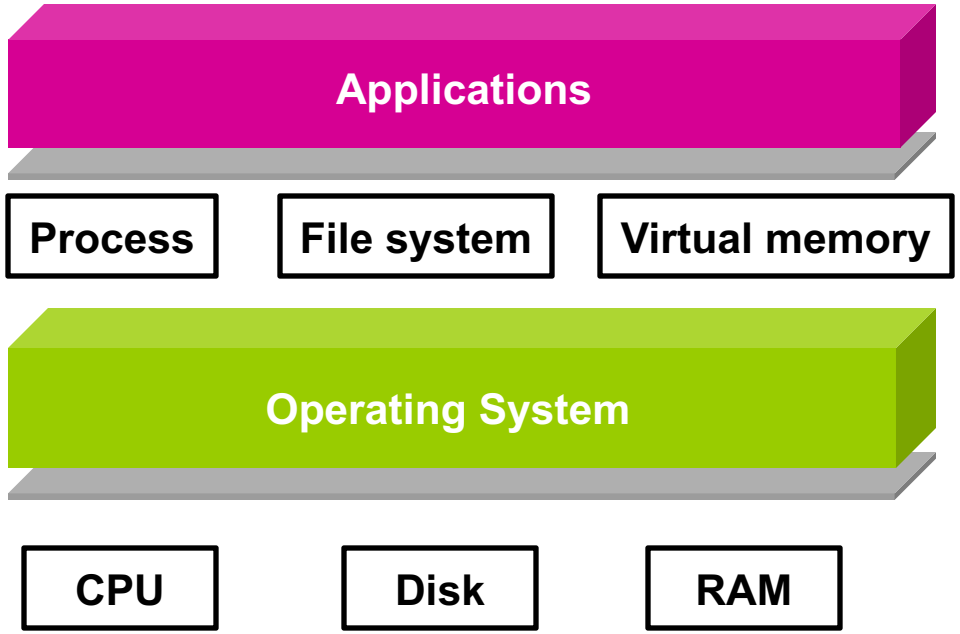
CSE 153
Design of Operating Systems

Winter 2023

Lecture 19: File Systems (1)—Disk drives

OS Abstractions



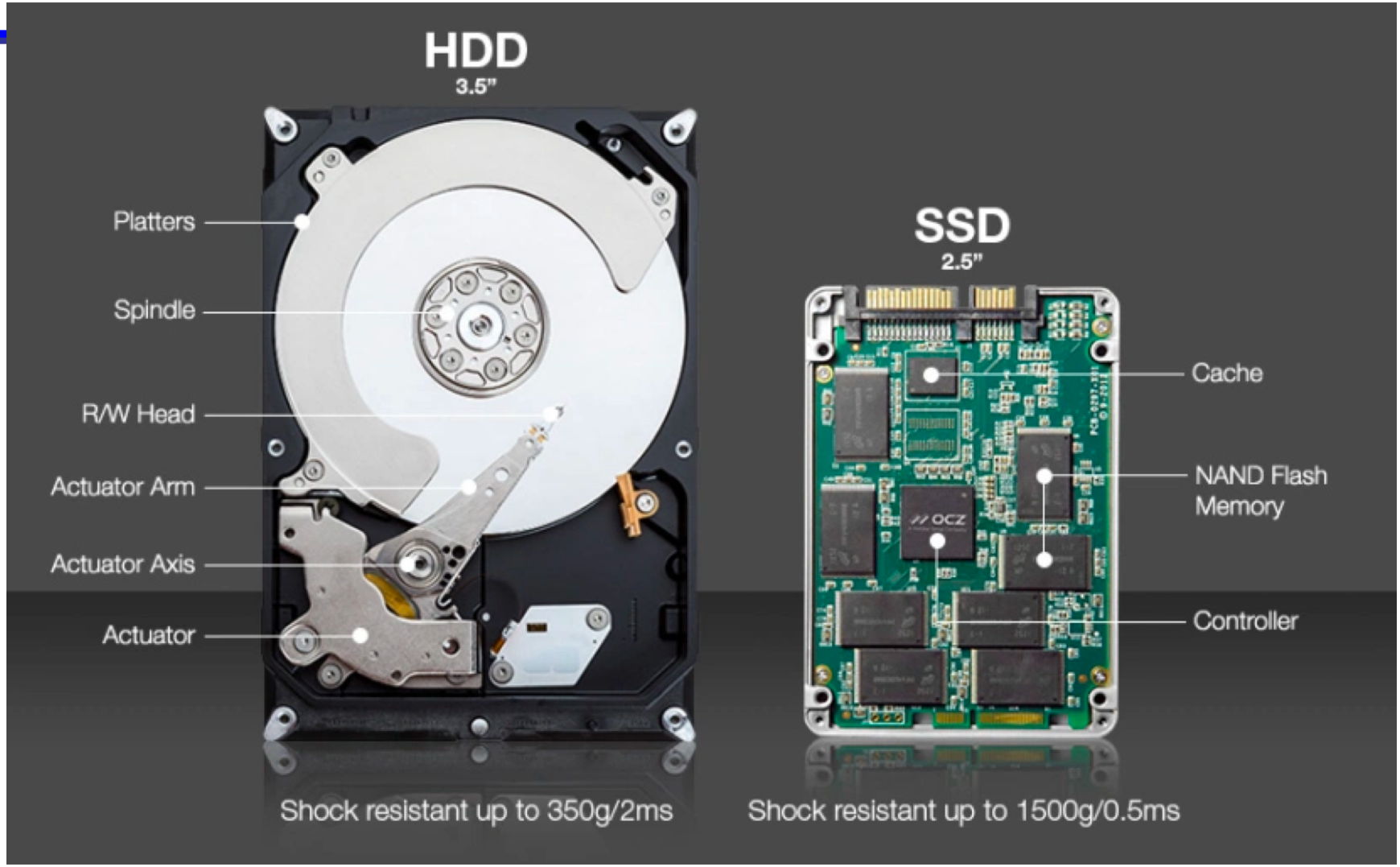
File Systems Agenda

- First we'll discuss properties of physical disks
  - Structure, Performance
  - Scheduling
- Then we'll discuss how to build file systems (next time)
  - Abstraction:
    - » Files
    - » Directories
    - » Sharing
    - » Protection
  - Implementation
    - » File System Layouts
    - » File Buffer Cache
    - » Read Ahead

