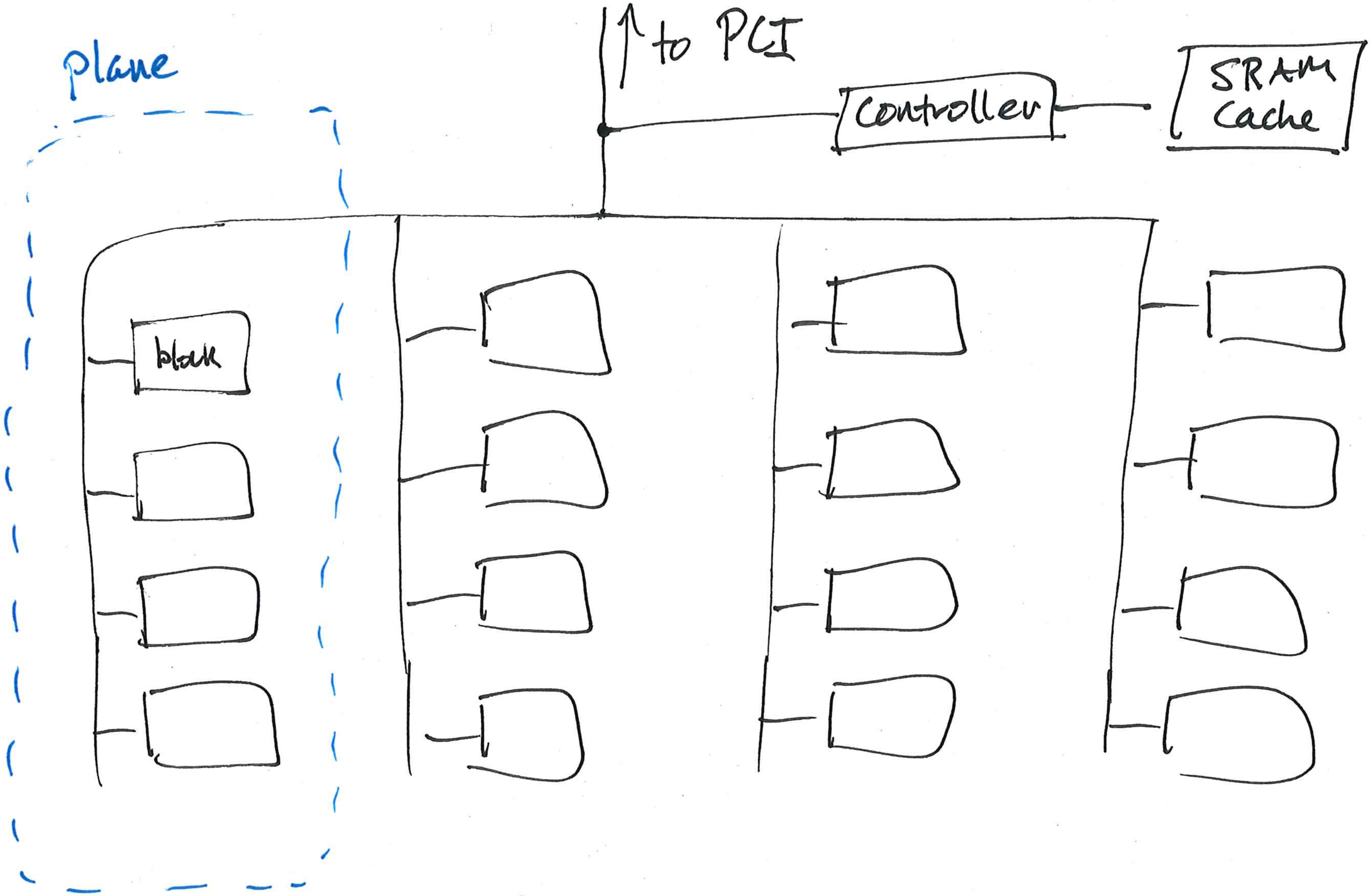
Scaling SSD Bandwidth



each plane operates in parallel

Striping

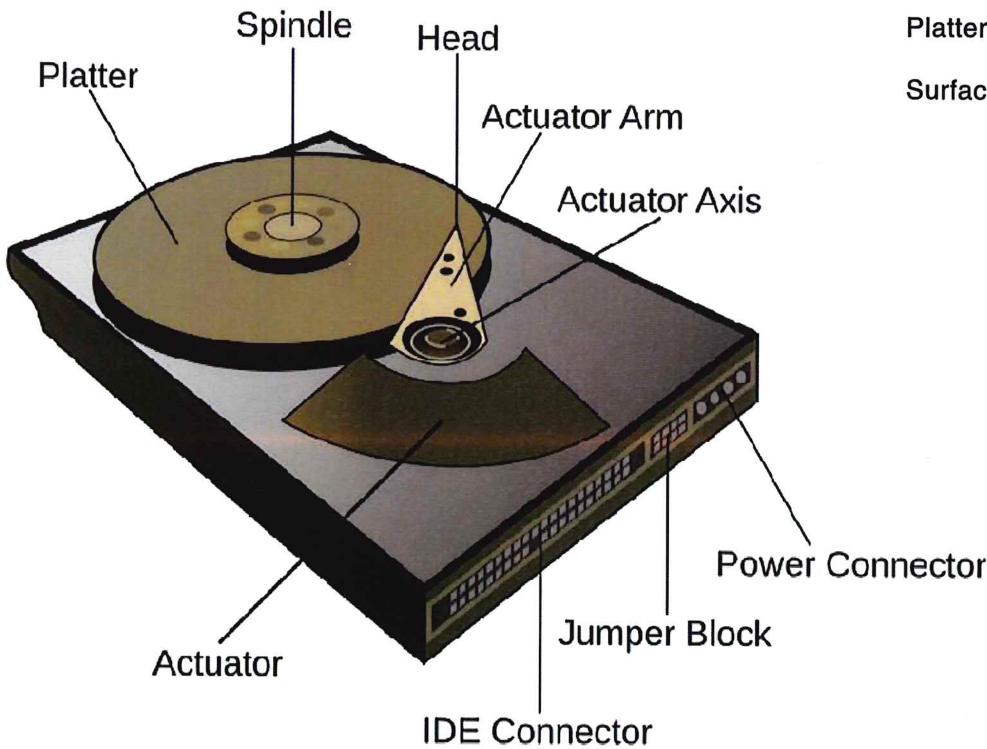
- data is striped across planes to improve perf
  - but that means data is spread across many erasure blocks → high fragmentation
- compaction of unrelated data in same plane can cause slowdowns for reads in that plane

