Mass Storage Systems

Readings

- Chapter 12.1-12.4, 12.7

Long-term Information Storage

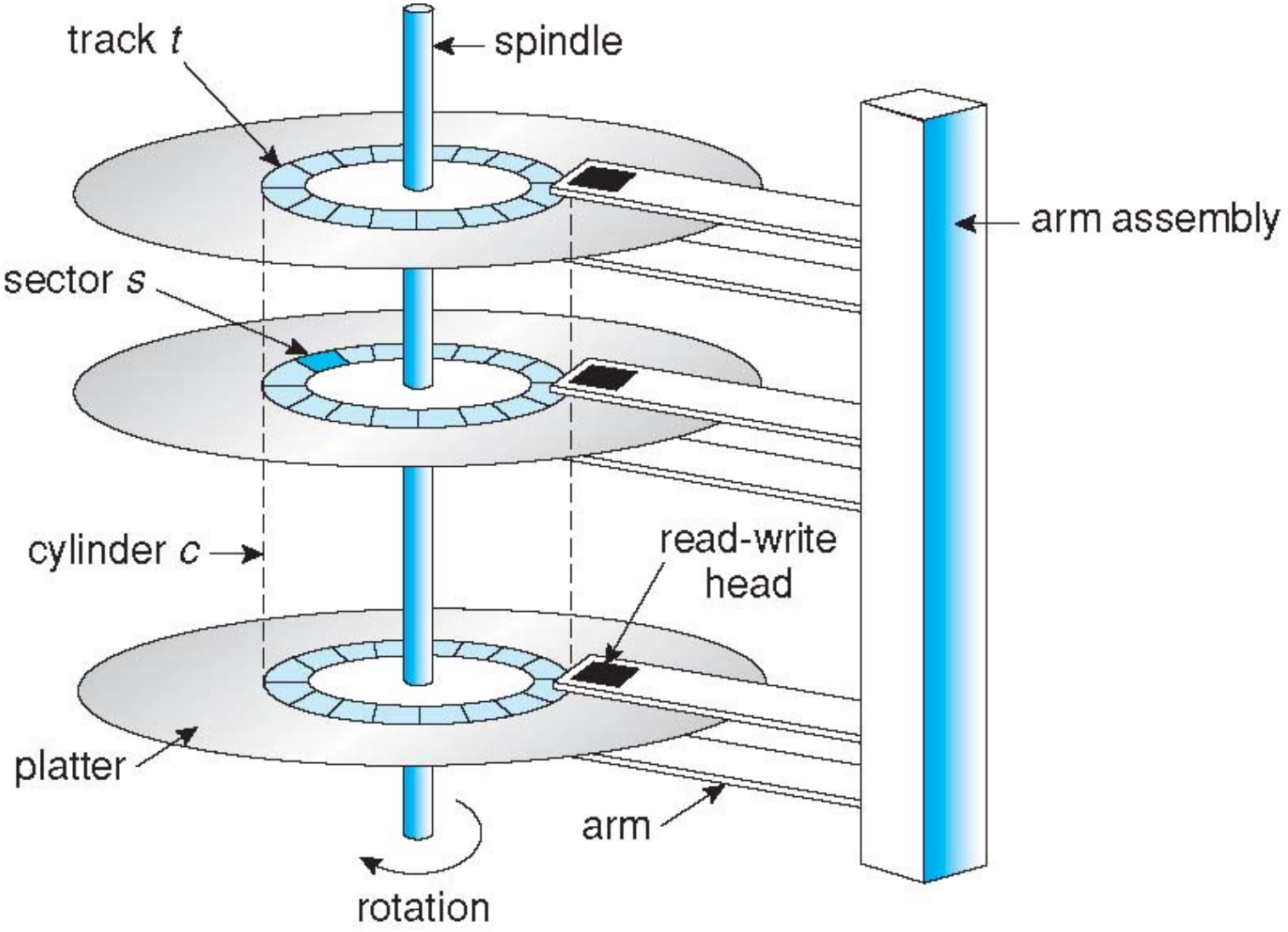
Three essential requirements:

- Must store large amounts of data
- Information stored must survive the termination of the process using it (persistence)
- Multiple processes must be able to access the information concurrently