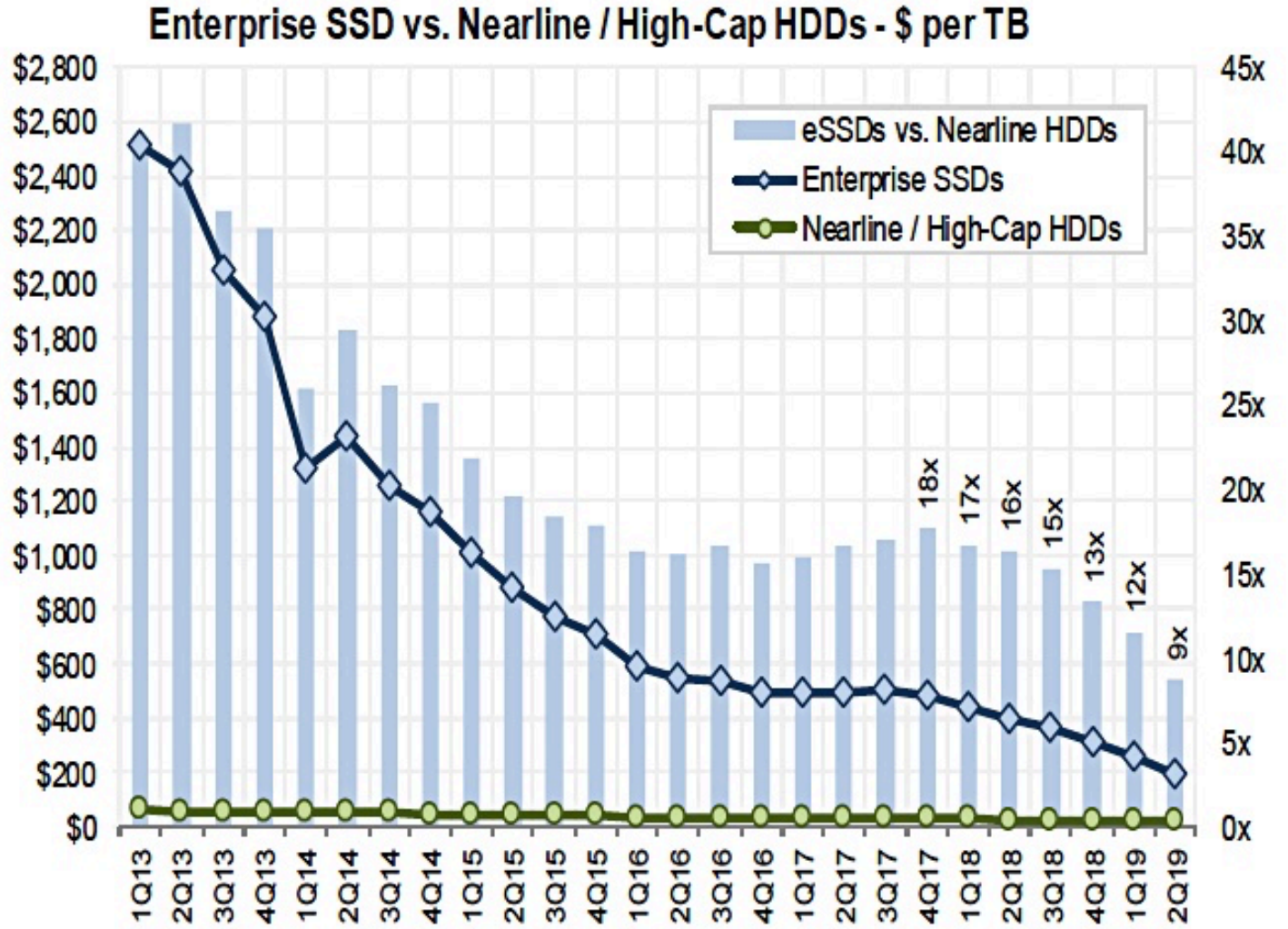
Disks and the OS

- Disks/SSDs are messy physical devices:
  - Disks: errors, bad blocks, missed seeks, etc.
  - SSD: limited wear cycles, block erase/write, …
- OS hides this mess from higher level software
  - Low-level device control (initiate a disk read, etc.)
- Os provides Higher-level abstractions (e.g., files)
  - OS maps them to the device and implements policies for
    - » Performance, reliability, protection, …

