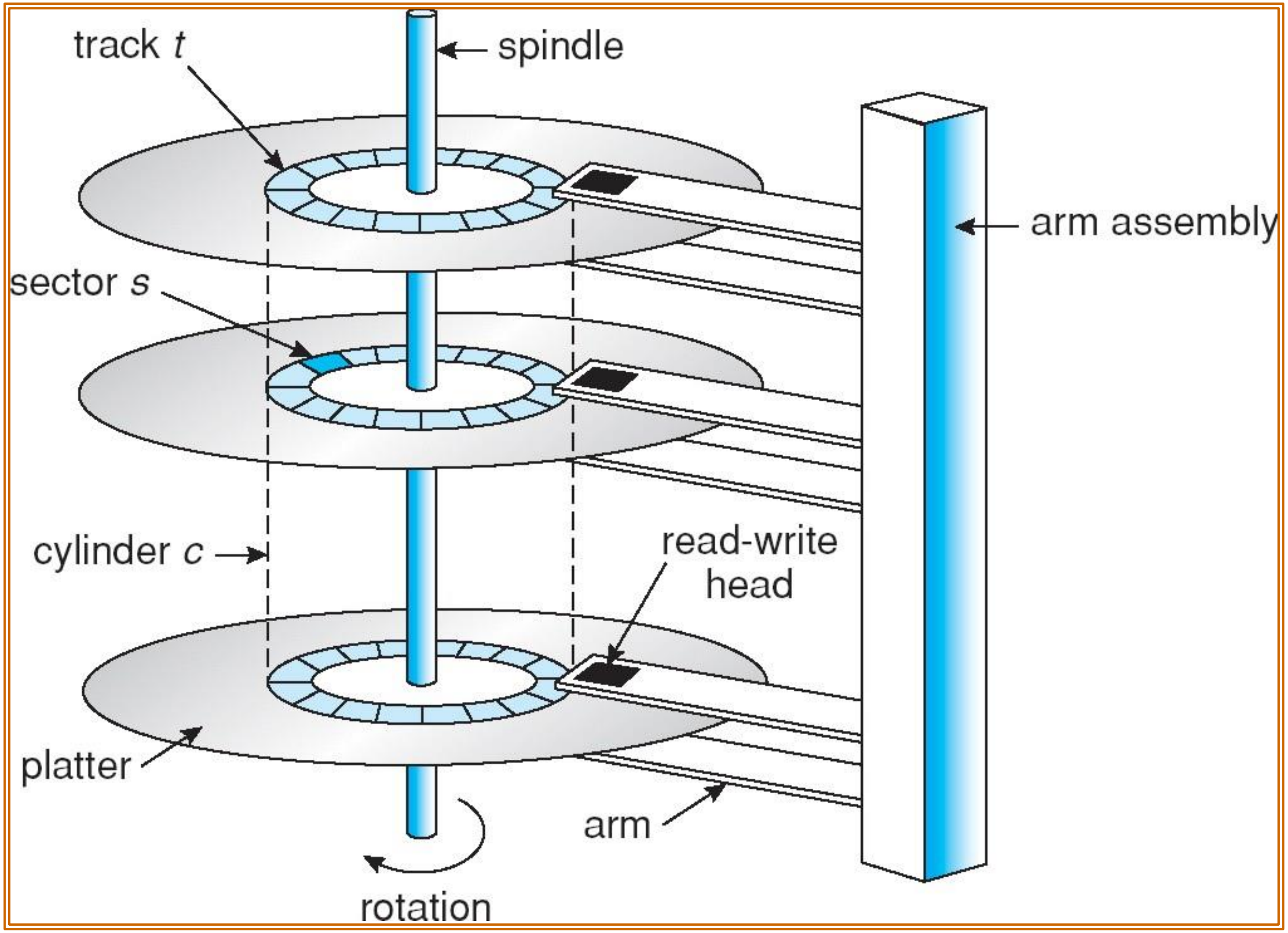
- Disk contains concentric **tracks.**
- Tracks are divided into **sectors**
- A **sector** is the smallest addressable unit in a disk.

Components of a Disk

- The platters spin (say, 90rps).
- The arm assembly is moved in or out to position a head on a desired track. Tracks under heads make a cylinder (imaginary!).
- Only one head reads/writes at any one time.
- Block size is a multiple of sector size (which is often fixed).



