Chapter 10

Mass Storage & Disk Structures

Disks

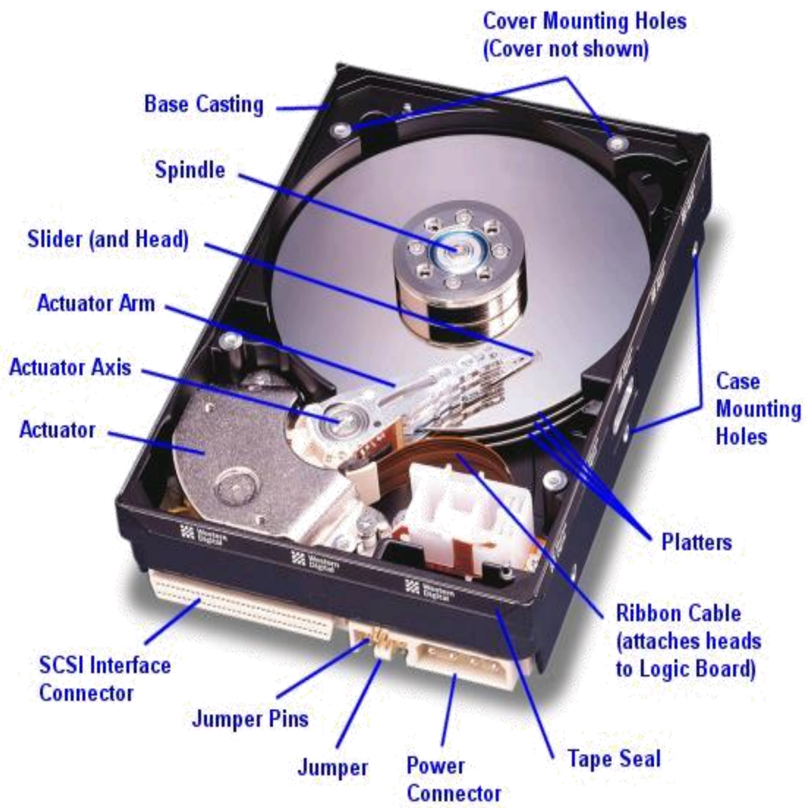
Form factor:
.5-1”× 4”× 5.7”
Storage:
18-73GB

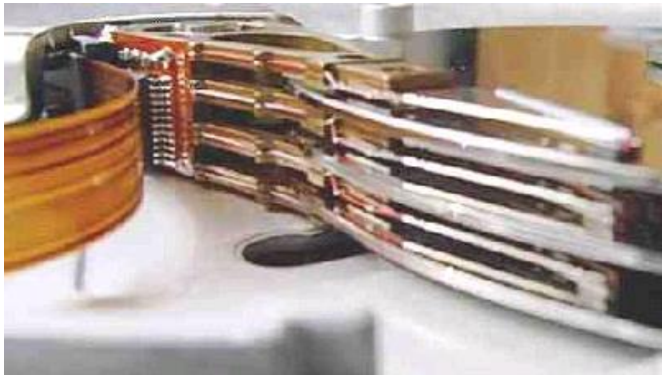
Form factor:
.4-.7” × 2.7” × 3.9”
Storage:
4-27GB

Form factor:
.2-.4” × 2.1” × 3.4”
Storage:
170MB-1GB

Old mainframe disks

Hard Disk Drives





Read/Write Head Side View

Western Digital Drive

http://www.storagereview.com/guide/IBM/Hitachi Microdrive

Solid State Drives

- Flash storage technology
  - Semiconductor technology
  - No moving parts
  - No spin-up time or noise
  - Very low power consumption
  - Low random access times
  - Very light-weight
  - Unaffected by magnetic fields
  - Shock and vibration resistant
  - Ability to handle extremes of temperature

Same interface and form factors as Existing hard disks.

SSD technology

- Uses semiconductor flash storage technology
  - NOR technology (mostly used in embedded systems)
  - NAND technology (thumb drives, SS Drives)
- Major difference from flash drives is the sophisticated controller on SSD’s.
- With all flash storage, each bit can be written only a finite number of times.