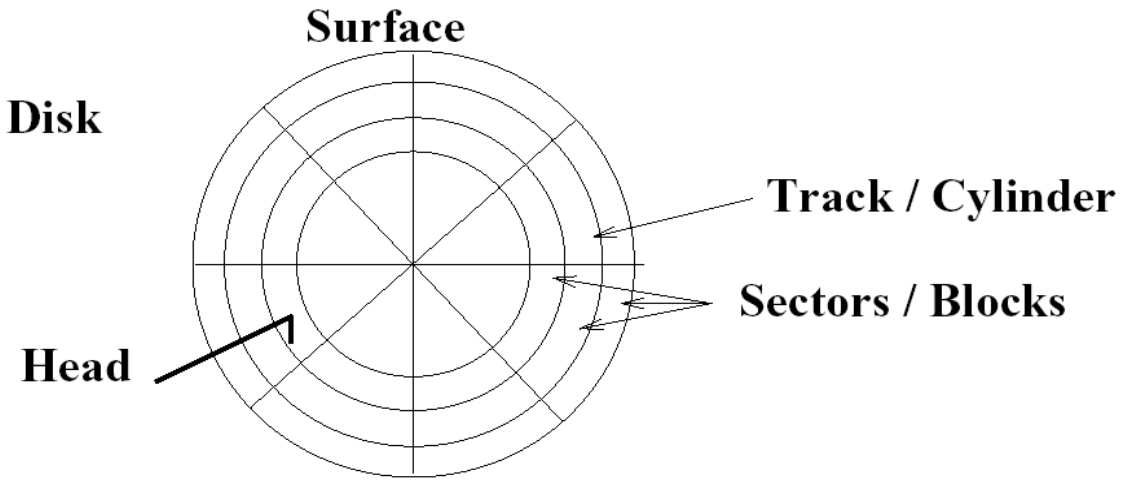
Examples of Mass Storage Srucutures

- Magnetic disks provide the bulk of secondary storage for modern computer systems (structure on next page)
- Magnetic tape was used as an early secondary-storage medium
  - Slow compared to magnetic disks and memory
  - Can hold large amounts of data
  - Read data sequentially
  - Mainly used for backup