Moving-head Disk Machanism



Disk Scheduling

- The operating system is responsible for using hardware efficiently — for the disk drives, this means having a fast access time and disk bandwidth.
- Access time has two major components
  - *Seek time* is the time for the disk are to move the heads to the cylinder containing the desired sector.
  - *Rotational latency* is the additional time waiting for the disk to rotate the desired sector to the disk head.
- Minimize seek time
- Seek time ≈ seek distance

Disk Scheduling (Cont.)

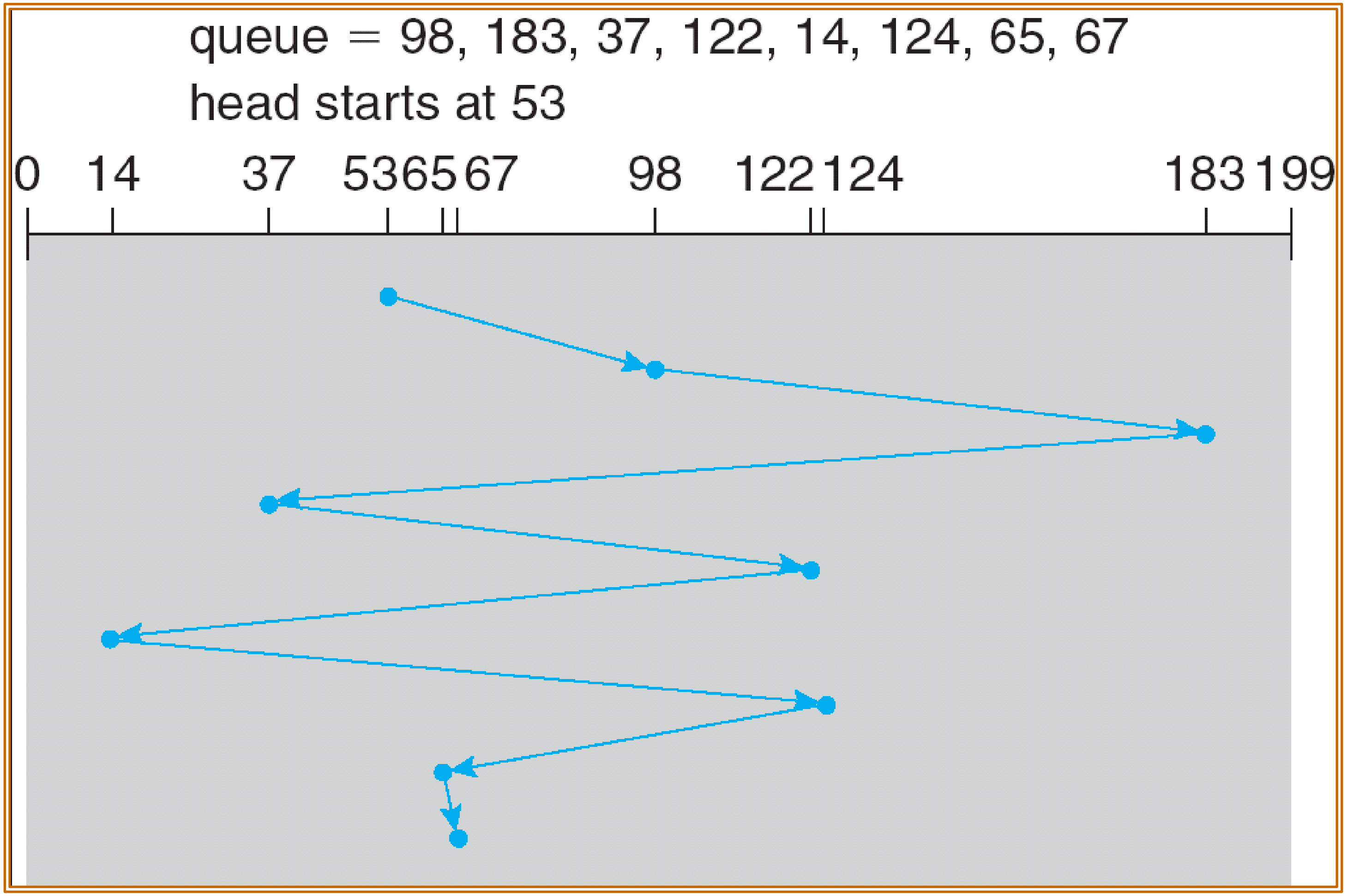
- Several algorithms exist to schedule the servicing of disk I/O requests.
- We illustrate them with a request queue (0-199).

