CS252
Graduate Computer Architecture

Lecture 5:
I/O Introduction: Storage Devices & RAID

January 31, 2001

Prof. David A. Patterson

Computer Science 252

Spring 2001

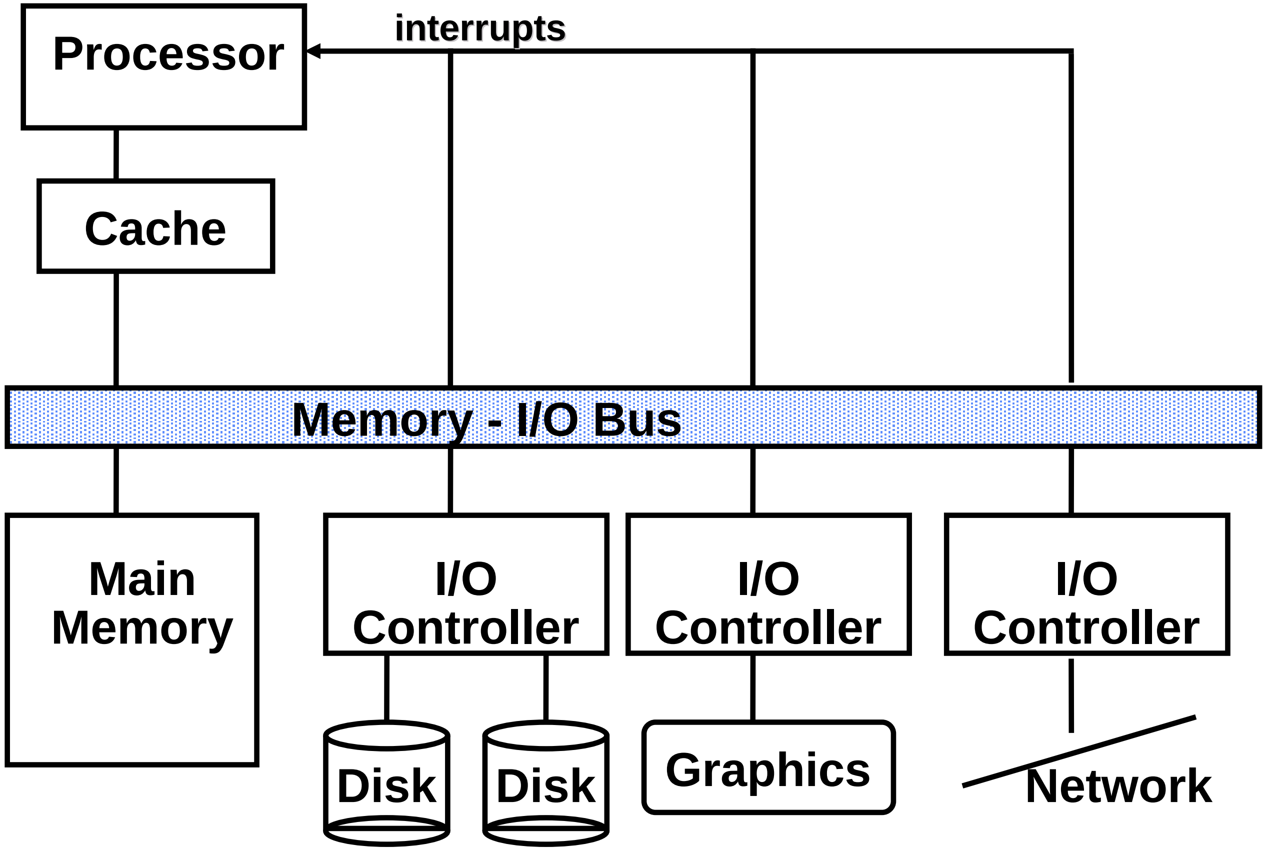
Motivation: Who Cares About I/O?

- CPU Performance: 60% per year
- I/O system performance limited by *mechanical* delays (disk I/O)
  - < 10% per year (IO per sec)
- Amdahl's Law: system speed-up limited by the slowest part!
  - 10% IO & 10x CPU => 5x Performance (lose 50%)
  - 10% IO & 100x CPU => 10x Performance (lose 90%)
- I/O bottleneck:
  - Diminishing fraction of time in CPU
  - Diminishing value of faster CPUs

Big Picture: Who cares about CPUs?

- Why still important to keep CPUs busy vs. IO devices ("CPU time"), as CPUs not costly?
  - Moore's Law leads to both large, fast CPUs but also to very small, cheap CPUs
  - 2001 Hypothesis: 600 MHz PC is fast enough for Office Tools?
  - PC slowdown since fast enough unless games, new apps?
- People care more about about storing information and communicating information than calculating
  - "Information Technology" vs. "Computer Science"
  - 1960s and 1980s: Computing Revolution
  - 1990s and 2000s: Information Age
- Next 3 weeks on storage and communication

I/O Systems


Processor
interrupts
Cache
Memory - I/O Bus
Main Memory
I/O Controller
I/O Controller
I/O Controller
Disk
Disk
Graphics
Network

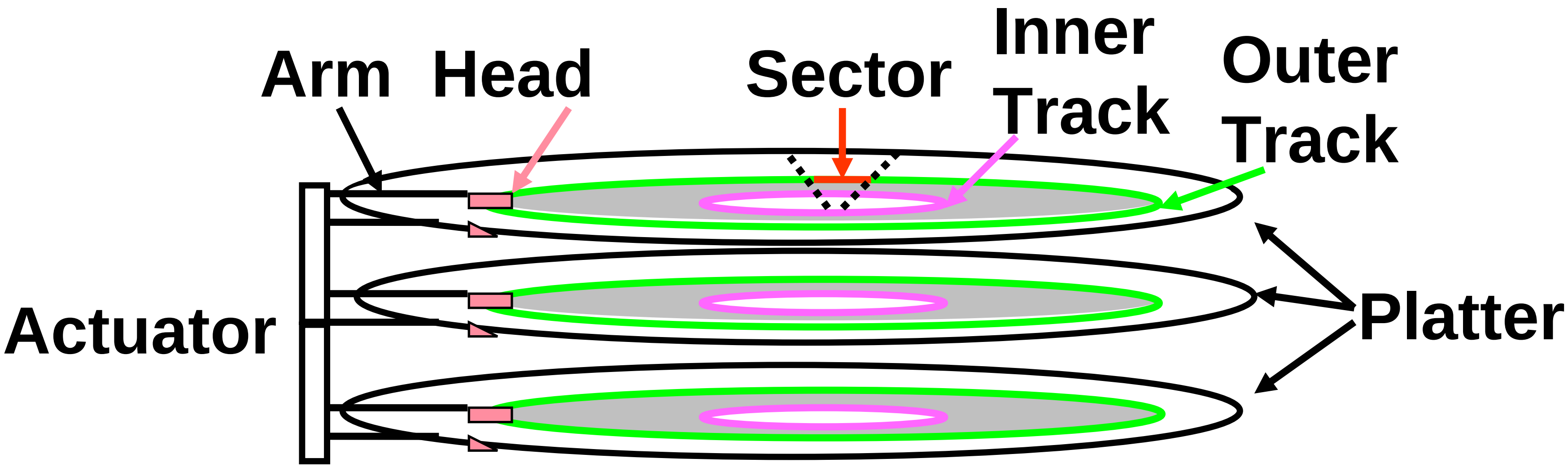
Storage Technology Drivers

- Driven by the prevailing computing paradigm
  - 1950s: migration from batch to on-line processing
  - 1990s: migration to ubiquitous computing
    - » computers in phones, books, cars, video cameras, …
    - » nationwide fiber optical network with wireless tails
- Effects on storage industry:
  - Embedded storage
    - » smaller, cheaper, more reliable, lower power
  - Data utilities
    - » high capacity, hierarchically managed storage

Outline

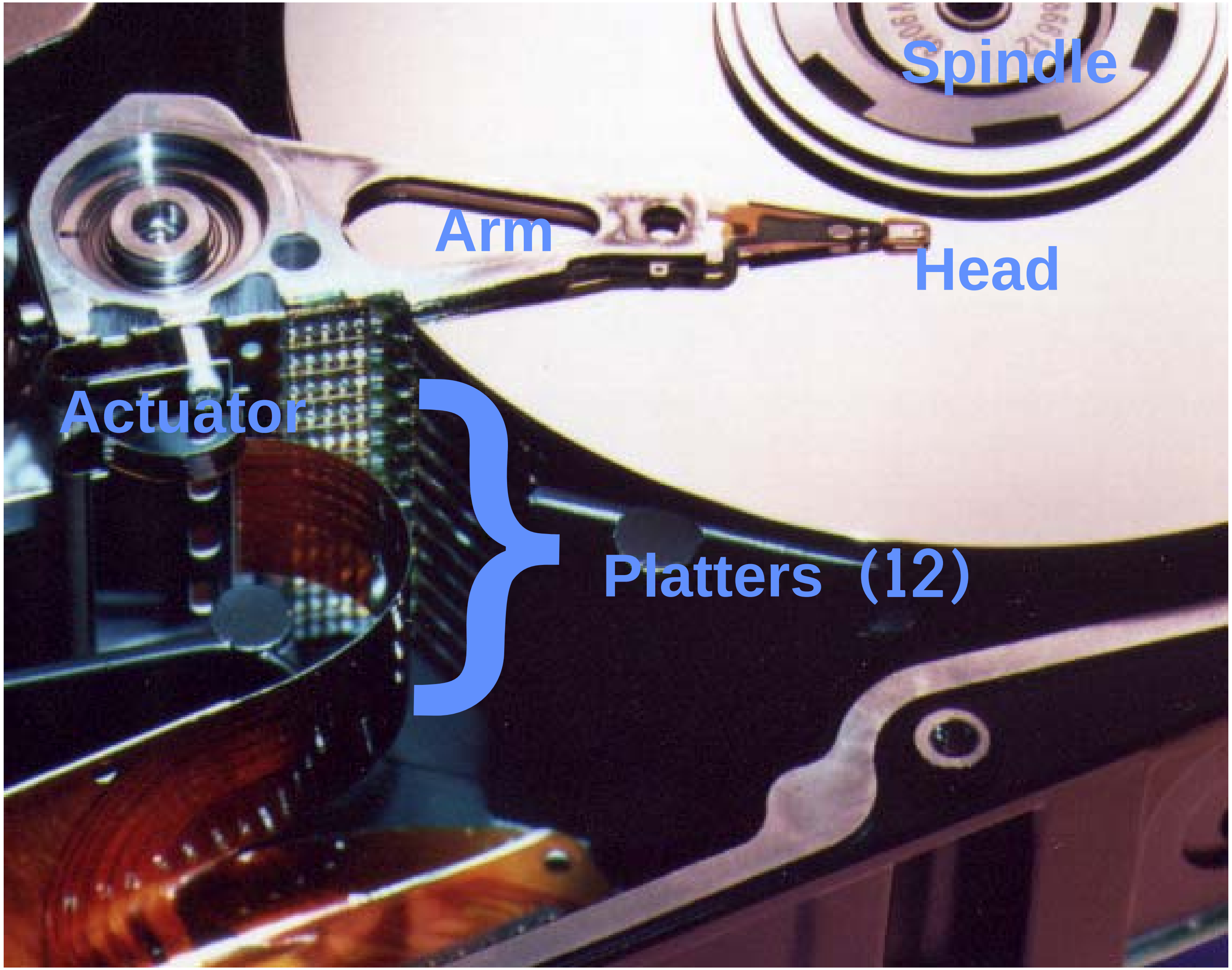
- Disk Basics
- Disk History
- Disk options in 2000
- Disk fallacies and performance
- Tapes
- RAID