Disk vs. SSD

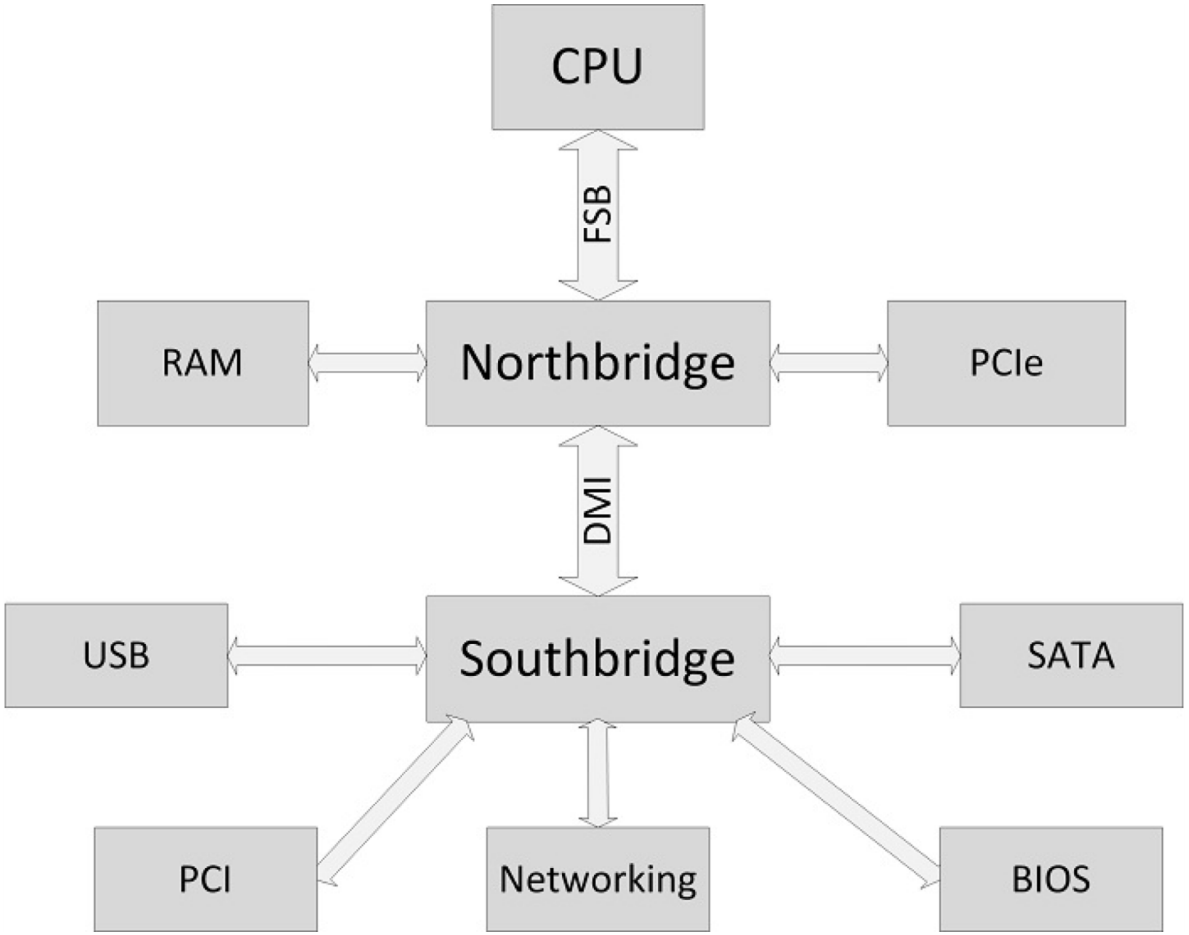

HDD
3.5"
Platters
Spindle
R/W Head
Actuator Arm
Actuator Axis
Actuator
Shock resistant up to 350g/2ms
SSD
2.5"
Cache
NAND Flash Memory
Controller
Shock resistant up to 1500g/0.5ms

