CS 4410
Operating Systems

Mass-Storage Structure

Summer 2011

Cornell University

Today

- How is data saved in the hard disk?
- Magnetic disk
- Disk speed parameters
- Disk Scheduling
- RAID Structure

Secondary Storage

- Save data permanently.
- Slower than memory.
- Cheaper and greater than memory.
- Magnetic Tapes
- Magnetic Disks

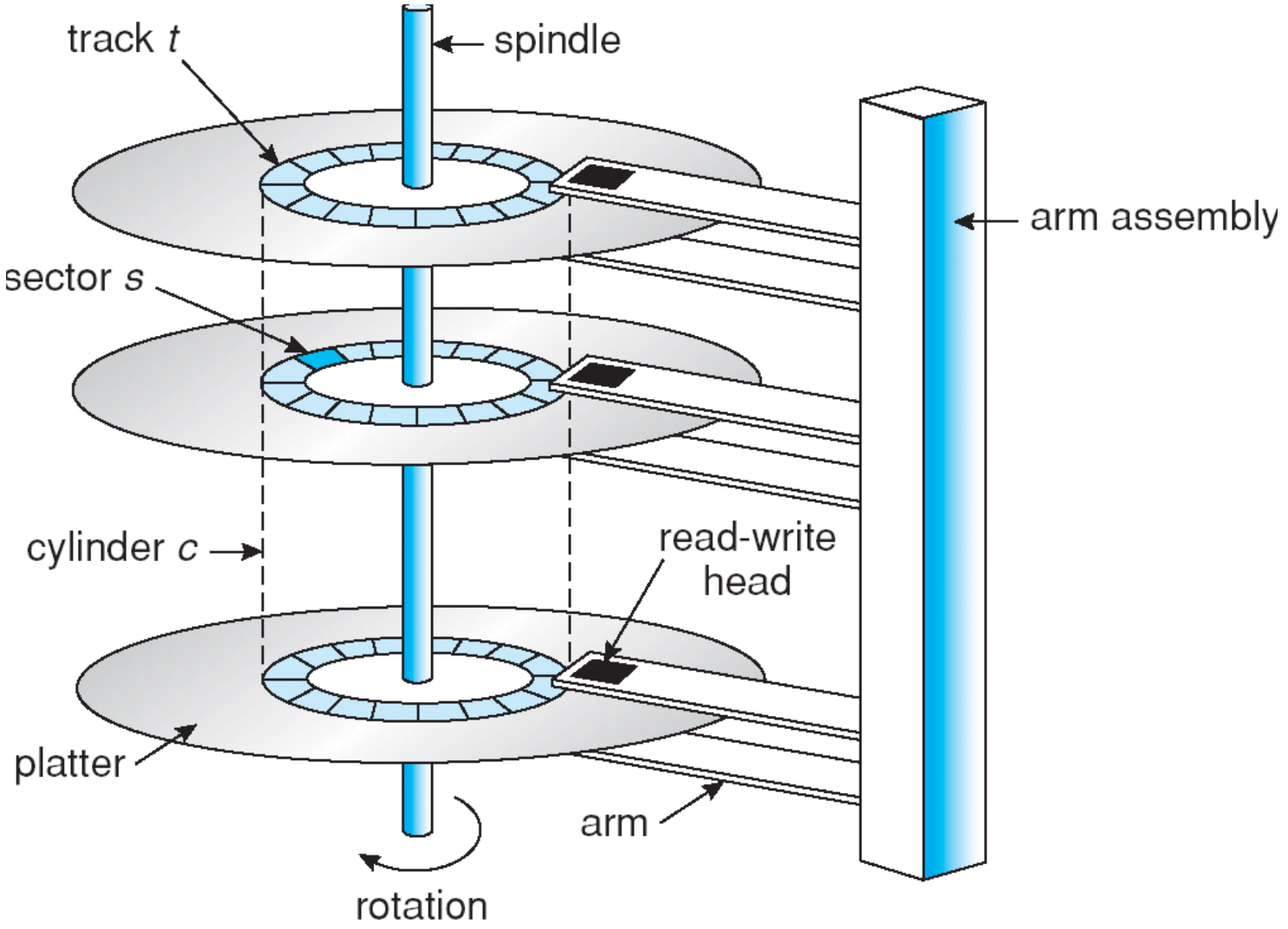
Magnetic Disks

Then

Magnetic Disks

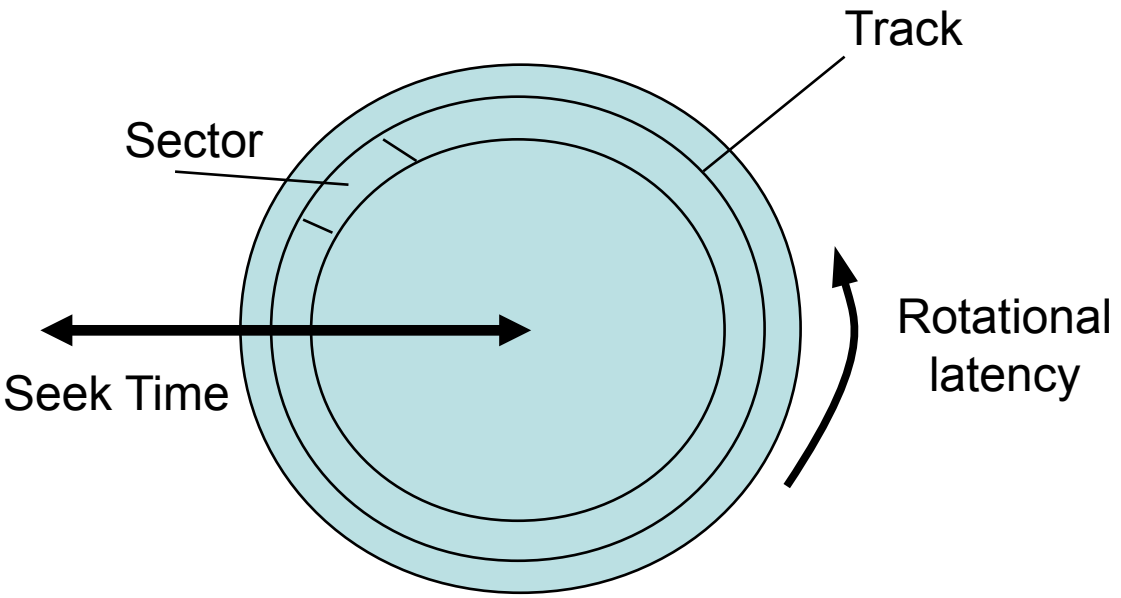
Now

Magnetic Disk: Internal



Disk Speed

- To read from disk, we must specify:
  - cylinder #, surface #, sector #, transfer size, memory address
- Disk speed has two parts:
  - Transfer rate: the rate at which data flow between the drive and the computer.
  - Positioning time:
    - Seek time: the time to move the disk arm to the desired cylinder.
    - Rotational latency: the time for the desired sector to rotate to the disk head.



Disks vs Memory

- Smallest write: sector
- Atomic write = sector
- Random access: 5ms
- Sequential access: 200MB/s
- Cost $.002MB
- Crash: no loss (“non-volatile”)