Disk Device Terminology

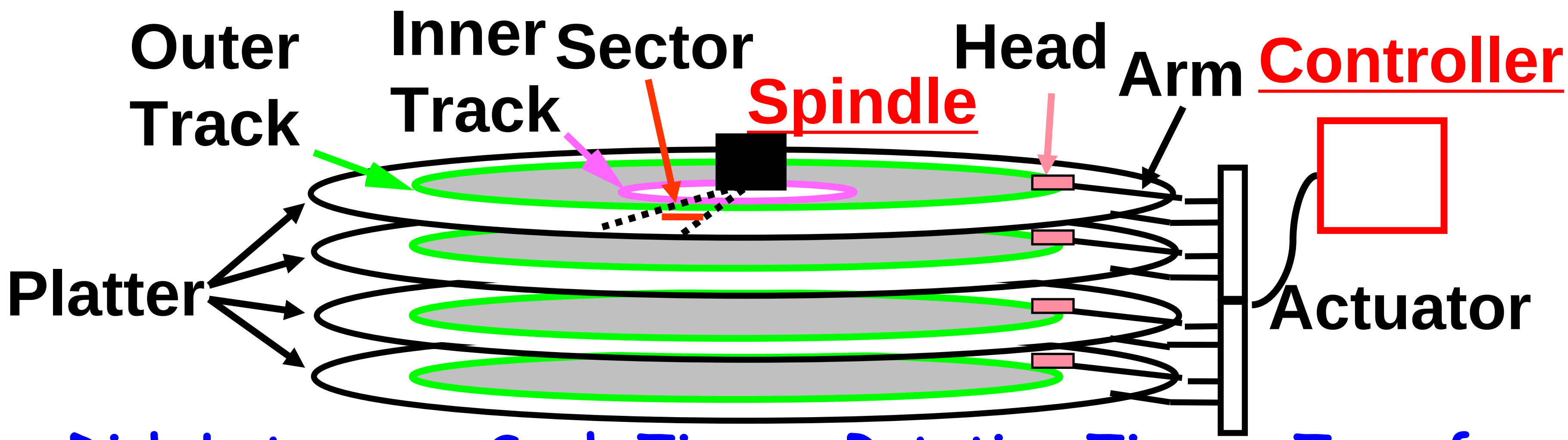


- Several platters, with information recorded magnetically on both surfaces (usually)
- Bits recorded in tracks, which in turn divided into sectors (e.g., 512 Bytes)
- Actuator moves head (end of arm,1/surface) over track ("seek"), select surface, wait for sector rotate under head, then read or write
  - "Cylinder": all tracks under heads

Photo of Disk Head, Arm, Actuator



Disk Device Performance



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

Disk Device Performance

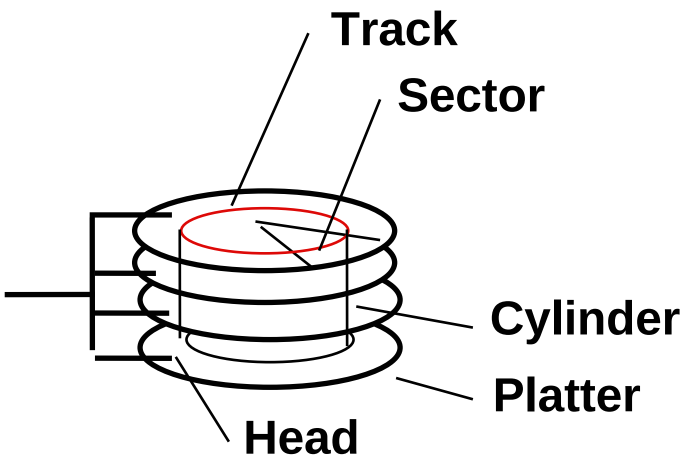
- Average distance sector from head?
- 1/2 time of a rotation
  - 10000 Revolutions Per Minute ⇒ 166.67 Rev/sec
  - 1 revolution = 1/ 166.67 sec ⇒ 6.00 milliseconds
  - 1/2 rotation (revolution) ⇒ 3.00 ms
- Average no. tracks move arm?
  - Sum all possible seek distances
    from all possible tracks / # possible
    - » Assumes average seek distance is random
  - Disk industry standard benchmark

Data Rate: Inner vs. Outer Tracks

- To keep things simple, orginally kept same number of sectors per track
  – Since outer track longer, lower bits per inch
- Competition ⇒ decided to keep BPI the same for all tracks ("constant bit density")
  ⇒ More capacity per disk
  ⇒ More of sectors per track towards edge
  ⇒ Since disk spins at constant speed, outer tracks have faster data rate
- Bandwidth outer track 1.7X inner track!
  – Inner track highest density, outer track lowest, so not really constant
  – 2.1X length of track outer / inner, 1.7X bits outer / inner

Devices: Magnetic Disks

- **Purpose:**
  - Long-term, nonvolatile storage
  - Large, inexpensive, slow level in the storage hierarchy
- **Characteristics:**
  - Seek Time (~8 ms avg)
    - » positional latency
    - » rotational latency
- **Transfer rate**
  - 10-40 MByte/sec
  - Blocks
- **Capacity**
  - Gigabytes
  - Quadruples every 2 years (aerodynamics)



7200 RPM = 120 RPS => 8 ms per rev
ave rot. latency = 4 ms
128 sectors per track => 0.25 ms per sector
1 KB per sector => 16 MB / s

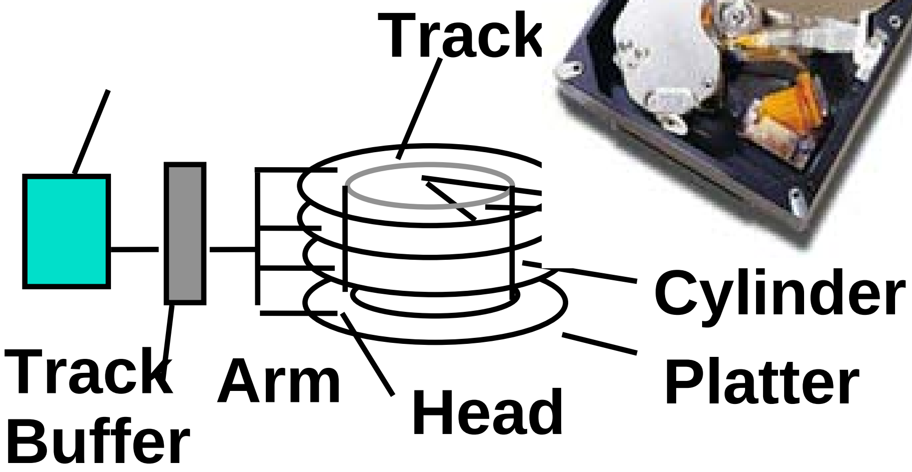
Response time
= Queue + Controller + $\underbrace{\text{Seek + Rot + Xfer}}_{\text{Service time}}$

Disk Performance Model /Trends

- Capacity
  - + 100%/year (2X / 1.0 yrs)
- Transfer rate (BW)
  - + 40%/year (2X / 2.0 yrs)
- Rotation + Seek time
  - – 8%/ year (1/2 in 10 yrs)
- MB/$
  - > 100%/year (2X / 1.0 yrs)

    Fewer chips + areal density

State of the Art: Barracuda 180



- 181.6 GB, 3.5 inch disk
- 12 platters, 24 surfaces
- 24,247 cylinders
- 7,200 RPM; (4.2 ms avg. latency)
- 7.4/8.2 ms avg. seek (r/w)
- 64 to 35 MB/s (internal)
- 0.1 ms controller time
- 10.3 watts (idle)

Latency =

per access: Queuing Time + Controller time + Seek Time + Rotation Time +

\+

per byte: Size / Bandwidth

source: www.seagate.com

Disk Performance Example (will fix later)

- Calculate time to read 64 KB (128 sectors) for Barracuda 180 X using advertised performance; sector is on outer track

Disk latency = average seek time + average rotational delay + transfer time + controller overhead