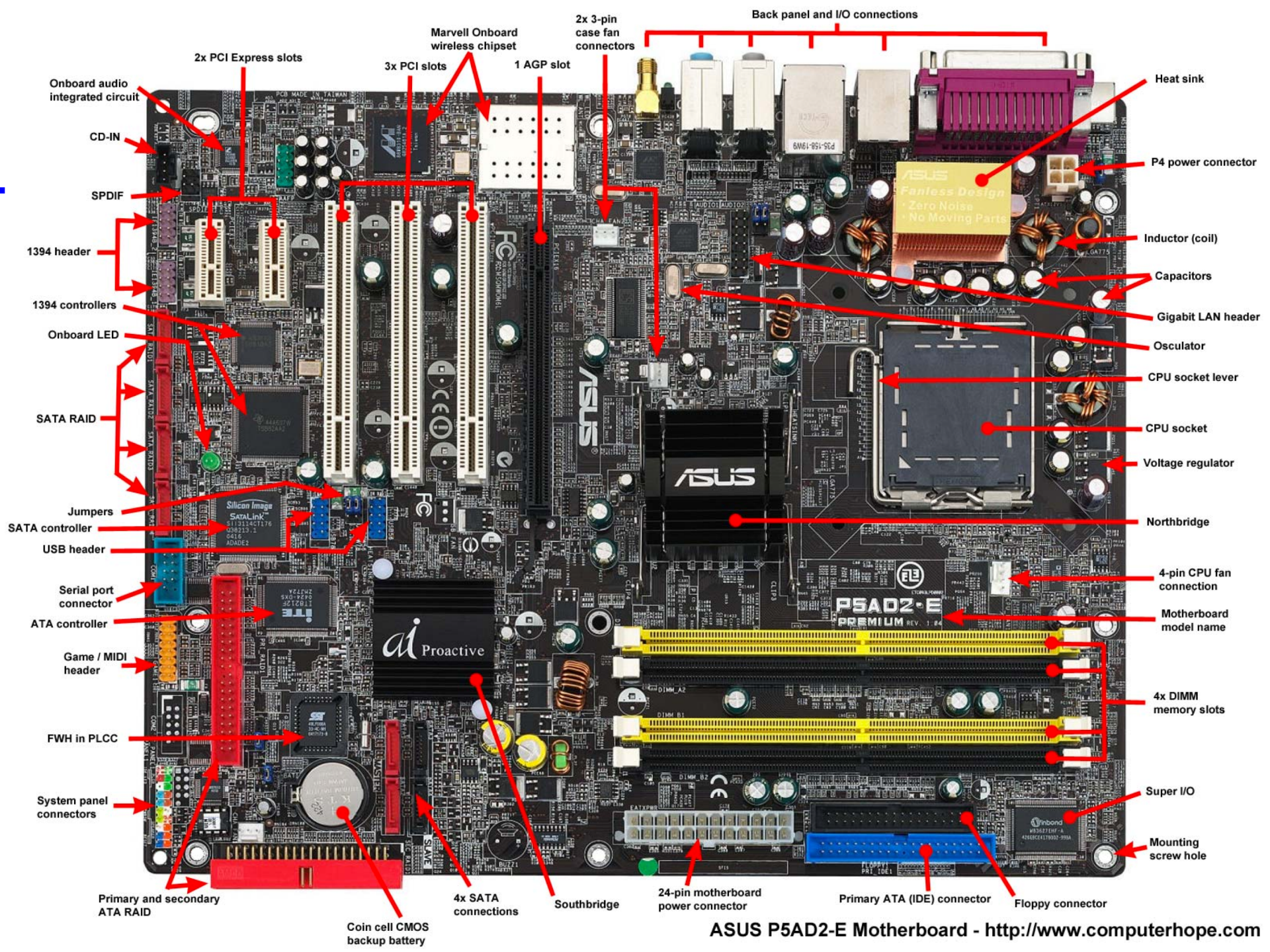
Gap is closing



Source: IDC; TrendFocus; Wells Fargo Securities, LLC

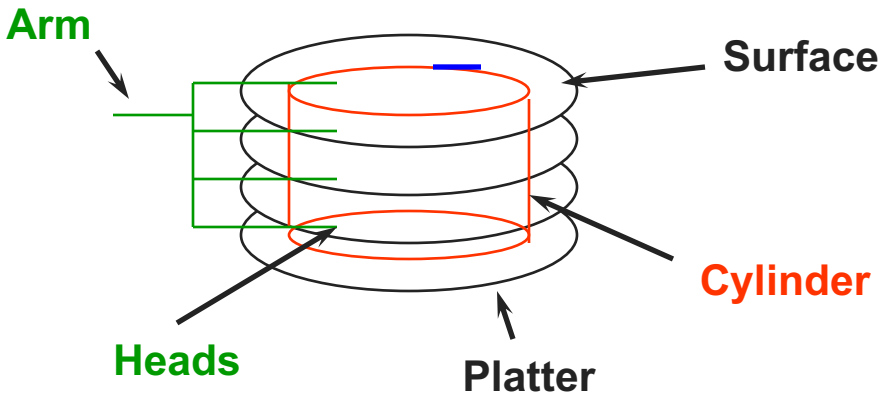
I/O and disk in the system

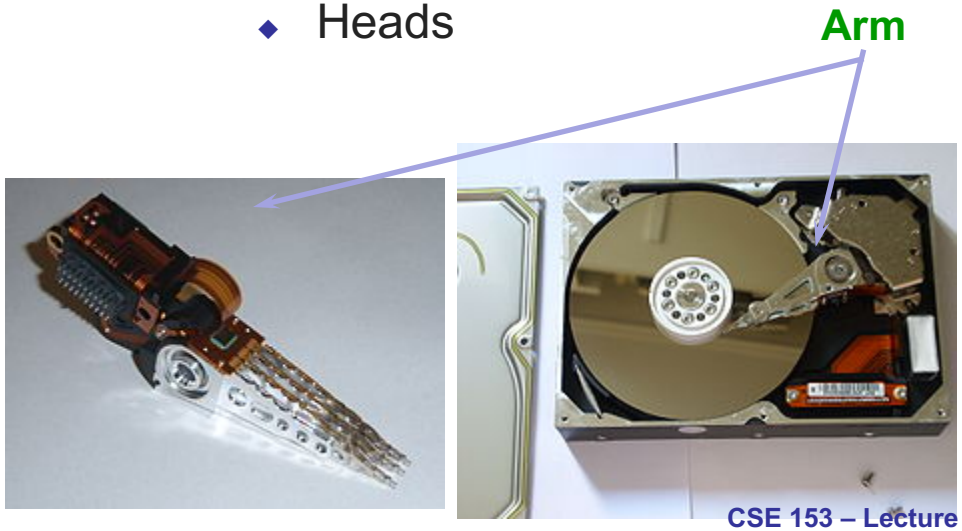


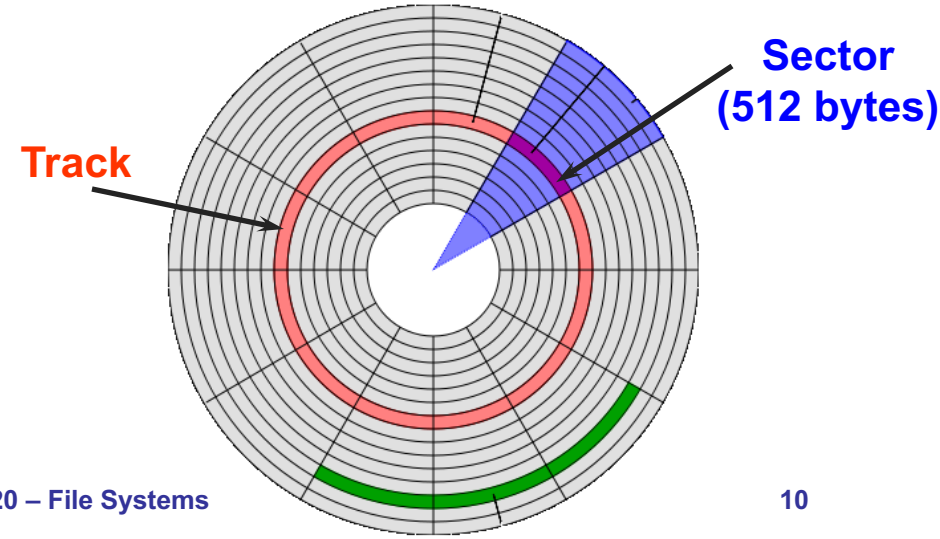

Back panel and I/O connections
2x 3-pin case fan connectors
Marvell Onboard wireless chipset
2x PCI Express slots
3x PCI slots
1 AGP slot
Onboard audio integrated circuit
CD-IN
SPDIF
1394 header
1394 controllers
Onboard LED
SATA RAID
Jumpers
SATA controller
USB header
Serial port connector
ATA controller
Game / MIDI header
FWH in PLCC
System panel connectors
Primary and secondary ATA RAID
Coin cell CMOS backup battery
4x SATA connections
Southbridge
24-pin motherboard power connector
Primary ATA (IDE) connector
Floppy connector
Heat sink
P4 power connector
Inductor (coil)
Capacitors
Gigabit LAN header
Osculator
CPU socket lever
CPU socket
Voltage regulator
Northbridge
4-pin CPU fan connection
Motherboard model name
4x DIMM memory slots
Super I/O
Mounting screw hole
ASUS P5AD2-E Motherboard - http://www.computerhope.com

Physical Disk Structure

- Disk components
  - Platters
  - Surfaces
  - Tracks
  - Sectors
  - Cylinders
  - Arm
  - Heads

