CS 61C: Great Ideas in Computer Architecture (Machine Structures)

Lecture 38: IO Disks

Instructor: Dan Garcia

http://inst.eecs.Berkeley.edu/~cs61c/sp13

Technology in the News

Seiki SE50UY04

50-inch 4k (3840x2160) display

Review

- Exceptions are "Unexpected" events
- Interrupts are asynchronous
  – can be used for interacting with I/O devices
- Need to handle in presence of pipelining, etc.

- Networks are another form of I/O

- Protocol suites allow networking of heterogeneous components
  – Another form of principle of abstraction

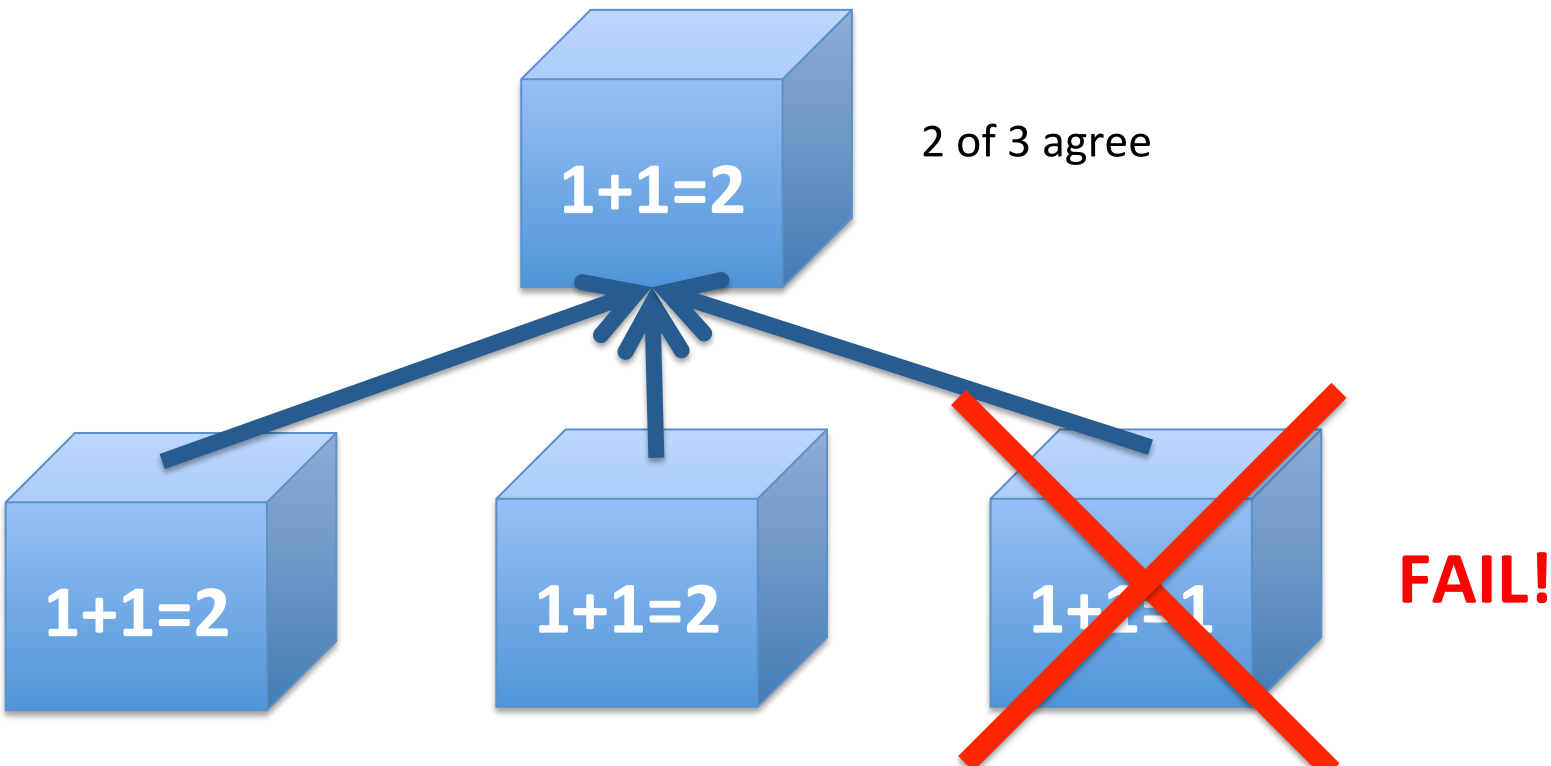
- Interested in Networking?
    - EE122 (CS-based in Fall, EE –based in Spring)