= 7.4 ms + 0.5 * 1/(7200 RPM) + 64 KB / (65 MB/s) + 0.1 ms

= 7.4 ms + 0.5 /(7200 RPM/(60000ms/M)) + 64 KB / (65 KB/ms) + 0.1 ms

= 7.4 + 4.2 + 1.0 + 0.1 ms = 12.7 ms

CS 252 Administrivia

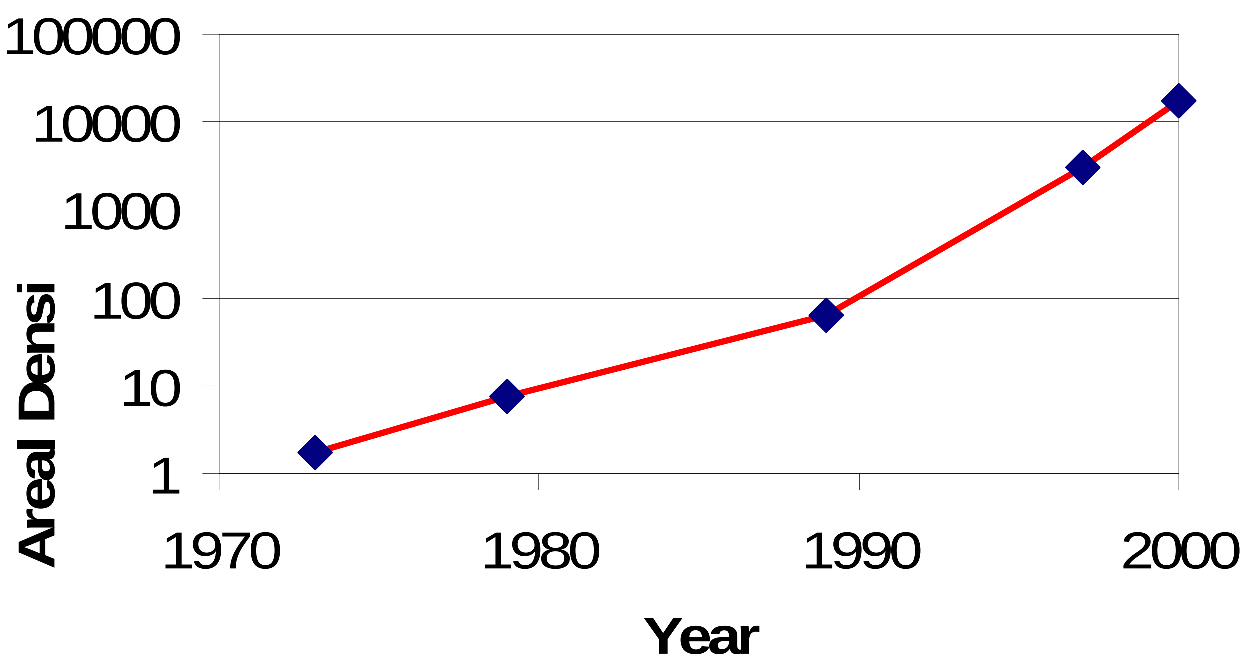
- We have a TA! Yu-jia Jin: yujia@ic.eecs
- Please send 1-2 paragraph summary of papers to him BEFORE CLASS Friday
  - J.GRAY, Turing Award Lecture: "What Next? A dozen remaining IT problems"; We will discuss Friday
  - Should have already turned in
    - » G. MOORE, "Cramming More Components onto Integrated Circuits"
    - » J. S. LIPTAY, "Structural Aspects of the System/360 Model 85, Part II: The Cache"
- Please fill out Third Edition chapter surveys for 1, 5, 6; 1,5 before Friday, 6 by next Wednesday
  - Link from 252 Web page (click on survey)
- Project suggestions are on web site; start looking
  - http://www.cs.berkeley.edu/~pattrsn/252S01/suggestions.html
- Office hours Wednesdays 11-12

Areal Density

- Bits recorded along a track
  - Metric is Bits Per Inch (BPI)
- Number of tracks per surface
  - Metric is Tracks Per Inch (TPI)
- Disk Designs Brag about bit density per unit area
  - Metric is Bits Per Square Inch
  - Called Areal Density
  - Areal Density = BPI x TPI

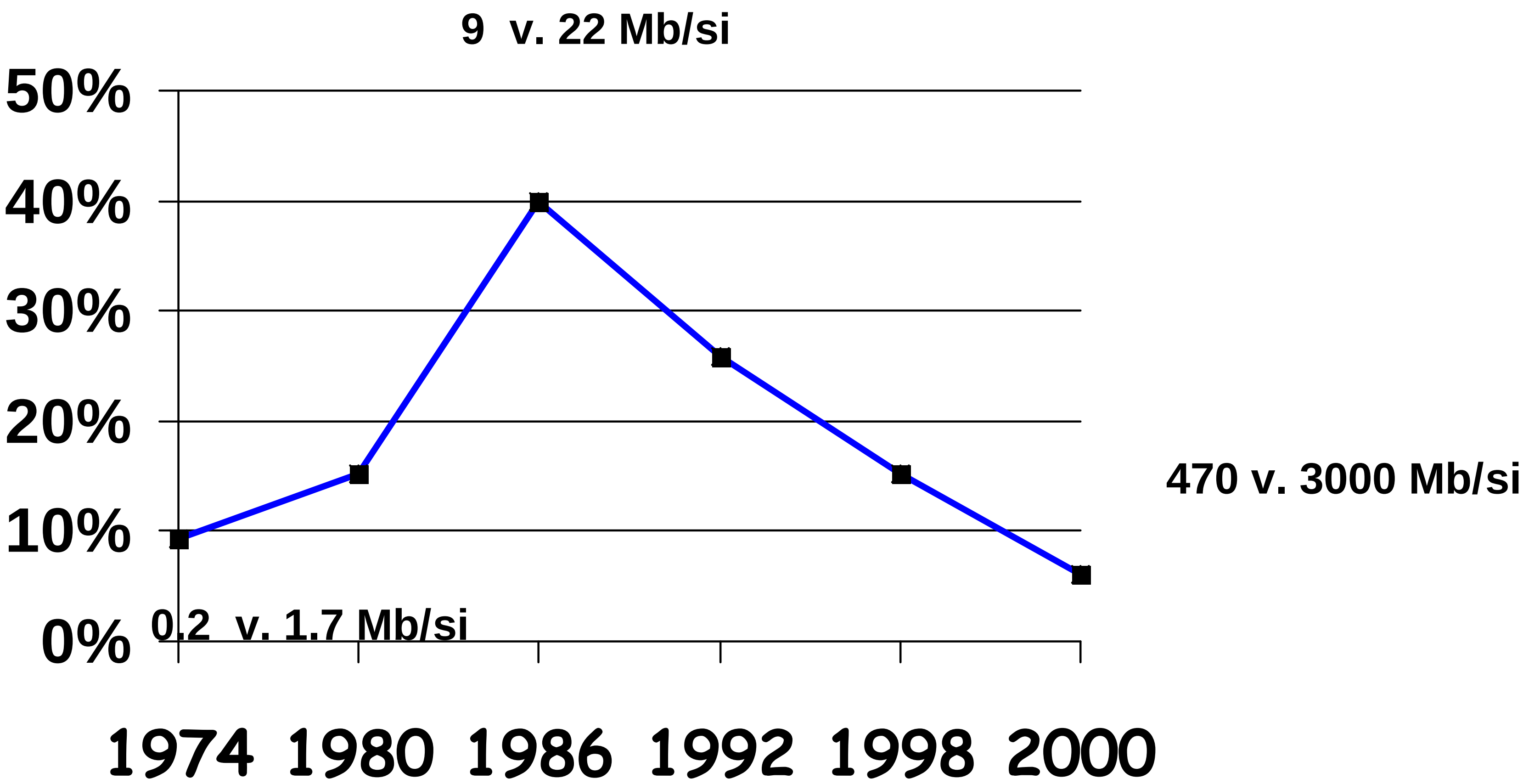
Areal Density

Year	Areal Density
1973	1.7
1979	7.7
1989	63
1997	3090
2000	17100



- Areal Density =_BPI x TPI
- Change slope 30%/yr to 60%/yr about 1991

MBits per square inch: DRAM as % of Disk over time



source: New York Times, 2/23/98, page C3,
"Makers of disk drives crowd even mroe data into even smaller spaces"

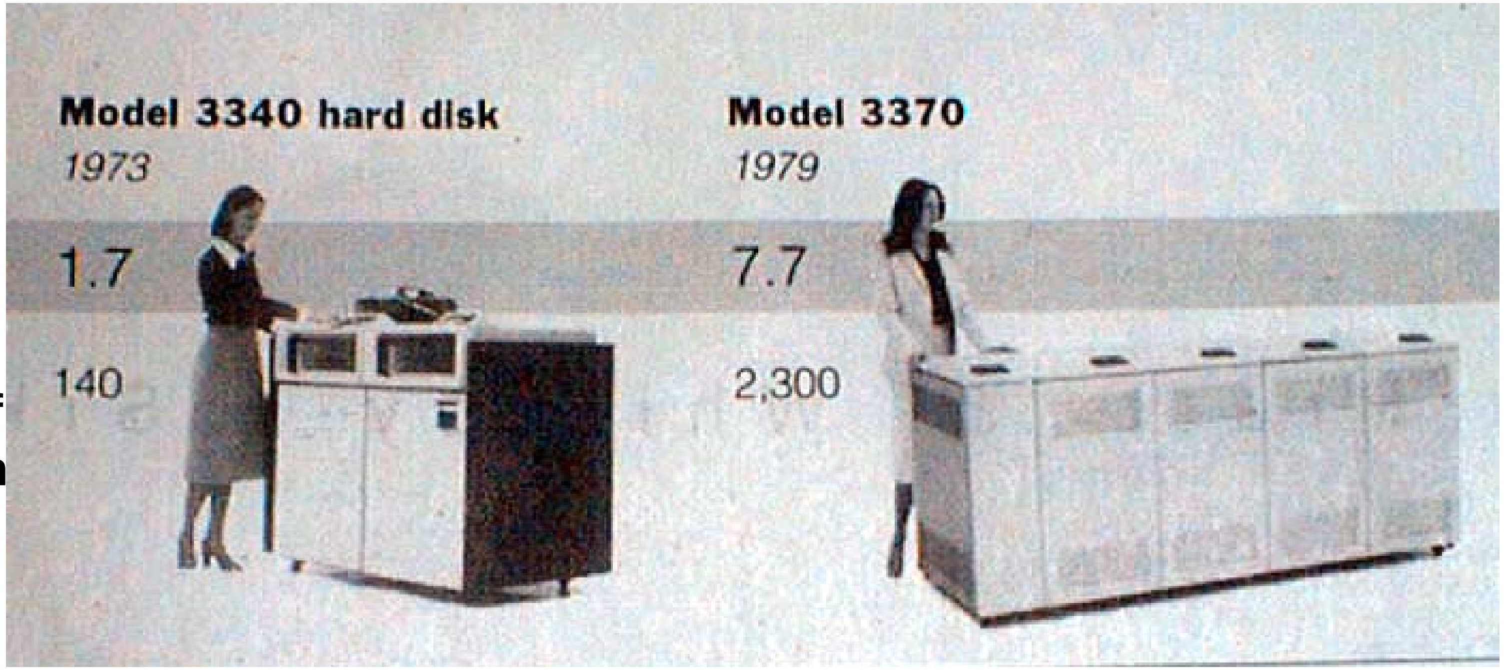
Historical Perspective

- 1956 IBM Ramac — early 1970s Winchester
  - Developed for mainframe computers, proprietary interfaces
  - Steady shrink in form factor: 27 in. to 14 in
- Form factor and capacity drives market, more than performance
- 1970s: Mainframes ⇒ 14 inch diameter disks
- 1980s: Minicomputers, Servers ⇒ 8",5 1/4" diameter
- PCs, workstations Late 1980s/Early 1990s:
  - Mass market disk drives become a reality
    - » industry standards: SCSI, IPI, IDE
  - Pizzabox PCs ⇒ 3.5 inch diameter disks
  - Laptops, notebooks ⇒ 2.5 inch disks
  - Palmtops didn't use disks,
    so 1.8 inch diameter disks didn't make it
- 2000s:
  - 1 inch for cameras, cell phones?