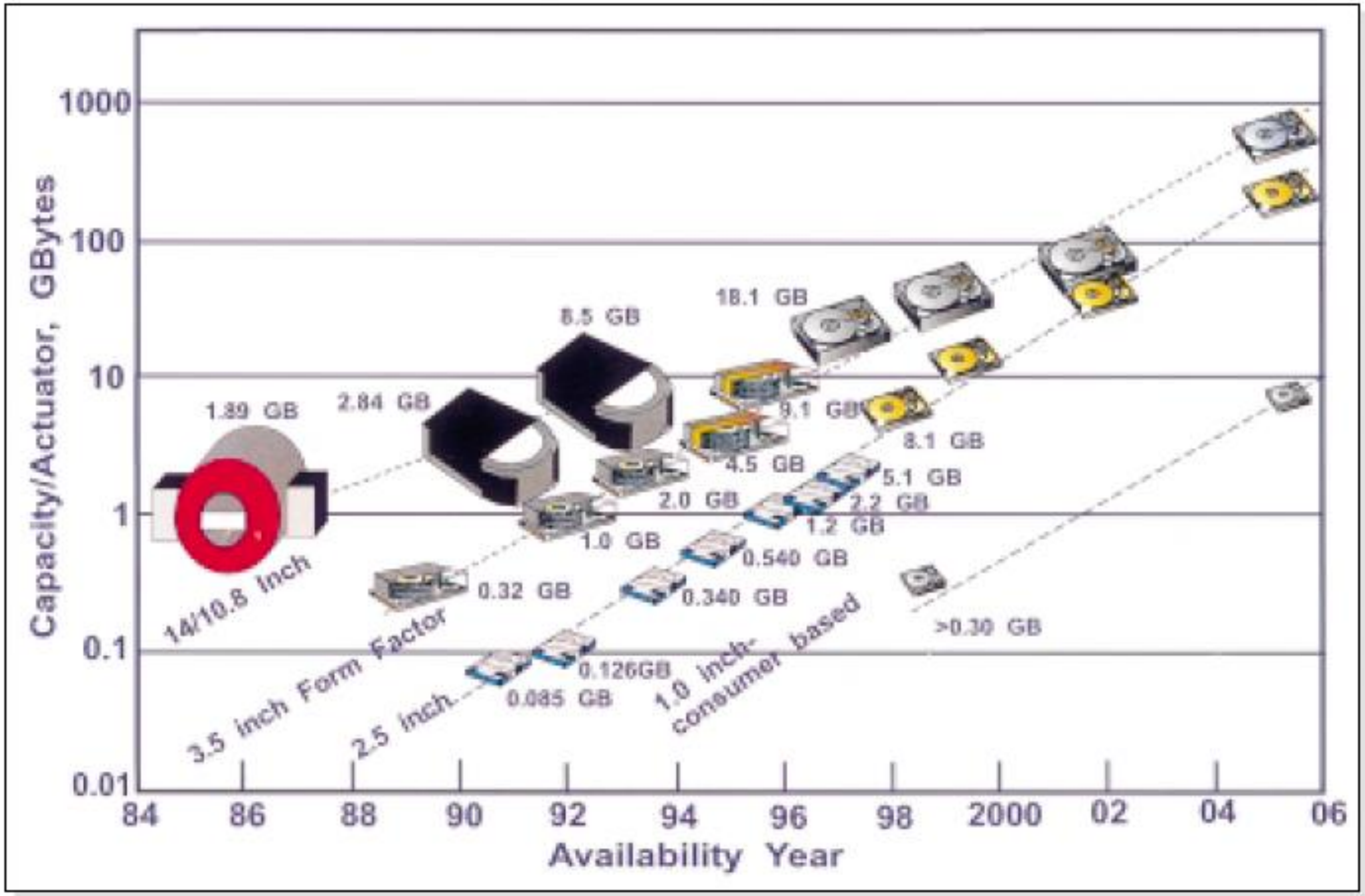
SSD Controller

- Every SSD contains an internal embedded processor (controller) that functions as a bridge between its NAND flash memory and a host (such as a computer).
- Controller responsible for the SSD’s performance and its features:
  - reading and writing
  - Erasing
  - Encryption
  - error checking and correction (ECC),
  - Bad block mapping
  - garbage collection
  - wear-leveling
- The Controller and its NAND non-volatile memory are the two primary components of all SSDs.
- The Controller is not to be confused with the actual I/O controller interface, which is typically a SATA interface used to physically attach the SSD to the host.

Disk Technology Trends

- Disks are getting smaller for similar capacity
  - Spin faster, less rotational delay, higher bandwidth
  - Less distance for head to travel (faster seeks)
  - Lighter weight (for portables)
- Disk data is getting denser
  - More bits/square inch
  - Tracks are closer together
  - Doubles density every 18 months
- Disks are getting cheaper ($/MB)
  - Factor of $\approx$ 2 per year since 1991
  - Head close to surface

Disk Technology Trends



- **From the paper: E. Growchowski. *Emerging Trends in Data Storage on Magnetic Hard Disk Drives*. In Datatech, pages 11-16. ICG Publishing, September 1998.**

Disk Technology Trends

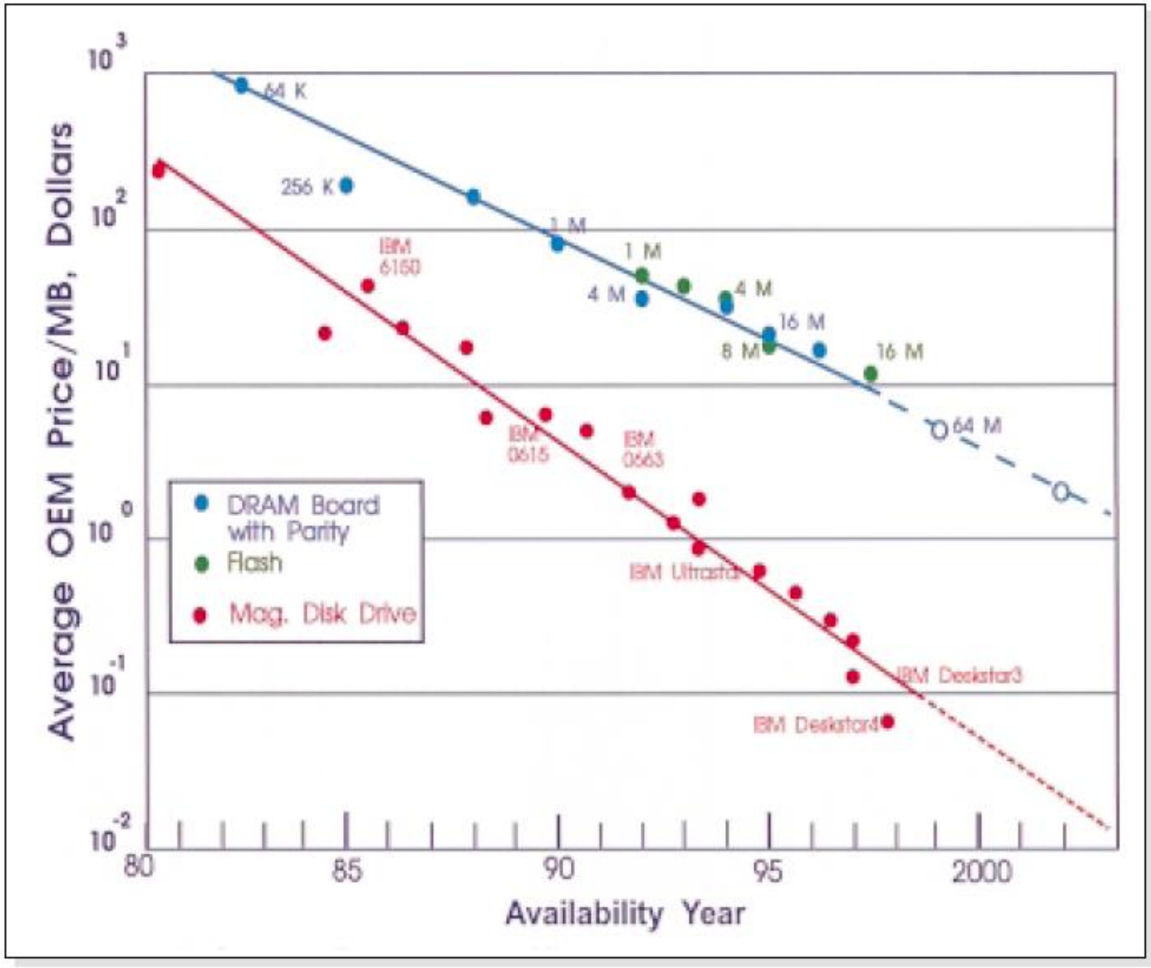
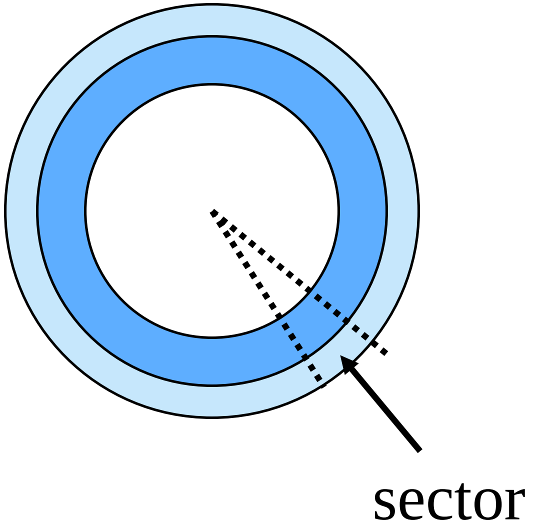