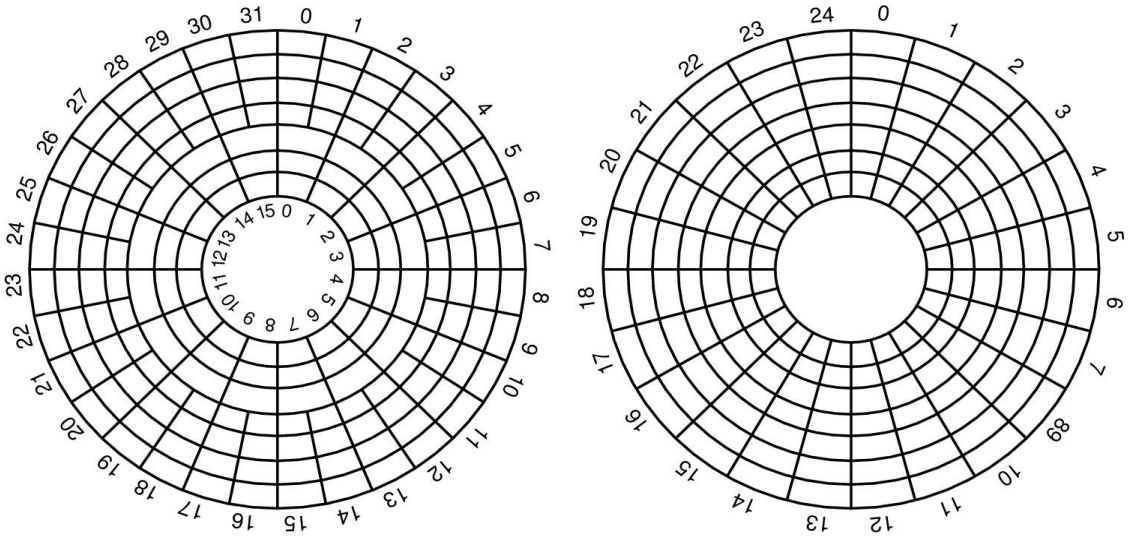
- (usually) bytes
- byte, word
- 50ns
- 200-1000MB/s
- $.10MB
- Contents gone (“volatile”)

Disk Structure

- Disk drives addressed as **1-dim arrays** of ***logical blocks***.
  - The logical block is the smallest unit of transfer.
  - Usually 512 bytes.
- This **array mapped** sequentially onto **disk sectors**.
  - Address 0 is 1st sector of 1st track of the outermost cylinder.
  - Addresses incremented within track, then within tracks of the cylinder, then across cylinders, from outermost to innermost.
- Translation is theoretically possible, but usually difficult.
  - Some sectors might be defective.
  - Number of sectors per track is not a constant.

Number of sectors per track

- **Uniform** Number of sectors per track.
  - Reduce bit density per track for outer layers.
  - Constant Linear Velocity.
  - Typically HDDs.
- **Non-uniform** Number of sectors per track.
  - Have more sectors per track on the outer layers.
  - Increase rotational speed when reading from outer tracks.
  - Constant Angular Velocity
  - Typically CDs, DVDs.