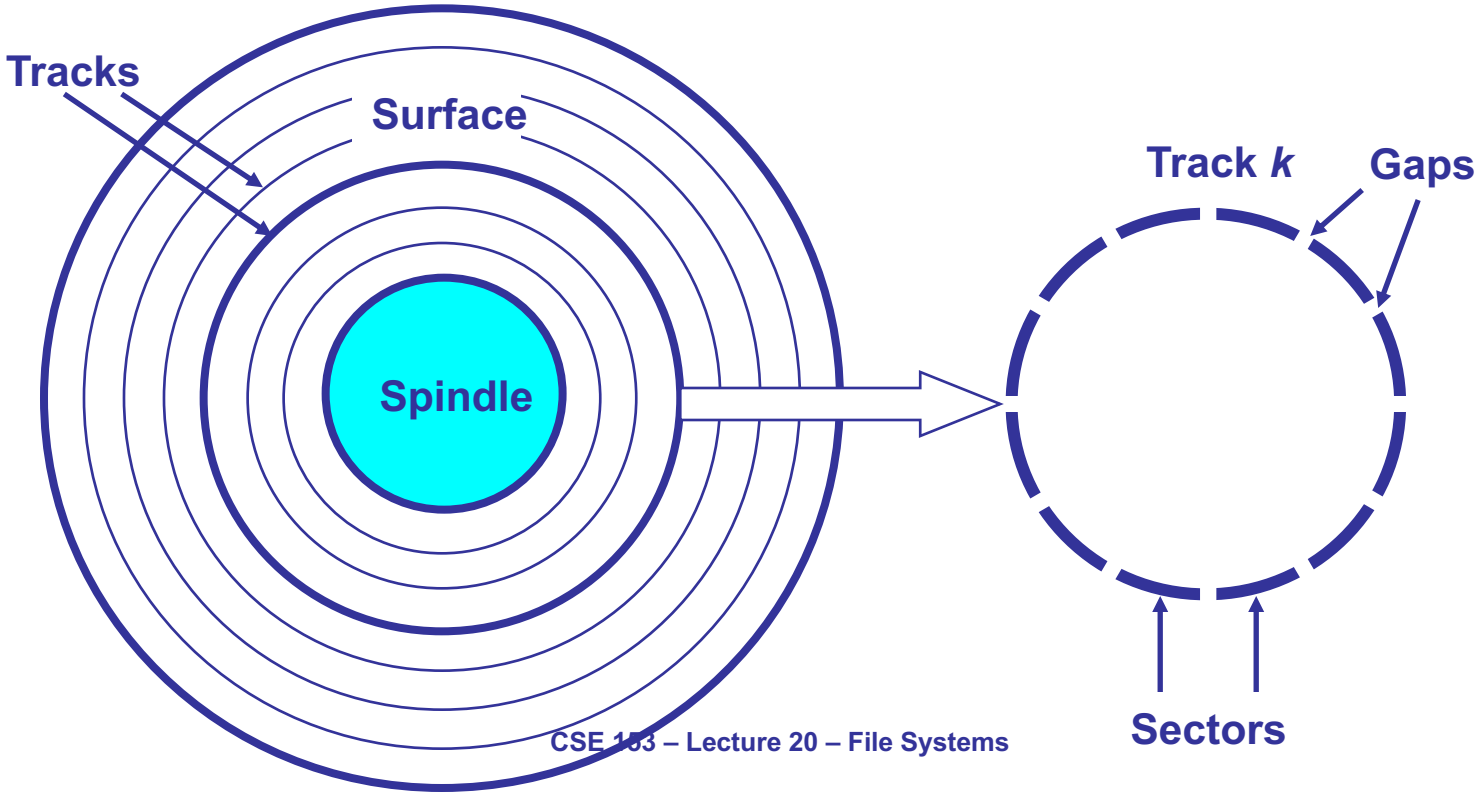






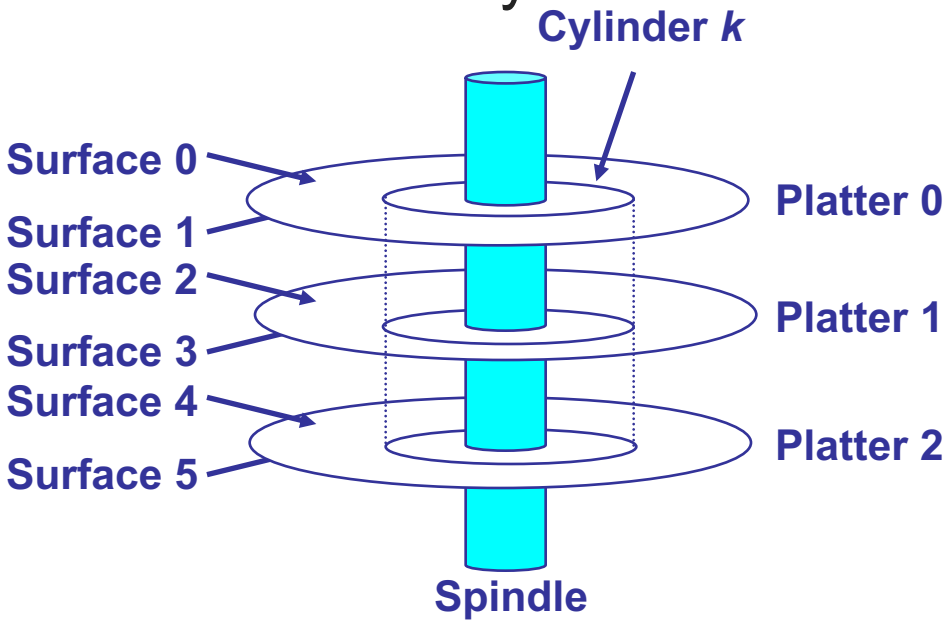
Disk Geometry

- Disks consist of platters, each with two surfaces.
- Each surface consists of concentric rings called tracks.
- Each track consists of sectors separated by gaps.



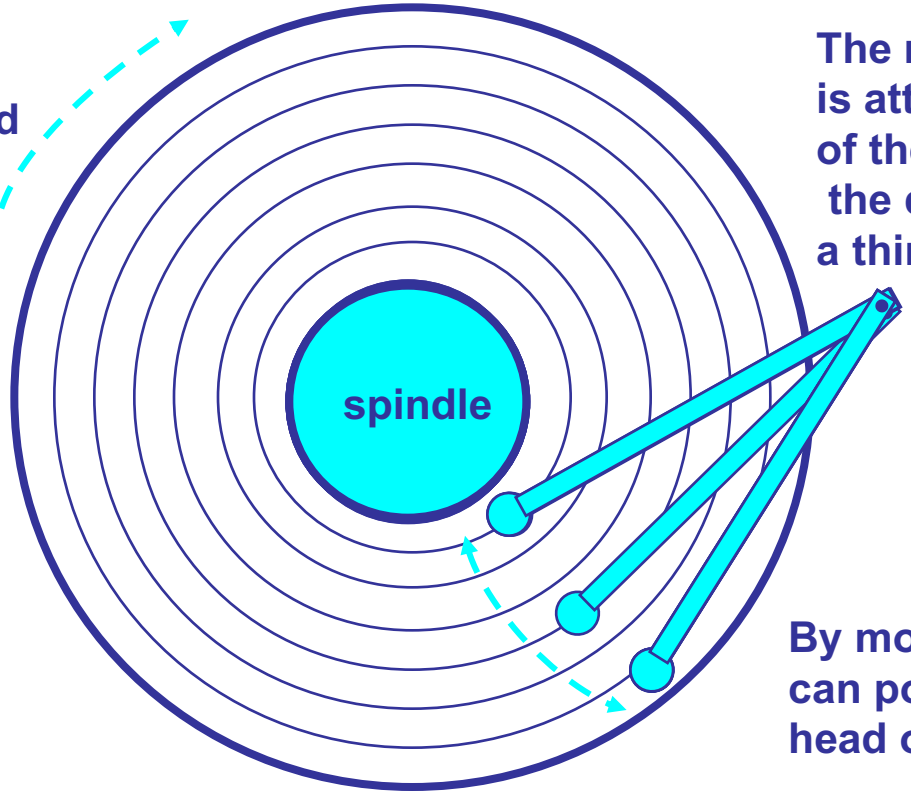
Disk Geometry (Muliple-Platter View)