Disk History

Data density Mbit/sq. in.

Capacity of Unit Shown Megabytes

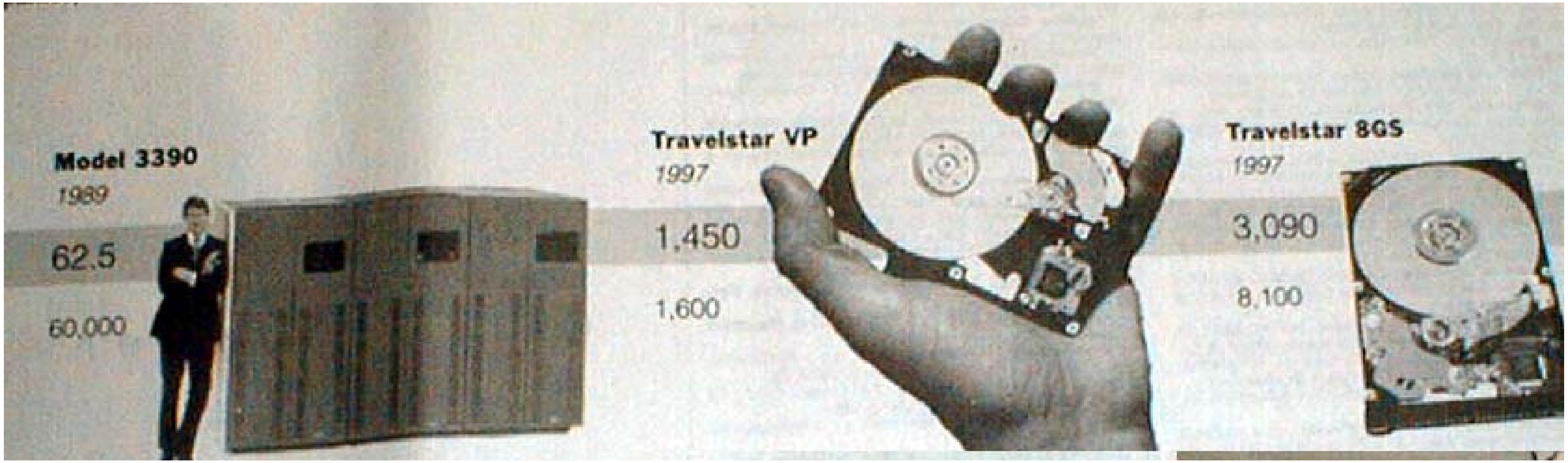


1973:
1. 7 Mbit/sq. in
140 MBytes

1979:
7. 7 Mbit/sq. in
2,300 MBytes

source: New York Times, 2/23/98, page C3,
"Makers of disk drives crowd even more data into even smaller spaces"

Disk History



1989:	1997:	1997:
63 Mbit/sq. in	1450 Mbit/sq. in	3090 Mbit/sq. in
60,000 MBytes	2300 MBytes	8100 MBytes

source: New York Times, 2/23/98, page C3,
"Makers of disk drives crowd even mroe data into even smaller spaces"

1 inch disk drive!

- **2000 IBM MicroDrive:**
  - 1.7" x 1.4" x 0.2"
  - 1 GB, 3600 RPM,
    5 MB/s, 15 ms seek
  - Digital camera, PalmPC?
- **2006 MicroDrive?**
- **9 GB, 50 MB/s!**
  - Assuming it finds a niche
    in a successful product
  - Assuming past trends continue

Disk Characteristics in 2000

	Seagate Cheetah ST173404LC Ultra160 SCSI	IBM Travelstar 32GH DJSA - 232 ATA-4	IBM 1GB Microdrive DSCM-11000
Disk diameter (inches)	3.5	2.5	1.0
Formatted data capacity (GB)	73.4	32.0	1.0
Cylinders	14,100	21,664	7,167
Disks	12	4	1
Recording Surfaces (Heads)	24	8	2
Bytes per sector	512 to 4096	512	512
Avg Sectors per track (512 byte)	~ 424	~ 360	~ 140
Max. areal density(Gbit/sq.in.)	6.0	14.0	15.2
	$828	**$447**	**$435**

Disk Characteristics in 2000

	Seagate Cheetah ST173404LC Ultra160 SCSI	IBM Travelstar 32GH DJSA - 232 ATA-4	IBM 1GB Microdrive DSCM-11000
Rotation speed (RPM)	10033	5411	3600
Avg. seek ms (read/write)	5.6/6.2	12.0	12.0
Minimum seek ms (read/write)	0.6/0.9	2.5	1.0
Max. seek ms	14.0/15.0	23.0	19.0
Data transfer rate MB/second	27 to 40	11 to 21	2.6 to 4.2
Link speed to buffer MB/s	160	67	13
Power idle/operating Watts	16.4 / 23.5	2.0 / 2.6	0.5 / 0.8

Disk Characteristics in 2000

	Seagate Cheetah ST173404LC Ultra160 SCSI	IBM Travelstar 32GH DJSA - 232 ATA-4	IBM 1GB Microdrive DSCM-11000
Buffer size in MB	4.0	2.0	0.125
Size: height x width x depth inches	1.6 x 4.0 x 5.8	0.5 x 2.7 x 3.9	0.2 x 1.4 x 1.7
Weight pounds	2.00	0.34	0.035
Rated MTTF in powered-on hours	1,200,000	(300,000?)	(20K/5 yr life?)
% of POH per month	100%	45%	20%
% of POH seeking, reading, writing	90%	20%	20%

Disk Characteristics in 2000

	Seagate Cheetah ST173404LC Ultra160 SCSI	IBM Travelstar 32GH DJSA - 232 ATA-4	IBM 1GB Microdri DSCM-11000
Load/Unload cycles (disk powered on/off)	250 per year	300,000	300,000
Nonrecoverable read errors per bits read	<1 per 10^{15}	< 1 per 10^{13}	< 1 per 10^{13}
Seek errors	<1 per 10^{7}	not available	not available
Shock tolerance: Operating, Not operating	10 G, 175 G	150 G, 700 G	175 G, 1500 G
Vibration tolerance: Operating, Not operating (sine swept, 0 to peak)	5-400 Hz @ 0.5G, 22-400 Hz @ 2.0G	5-500 Hz @ 1.0G, 2.5-500 Hz @ 5.0G	5-500 Hz @ 1G, 500 Hz @ 5G

Fallacy: Use Data Sheet "Average Seek" Time

- Manufacturers needed standard for fair comparison ("benchmark")
  - Calculate all seeks from all tracks, divide by number of seeks => "average"
- Real average would be based on how data laid out on disk, where seek in real applications, then measure performance
  - Usually, tend to seek to tracks nearby, not to random track
- Rule of Thumb: observed average seek time is typically about 1/4 to 1/3 of quoted seek time (i.e., 3X-4X faster)
  - Barracuda 180 X avg. seek: 7.4 ms $\Rightarrow$ 2.5 ms

Fallacy: Use Data Sheet Transfer Rate

- Manufacturers quote the speed off the data rate off the surface of the disk
- Sectors contain an error detection and correction field (can be 20% of sector size) plus sector number as well as data
- There are gaps between sectors on track
- Rule of Thumb: disks deliver about 3/4 of internal media rate (1.3X slower) for data
- For example, Barracuda 180X quotes
  64 to 35 MB/sec internal media rate

$\Rightarrow$ 47 to 26 MB/sec external data rate (74%)

Disk Performance Example

- Calculate time to read 64 KB for UltraStar 72 again, this time using 1/3 quoted seek time, 3/4 of internal outer track bandwidth; (12.7 ms before)

Disk latency = average seek time + average rotational delay + transfer time + controller overhead

= (0.33 * 7.4 ms) + 0.5 * 1/(7200 RPM)
+ 64 KB / (0.75 * 65 MB/s) + 0.1 ms

= 2.5 ms + 0.5 /(7200 RPM/(60000ms/M))
+ 64 KB / (47 KB/ms) + 0.1 ms

= 2.5 + 4.2 + 1.4 + 0.1 ms = 8.2 ms (64% of 12.7)

Future Disk Size and Performance

- Continued advance in capacity (60%/yr) and bandwidth (40%/yr)
- Slow improvement in seek, rotation (8%/yr)
- Time to read whole disk

Year	Sequentially	Randomly (1 sector/seek)
1990	4 minutes	6 hours
2000	12 minutes	1 week(!)

- 3.5" form factor make sense in 5 yrs?
  - What is capacity, bandwidth, seek time, RPM?
  - Assume today 80 GB, 30 MB/sec, 6 ms, 10000 RPM

Tape vs. Disk

- Longitudinal tape uses same technology as hard disk; tracks its density improvements
- Disk head flies above surface, tape head lies on surface
- Disk fixed, tape removable
- Inherent cost-performance based on geometries:
  fixed rotating platters with gaps
  (random access, limited area, 1 media / reader)

vs.