Review - 6 Great Ideas in Computer Architecture

1. Layers of Representation/Interpretation
2. Moore's Law
3. Principle of Locality/Memory Hierarchy
4. Parallelism
5. Performance Measurement & Improvement
6. **Dependability via Redundancy**

Review - Great Idea #6: Dependability via Redundancy

- Redundancy so that a failing piece doesn't make the whole system fail



Increasing transistor density reduces the cost of redundancy

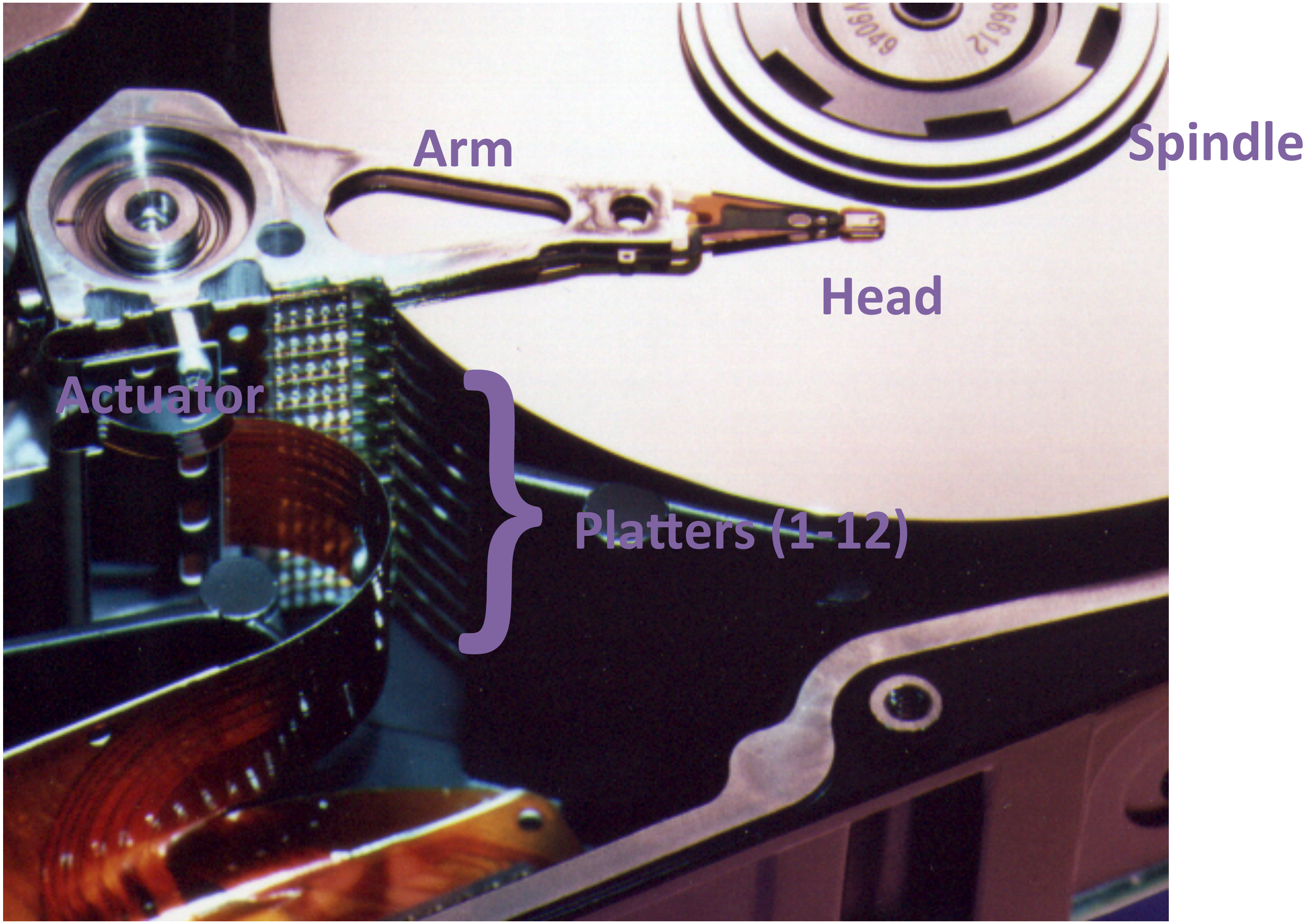
Review - Great Idea #6: Dependability via Redundancy

- Applies to everything from datacenters to memory
  - Redundant datacenters so that can lose 1 datacenter but Internet service stays online
  - Redundant routes so can lose nodes but Internet doesn't fail
  - Redundant disks so that can lose 1 disk but not lose data (Redundant Arrays of Independent Disks/RAID)
  - Redundant memory bits of so that can lose 1 bit but no data (Error Correcting Code/ECC Memory)