Disk Scheduling

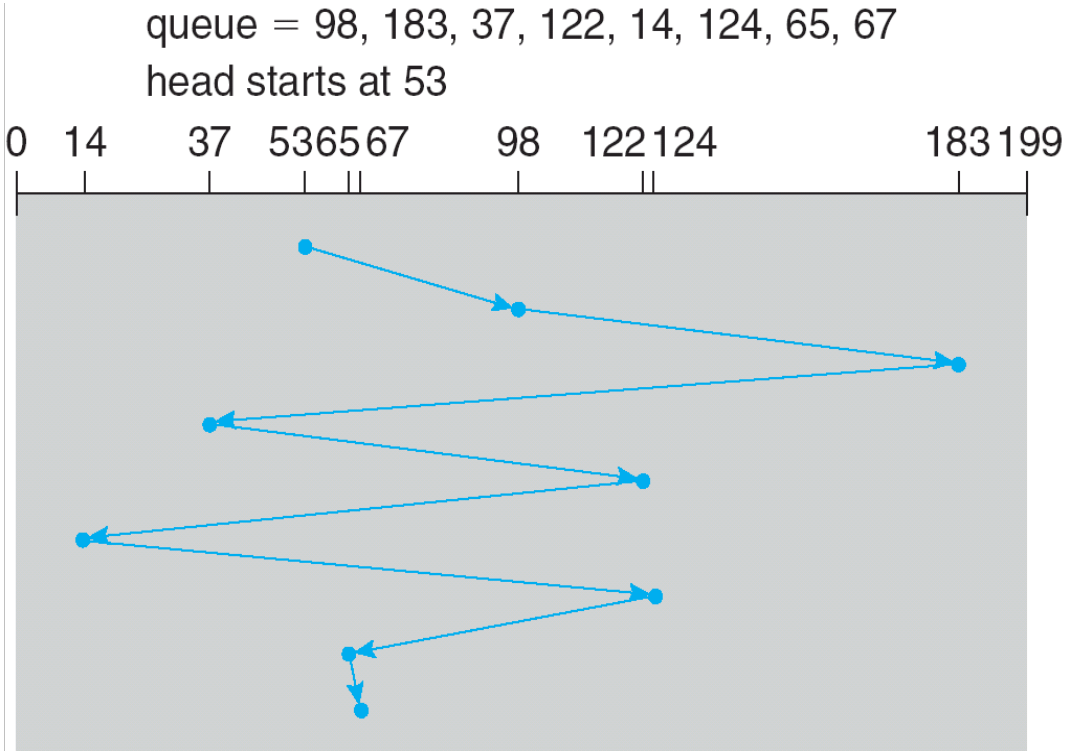
- Whenever a process needs to read or write to the disk:
  - It issues a system call to the OS.
  - If the controller is available, the request is served.
  - Else, the request is placed in the pending requests queue of the driver.
  - When a request is completed, the OS decides which is the next request to service.
  - How does the OS make this decision? On which criteria?

Disk Scheduling

- The OS tries to **use the disk efficiently**.
- Target: **Small access time** and **large bandwidth**.
- The target can be achieved by **managing the order** in which disk I/O requests are serviced.
- Different algorithms can be used.