- Aligned tracks form a cylinder.



Disk Operation (Single-Platter View)

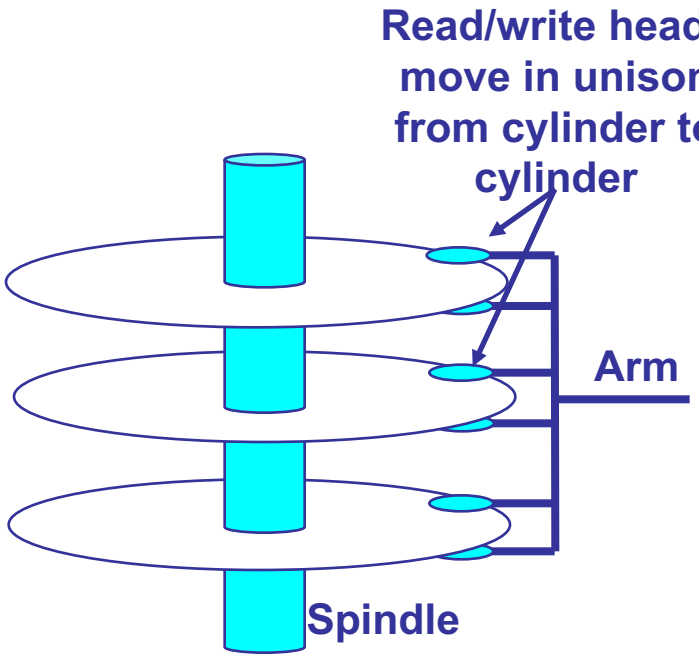
The disk surface spins at a fixed rotational rate



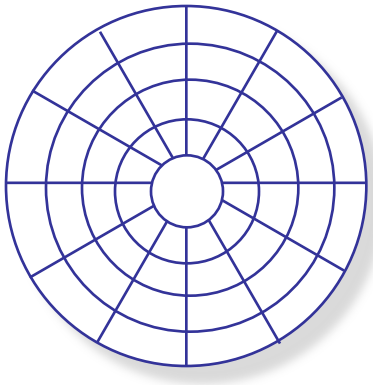
The read/write *head* is attached to the end of the *arm* and flies over the disk surface on a thin cushion of air.

By moving radially, the arm can position the read/write head over any track.

Disk Operation (Multi-Platter View)



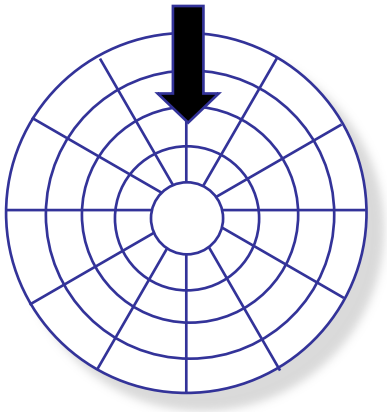
Disk Structure - top view of single platter