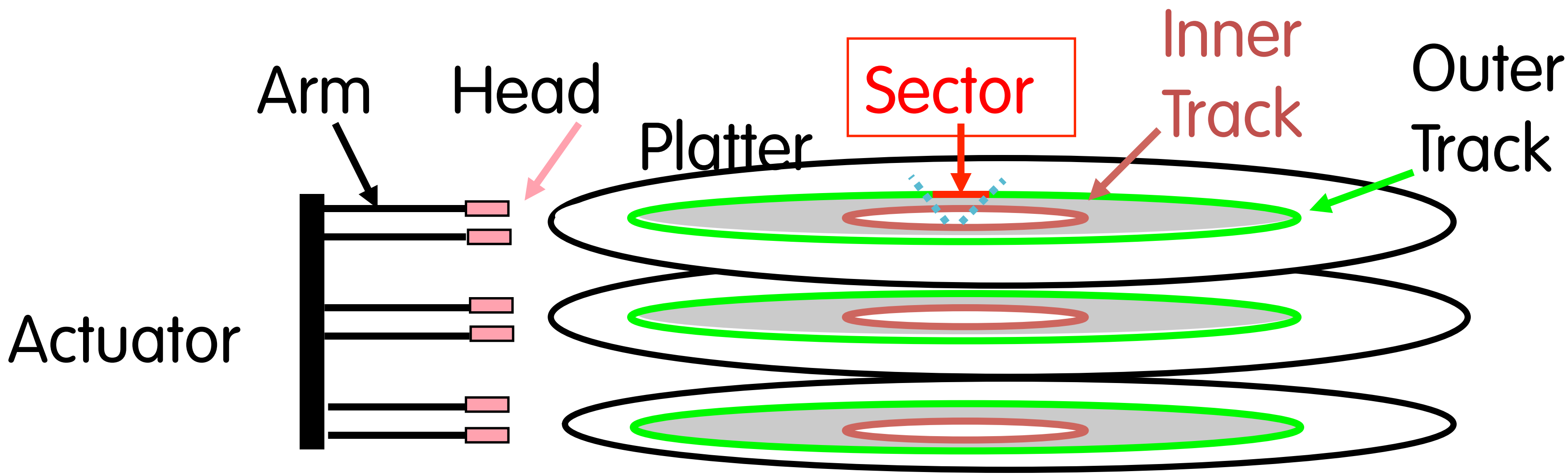
Magnetic Disk – common I/O device

- A kind of computer memory
  - Information stored by magnetizing ferrite material on surface of rotating disk
    - similar to tape recorder except digital rather than analog data
- Nonvolatile storage
  - retains its value without applying power to disk.
- Two Types
  - Floppy disks – slower, less dense, removable.
  - Hard Disk Drives (HDD) – faster, more dense, non-removable.
- Purpose in computer systems (Hard Drive):
  - Long-term, inexpensive storage for files
  - "Backup" for main-memory. Large, inexpensive, slow level in the memory hierarchy (virtual memory)

Photo of Disk Head, Arm, Actuator



Disk Device Terminology



- Several platters, with information recorded magnetically on both surfaces (usually)
- **Bits recorded in tracks, which in turn divided into sectors (e.g., 512 Bytes)**
- **Actuator moves head (end of arm) over track ("seek"), wait for sector rotate under head, then read or write**

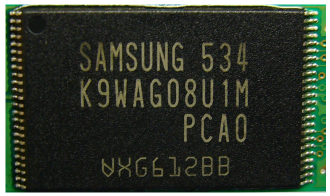
Where does Flash memory come in?

- Microdrives and Flash memory (e.g., CompactFlash going head-to-head
  - Both non-volatile (no power, data ok)
  - Flash benefits: durable & lower power (no moving parts, need to spin µdrives up/down)
  - Flash limitations: finite number of write cycles (wear on the insulating oxide layer around the charge storage mechanism). Most ≥ 100K, some ≥ 1M W/erase cycles.
- How does Flash memory work?
  - NMOS transistor with an additional conductor between gate and source/drain which "traps" electrons. The presence/absence is a 1 or 0.

en.wikipedia.org/wiki/Ipod www.apple.com/ipod

What does Apple put in its iPods?