Figure 7. Projection of average price per megabyte for HDDs and DRAMs.

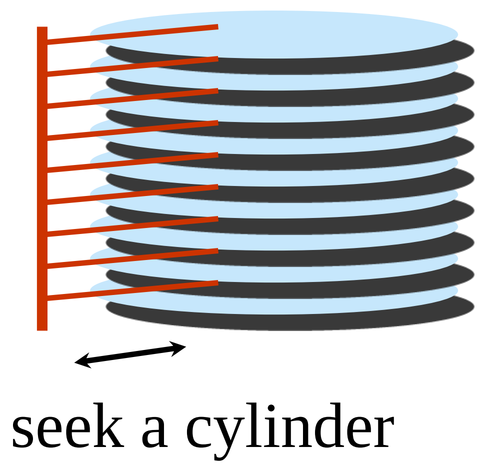
Disk Organization

- Disk surface
  - Circular disk coated with magnetic material
- Tracks
  - Concentric rings around disk surface, bits laid out serially along each track
- Sectors
  - Each track is split into arc of track (min unit of transfer)



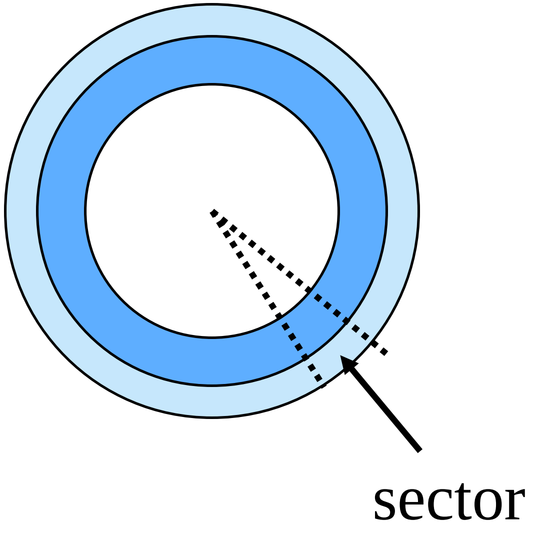
More on Disks

- CD’s and floppies come individually, but magnetic disks come organized in a disk pack
- Cylinder
  - Certain track of the platter
- Disk arm
  - Seek the right cylinder

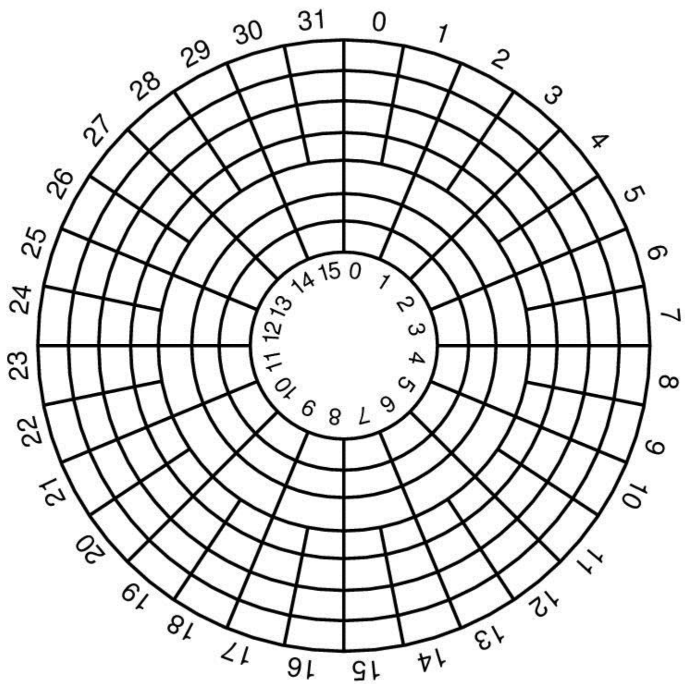


Disk Organization As Fiction

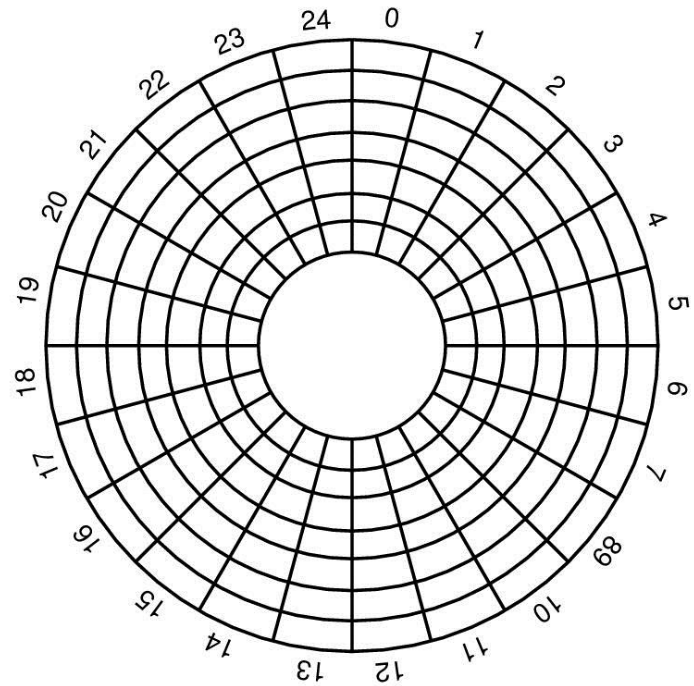
- Fixed arc implies inefficiency
  - short inner sectors, long outer sectors
- Reality
  - More sectors on outer tracks
  - Disks map transparently