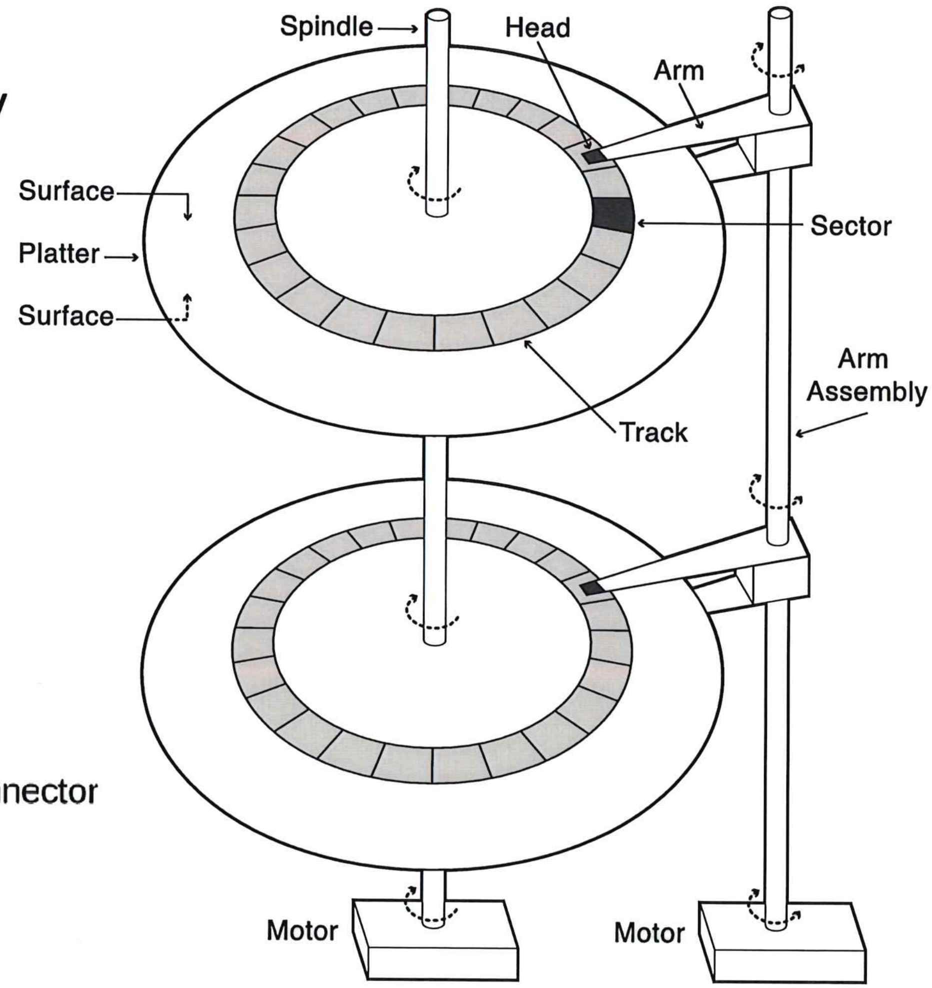
Solution: Zoned namespaces

- allocate separate regions of complete erasure blocks
  - restrict to sequential writes, (random reads ok)
  - no need for compaction, no spare capacity

Spinning Disks

Magnetic Disk Anatomy





18.1

Disk tracks

- ~1 μm wide
  - wavelength of light ≈ 500 nm = 0.5 μm
  - ~100K tracks on 2.5" disk
- Tracks separated by guard regions
  - bit-level corruption does happen
- More sectors per track near outside of disk
  → more bandwidth
  - only outer half of disk is used (= 75% total area)
- Reading a complete track <u>much</u> faster than reading same # of random sectors

18.2

Sectors

- 512 bytes typical
- one sector is the smallest unit read/written
- sector contains error correcting codes
- spare sectors: remap bad sectors
- "slip sparing": remap all sectors after bad sector
  - preserves sequential behavior
- track skewing: logically first sector of each track is offset → sequential reads across tracks

Disk performance

disk latency = seek + rotation + transfer

- seek: - move arm over track (1-20 ms) avg: $\frac{1}{3}$rd of max
  - head must "settle" to align exactly over track
  - head switch (ie, surface/platter switch) ≈ track switch
- rotate: - wait for desired sector to rotate under head
  - on average, about half a rotation
  - full rotation = 4-15 ms depending on RPM
- transfer: - read/write the desired sectors
  - 100-250 MB/s disk head transfer rate
  - may be limited by host transfer rate

Example

suppose a disk has avg seek time of 8 ms

avg rotational time of 4 ms

transfer rate 250 MB/s

7200 RPM
= 8.3 ms/rotation

How long to complete 100 random 4KB reads?

$4\text{KB} / 250\text{ MB/s} = 16\ \mu s$

$100 \times (8_{ms} + 4_{ms} + 0.016\,ms) = 1200\ ms = 1.2\ s$

note transfer time negligible (!)

How long to read 100 sequential

transfer time = $400\ \text{KB} / 250\ \text{MB} = 1.6\ ms$

total = $8 + 4 + 1.6 = 13.6\ ms$

≈ 1 random read (!)

Efficient Disk Usage

How large a transfer needed to achieve 80% efficiency?

12 ms to reach first sector

assume x rotations needed, 8.3 ms / rotation

$$0.8 \times (12\text{ms} + 8.3x) = 8.3x$$

$$12\text{ms} = \frac{1}{4} \cdot 8.3x$$

$$x = 5.7 \text{ rotations} = 48\text{ms}$$

$$250 \text{ MB/s} \cdot 48\text{ms} = 12\text{MB}$$

Disk Scheduling

- what order should we process outstanding requests in?

FIFO? not optimal

bad workload:

Shortest seek time first?

not optimal!

bad work load