Surface organized into tracks

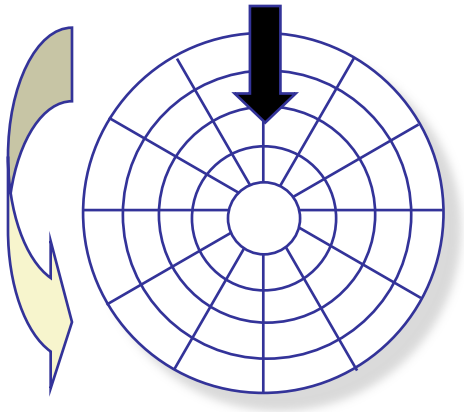
Tracks divided into sectors

Disk Access



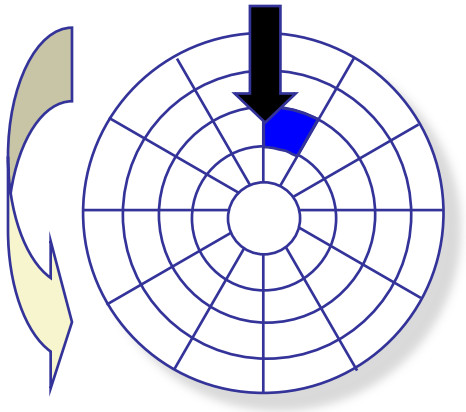
Head in position above a track

Disk Access



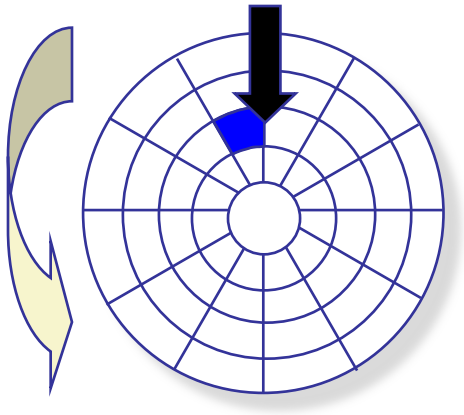
Rotation is counter-clockwise

Disk Access – Read



About to read blue sector

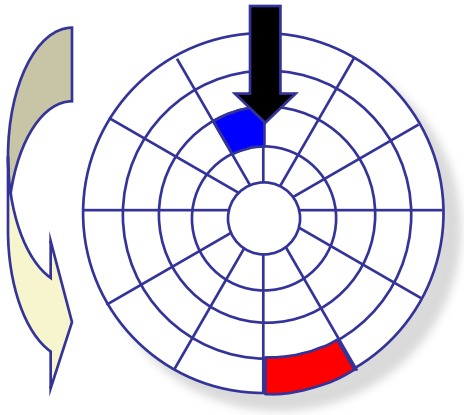
Disk Access – Read



After **BLUE** read

After reading blue sector

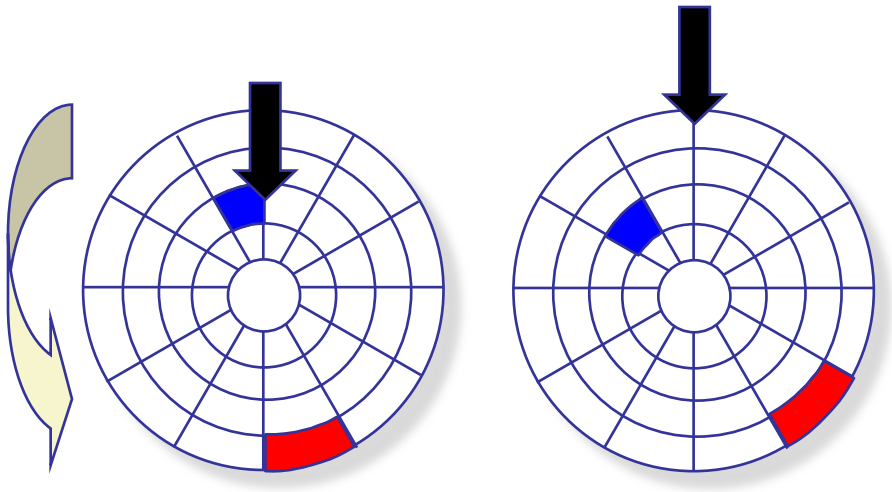
Disk Access – Read



After BLUE read

Red request scheduled next

Disk Access – Seek

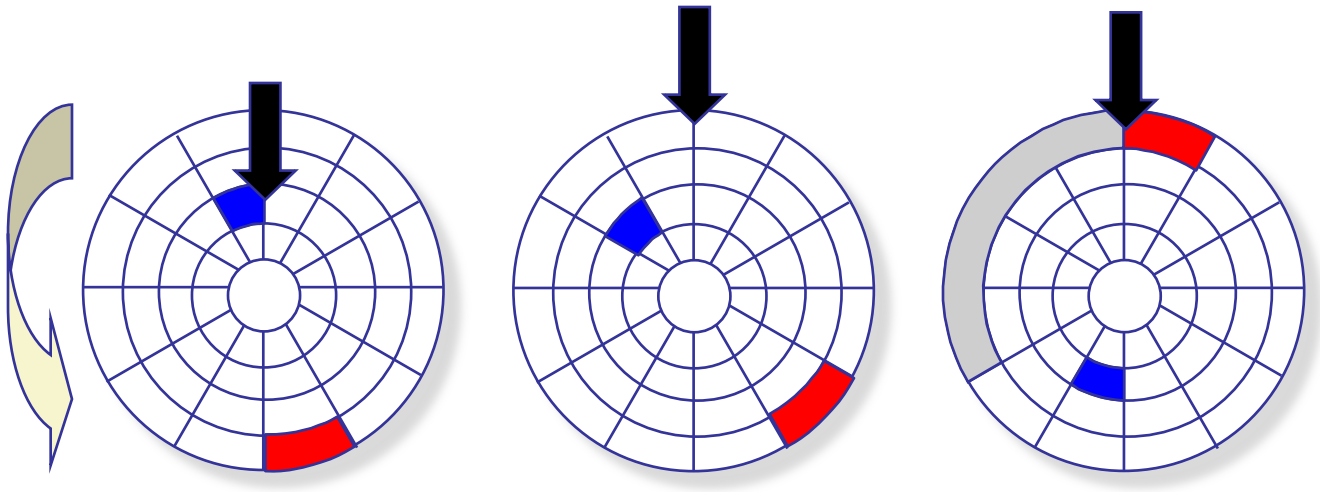


After BLUE read

Seek for RED

Seek to red's track

Disk Access – Rotational Latency



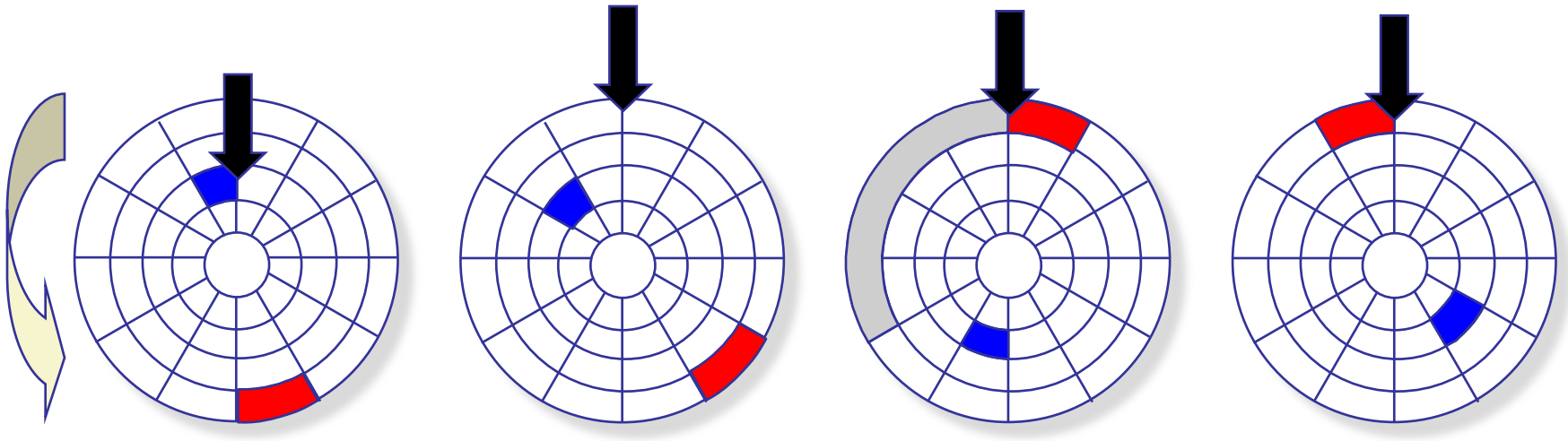
After BLUE read

Seek for RED

Rotational latency

Wait for red sector to rotate around

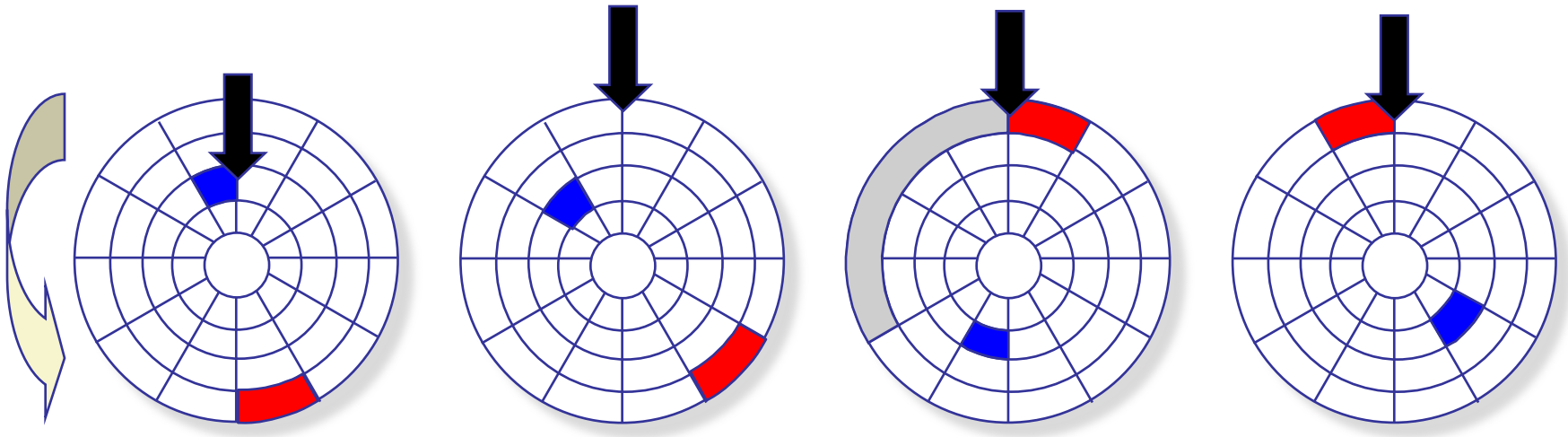
Disk Access – Read



After BLUE read | Seek for RED | Rotational latency | After RED read

Complete read of red

Disk Access – Service Time Components



Disk Access Time

- Average time to access some target sector approximated by :
  - Taccess = Tavg seek + Tavg rotation + Tavg transfer
- Seek time (Tavg seek)
  - Time to position heads over cylinder containing target sector.
  - Typical Tavg seek is 3—9 ms
- Rotational latency (Tavg rotation)
  - Time waiting for first bit of target sector to pass under r/w head.
  - Tavg rotation = 1/2 x 1/RPMs x 60 sec/1 min
  - Typical Tavg rotation = 7200 RPMs
- Transfer time (Tavg transfer)
  - Time to read the bits in the target sector.
  - Tavg transfer = 1/RPM x 1/(avg # sectors/track) x 60 secs/1 min.

Disk Access Time Example

- Given:
  - Rotational rate = 7,200 RPM
  - Average seek time = 9 ms.
  - Avg # sectors/track = 400.
- Derived:
  - Tavg rotation = 1/2 x (60 secs/7200 RPM) x 1000 ms/sec = 4 ms.
  - Tavg transfer = 60/7200 RPM x 1/400 secs/track x 1000 ms/sec = 0.02 ms
  - Taccess = 9 ms + 4 ms + 0.02 ms
- Important points:
  - Access time dominated by seek time and rotational latency.
  - First bit in a sector is the most expensive, the rest are free.
  - SRAM access time is about 4 ns/doubleword, DRAM about 60 ns