Disk Hardware



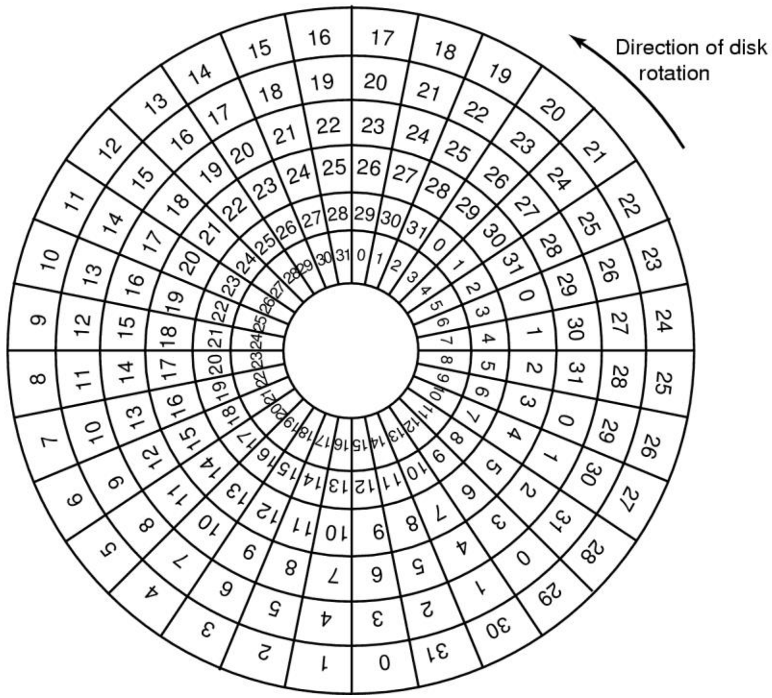
Physical geometry



Virtual geometry

- To hide the complexity of the physical geometry, most modern disks present a virtual view of the disk to the OS.
- Controller maps virtual address to physical address.

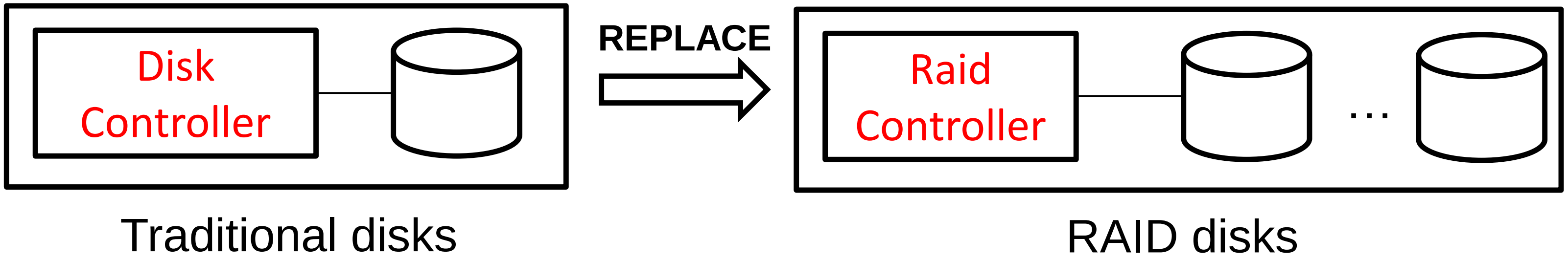
Disk Formatting



- Cylinder skew is used to improve performance:
  - Sector 0 in each cylinder is offset
  - Allows disk to read multiple tracks in one continuous operation
    - By allowing time for the R/W head to change tracks.

RAID

- Parallel I/O
- RAID: Redundant Array of Independent Disks
- Relies on redundancy
- Collection of disks arranged in a specific way to obtain more speed and/or reliability



The rest of the computer should not be able to tell the difference

RAID structures

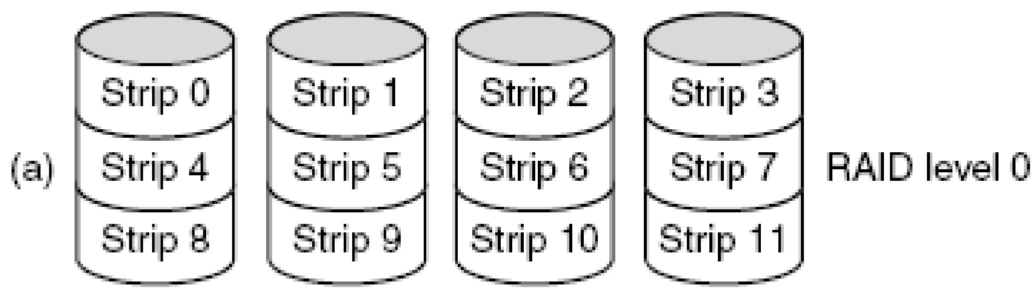
- Redundancy is achieved via duplication of data
- Parallelism is achieved via striping
- Striping can be at the bit level or block level
  - Bit level: bits of a byte distributed across multiple physical disks
  - Block level: blocks are distributed across multiple physical disks

RAID

- Redundant Array of Inexpensive (Independent) Disks
  - Use multiple smaller disks (c.f. one large disk)
  - Parallelism improves performance
  - Plus extra disk(s) for redundant data storage
- Provides fault tolerant storage system
  - Especially if failed disks can be “hot swapped”

RAID

- RAID 0
  - No redundancy ("AID"?)
    - Just stripe data over multiple disks
  - But it does improve performance (parallel access)