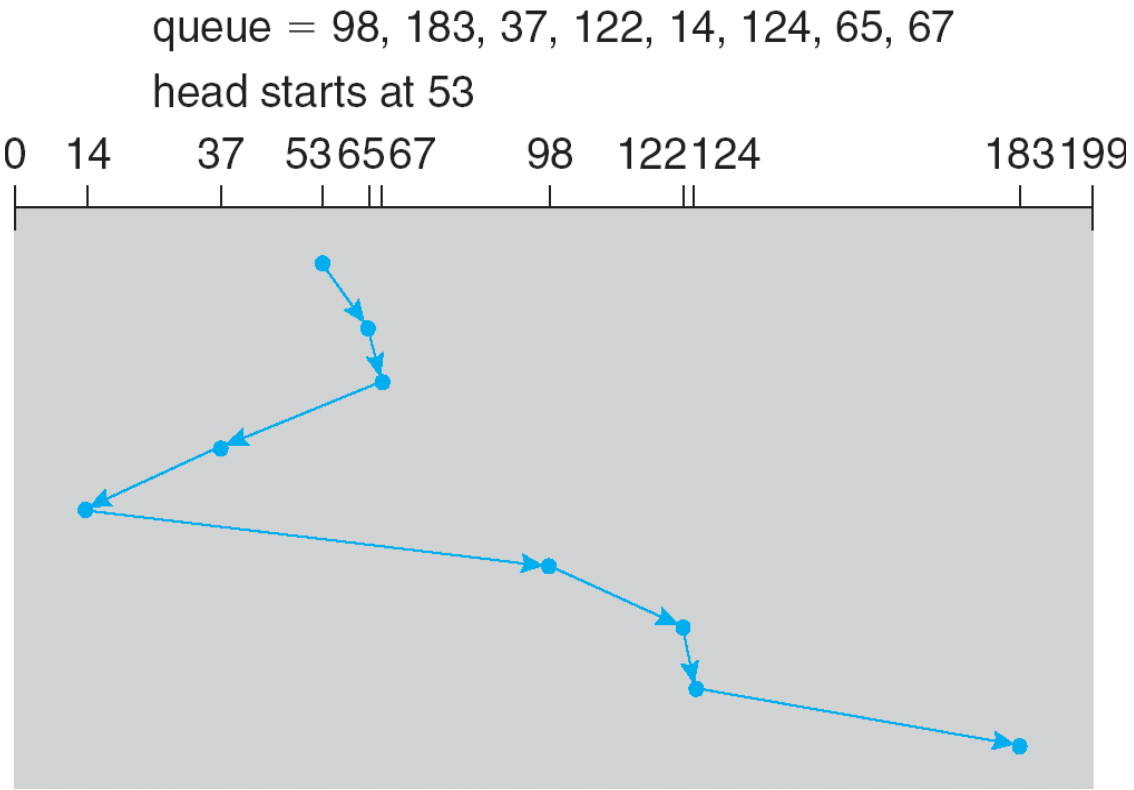
FCFS

- Consider a disk queue with requests for I/O to blocks on cylinders:
  - 98, 183, 37, 122, 14, 124, 65, 67
- The disk head is initially at cylinder 53.
- Total head movement of 640 cylinders



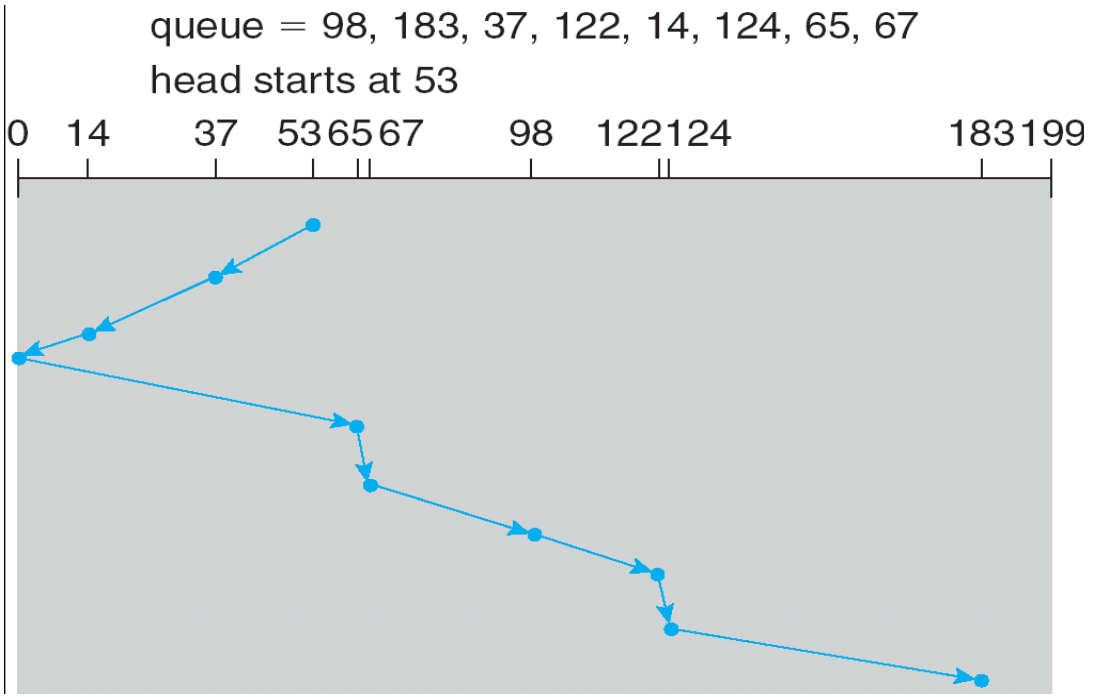
SSTF

- Selects request with minimum seek time from current head position
- SSTF scheduling is a form of SJF scheduling
  - May cause starvation of some requests.
- Total head movement of 236 cylinders