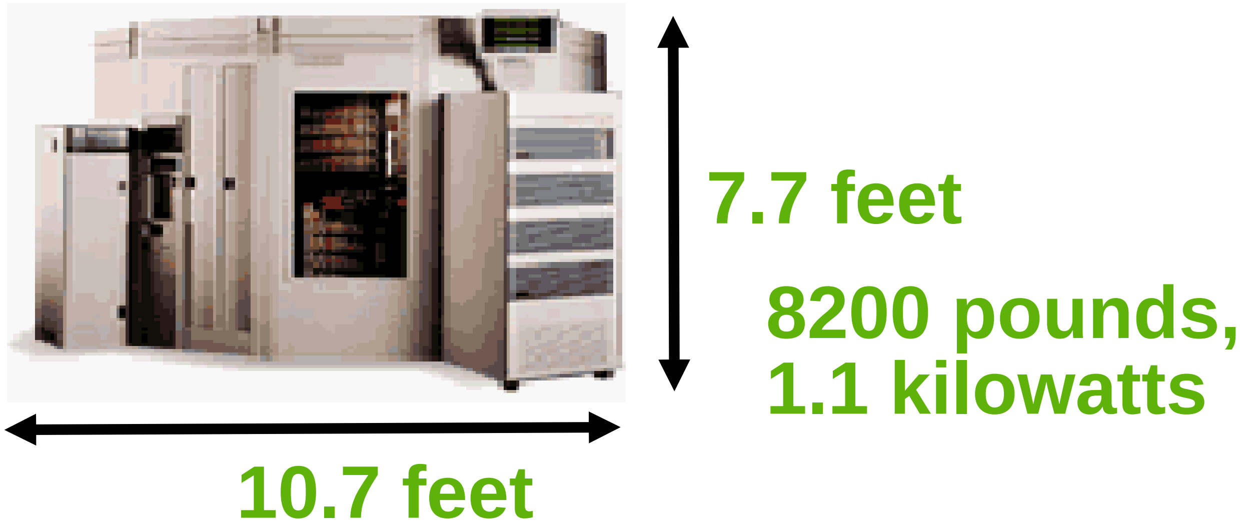
  removable long strips wound on spool
  (sequential access, "unlimited" length, multiple / reader)

- Helical Scan (VCR, Camcoder, DAT)
  Spins head at angle to tape to improve density

Current Drawbacks to Tape

- Tape wear out:
  - Helical 100s of passes to 1000s for longitudinal
- Head wear out:
  - 2000 hours for helical
- Both must be accounted for in economic / reliability model
- Bits stretch
- Readers must be compatible with multiple generations of media
- Long rewind, eject, load, spin-up times; not inherent, just no need in marketplace
- Designed for archival

Automated Cartridge System: StorageTek Powderhorn 9310



- 6000 x 50 GB 9830 tapes = 300 TBytes in 2000 (uncompressed)
  - Library of Congress: all information in the world; in 1992, ASCII of all books = 30 TB
  - Exchange up to 450 tapes per hour (8 secs/tape)
- 1.7 to 7.7 Mbyte/sec per reader, up to 10 readers

Library vs. Storage

- Getting books today as quaint as the way I learned to program
  - punch cards, batch processing
  - wander thru shelves, anticipatory purchasing
- Cost $1 per book to check out
- $30 for a catalogue entry
- 30% of all books never checked out
- Write only journals?
- Digital library can transform campuses

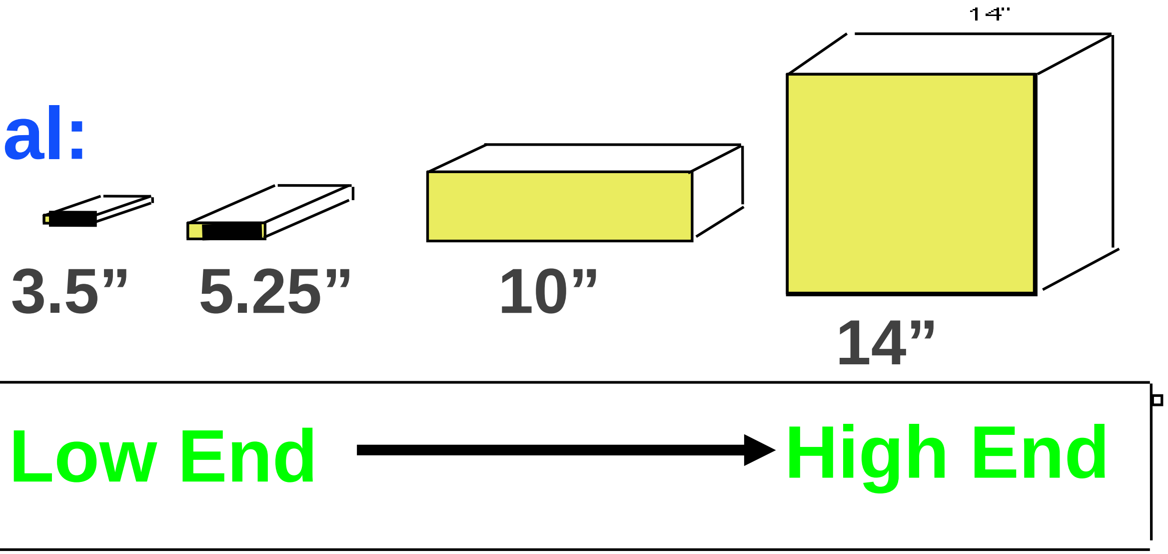
Whither tape?

- Investment in research:
  - 90% of disks shipped in PCs; 100% of PCs have disks
  - ~0% of tape readers shipped in PCs; ~0% of PCs have disks
- Before, N disks / tape; today, N tapes / disk
  - 40 GB/DLT tape (uncompressed)
  - 80 to 192 GB/3.5" disk (uncompressed)
- Cost per GB:
  - In past, 10X to 100X tape cartridge vs. disk
  - Jan 2001: 40 GB for $53 (DLT cartridge), $2800 for reader
  - $1.33/GB cartridge, $2.03/GB 100 cartridges + 1 reader
  - ($10995 for 1 reader + 15 tape autoloader, $10.50/GB)
  - Jan 2001: 80 GB for $244 (IDE,5400 RPM), $3.05/GB
  - Will $/GB tape v. disk cross in 2001? 2002? 2003?
- Storage field is based on tape backup; what should we do? Discussion if time permits?

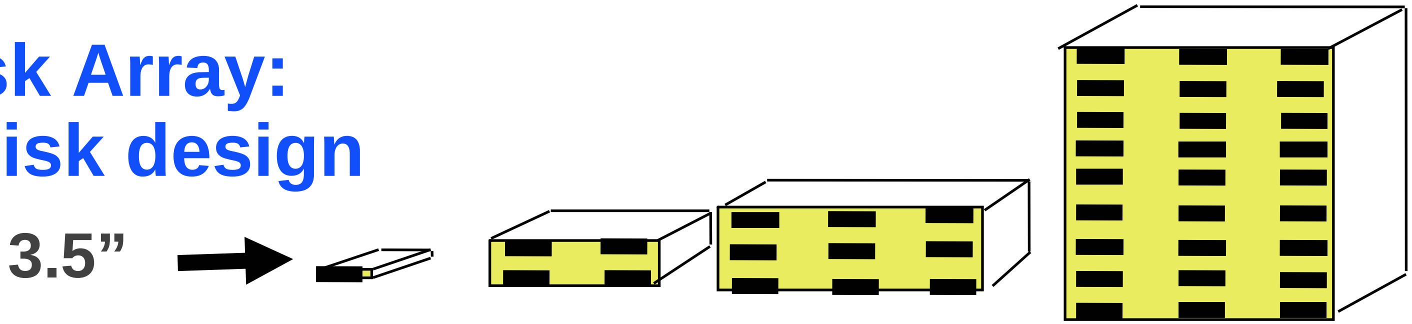
Use Arrays of Small Disks?

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

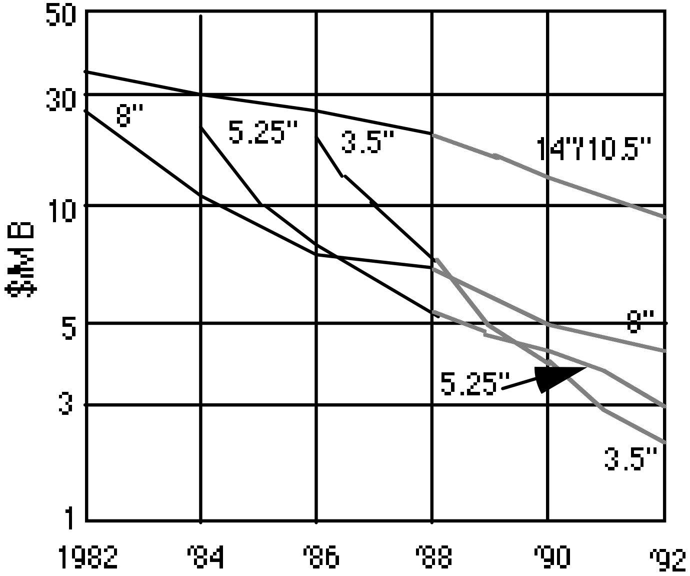
Conventional:
4 disk designs

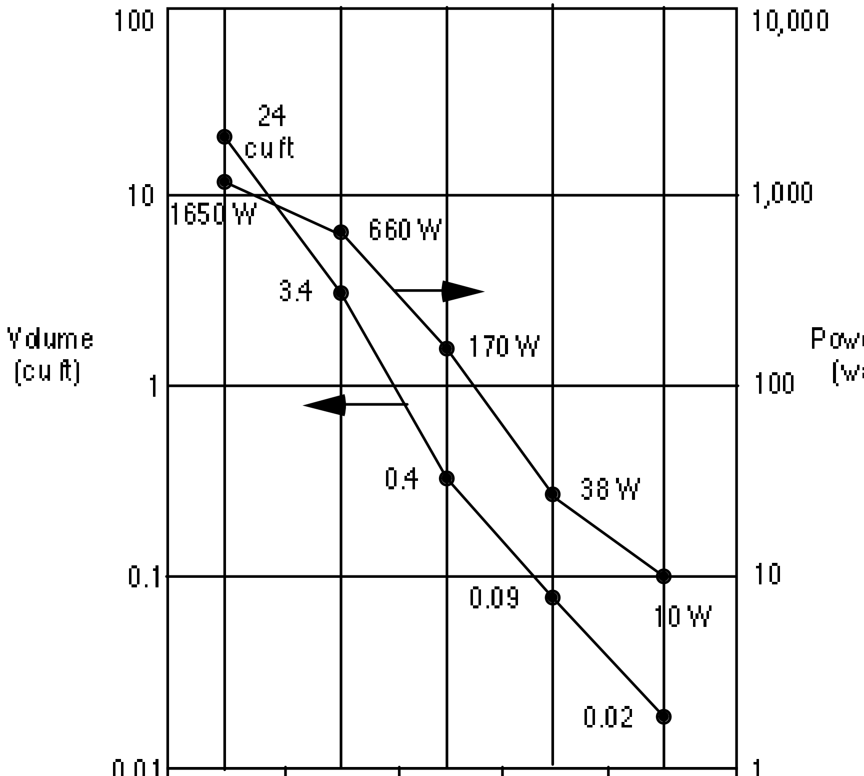


Disk Array:
1 disk design



Advantages of Small Formfactor Disk Drives





	14"	10.5"	8"	5.25"	3.5"
MB	7,500 F	1,320 F	1,000	765	200
MB/watt	4.54	2.0	5.88	20.13	20.0
MB/cu ft	313	388	2,500	8,500	10,000

Low cost/MB
High MB/volume
High MB/watt
Low cost/Actuator

Cost and Environmental Efficiencies

Replace Small Number of Large Disks with Large Number of Small Disks! (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 IOs/s	6X
MTTF	250 KHrs	50 KHrs	??? Hrs	
Cost	$250K	$2K	$150K	

Disk Arrays have potential for large data and I/O rates, high MB per cu. ft., high MB per KW, but what about reliability?