98, 183, 37, 122, 14, 124, 65, 67

Head pointer 53

FCFS

Illustration shows total head movement of 640 cylinders.



queue = 98, 183, 37, 122, 14, 124, 65, 67

head starts at 53

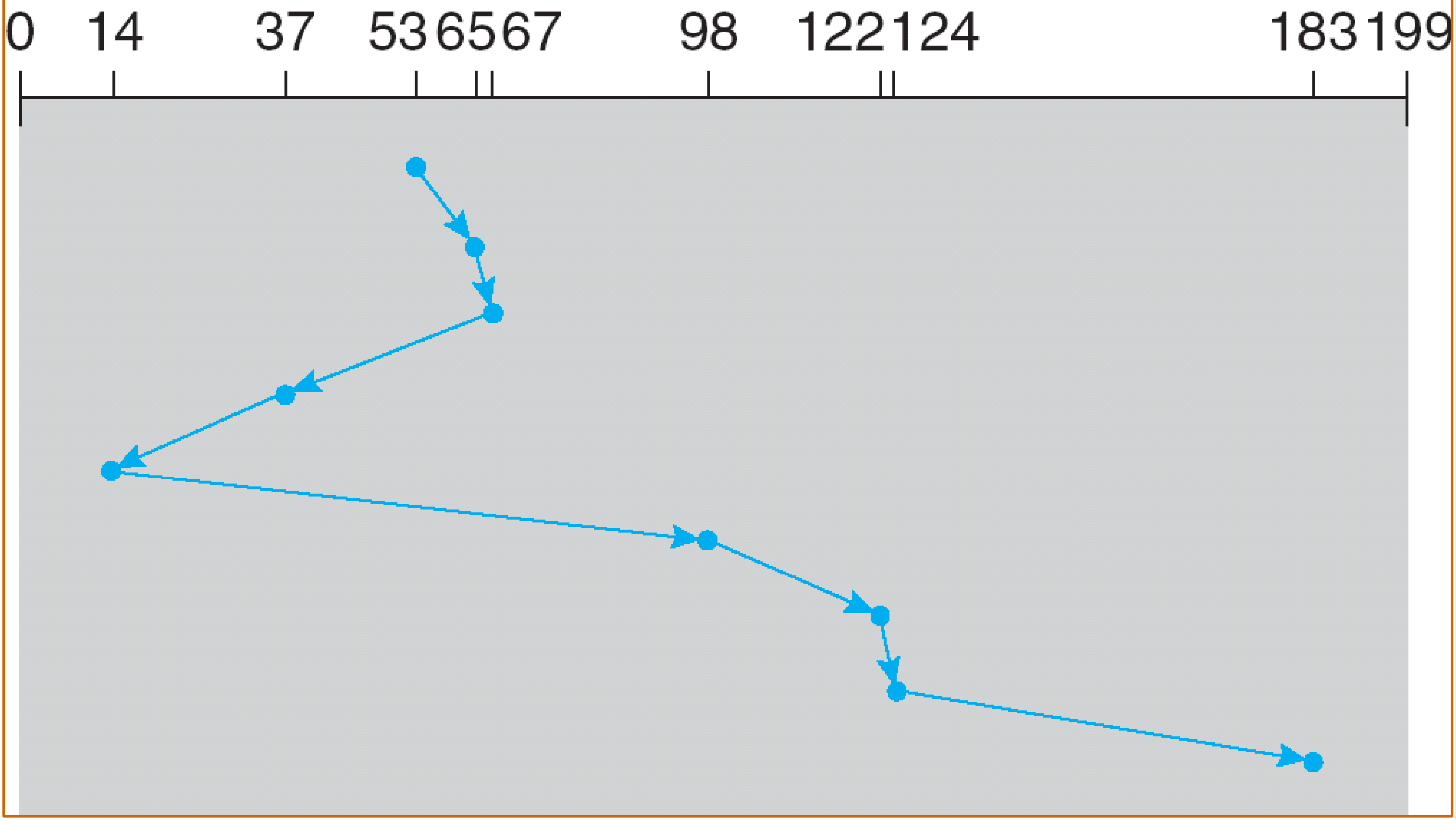
SSTF

- Selects the request with the minimum seek time from the current head position.
- SSTF scheduling is a form of SJF scheduling; may cause starvation of some requests.
- Illustration shows total head movement of 236 cylinders.