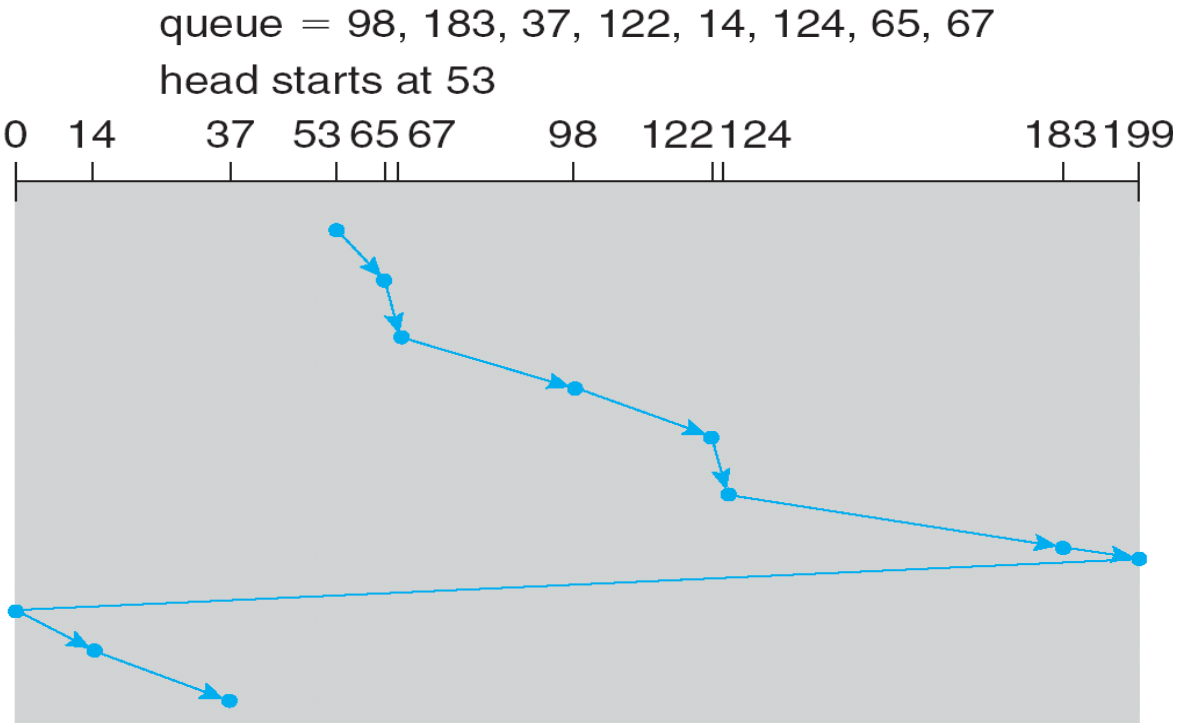
SCAN

- The disk arm starts at one end of the disk.
  - Moves toward the other end, servicing requests.
  - Head movement is reversed when it gets to the other end of disk.
  - Servicing continues.
- Total head movement of 208 cylinders



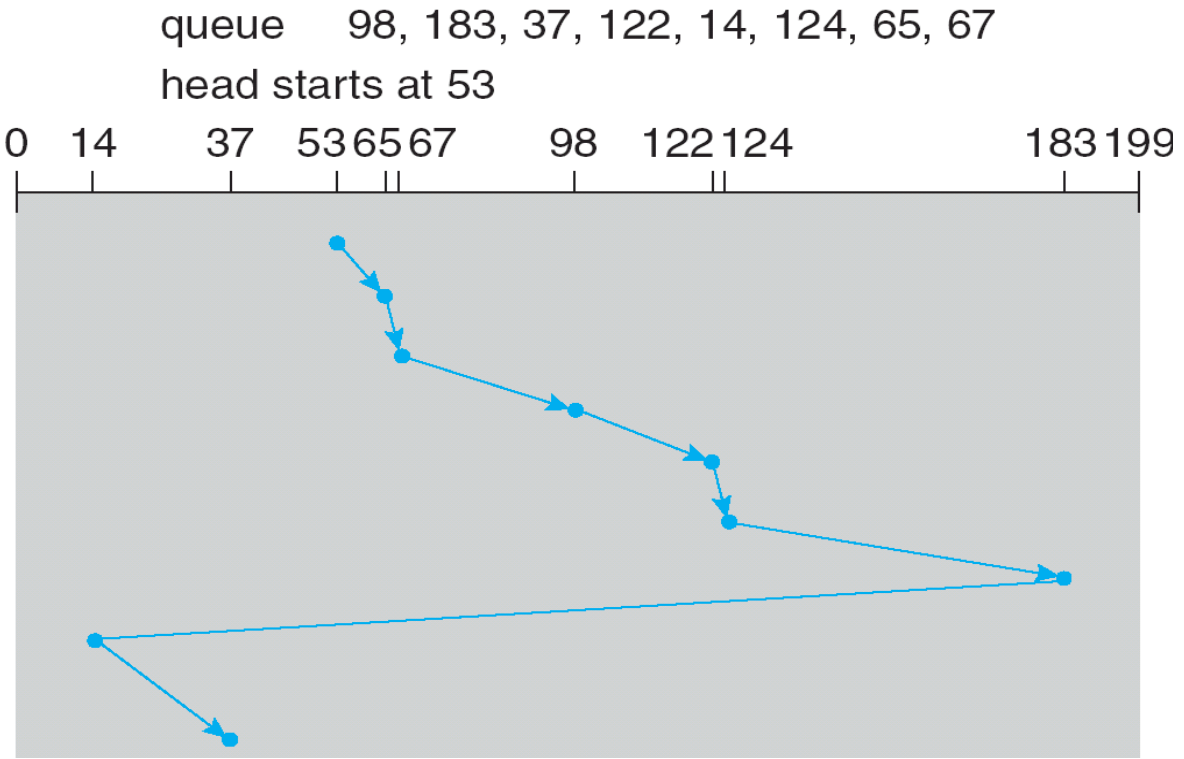
C-SCAN

- 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 it immediately returns to the beginning of the disk.