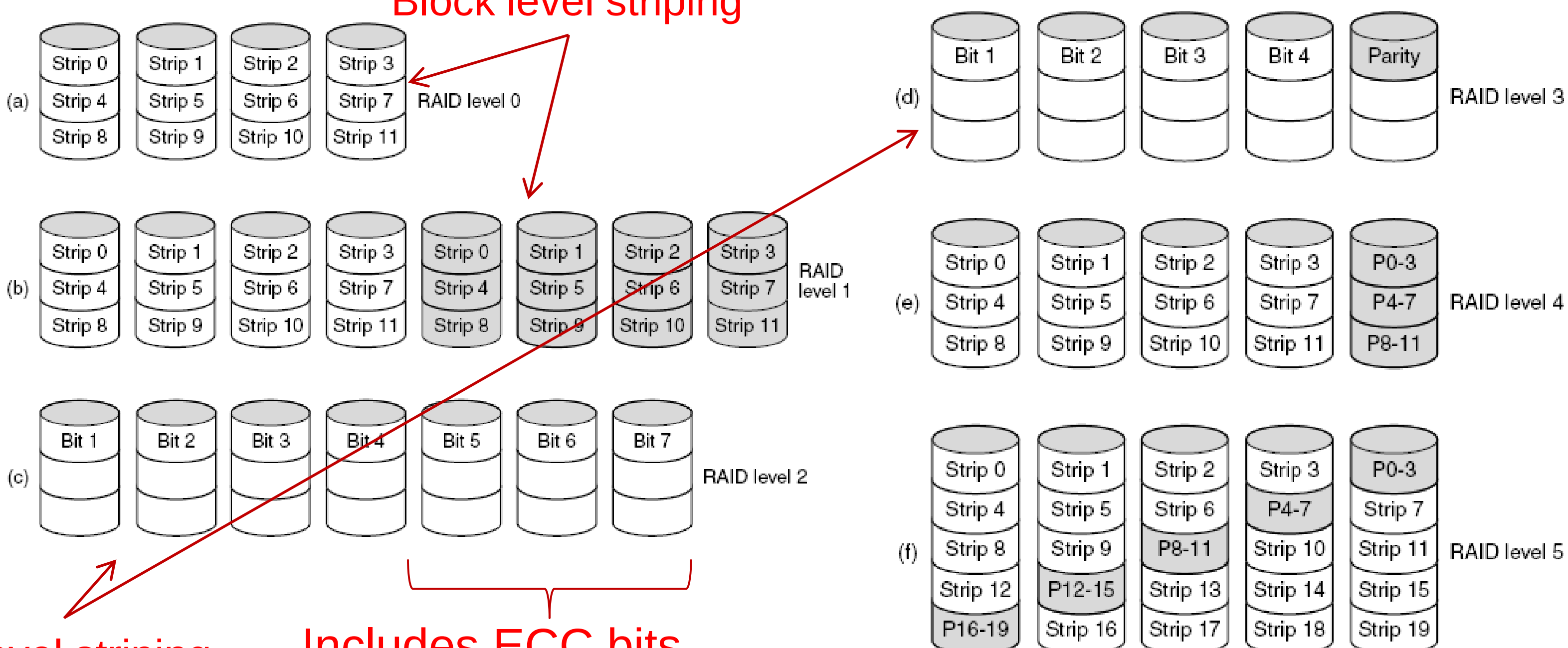


RAID

- RAID levels 0 through 5.
  Backup and parity drives are shown shaded.

1 parity bit disk
Can utilize disk controller ability to detect damaged sectors

Block level striping

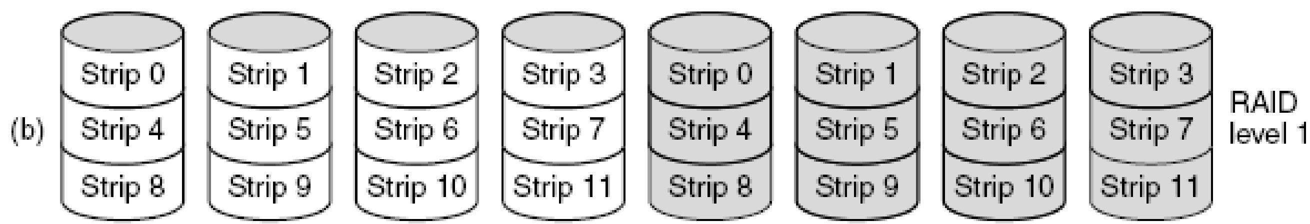


Bit level striping

Includes ECC bits
One disk per ECC bit

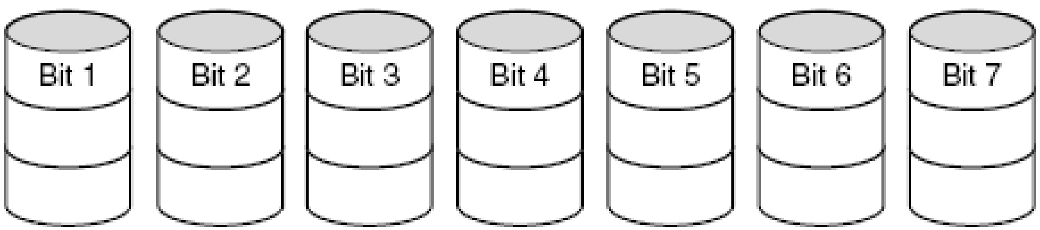
RAID 1 & 2

- RAID 1: Mirroring
  - N + N disks, replicate data
    - Write data to both data disk and mirror disk
    - On disk failure, read from mirror



RAID 1 & 2

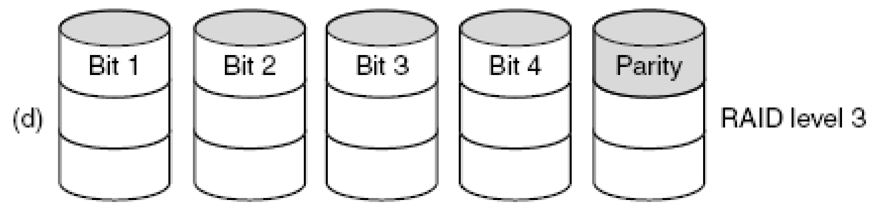
- RAID 2: Error correcting code (ECC)
  - N + E disks (e.g., 10 + 4)
  - Split data at bit level across N disks
  - Generate E-bit ECC
  - Too complex, not used in practice



(c) RAID level 2

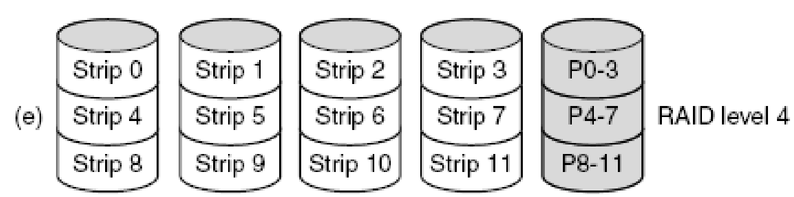
RAID 3: Bit-Interleaved Parity

- N + 1 disks
  - Data striped across N disks at byte level
  - Redundant disk stores parity
  - Read access
    - Read all disks
  - Write access
    - Generate new parity and update all disks
  - On failure
    - Use parity to reconstruct missing data
- Not widely used