SSTF (Cont.)

queue = 98, 183, 37, 122, 14, 124, 65, 67

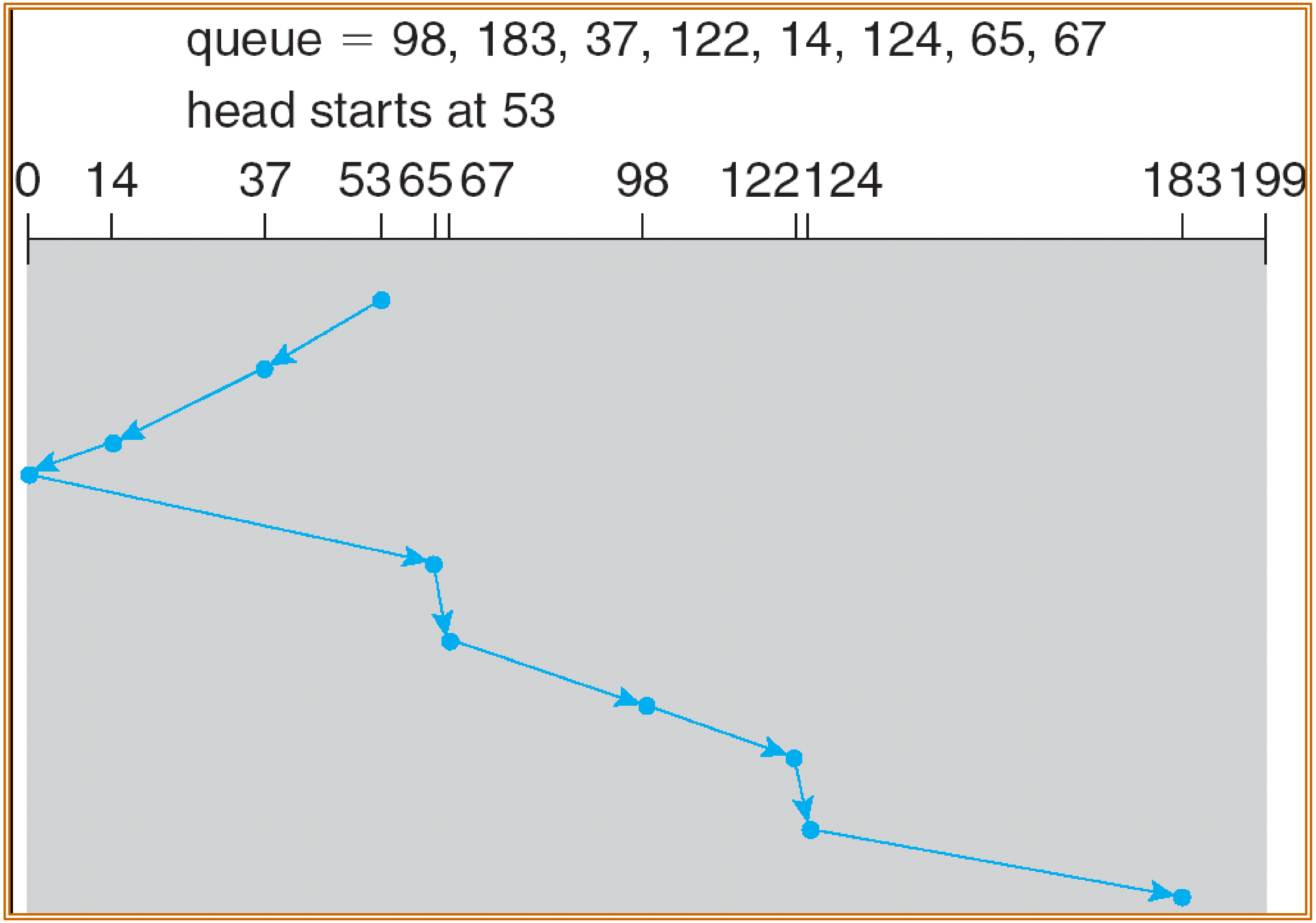
head starts at 53



SCAN

- The disk arm starts at one end of the disk, and moves toward the other end, servicing requests until it gets to the other end of the disk, where the head movement is reversed and servicing continues.
- Sometimes called the *elevator algorithm*.
- Illustration shows total head movement of 208 cylinders.

SCAN (Cont.)