C-LOOK

- Arm only goes as far as last request in each direction.
  - Then reverses direction immediately.



RAID Structure

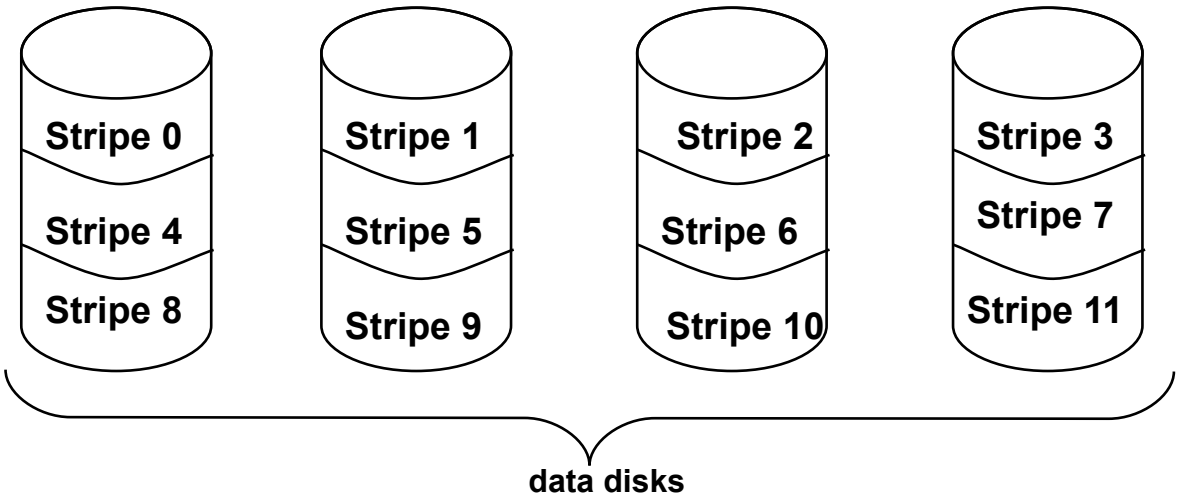
- Disks are improving, but not as fast as CPUs.
  - 1970s seek time: 50-100 ms.
  - 2000s seek time: <5 ms.
  - Factor of 20 improvement in 3 decades
- We can use multiple disks for improving performance.
- By Striping files across multiple disks (placing parts of each file on a different disk), parallel I/O can improve access time.
- Striping reduces reliability.
  - 100 disks have 1/100th mean time between failures of one disk
- So, we need Striping for performance, but we need something to help with reliability / availability.
- To improve reliability, we can add redundant data to the disks, in addition to Striping

RAID Structure

- A RAID is a Redundant Array of Independent Disks.
- Disks are small and cheap, so it's easy to put lots of disks in one box for increased storage, performance, and availability.
- **Data plus some redundant information is Striped across the disks in some way.**

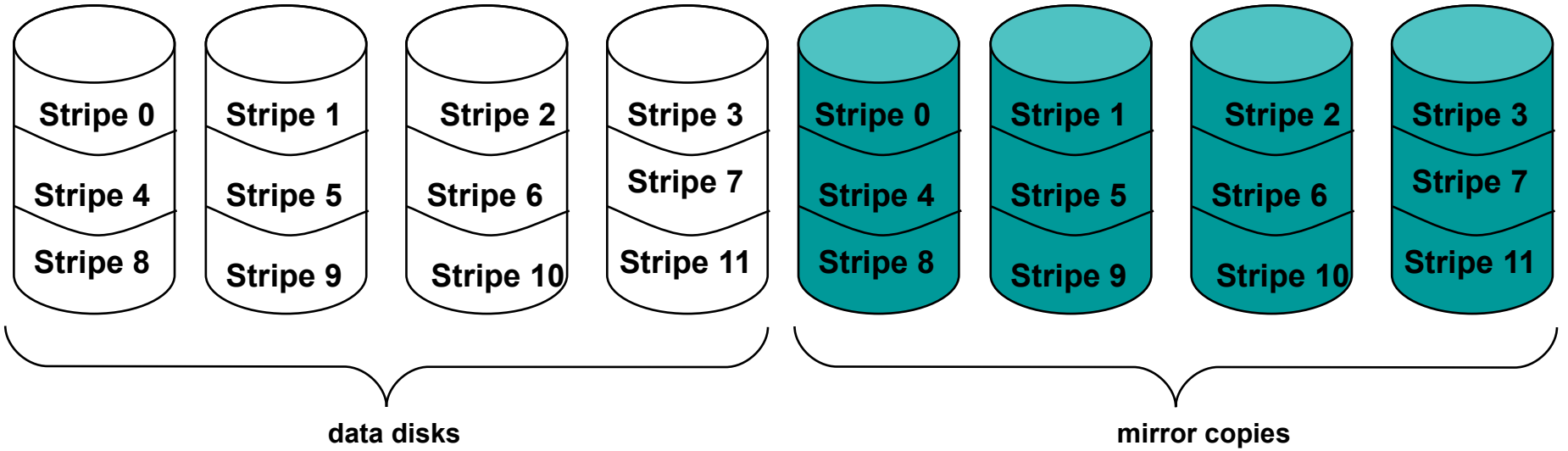
Raid Level 0

- Level 0 is non-redundant disk array.
- Files are Striped across disks, no redundant info.
- High read throughput.
- Best write throughput (no redundant info to write).
- Any disk failure results in data loss.