Array Reliability

- Reliability of N disks = Reliability of 1 Disk ÷ N

  50,000 Hours ÷ 70 disks = 700 hours

  Disk system MTTF: Drops from 6 years to 1 month!

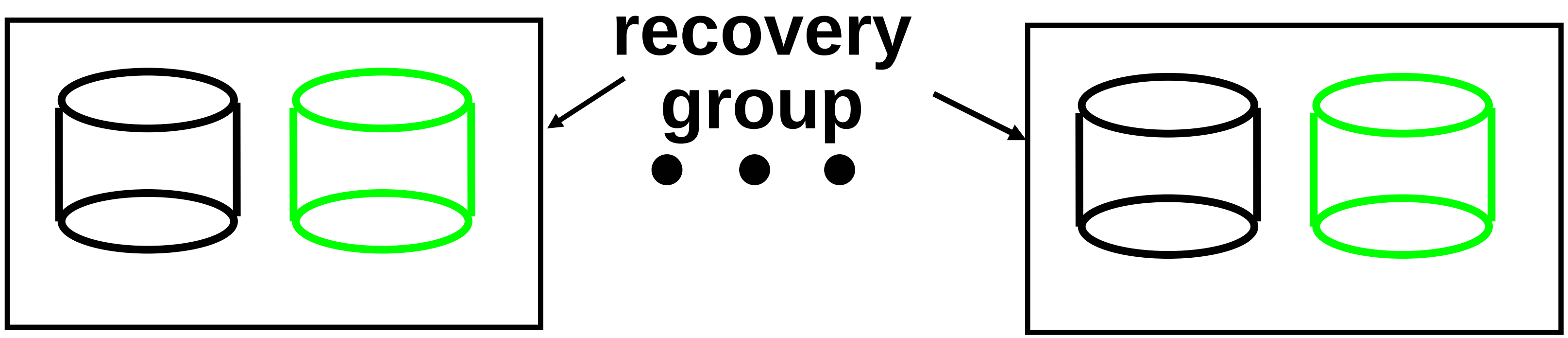
- Arrays (without redundancy) too unreliable to be useful!

Hot spares support reconstruction in parallel with access: very high media availability can be achieved

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

Redundant Arrays of Inexpensive Disks
RAID 1: Disk Mirroring/Shadowing



- Each disk is fully duplicated onto its "<u>mirror</u>"

  Very high availability can be achieved
- Bandwidth sacrifice on write:

  Logical write = two physical writes
  - Reads may be optimized
- Most expensive solution: 100% capacity overhe

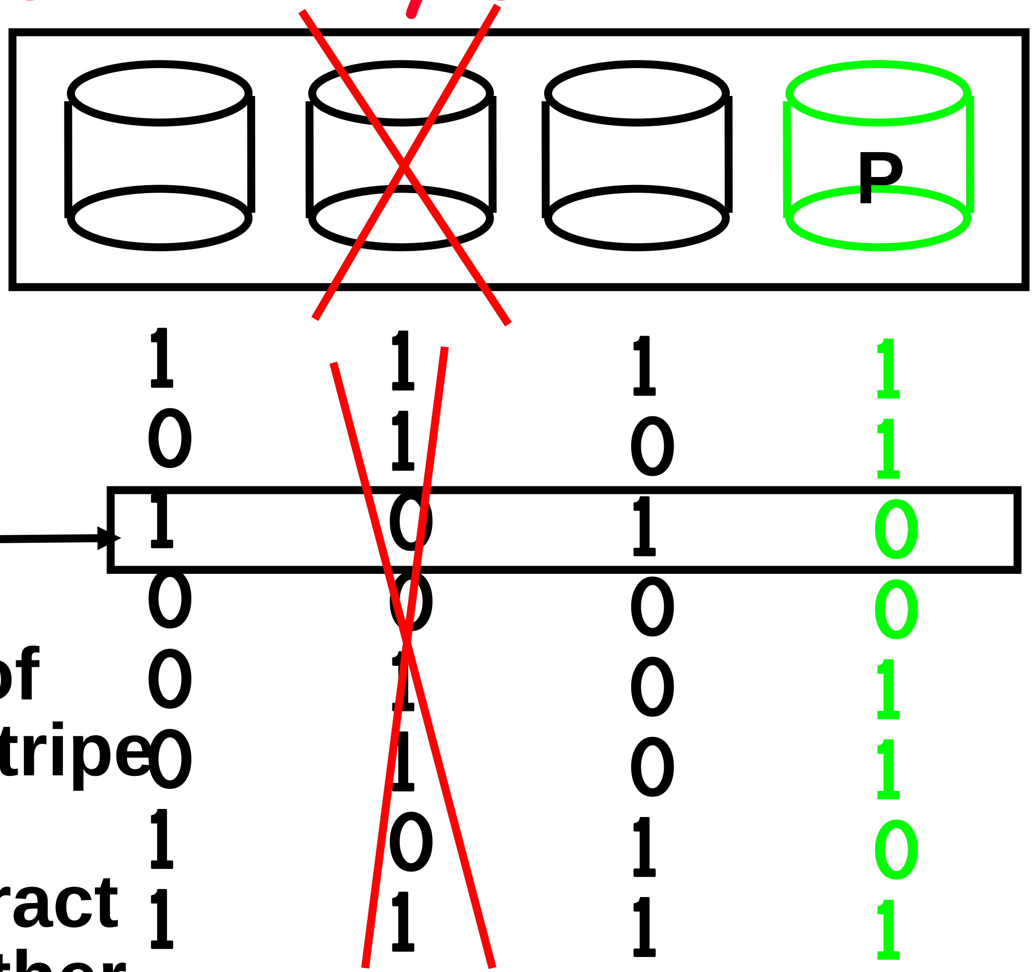
- (RAID 2 not interesting, so skip)

Redundant Array of Inexpensive Disks
RAID 3: Parity Disk

10010011
11001101
10010011
. . .

logical record

Striped physical records



- **P contains sum of other disks per stripe mod 2 ("parity")**
- **If disk fails, subtract P from sum of other disks to find missing information**

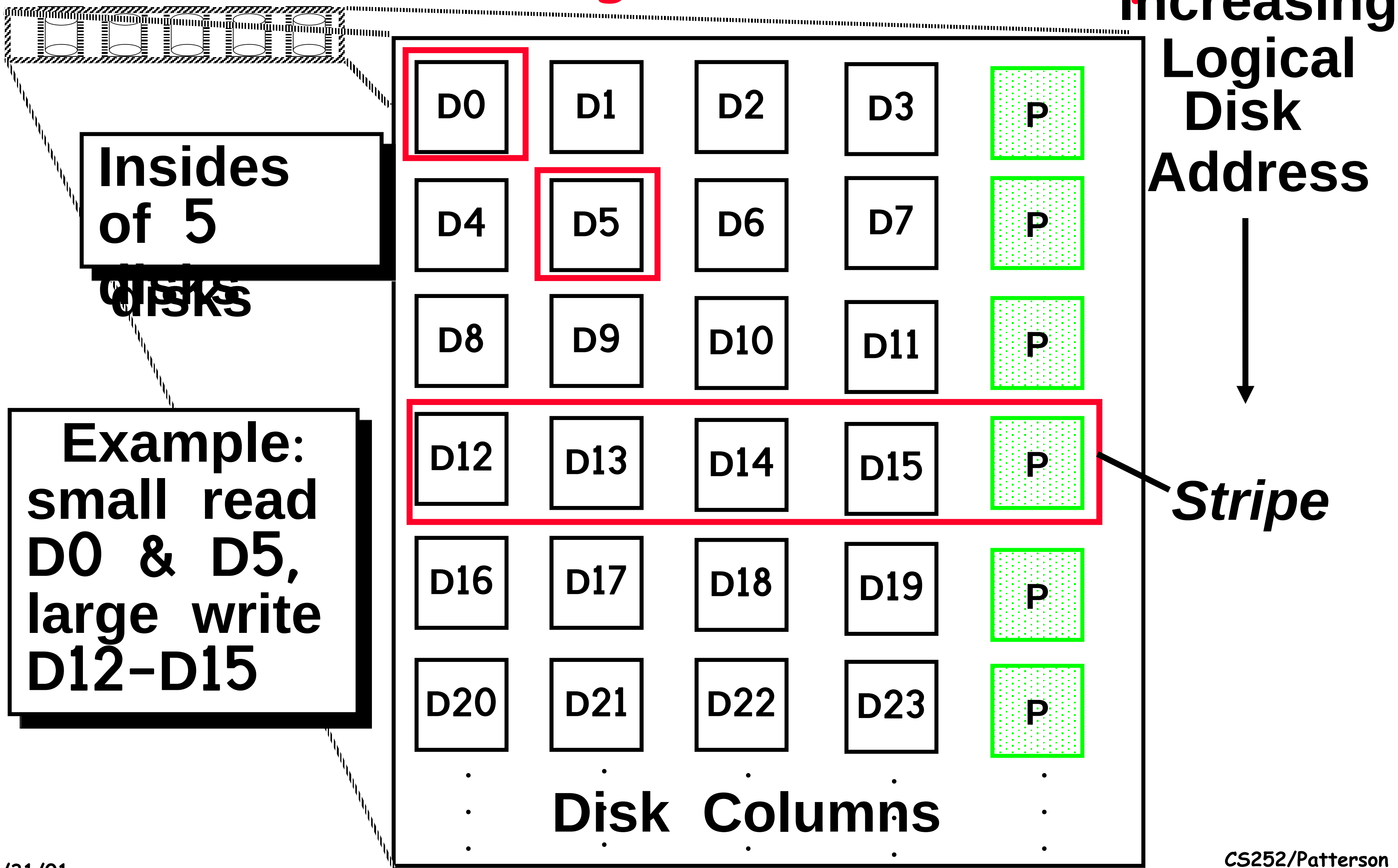
RAID 3

- Sum computed across recovery group to protect against hard disk failures, stored in P disk
- Logically, a single high capacity, high transfer rate disk: good for large transfers
- Wider arrays reduce capacity costs, but decreases availability
- 33% capacity cost for parity in this configuration

Inspiration for RAID 4

- RAID 3 relies on parity disk to discover errors on Read
- But every sector has an error detection field
- Rely on error detection field to catch errors on read, not on the parity disk
- Allows independent reads to different disks simultaneously