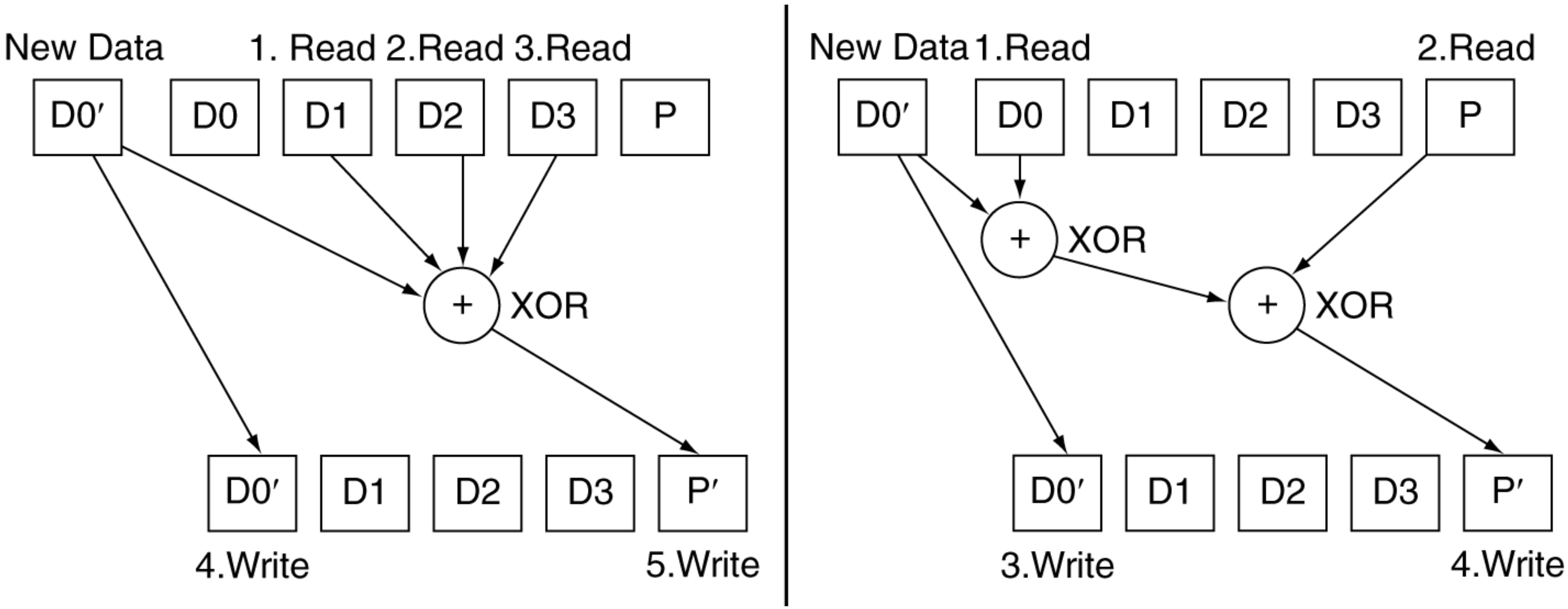


RAID 4: Block-Interleaved Parity

- N + 1 disks
  - Data striped across N disks at block level
  - Redundant disk stores parity for a group of blocks
  - Read access
    - Read only the disk holding the required block
  - Write access
    - Just read disk containing modified block, and parity disk
    - Calculate new parity, update data disk and parity disk
  - On failure
    - Use parity to reconstruct missing data
- Not widely used

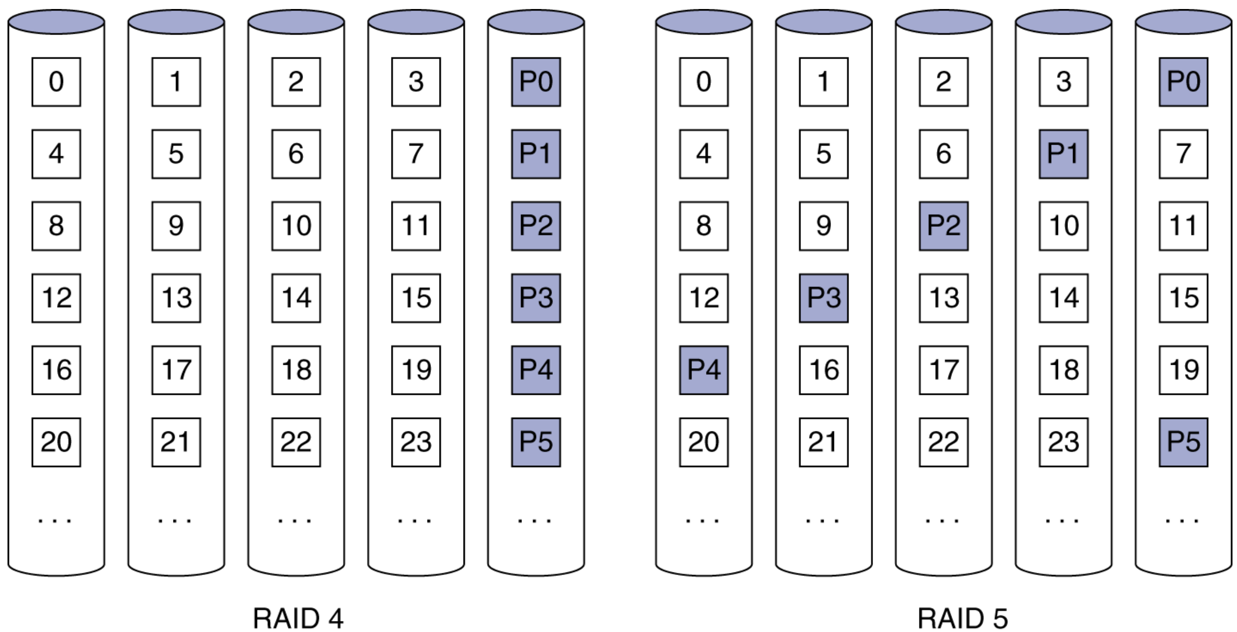


RAID 3 vs RAID 4



RAID 5: Distributed Parity

- N + 1 disks
  - Like RAID 4, but parity blocks distributed across disks
    - Avoids parity disk being a bottleneck
- Widely used



RAID Summary

- RAID can improve performance and availability
  - High availability requires hot swapping
- Assumes independent disk failures
  - Too bad if the building burns down!
- See “Hard Disk Performance, Quality and Reliability”
  - http://www.pcguide.com/ref/hdd/perf/index.htm