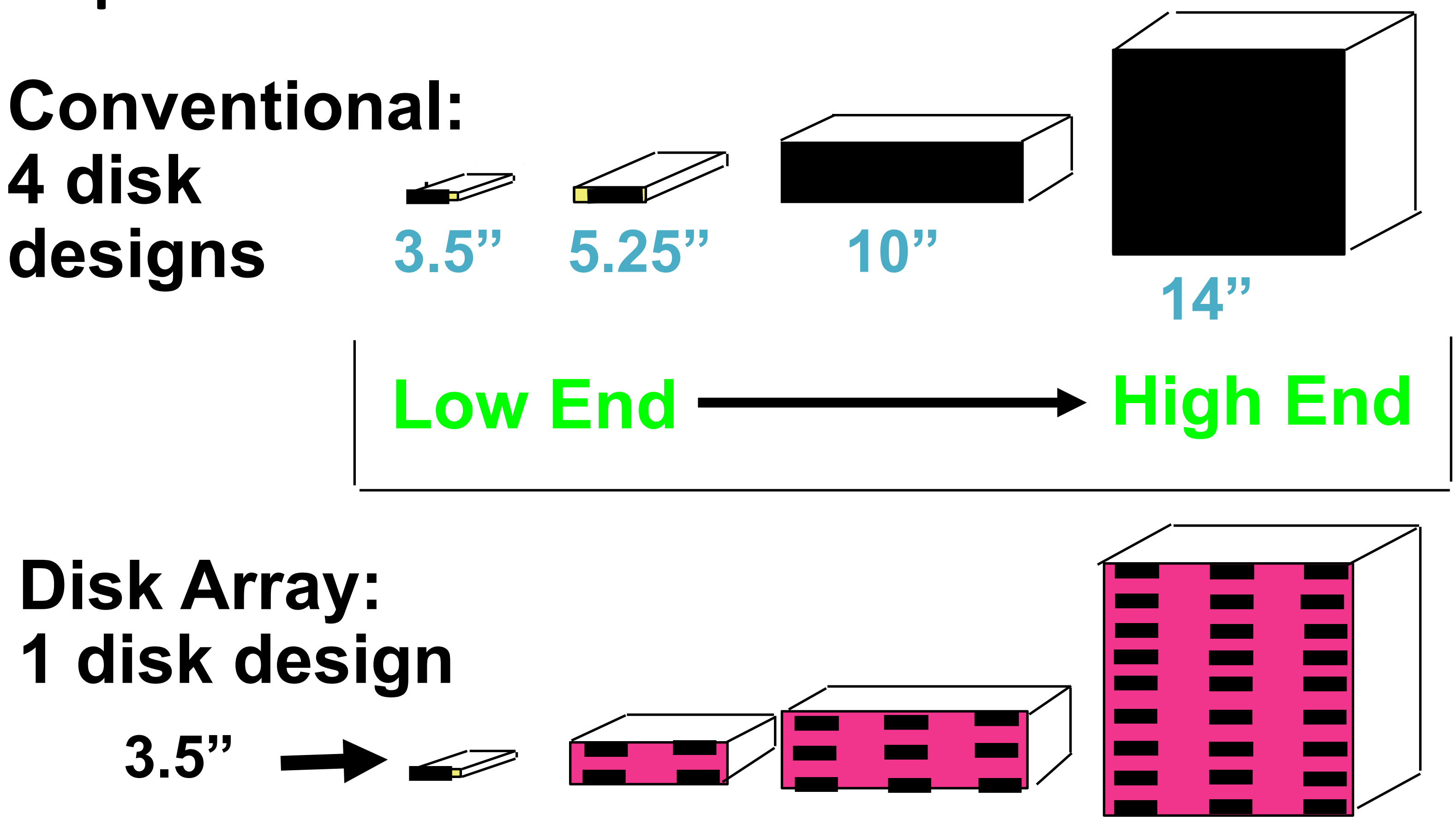
Toshiba flash 2 GB	Samsung flash 16 GB	Toshiba 1.8-inch HDD 80, 120, 160 GB	Toshiba flash 32, 64 GB



shuffle, nano, classic, touch

Use Arrays of Small Disks…

- **Katz and Patterson asked in 1987:**
  - **Can smaller disks be used to close gap in performance between disks and CPUs?**



Replace Small # of Large Disks with Large # of Small!

(1988 Disks)

	IBM 3390K	IBM 3.5" 0061
Capacity	20 GBytes	320 MBytes
Volume	97 cu. ft.	0.1 cu. ft.
Power	3 KW	11 W
Data Rate	15 MB/s	1.5 MB/s
I/O Rate	600 I/Os/s	55 I/Os/s
MTTF	250 KHrs	50 KHrs
Cost	$250K	$2K

Disk Arrays potentially high performance, high MB per cu. ft., high MB per KW, <u>but what about reliability?</u>

Replace Small # of Large Disks with Large # of Small!