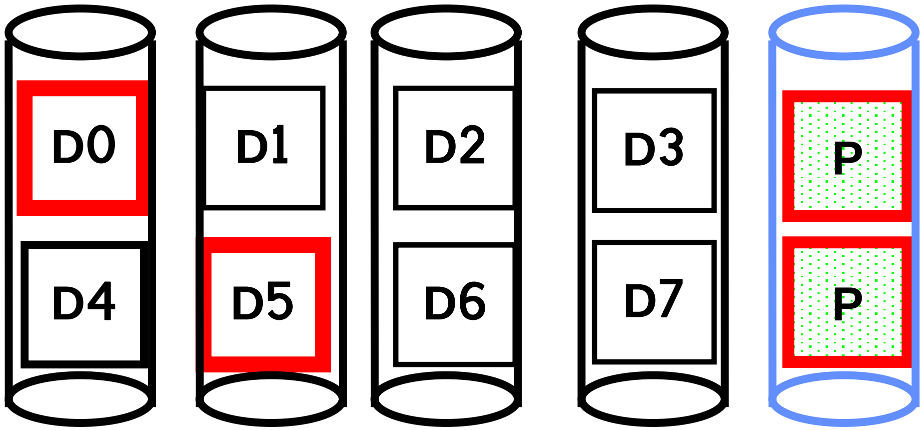
Redundant Arrays of Inexpensive Disks
RAID 4: High I/O Rate Parity



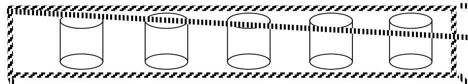
Example: small read D0 & D5, large write D12-D15

Inspiration for RAID 5

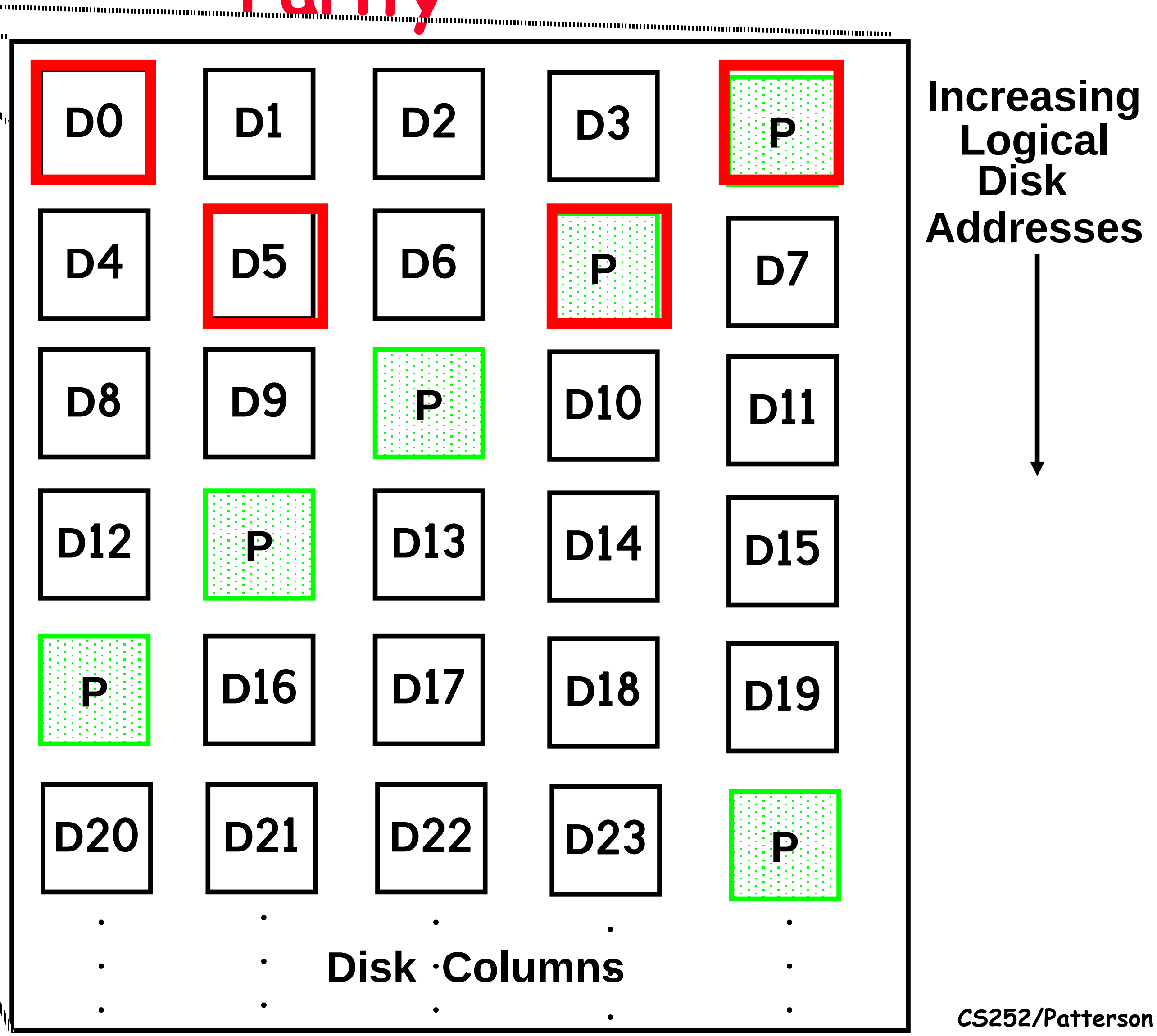
- RAID 4 works well for small reads
- Small writes (write to one disk):
  - Option 1: read other data disks, create new sum and write to Parity Disk
  - Option 2: since P has old sum, compare old data to new data, add the difference to P
- Small writes are limited by Parity Disk: Write to D0, D5 both also write to P disk