queue = 98, 183, 37, 122, 14, 124, 65, 67

head starts at 53

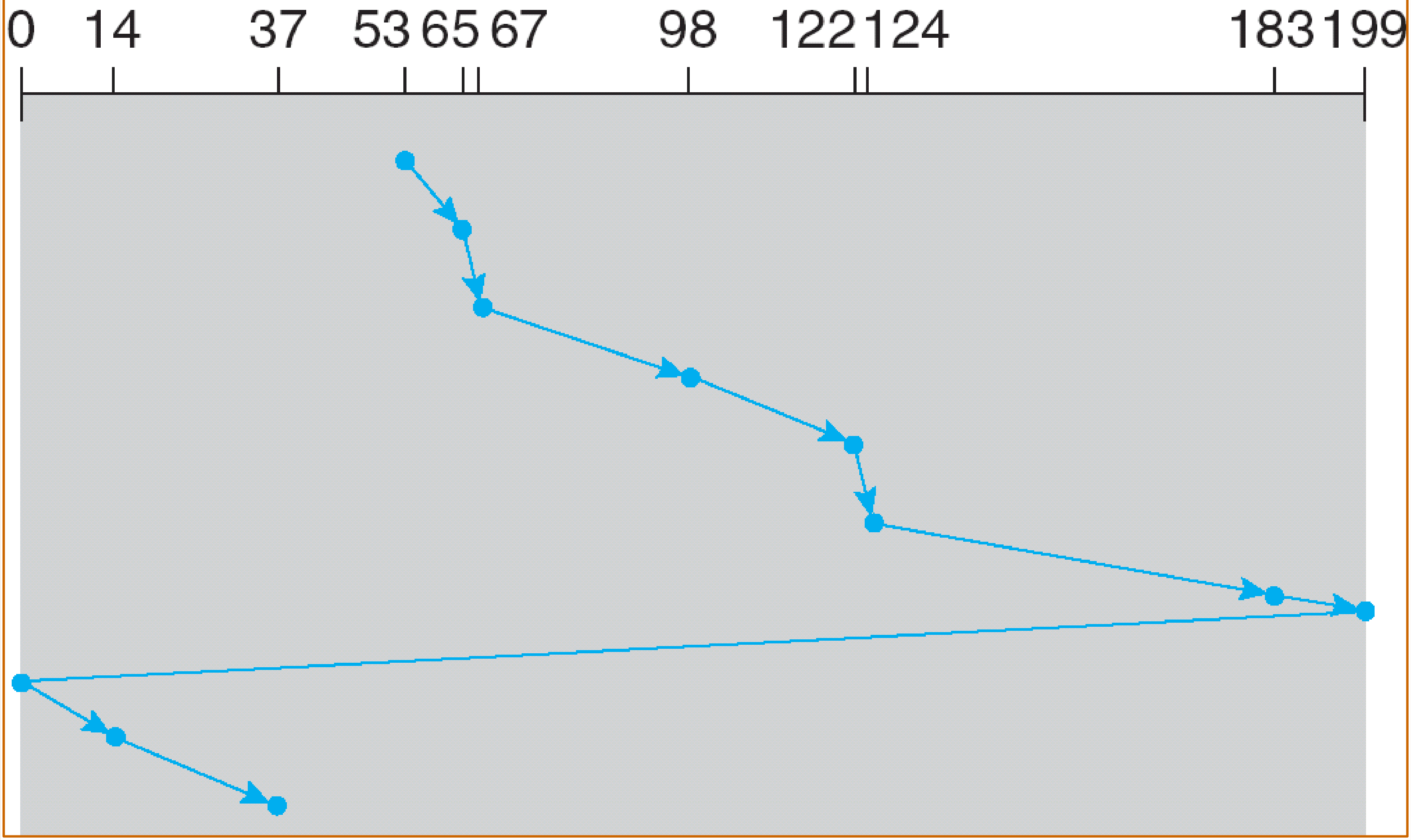
C-SCAN

- Provides a more uniform wait time than SCAN.
- The head moves from one end of the disk to the other. servicing requests as it goes. When it reaches the other end, however, it immediately returns to the beginning of the disk, without servicing any requests on the return trip.
- Treats the cylinders as a circular list that wraps around from the last cylinder to the first one.

C-SCAN (Cont.)

