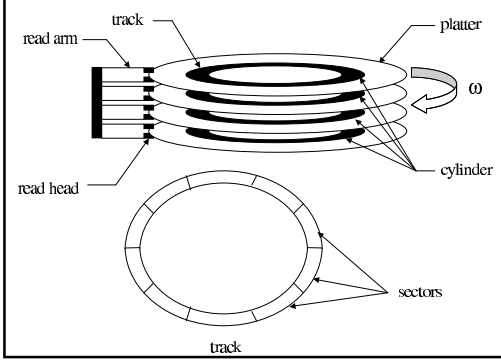
Disk Basics

CS 537 - Introduction to Operating Systems

Hard Drive Technology



Hard Drive Technology

- Modern Statistics
  - capacity: 20.5 GB
  - sector size: 512 bytes
  - speed: 7200 RPM
  - platters: 3
  - heads: 6
  - cylinders: 17,688
  - media transfer rate: 35 MB/sec
  - interface transfer rate: 14.1 - 23.4 MB/sec
  - cost: $127.00

Hard Drive Terminology

- Seek Latency
  - time to move the read head to correct cylinder
    - avg = 9.0 ms
    - track-to-track = 2.2 ms
    - full-track = 15 ms
- Rotational Latency
  - time for disk to rotate to needed sector
    - avg = 4.17 ms
- Transfer Latency
  - time to actually read and send the data
    - ≈50 μs for 1 KB of data

Disk Access

- Cannot access any random byte on a disk
  - this would be called random access (RAM)
- However, don't have to access them sequentially
  - have to go through all data to get to desired data (tape drive)
- Disks must be accessed per sector
  - each sector is 512 bytes
  - this is called direct access

Reading from Disk

- To read a single byte from disk
  - provide address of sector
    - cylinder, head, sector number
  - seek to location of data on disk
  - read entire sector into memory (512 bytes)
  - index into memory to grab the necessary byte

Writing to Disk

- To write a single byte out to disk
  - read entire sector into memory
  - modify the necessary byte in memory
  - seek back to disk location
  - write the entire sector back to disk
- Notice 2 times avg. latency required to write a single block

Disk Performance

- Disk is a mechanical device
  - extremely SLOW!!
- Can use DMA and context switching to hide latency
  - this will only help so much
- Want to interact with physical disk as little as possible
- Reading and writing a single byte is a bad idea

Prefetching Blocks

- If the user wants data in sector 23, odds are they will eventually want data in sector 24
  - this assumes intelligent placement of data on disk to start with - more on this later
- Amortize the latencies by reading in sector 23 and sector 24
  - when user does actually want sector 24, already in memory
- Most disks are read in a page at a time
  - 4 KB of data (8 sectors)

Disk Buffering

- If byte 55 of sector 26 is modified, high probability byte 56 of sector 27 will also be modified
- Usually wait awhile to write back a sector that is currently in memory
  - hopefully wait until all data that will be modified is modified
  - again, allows amortization of latencies over a large number of writes