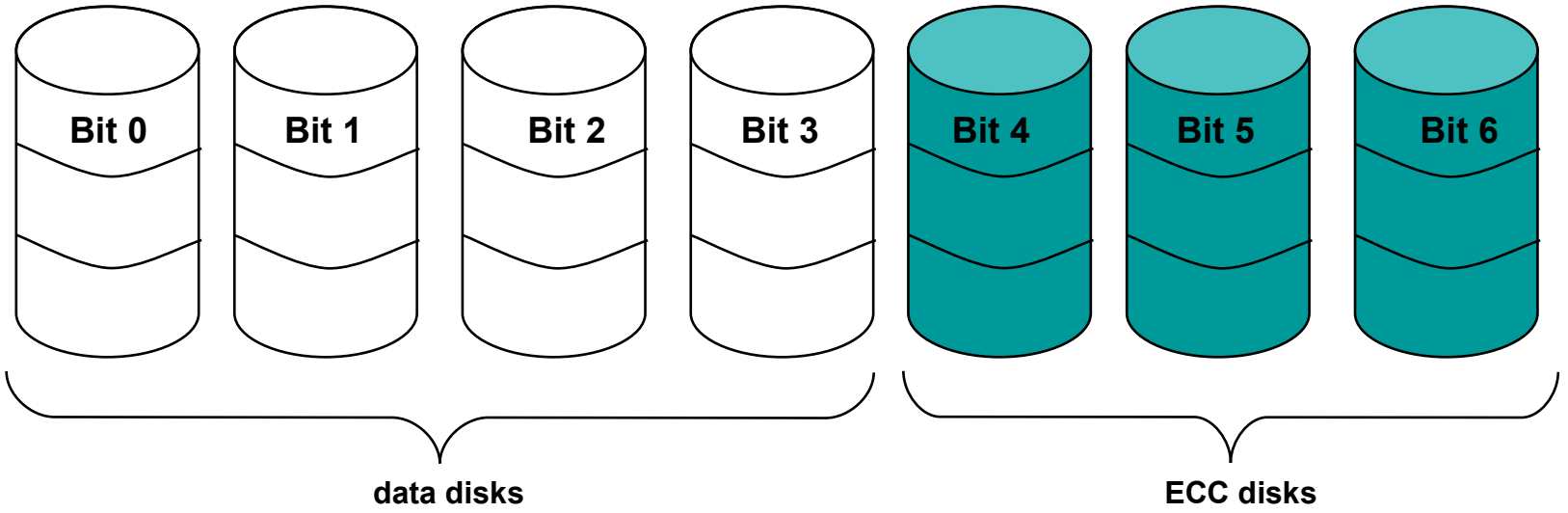
Raid Level 1

- Mirrored Disks
- Data is written to two places.
  - On failure, just use surviving disk.
- On read, choose fastest to read.
  - Write performance is same as single drive, read performance is 2x better.
- Expensive



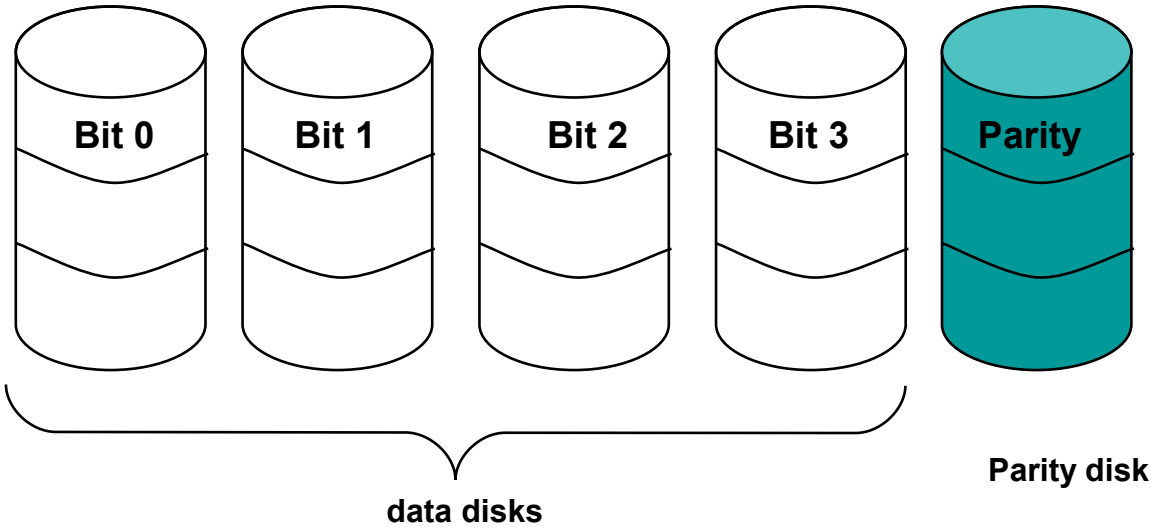
Raid Level 2

- Bit-level Striping with ECC codes for error correction.
- All 7 disk arms are synchronized and move in unison.
- Complicated controller.
- Single access at a time.
- Tolerates only one error, but with no performance degradation.