queue = 98, 183, 37, 122, 14, 124, 65, 67

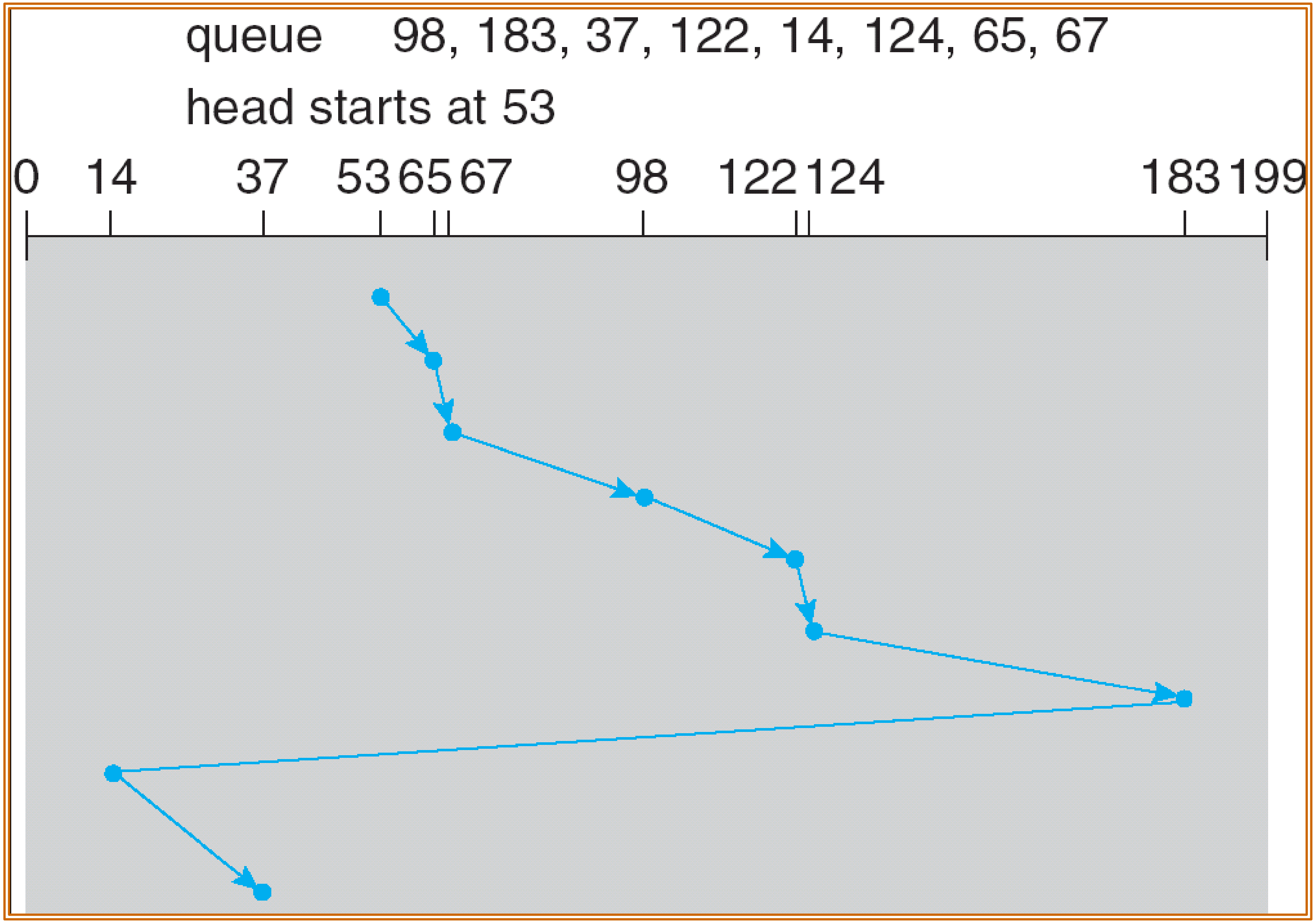
head starts at 53



C-LOOK

- Version of C-SCAN
- Arm only goes as far as the last request in each direction, then reverses direction immediately, without first going all the way to the end of the disk.

C-LOOK (Cont.)


queue 98, 183, 37, 122, 14, 124, 65, 67
head starts at 53
0
14
37
53
65
67
98
122
124
183
199

Selecting a Disk-Scheduling Algorithm

- SSTF is common and has a natural appeal
- SCAN and C-SCAN perform better for systems that place a heavy load on the disk.
- Performance depends on the number and types of requests.
- Requests for disk service can be influenced by the file-allocation method.
- The disk-scheduling algorithm should be written as a separate module of the operating system, allowing it to be replaced with a different algorithm if necessary.
- Either SSTF or LOOK is a reasonable choice for the default algorithm.

Disk Management

- *Low-level formatting*, or *physical formatting* — Dividing a disk into sectors that the disk controller can read and write.
- To use a disk to hold files, the operating system still needs to record its own data structures on the disk.
  - *Partition* the disk into one or more groups of cylinders.
  - *Logical formatting* or "making a file system".
- Boot block initializes system.
  - The bootstrap is stored in ROM.
  - *Bootstrap loader* program.
- Methods such as *sector sparing* used to handle bad blocks.

Types of Devices

- • Despite the multitude of devices that appear (and
- disappear) in the marketplace and the swift rate of
- change in device technology, the Device Manager
- must manage every peripheral device of the system.
- • It must maintain a delicate balance of supply and
- demand – balancing the system’s finite supply of
- devices with users’ almost infinite demand for them.

- Device management involves four basic functions:
- – Monitoring the status of each device, such as storage drives, printers, and other peripheral devices;
- – Enforcing preset policies to determine which process will get a device and for how long;
- – Allocating the devices;
- – Deallocating them at two levels:
- • At the process (or task) level when an I/O command has been executed and the device is temporarily released;