(1988 Disks)

	IBM 3390K	IBM 3.5" 0061	x70	
Capacity	20 GBytes	320 MBytes	23 GBytes	
Volume	97 cu. ft.	0.1 cu. ft.	11 cu. ft.	9X
Power	3 KW	11 W	1 KW	3X
Data Rate	15 MB/s	1.5 MB/s	120 MB/s	8X
I/O Rate	600 I/Os/s	55 I/Os/s	3900 I/Os/s	6X
MTTF	250 KHrs	50 KHrs	??? Hrs	
Cost	$250K	$2K	$150K	

Disk Arrays potentially high performance, high MB per cu. ft., high MB per KW, but what about reliability?

Array Reliability

- Reliability - whether or not a component has failed
  - measured as Mean Time To Failure (MTTF)
- Reliability of N disks
  = Reliability of 1 Disk ÷ N
  (assuming failures independent)
  - 50,000 Hours ÷ 70 disks = 700 hour
- Disk system MTTF:
  Drops from 6 years to 1 month!
- Disk arrays too unreliable to be useful!

Redundant Arrays of (Inexpensive) Disks

- Files are "striped" across multiple disks
- Redundancy yields high data availability
  - Availability: service still provided to user, even if some components failed
- Disks will still fail
- Contents reconstructed from data redundantly stored in the array

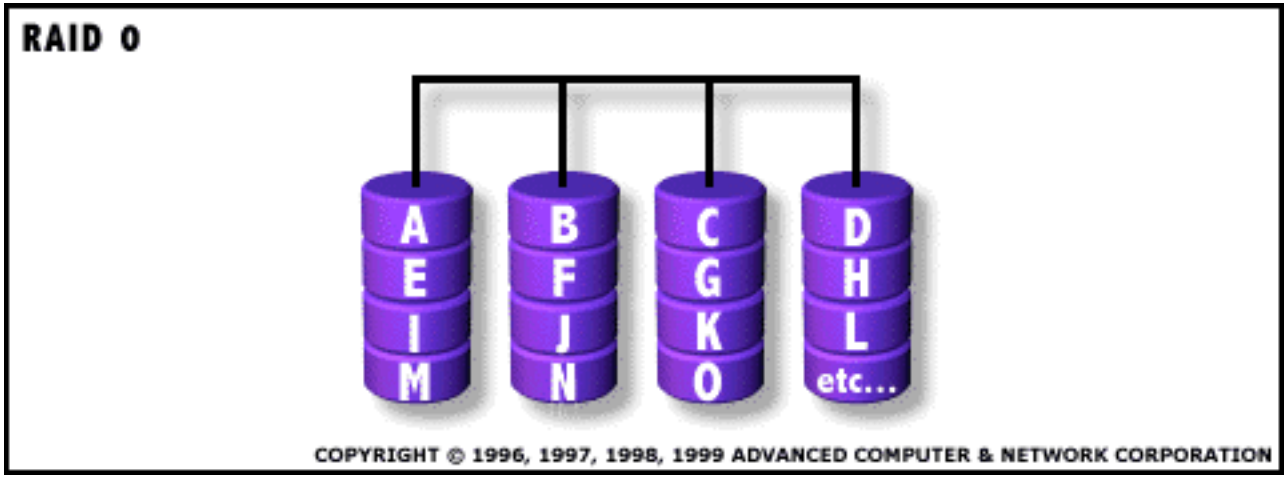
  ⇒ Capacity penalty to store redundant info