Disk Hardware

Parameter	IBM 360-KB floppy disk	WD 18300 hard disk
Number of cylinders	40	10601
Tracks per cylinder	2	12
Sectors per track	9	281 (avg)
Sectors per disk	720	35742000
Bytes per sector	512	512
Disk capacity	360 KB	18.3 GB
Seek time (adjacent cylinders)	6 msec	0.8 msec
Seek time (average case)	77 msec	6.9 msec
Rotation time	200 msec	8.33 msec
Motor stop/start time	250 msec	20 sec
Time to transfer 1 sector	22 msec	17 μsec

Disk parameters for the original IBM PC floppy disk and a Western Digital WD 18300 hard disk

Disk Examples (Summarized Specs)

	Seagate Barracuda	IBM Ultrastar 72ZX
Capacity, Interface & Configuration		
Formatted Gbytes	28	73.4
Interface	Ultra ATA/66	Ultra160 SCSI
Platters / Heads	4 / 8	11/22
Bytes per sector	512	512-528
Performance		
Max Internal transfer rate (Mbytes/sec)	40	53
Max external transfer rate (Mbytes/sec)	66.6	160
Avg Transfer rate(Mbytes/sec)	> 15	22.1-37.4
Multisegmented cache (Kbytes)	512	16,384
Average seek, read/write (msec)	8	5.3
Average rotational latency (msec)	4.16	2.99
Spindle speed (RPM)	7,200	10,000

Disk Performance

- Seek
  - Position heads over cylinder, typically 5.3 – 8 ms
- Rotational delay
  - Wait for a sector to rotate underneath the heads
  - Typically 8.3 – 6.0 ms (7,200 – 10,000RPM) or ½ rotation takes 4.15-3ms
- Transfer bytes
  - Average transfer bandwidth (15-37 Mbytes/sec)
- Performance of transfer 1 Kbytes
  - Seek (5.3 ms) + half rotational delay (3ms) + transfer (0.04 ms)
  - Total time is 8.34ms or 120 Kbytes/sec!
- What block size can get 90% of the disk transfer bandwidth?

Disk Behaviors

- There are more sectors on outer tracks than inner tracks
  - Read outer tracks: 37.4MB/sec
  - Read inner tracks: 22MB/sec
- Seek time and rotational latency dominates the cost of small reads
  - A lot of disk transfer bandwidth are wasted
  - Need algorithms to reduce seek time!

Block Size (Kbytes)	% of Disk Transfer Bandwidth
1Kbytes	0.5%
8Kbytes	3.7%
256Kbytes	55%
1Mbytes	83%
2Mbytes	90%

Disk Arm Scheduling Algorithms

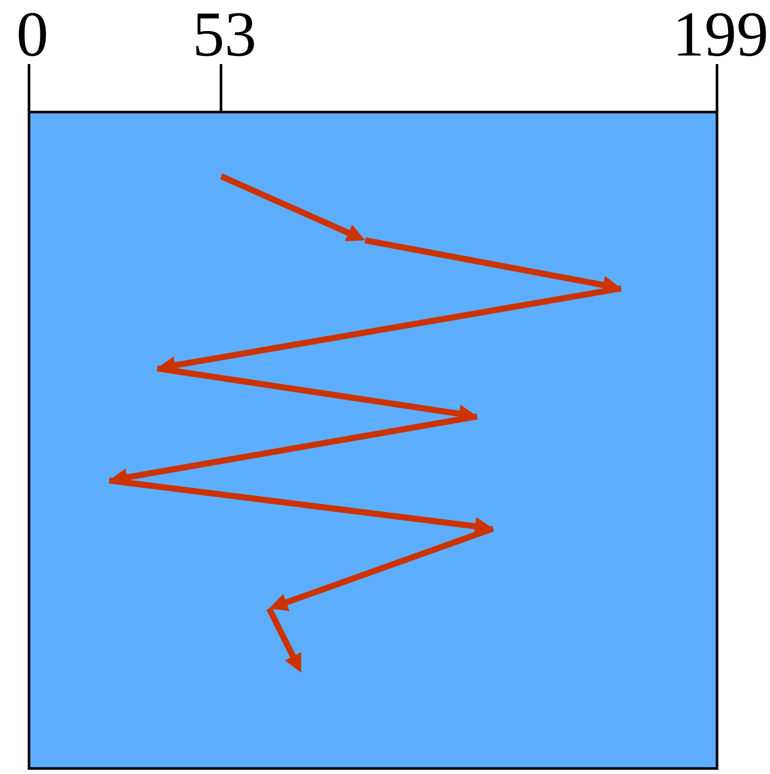
- Time required to read or write a disk block determined by 3 factors
  1. Seek time
  2. Rotational delay
  3. Actual transfer time
- Seek time dominates
- Error checking is done by controllers
- Peak bandwidth high, but rarely achieved
- Need to mitigate disk performance impact
  – Do extra calculations to speed up disk access
    - Schedule requests to shorten seeks
  – Move some disk data into main memory – file system caching

Disk Arm Scheduling

- Which disk request is serviced first?
  - FCFS
  - Shortest seek time first
  - Elevator (SCAN)
  - C-SCAN (Circular SCAN)
- Look familiar?

FIFO (FCFS) order

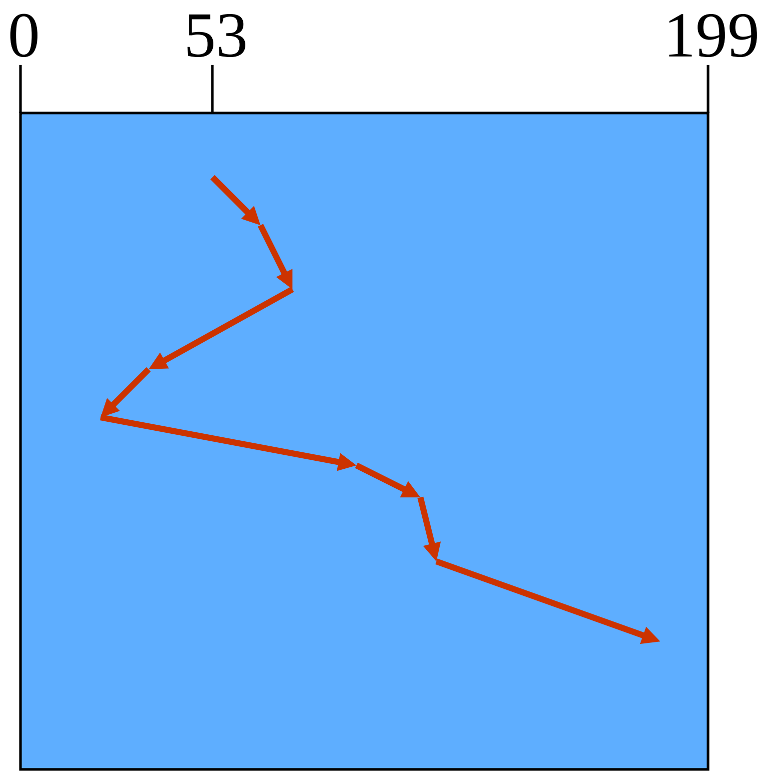
- Method
  - First come first serve
- Pros
  - Fairness among requests
  - In the order applications expect
- Cons
  - Arrival may be on random spots on the disk (long seeks)
  - Wild swing can happen



