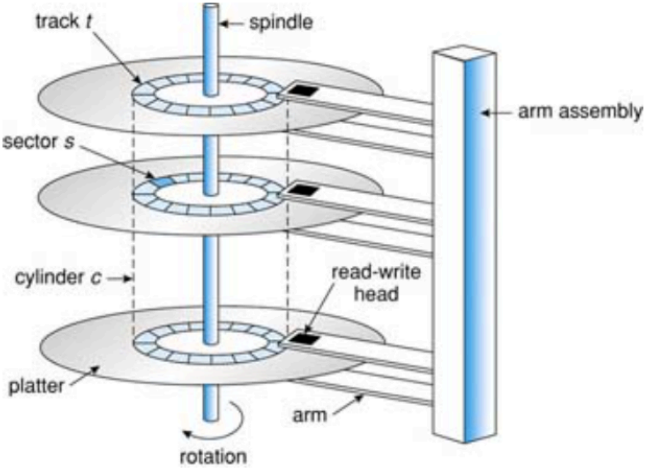
ECE 344: Operating Systems
Lecture 28

Hard Disk Drives
1.0.0

Jon Eyolfson
November 24, 2022

The Structure of a Hard Disk Drive (aka HDD)



Access Speed Depends on Locality

Sectors on same track can be read continuously

Switching tracks needs repositioning of the arm

Repositioning the arm is expensive)

You Physically Address a HDD Using Cylinder-Head-Sector (CHS)

Data has the following Coordinates (multi-dimensional polar coodinates):

- Platter: which revolving platter (addressed as head) [z Axis]
- Track: which track lane on platter (historically cylinder) [$||r||$]
- Sector: which sector on track [Θ]

The historical CHS has an approximate 8 GB limit of addressable space

(512 bytes/sector)×(63 sectors/track)×(255 heads (tracks/cylinder))
×(1024 cylinders)

LBA (Logical Block Addressing) uses one number to address any block and is not limited to 8 GB

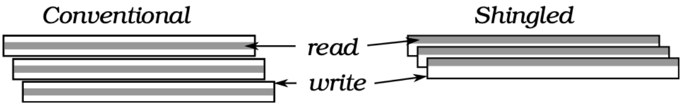
Shingled Magnet Recording (SMR)

The write head only writes in the center of a track, and has unused padding

You can't write to this padding without destroying neighboring tracks

SMR however, allows you to write over the padding, if you do it sequentially

Drive performance may suffer, but it's easier to increase capacity



HDDs Have Latencies Dependent on the Distance Travelled

Rotational delay: physically rotate the disk to get to the correct sector

- Typically 4-8 ms (average delay is half of a full rotation)

Seek time: moving the disk arm to get to the correct track

- Typically 0.5-2 ms

Transfer time: how long it takes to read bytes from the disk

- Typically the maximum transfer speed is 125 MB/s

Calculating Transfer Rate

The total time, T, is equal to

rotational delay + seek time + transfer time

The transfer rate, R, is equal to

Size of the transfer / T

What is the transfer rate of

Large sequential accesses?

Small random accesses?

We Should Use HDDs Sequentially Whenever Possible

	HDD 1	HDD 2
Rotational speed	7,200 RPM	15,000 RPM
Rotational latency (ms)	4.2	2.0
Average seek (ms)	9	4
Max transfer	105 MB/s	125 MB/s
Platters	4	4
Interface	SATA	SCSI

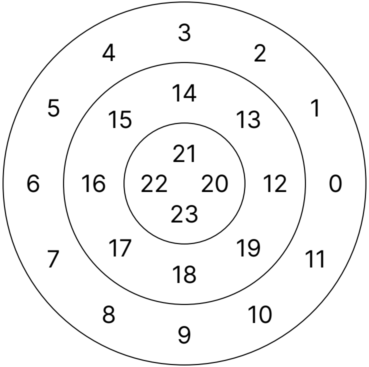
Sequential 100 MB read:

- HDD 1, T = 950 ms, R = 105 MB/s
- HDD 2, T = 800 ms, R = 125 MB/s

Random 4 KB read:

- HDD 1, T = 13.2 ms, R = 0.31 MB/s
- HDD 2, T = 6 ms, R = 0.66 MB/s

Logical Mapping Could Place All Sectors Next to Each Other



You may want to offset the sectors in different tracks so the head has time to settle

Track skew allows the disk to be efficient with minimal head movement

You May Want More Flexibility Than the Default Mapping

Pros

- Simple to program
- Default mapping reduces seek time for sequential access

Cons

- Filesystem can’t inspect or try to optimize the mapping
- Trying to reverse the mapping is difficult
    - Number of sectors per track changes
    - Disk silently remaps bad sectors

A Cache Can Significantly Speed Up Disk Transfers

Disks have some internal memory (WD Red - 64 MB) for caching

Implement a read-ahead “track buffer”

- Read the entire contents of the track into memory during the rotational delay

Write caching with volatile memory

- Write back: claim data is written to disk
  - Fast, but there’s data loss if there’s a power failure
  - Write through: acknowledge after data is physically written

We Can Schedule Disk Accesses

We want to minimize the time the disk moves without reading or writing data

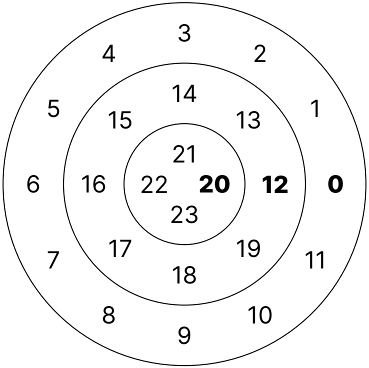
FCFS: schedule requests in the order received

- Fair, but it has a high seek and rotation cost

SSTF: shortest seek time first

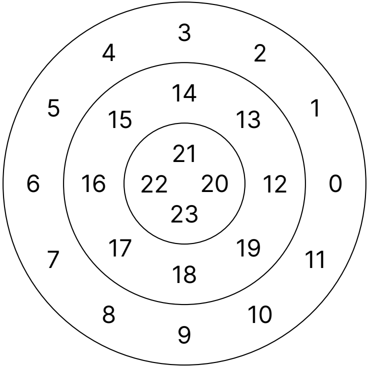
- Handle the nearest cylinder/sector next
  - Pro: reduces arm movement (seek time)
  - Con: unfair, can starve some requests

Elevator (aka SCAN or C-SCAN) Sweeps Across the Disk



If a request comes in for a track already serviced this sweep, queue it for the next

Elevator (or SSTF) Ignores Rotation



Shortest positioning time first (SPTF) is often the best strategy

The OS and disk need to work together to implement this

Disks Enable Persistence

We explored two HDDs today:

- Magnetic disks have poor random access (need to be scheduled)
- Shortest Positioning Time First (SPTF) is the best scheduling for throughput