  ⇒ Bandwidth penalty to update redundant info

RAID : Redundant Array of Inexpensive Disks

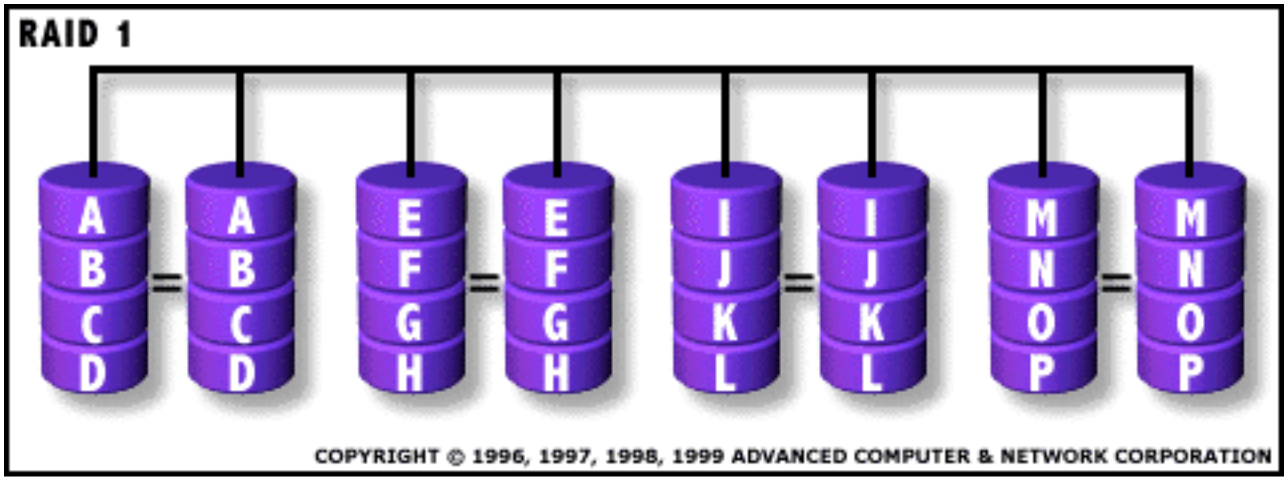
- Invented @ Berkeley (1989)
- A multi-billion industry
  80% non-PC disks sold in RAIDs
- Idea:
  – Files are "striped" across multiple disks
  – Redundancy yields high data availability
    - Disks will still fail
  – Contents reconstructed from data redundantly stored in the array
    - ⇒ Capacity penalty to store redundant info
    - ⇒ Bandwidth penalty to update redundant info

“RAID 0”: No redundancy = “AID”



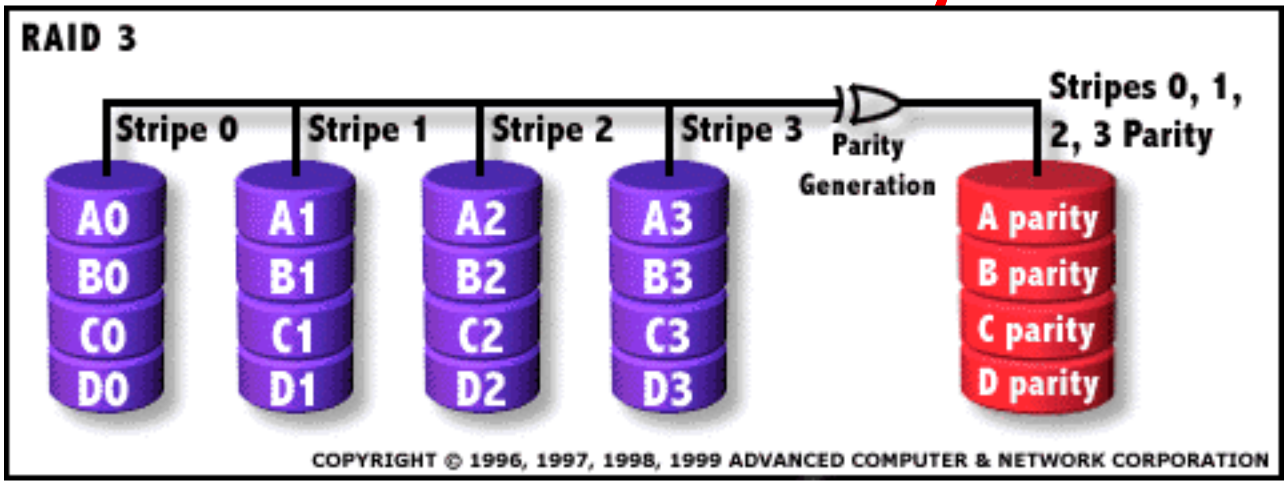
- Assume have 4 disks of data for this example, organized in blocks
- Large accesses faster since transfer from several disks at once

RAID 1: Mirror data



- Each disk is fully duplicated onto its "mirror"
  - Very high availability can be achieved
- Bandwidth reduced on write:
  - 1 Logical write = 2 physical writes
- Most expensive solution: 100% capacity overhead