Moving-head Disk Mechanism



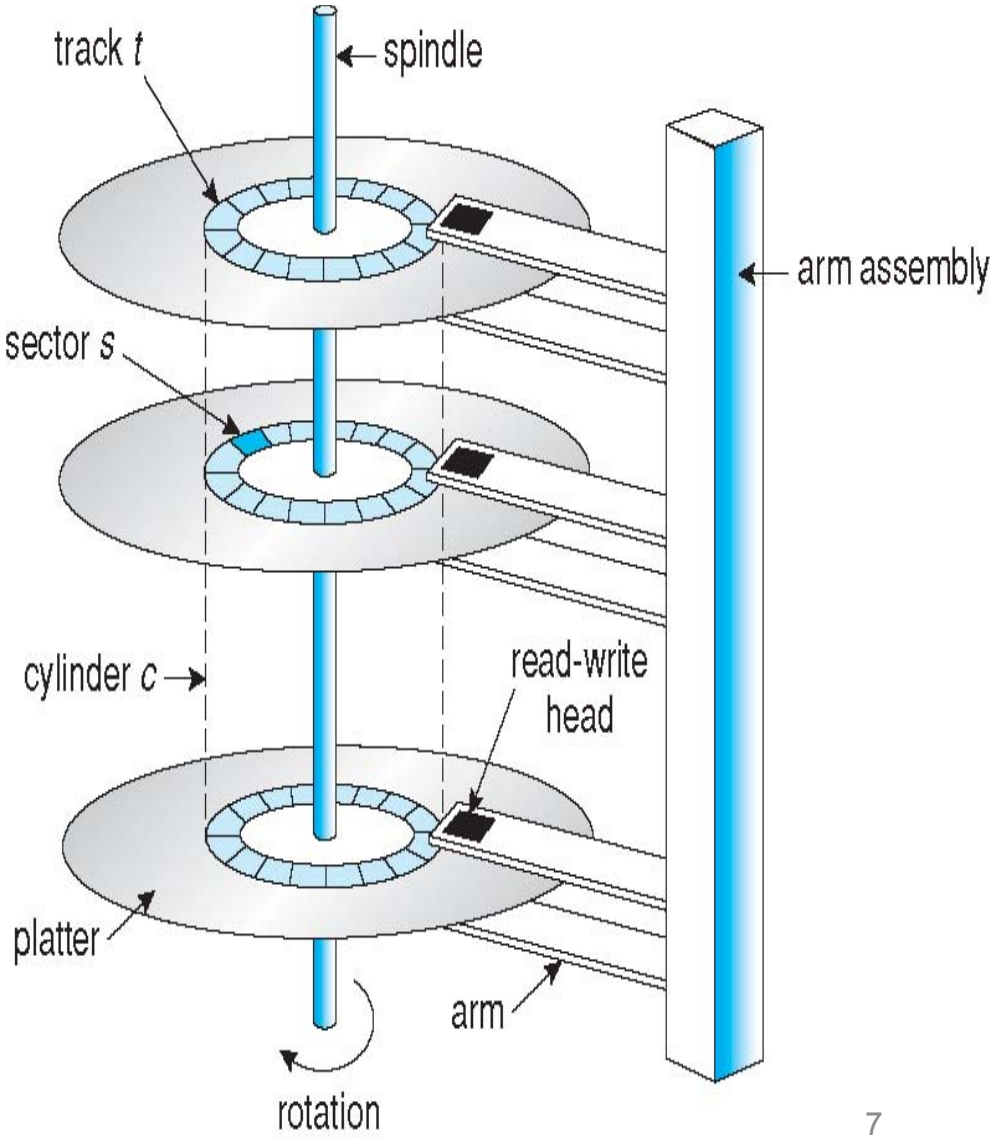
Disk Surface Layout



- Tracks: concentric rings on platter (see above)
  - bits laid out serially on tracks
- Tracks split into sectors
- Sectors may be grouped into blocks
- Addressable unit is typically a block

Disk Pack: Multiple Disks

- Think of disks as a stack of platters
- Use both sides of platters
- Two read-write heads at end of each arm
- Cylinders = matching sectors on each surface



Reading/Writing

- Position the read/write heads over the correct cylinder
- Rotate the disk platter until the desired sector is underneath the read/write head

Cost of Disk Operations

- Access Time: Composed of the following:
  - Seek time: The time to position head over correct cylinder
  - Rotational time: The time for correct sector to rotate under disk head
  - Transfer time: The time to transfer data
- Usually the seek time dominates
- Reducing seek time can improve system performance substantially
- Note: The transfer time for consecutive sectors within a track can be very fast

I/O Busses

- A disk drive is attached to a computer by a set of wires called an I/O bus.
- Types of busses available:
  - Enhanced integrated drive electrics (EDIE)
  - Advanced Technology Attachment (ATA)
  - Serial ATA (SATA)
  - Universal serial bus (USB)
  - Small computer-systems interface (SCI)

I/O Controllers

- The data transfers on a bus are carried out by special electronic processors called controllers
- The host controller is the controller at the computer end of the bus
- A disk controller is built into each disk drive

I/O Controllers

- To perform a disk I/O operation, the computer places a command into the host controller
- The host controller sends command to the disk controller
  - The command either requests a read or write to a block/sector
- The disk controller operates on the disk-drive hardware to carry out the command
- Disk controllers have caches
  - Can write to cache and then to disk; writer can return before write to disk