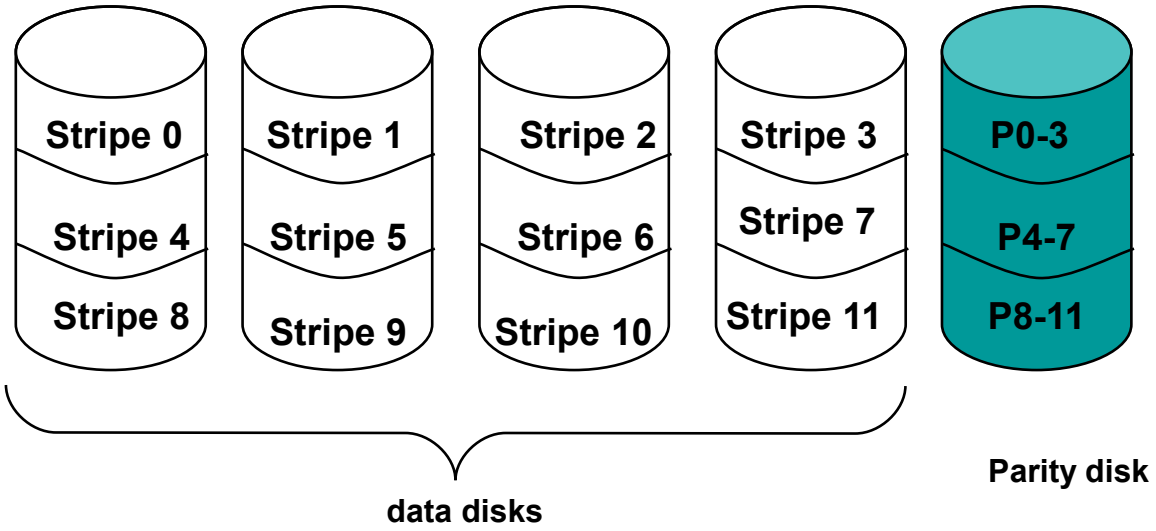
Raid Level 3

- Use a parity disk.
  - Each bit on the parity disk is a parity function of the corresponding bits on all the other disks.
- A read accesses all the data disks.
- A write accesses all data disks plus the parity disk.
- On disk failure, read remaining disks plus parity disk to compute the missing data.



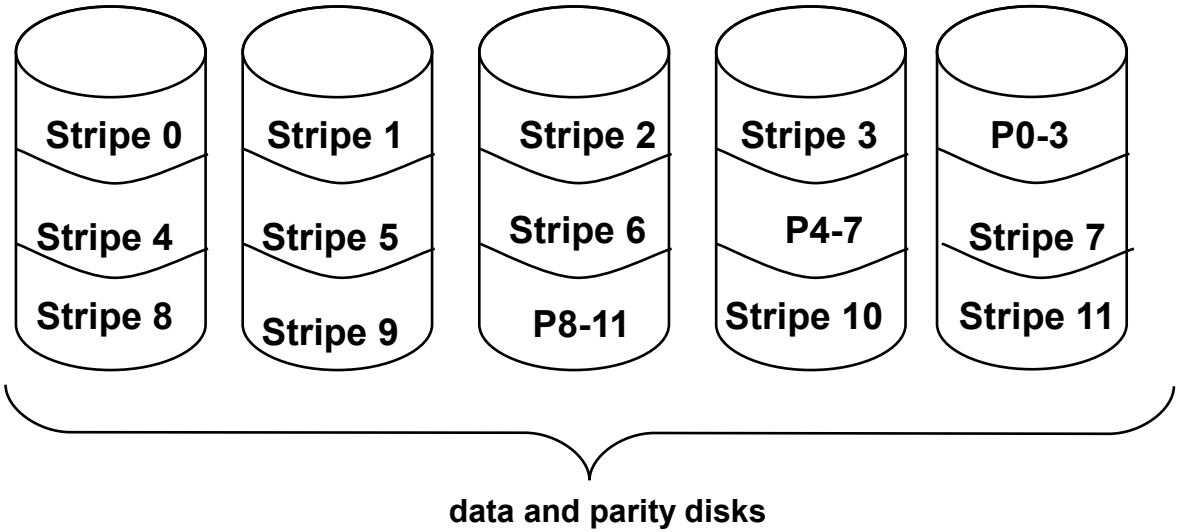
Raid Level 4

- Combines Level 0 and 3 – block-level parity with Stripes.
- A read accesses all the data disks.
- A write accesses all data disks plus the parity disk.
- Heavy load on the parity disk.



Raid Level 5

- Block Interleaved Distributed Parity.
- Like parity scheme, but distribute the parity info over all disks (as well as data over all disks).
- Better read performance, large write performance.



Today

- How is data saved in the hard disk?
- Magnetic disk
- Disk speed parameters
- Disk Scheduling
- RAID Structure