Redundant Arrays of Inexpensive Disks RAID 5: High I/O Rate Interleaved Parity



Independent writes possible because of interleaved parity

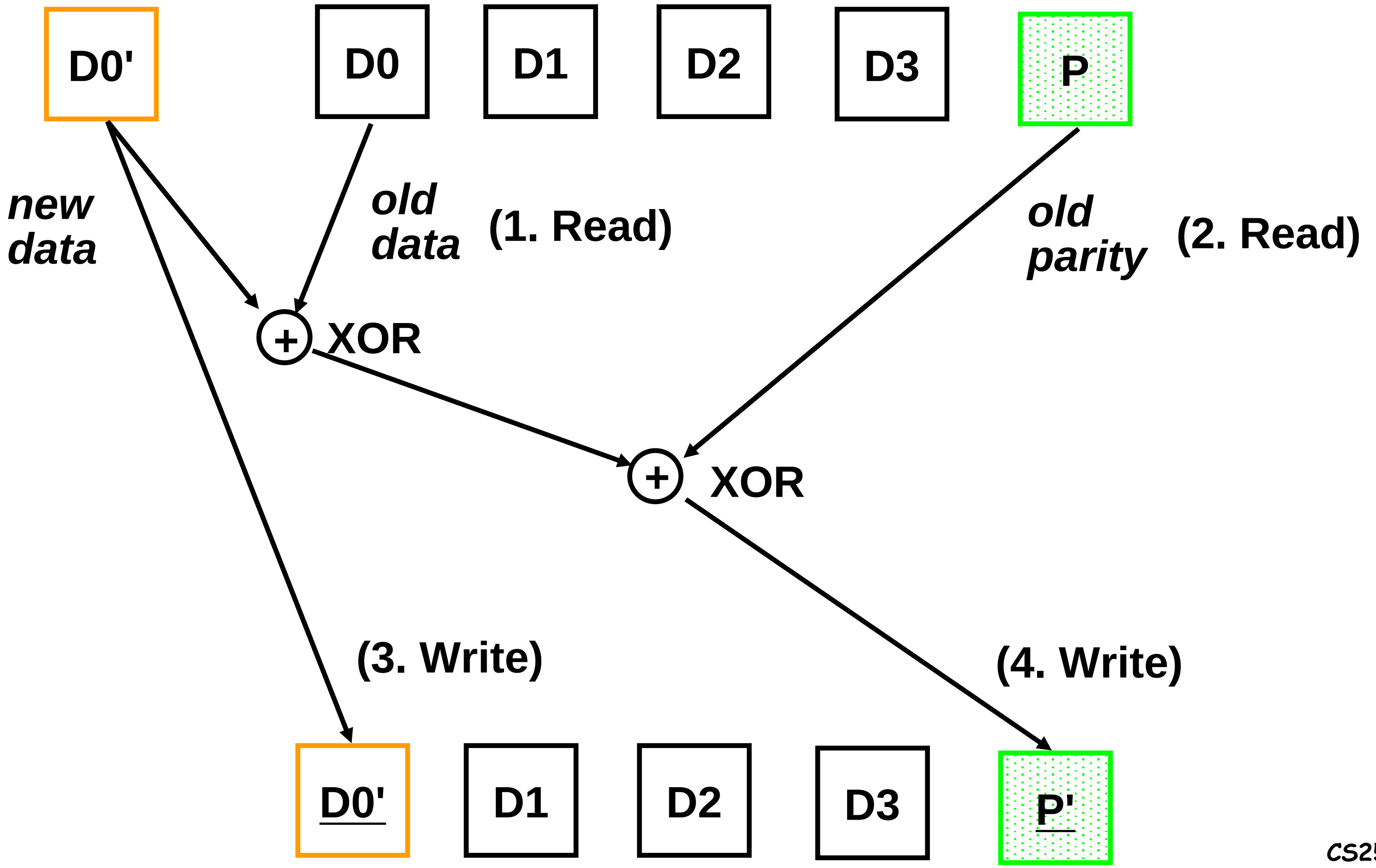


Example: write to D0, D5 uses disks 0, 1, 3, 4

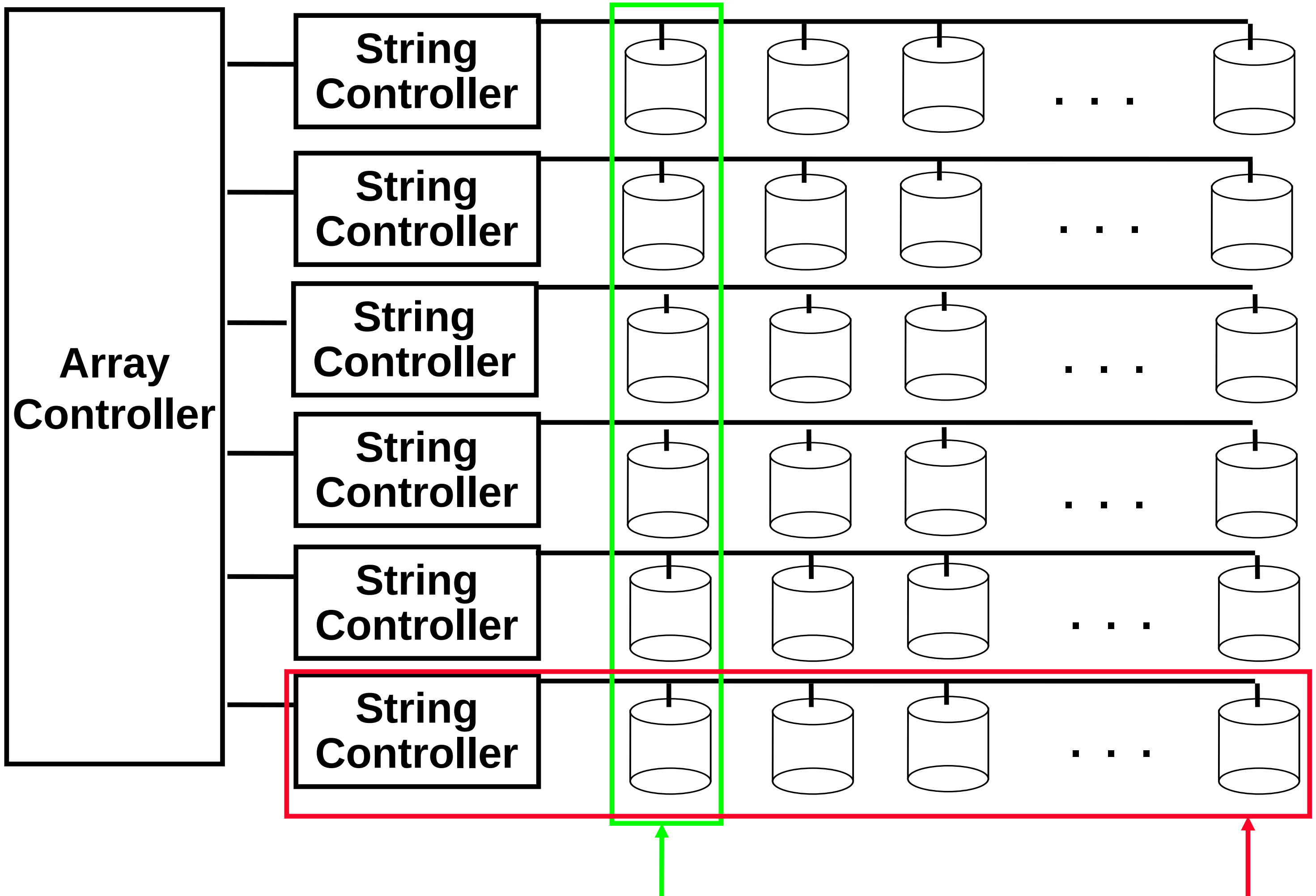
Problems of Disk Arrays: Small Writes

RAID-5: Small Write Algorithm

1 Logical Write = 2 Physical Reads + 2 Physical Writes



System Availability: Orthogonal RAIDs

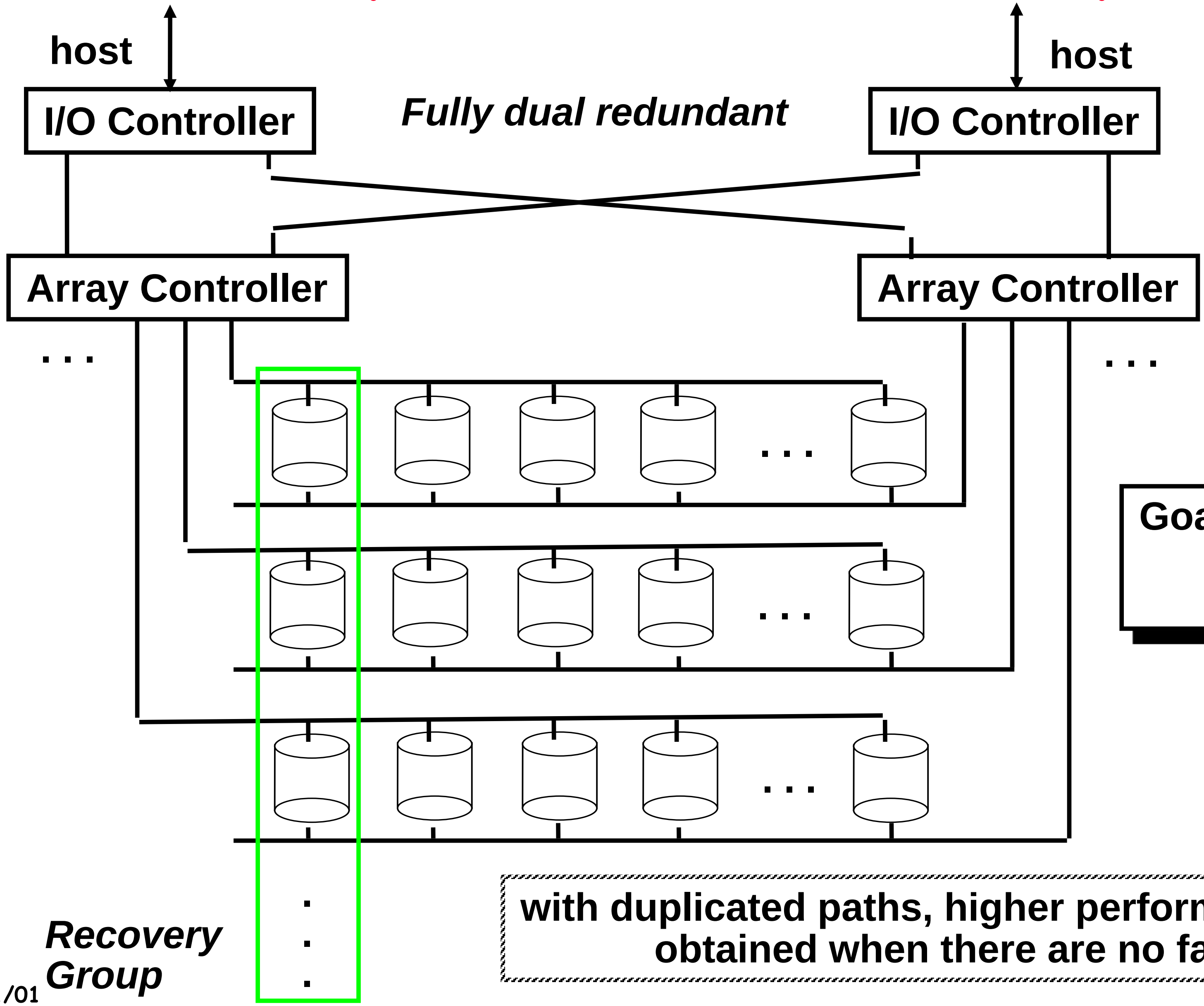


Data Recovery Group: **unit of data redundancy**

Redundant Support Components: **fans, power supplies, controller, cables**

End to End Data Integrity: **internal parity protected data paths**

System-Level Availability



Goal: No Single Points of Failure

with duplicated paths, higher performance can be obtained when there are no failures

Berkeley History: RAID-I

- RAID-I (1989)
  - Consisted of a Sun 4/280 workstation with 128 MB of DRAM, four dual-string SCSI controllers, 28 5.25-inch SCSI disks and specialized disk striping software
- Today RAID is $19 billion dollar industry, 80% nonPC disks sold in RAIDs

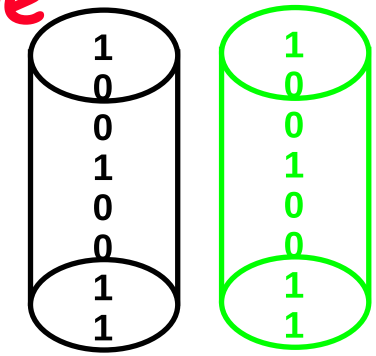
Summary: RAID Techniques: Goal was performance, popularity due to reliability of storage

- ***Disk Mirroring, Shadowing (RAID 1)***

  Each disk is fully duplicated onto its "shadow"

  Logical write = two physical writes

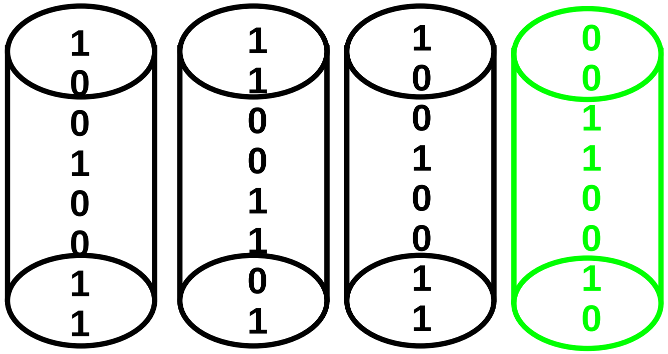
  100% capacity overhead



- ***Parity Data Bandwidth Array (RAID 3)***

  Parity computed horizontally

  Logically a single high data bw disk

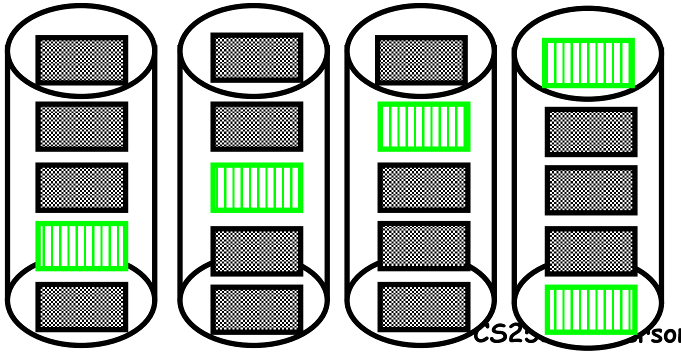


- ***High I/O Rate Parity Array (RAID 5)***

  Interleaved parity blocks

  Independent reads and writes

  Logical write = 2 reads + 2 writes



Summary Storage

- Disks:
  - Extraodinary advance in capacity/drive, $/GB
  - Currently 17 Gbit/sq. in. ; can continue past 100 Gbit/sq. in.?
  - Bandwidth, seek time not keeping up: 3.5 inch form factor makes sense? 2.5 inch form factor in near future? 1.0 inch form factor in long term?
- Tapes
  - No investment, must be backwards compatible
  - Are they already dead?
  - What is a tapeless backup system?