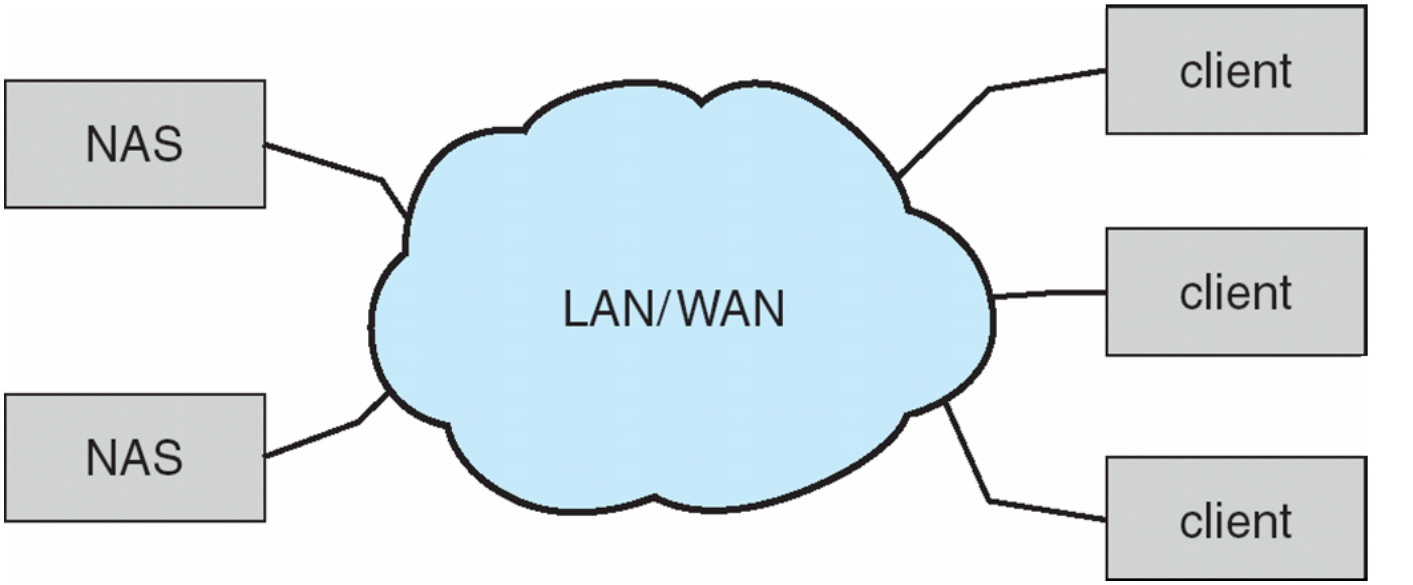
Disk Attachment

- The primary focus of the discussion has been on disks accessed through busses
- Other types of storage can be accessed remotely over a data network

Networked Attached Storage

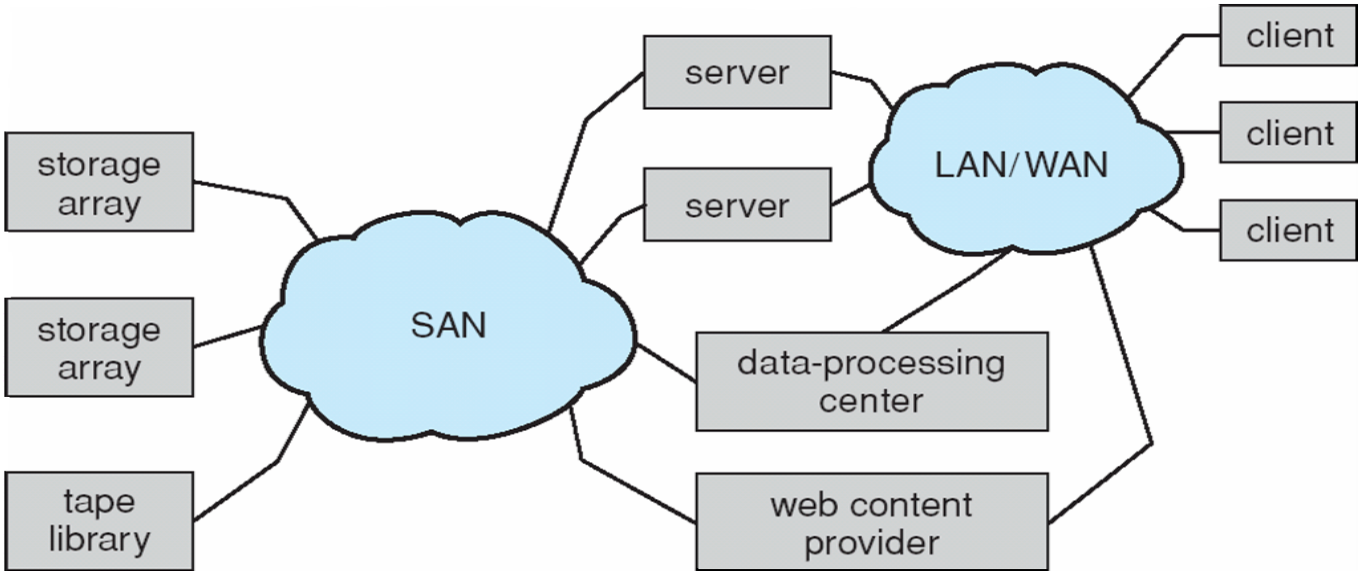
- Network-attached storage (NAS) is storage made available over a network rather than over a local connection (such as a bus)
- Often a set of disks (storage array) are placed together
- NFS and CIFS are common protocols
- Implemented via remote procedure calls (RPCs) between host and storage
- Recent iSCSI protocosl uses IP network to carry the SCSI protocol