98, 183, 37, 122, 14, 124, 65, 67

SSTF (Shortest Seek Time First)

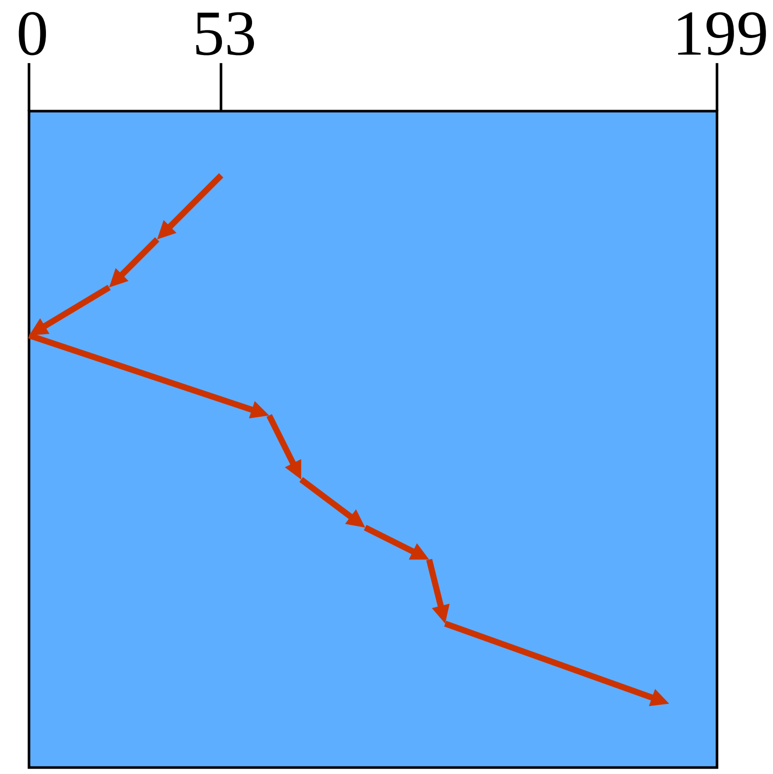
- Method
  - Pick the one closest to current head position on disk
- Pros
  - Try to minimize seek time
- Cons
  - Starvation



98, 183, 37, 122, 14, 124, 65, 67
(65, 67, 37, 14, 98, 122, 124, 183)

Elevator (SCAN)

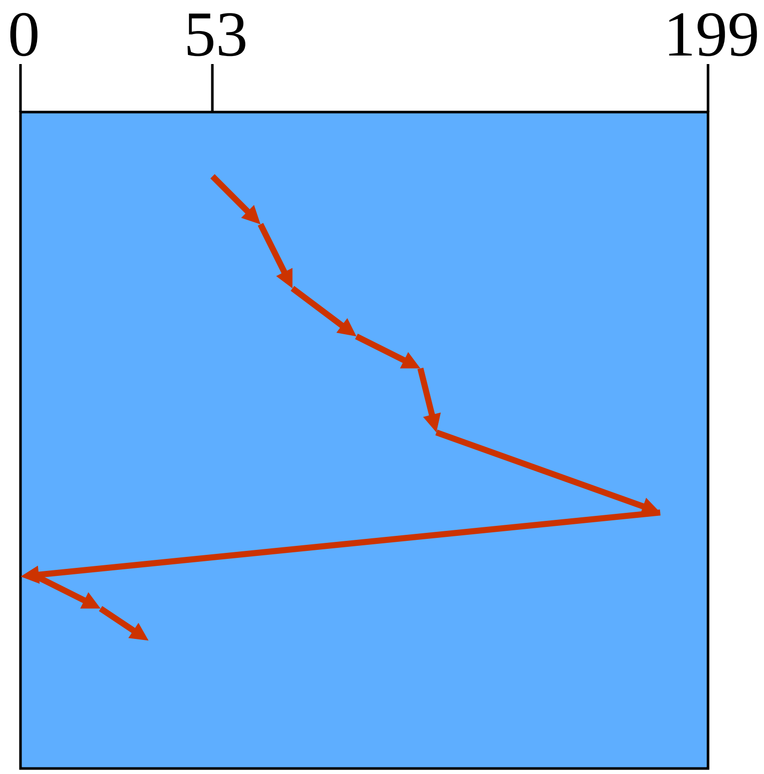
- Method
  - Take the closest request in the current direction of travel
  - Real implementations do not go to the end (called LOOK)
- Pros
  - Bounded time for each request
- Cons
  - Request at the other end will take a while



98, 183, 37, 122, 14, 124, 65, 67
(37, 14, 65, 67, 98, 122, 124, 183)

C-SCAN (Circular SCAN)

- Method
  - Like SCAN
  - But, wrap around
  - Real implementation doesn't go to the end (C-LOOK)
- Pros
  - Uniform service time
- Cons
  - Do nothing on the return



98, 183, 37, 122, 14, 124, 65, 67
(65, 67, 98, 122, 124, 183, 14, 37)

Disk Versus Memory

Memory

- Latency in 10’s of processor cycles
- Transfer rate 300+MB/s

Disk

- Latency in milliseconds (millions of processor cycles)
- Transfer rate 5-50MB/s

On-Disk Caching

- Method
  - Put RAM on disk controller to cache blocks
    - Seagate ATA disk has 0.5MB, IBM Ultra160 SCSI has 16MB
    - Some of the RAM space stores “firmware” (an OS)
  - Blocks are replaced usually in LRU order
- Pros
  - Good for reads if you have locality
- Cons
  - Expensive
  - Need to deal with reliable writes