RAID 3: Parity

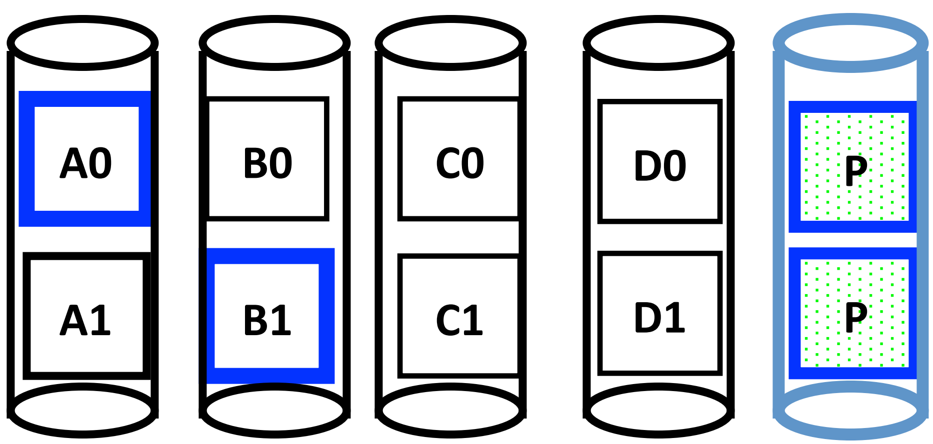


- Spindles synchronized, each sequential byte on a diff. drive
- Parity computed across group to protect against hard disk failures, stored in P disk
- Logically, a single high capacity, high transfer rate disk
- 25% capacity cost for parity in this example vs. 100% for RAID 1 (5 disks vs. 8 disks)
- Q: How many drive failures can be tolerated?

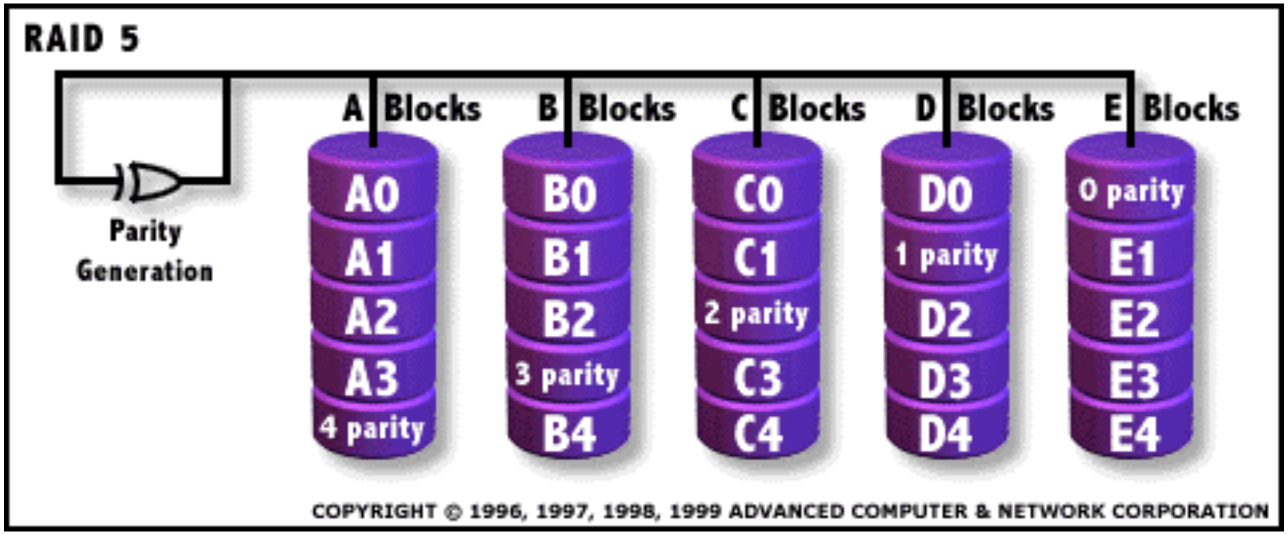
Inspiration for RAID 5 (RAID 4 block-striping)

- Small writes (write to one disk):
  - Option 1: read other data disks, create new sum and write to Parity Disk (access all disks)
  - Option 2: since P has old sum, compare old data to new data, add the difference to P:
    1 logical write = 2 physical reads + 2 physical writes to 2 disks
- Parity Disk is bottleneck for Small writes: Write to A0, B1 → both write to P disk

(RAID 4 block-striping)



RAID 5: Rotated Parity, faster small writes



- Independent writes possible because of interleaved parity
  - Example: write to A0, B1 uses disks 0, 1, 4, 5, so can proceed in parallel
  - Still 1 small write = 4 physical disk accesses

Peer Instruction

1. RAID 1 (mirror) and 5 (rotated parity) help with performance and availability
2. RAID 1 has higher cost than RAID 5
3. Small writes on RAID 5 are slower than on RAID 1

	123
A:	FFF
B:	FFT
B:	FTF
C:	FTT
C:	TFF
D:	TFT
D:	TTF
E:	TTT

Peer Instruction Answer

1. All RAID (0-5) helps with performance, only RAID0 doesn't help availability. TRUE
2. Surely! Must buy 2x disks rather than 1.25x (from diagram, in practice even less) TRUE
3. RAID5 (2R,2W) vs. RAID1 (2W). Latency worse, throughput (ll writes) better. TRUE