Network-Attached Storage



Storage Area Network

- Common in large storage environments (and becoming more common)
- Multiple hosts attached to multiple storage arrays - flexible



Disk Scheduling

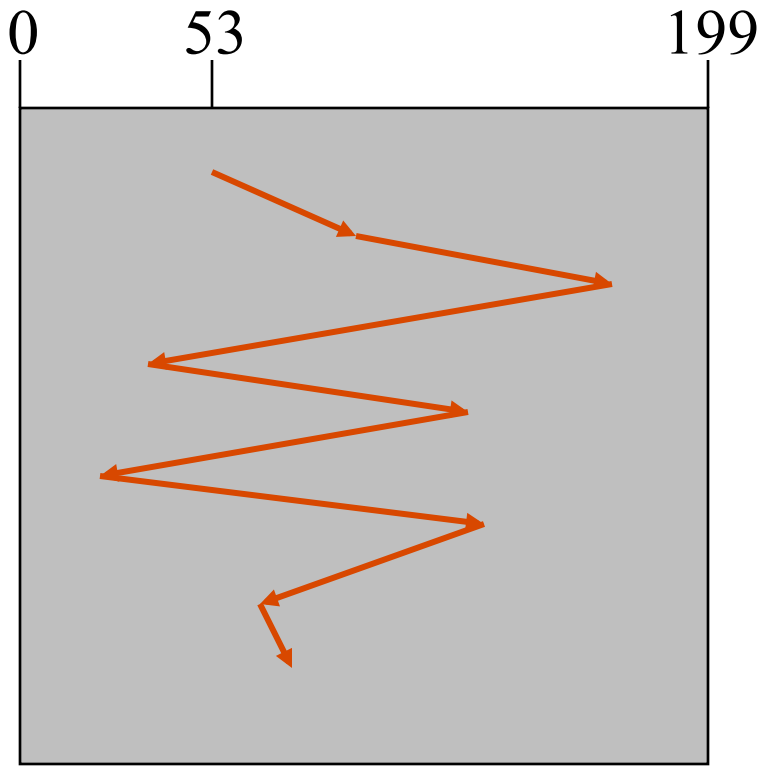
- The disk accepts requests
  - What sort of disk arm scheduling algorithm is needed?
- The access time depends on the order in which disk I/O requests are serviced
- I/O requests include information such as:
  - Is the operation input/output
  - Disk address
  - Memory address

Disk Scheduling: Data Structure

- Software for disks maintain a table called the pending request table
- Table is indexed by cylinder number
- All requests for each cylinder are chained together in a linked list headed by the table entries