    - » Disk is about 40,000 times slower than SRAM,
    - » 2,500 times slower then DRAM.

Disks Heterogeneity

- Seagate Barracuda 3.5" (workstation)
  - capacity: 250 - 750 GB
  - rotational speed: 7,200 RPM
  - sequential read performance: 78 MB/s (outer) - 44 MB/s (inner)
  - seek time (average): 8.1 ms
- Seagate Cheetah 3.5" (server)
  - capacity: 73 - 300 GB
  - rotational speed: 15,000 RPM
  - sequential read performance: 135 MB/s (outer) - 82 MB/s (inner)
  - seek time (average): 3.8 ms
- Seagate Savvio 2.5" (smaller form factor)
  - capacity: 73 GB
  - rotational speed: 10,000 RPM
  - sequential read performance: 62 MB/s (outer) - 42 MB/s (inner)
  - seek time (average): 4.3 ms

Logical Disk Blocks

- Modern disks present a simpler abstraction:
  - The set of available sectors is modeled as a sequence of b-sized logical blocks (0, 1, 2, ...)

- Mapping between logical and actual (physical) sectors
  - Maintained by a device called disk controller.
  - Converts requests for logical blocks into (surface,track,sector)
  - Allows controller to set aside spare cylinders for each zone.
  - Accounts for the difference in “formatted capacity” and “maximum capacity”.

Disk Scheduling

- Because seeks are so expensive (milliseconds!), OS schedules requests that are queued waiting for the disk
  - FCFS (do nothing)
    - » Reasonable when load is low
    - » Does nothing to minimize overhead of seeks
  - SSTF (shortest seek time first)
    - » Minimize arm movement (seek time), maximize request rate
    - » Favors middle blocks, potential starvation of blocks at ends
  - SCAN (elevator)
    - » Service requests in one direction until done, then reverse
    - » Long waiting times for blocks at ends
  - C-SCAN
    - » Like SCAN, but only go in one direction (typewriter)

Disk Scheduling (2)

- In general, unless there are request queues, disk scheduling does not have much impact
  - Important for servers, less so for PCs
- Modern disks often do the disk scheduling themselves
  - Disks know their layout better than OS, can optimize better
  - Ignores, undoes any scheduling done by OS