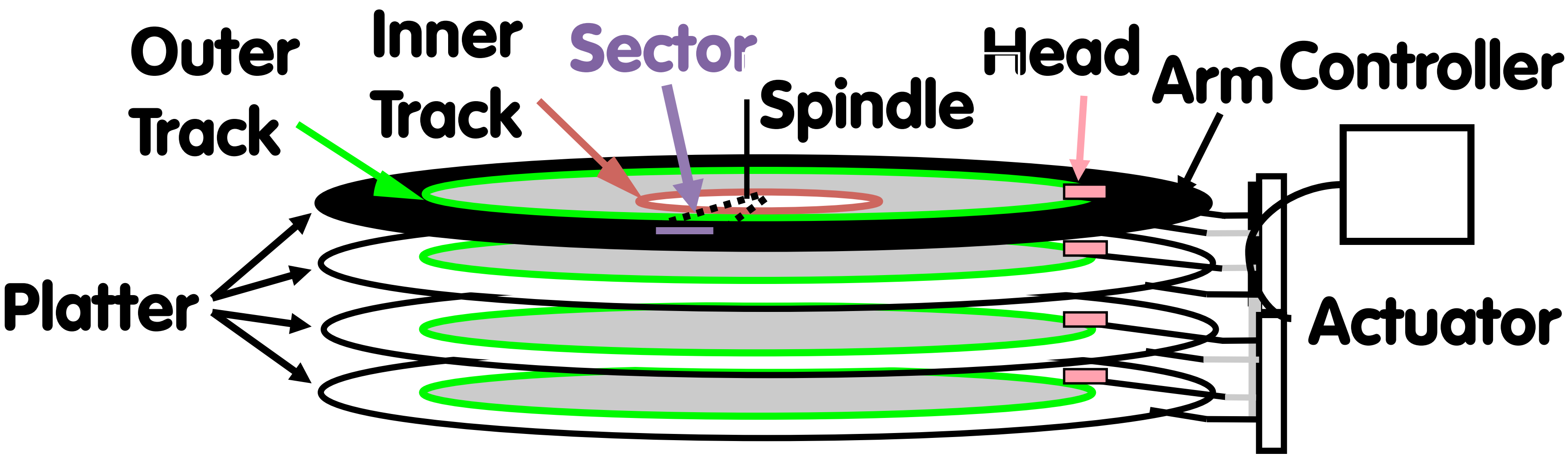
1. RAID 1 (mirror) and 5 (rotated parity) help with performance and availability
2. RAID 1 has higher cost than RAID 5
3. Small writes on RAID 5 are slower than on RAID 1

	123
A:	FFF
B:	FFT
B:	FTF
C:	FTT
C:	TFF
D:	TFT
D:	TTF
E:	TTT

“And in conclusion…”

- I/O gives computers their 5 senses
- I/O speed range is 100-million to one
- Processor speed means must synchronize with I/O devices before use: Polling vs. Interrupts
- Networks are another form of I/O
- Protocol suites allow networking of heterogeneous components
  - Another form of principle of abstraction
- RAID
  - Higher performance with more disk arms per $
  - More disks == More disk failures
  - Different RAID levels provide different cost/speed/reliability tradeoffs

Bonus: Disk Device Performance (1/2)



- **Disk Latency = Seek Time + Rotation Time + Transfer Time + Controller Overhead**
  - Seek Time? depends on no. tracks to move arm, speed of actuator
  - Rotation Time? depends on speed disk rotates, how far sector is from head
  - Transfer Time? depends on data rate (bandwidth) of disk (f(bit density,rpm)), size of request

Bonus: Disk Device Performance (2/2)

- Average distance of sector from head?
- 1/2 time of a rotation
  - 7200 Revolutions Per Minute $\Rightarrow$ 120 Rev/sec
  - 1 revolution = 1/120 sec $\Rightarrow$ 8.33 milliseconds
  - 1/2 rotation (revolution) $\Rightarrow$ 4.17 ms
- Average no. tracks to move arm?
  - Disk industry standard benchmark:
    - Sum all time for all possible seek distances from all possible tracks / # possible
    - Assumes average seek distance is random
- Size of Disk cache can strongly affect perf!
  - Cache built into disk system, OS knows nothing

BONUS : Hard Drives are Sealed. Why?

- The closer the head to the disk, the smaller the "spot size" and thus the denser the recording.
  - Measured in Gbit/in2. ~60 is state of the art.
- Disks are sealed to keep the dust out.
  - Heads are designed to "fly" at around 5-20nm above the surface of the disk.
  - 99.999% of the head/arm weight is supported by the air bearing force (air cushion) developed between the disk and the head.