First-Come, First-Served (FCFS) order

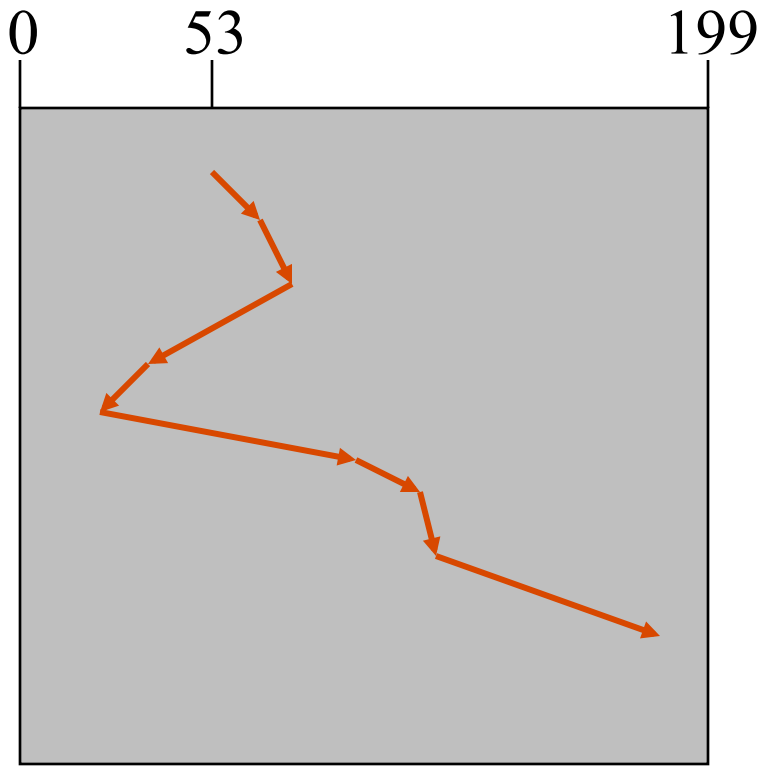
- Method
  - First come first serve
- Pros
  - Fairness among requests
  - In the order applications expect
- Cons
  - Arrival may be on random spots on the disk (long seeks)
  - Wild swings can happen



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

SSTF (Shortest Seek Time First)

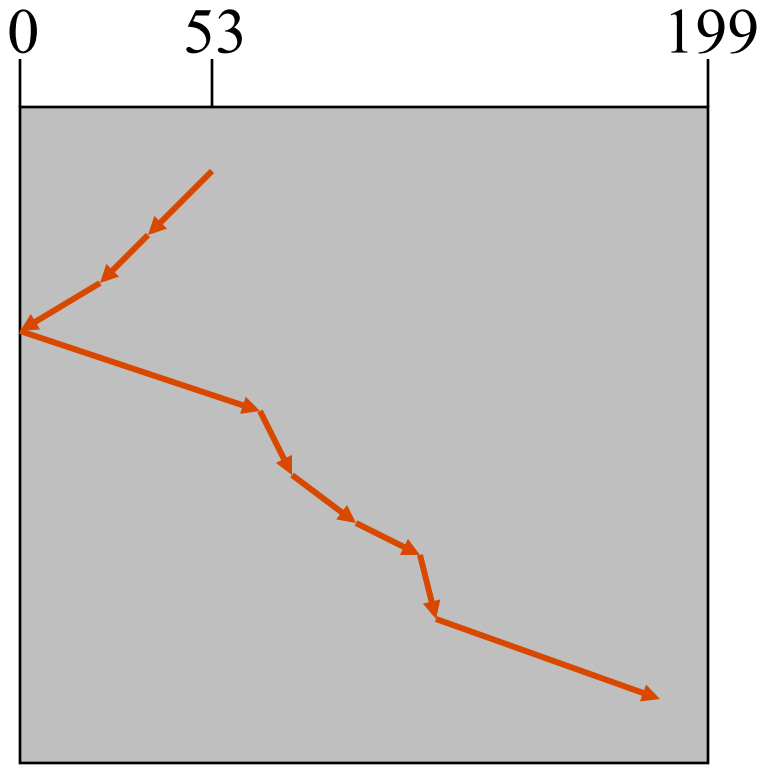
- Method
  - Pick the one closest on disk
  - Rotational delay is in calculation
- Pros
  - Try to minimize seek time
- Cons
  - Starvation



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 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

- Variant of SCAN
- Like SCAN, C-SCAN moves the head from one end of the disk to the other
- When the head reaches the other end, it immediately returns to the start of the disk without servicing requests on the return-trip

Optimization

- Some disk controllers provide a way for the software to inspect the current sector number under the read
- If there are two or more requests for the same cylinder that are pending:
  - The driver can issue a request for the sector that will pass under the head next
  - The information is known from the pending request table
- Caching is needed to hold the additional data

