- [x] 1. Last time
- [x] 2. Disks (HDDs)
- [ ] 3. Intro to file systems

---

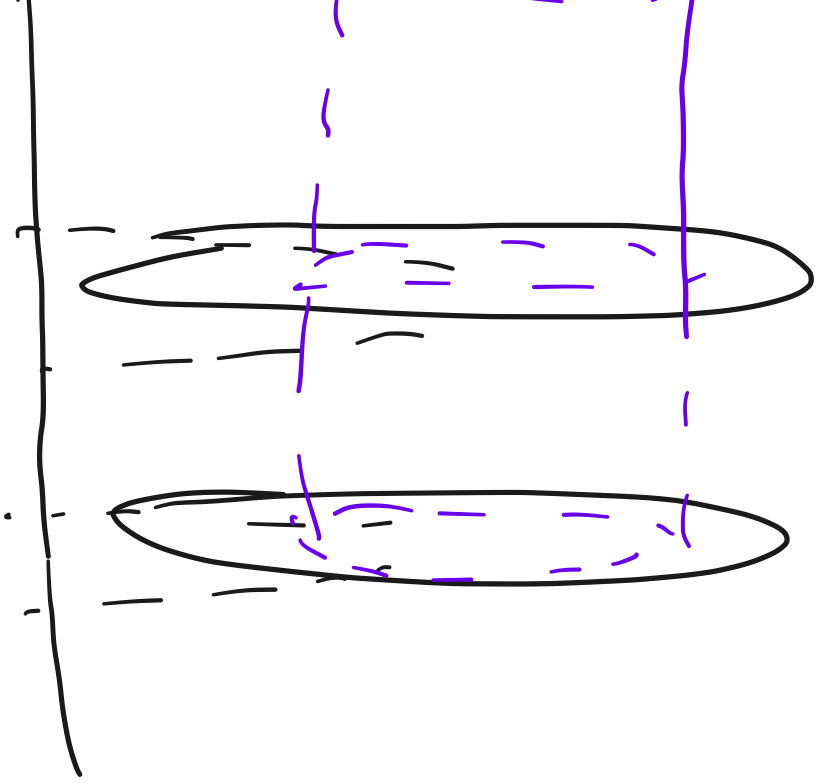
2. Disks

- [x] A. What is a disk?
- [x] B. Geometry
- [x] C. Performance
- [x] D. Common numbers
- [x] E. Interface to disk
- [x] F. Performance II
- [x] G. Disk scheduling (performance III)
- [x] H. Technology + systems trends

$/GB

- stack of magnetic platters
- rotate together on central spindle

3600 - 15,000 RPM (=60 - 250 rot/sec)



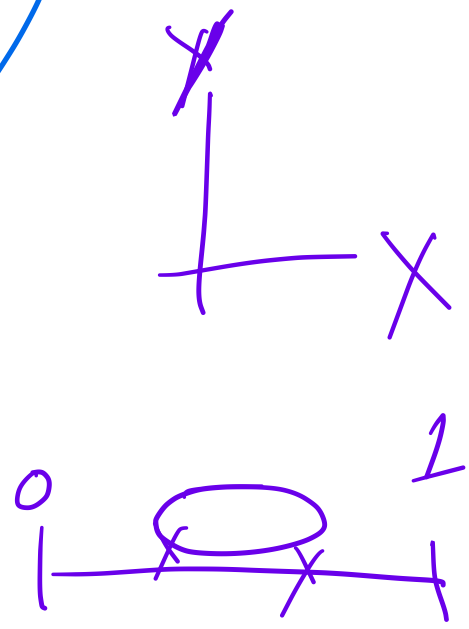
Arms move together

Arms contain disk heads; heads read + write to platters

Geometry

- track: circle on a platter
- sector: chunk of a track
- cylinder: all tracks of fixed radius on all platters
- generally only one head active at a time
- disk positioning system: moves head to a track and keeps it there

- seek: 4 phases: speedup, coast, slowdown, settle



Performance

Components of total transfer time:
- rotational delay
- seek delay
- transfer time

"Avg seek time"
- "time to seek 1/3 of the disk", but manufacturers might report:

1/3 of "time to seek the whole disk"

Common #s

Capacity: TBs common (10^{12} bytes vs. 2^{40} bytes)

