Parts of a drive:

What atomic guarantees do we have?

What speed assumptions are generally true?

10,000 rpm  nearby sectors are faster

How long does an access to sector 0 take?

Multiple Tracks

Seek Track To Track

Figure 37.3: Three Tracks Plus A Head (Right: With Seek)

Disk Cache:

Buffer cache being read vs the data being written? writeback vs. write-through faster so far
Disk transfer metrics

Time for I/O: $T_{\text{seek}} + T_{\text{rotate}} + T_{\text{transfer}} + T_{\text{I/O}}$
Rate of I/O: $\frac{\text{Sum of transfer}}{T_{\text{I/O}}}$

Workloads:
- Most likely access is sequential
- Random

Let's compare two different drives

<table>
<thead>
<tr>
<th></th>
<th>Cheetah</th>
<th>Barracuda</th>
</tr>
</thead>
<tbody>
<tr>
<td>Capacity</td>
<td>300 GB</td>
<td>1 TB</td>
</tr>
<tr>
<td>RPM</td>
<td>15,000</td>
<td>7,200</td>
</tr>
<tr>
<td>Average Seek</td>
<td>4 ms</td>
<td>9 ms</td>
</tr>
<tr>
<td>Max Transfer</td>
<td>125 MB/s</td>
<td>105 MB/s</td>
</tr>
<tr>
<td>Platters</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Cache</td>
<td>16 MB</td>
<td>16/32 MB</td>
</tr>
<tr>
<td>Connects via</td>
<td>SCSI</td>
<td>SATA</td>
</tr>
</tbody>
</table>

HW: given the book’s calculation of the time needed, how much is the Cheetah worth to you?

Take-away: Cheetah is faster & small

Q: Why does the OS care...? A: It can perform Disk Scheduling

How: FCFS

Why: simple & fair

Why not: inefficient

SSTF: Shortest Seek Time First

Idea: sort outstanding requests by seek time efficiently

Issue #1: starvation

Issue #2: ignore rotation time

not fair
Elevator (a.k.a. SCAN or C-SCAN)

Idea:

Variations:
- F-SCAN: Freeze Scan
- C-SCAN: Circular Scan (0, 1, 2, 3, 4, ..., 000) but less efficient and fair

Issue:

SPTF: Shortest Positioning Time First

Issue:

Solution:

Final Thoughts

- merge good use 42, 8, 43
  - drin
  - why not wait
  - Predictive disk reads
Raid

Read 2

$S_1$ $S_2$ $S_3$

data d d parity info

even 1 0 1 0

1 1 1 1

Do D1 D2 P3 D5
0 1 2 P0
3 4 P1 5
6 P2 7 8

P3