- • At the job level when the job is finished and the device is permanently released.

Types of Devices (cont’d)

- • The system’s peripheral devices generally fall into
- one of three categories:
- – Dedicated
- – Shared
- – Virtual
- • The differences are a function of the characteristics of the devices, as well as how they’re managed by the Device Manager.

- **Dedicated Devices**
- – Are assigned to only one job at a time.
- – They serve that job for the entire time the job is activen or until it releases them.
- – Some devices demand this kind of allocation scheme, because it would be awkward to let several users share them.
- • Example: tape drives, printers, and plotters
- – Disadvantages
- • They must be allocated to a single user for the duration of a job's execution, which can be quite inefficient, even though the device is not used 100% of the time.

- **Shared Devices**
- – Can be assigned to several processes.
- – For example – a disk (DASD) can be shared by several processes at the same time by interleaving their requests;
- • This interleaving must be carefully controlled by the Device Manager– All conflicts must be resolved based on predetermined policies.

- **Virtual Devices**
- – A combination of the first two types;
- – They’re dedicated devices that have been transformed into shared devices.
- • Example: printer
- – Converted into a shareable device through a spooling

program that reroutes all print requests to a disk.

- – Only when all of a job’s output is complete, and the

printer is ready to print out the entire document, is the output sent to the printer for printing.

- – Because disks are shareable devices, this technique can convert one printer into several virtual printers,

thus improving both its performance and use.

Types of Devices (cont'd.)

- • Regardless of the specific attributes of the device,the most important differences among them are speed and degree of sharability.
- • Storage media are divided into two groups:
- – **Sequential Access Media**
- • Store records sequentially, one after the other.
- – **Direct Access Storage Devices (DASD)**
- • Can store either sequential or direct access files.
- • There are vast differences in their speed and sharability.