Platters: 8

... of thousands or more

Number of cylinders: tens of thousands

Sectors per track: ~1000

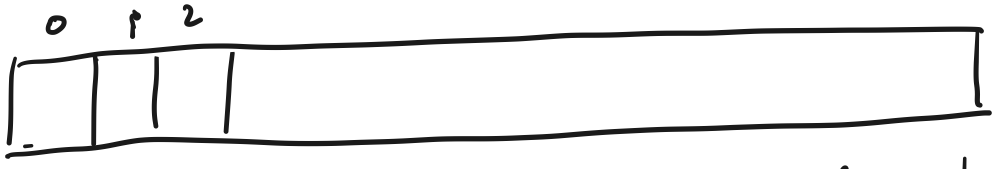
RPM: 10,000

Transfer rate: 50-150 MB/s

MTBF: ~1 million hours

How driver interfaces to disk

Sectors



Interface to disk is a linear array of sectors

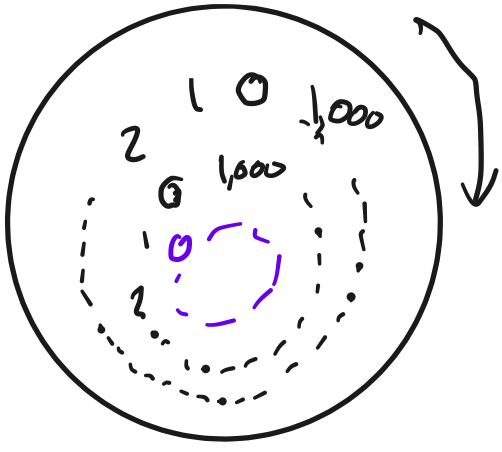
Sector: 512 bytes, moving to 4KB written atomically

Disk does some cool things under the hood (invisibly to OS)

Zoning

Skewing

Sparing



Disk performance example

Spindle speed: 12,000 RPM

Avg. seek time: 12 ms

Transfer rate: 128 MB/s

Sector: : 512 bytes

12,000 RPM = $\frac{1200}{6}$ = 200 rot/sec

$\frac{12{,}000 \text{ rot}}{\text{min}} \cdot \frac{1 \text{ min}}{60 \text{ sec}} = \frac{200 \text{ rot}}{\text{sec}}$

$\frac{1 \text{ s}}{128 \text{ MB}} \times 512 \text{ B} = \frac{2^9}{2^{27}} = 2^{-18} \text{ s} = 4 \mu\text{s}$

(a) What is the throughput if doing 500 sector reads, spread randomly over the disk and serviced in FIFO order?

(b) Same question, but now the reads are sequential

(a)

$\text{TPUT} = \frac{\text{bytes}}{\text{time}} = \frac{500 \times 512}{500(\text{rot} + \text{seek} + \text{transfer})}$

$= \frac{500 \times 512}{500(8.5\text{ms} + 12\text{ms} + 4\mu\text{s})}$

$= \frac{512}{14.5\text{ms}} \approx \frac{512 \times 70 \text{ B}}{1 \text{ s}} \approx \boxed{\frac{35 \text{ kB}}{\text{s}}}$

(4) $TPUT = \frac{bytes}{time} = \frac{500 \times 512 \text{ bytes}}{2.5ms + 12ms + \frac{1s}{128MB} \times 500 \times 512 bytes} \approx \frac{512^2}{2.5 + 12 + \frac{1}{128} \cdot 512^2}$

$= \frac{2^{18} bytes}{14.5ms + \frac{2^{18} bytes \cdot s}{2^7 bytes}} = \frac{2^{18} bytes}{14.5ms + 2^{-9} s.}$

$= \frac{2^{18} bytes}{14.5ms + \approx 2ms}$

$= \frac{2^{18} bytes}{16.5ms}$

$\approx \frac{2^{18} bytes \times 60}{1000 ms} \approx \frac{2^{18} \cdot 2^6}{1s}$

$\approx \frac{2^{24}}{s} = \boxed{\frac{16MB}{s}}$

Intro to file systems

What does a FS do?

- provide persistence
- create a way to name data on the disk

FS: can be implemented in lots of places

- We focus on the disk, generalize later

Note: disk is the 1st thing we've seen that is **both** modifiable **and** persistent.

Files

What is a file?

From user's view: a named, contiguous run of bytes

From FS's view: collection of disk blocks

Job of a FS:

map {file, offset in file} $\xrightarrow{FS}$ disk address

operations:

create(file), delete(file), read(), write()

Goal: operations have as few disk accesses as possible and minimal space overhead