Mass Storage

- Many systems today need to store many large amounts of data
  - Can you say zettabytes?
- Don't want to use single, large disk
  - too expensive
  - failures could be catastrophic
- Would prefer to use many smaller disks

RAID Structure

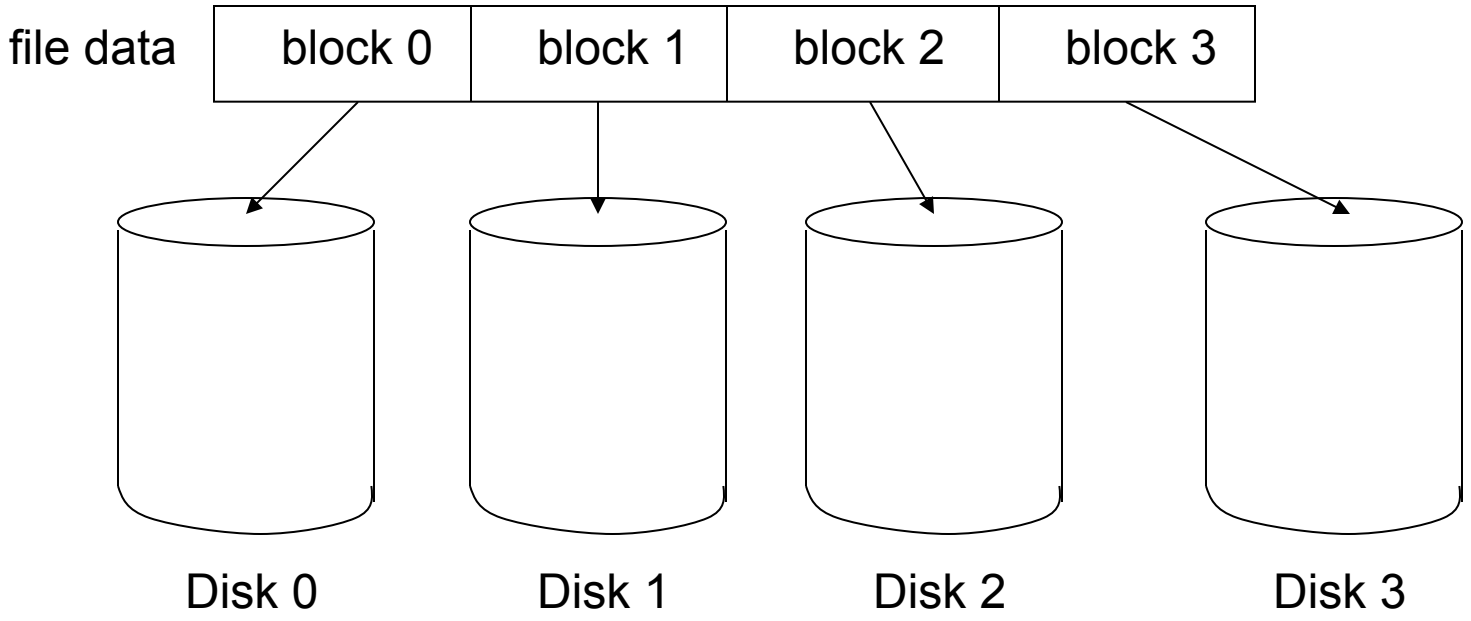
- Using multiple disks attached to a computer system has these benefits:
  - Improve the rate at which data can be read or written
  - Improve reliability of data storage
    - Redundant information can be stored on multiple disks
- RAID – multiple disk drives provides reliability via redundancy.

RAID Structure

- Improve performance via parallelism
  - Striping,mirroring
- Improve reliability via information redundancy
  - Error correcting codes

Striping

- Take file data and map it to different disks (interleaving)
- Allows for reading data in parallel



Striping

- Example
  - Assume you have 4 disks.
  - Bit interleaving means that bit N is on disk (N mod 4)
  - Byte interleaving means that byte N is on disk (N mod 4).
  - Block interleaving means that block N is on disk (N mod 4).
- All reads and writes involve all disks, which is great for large transfers

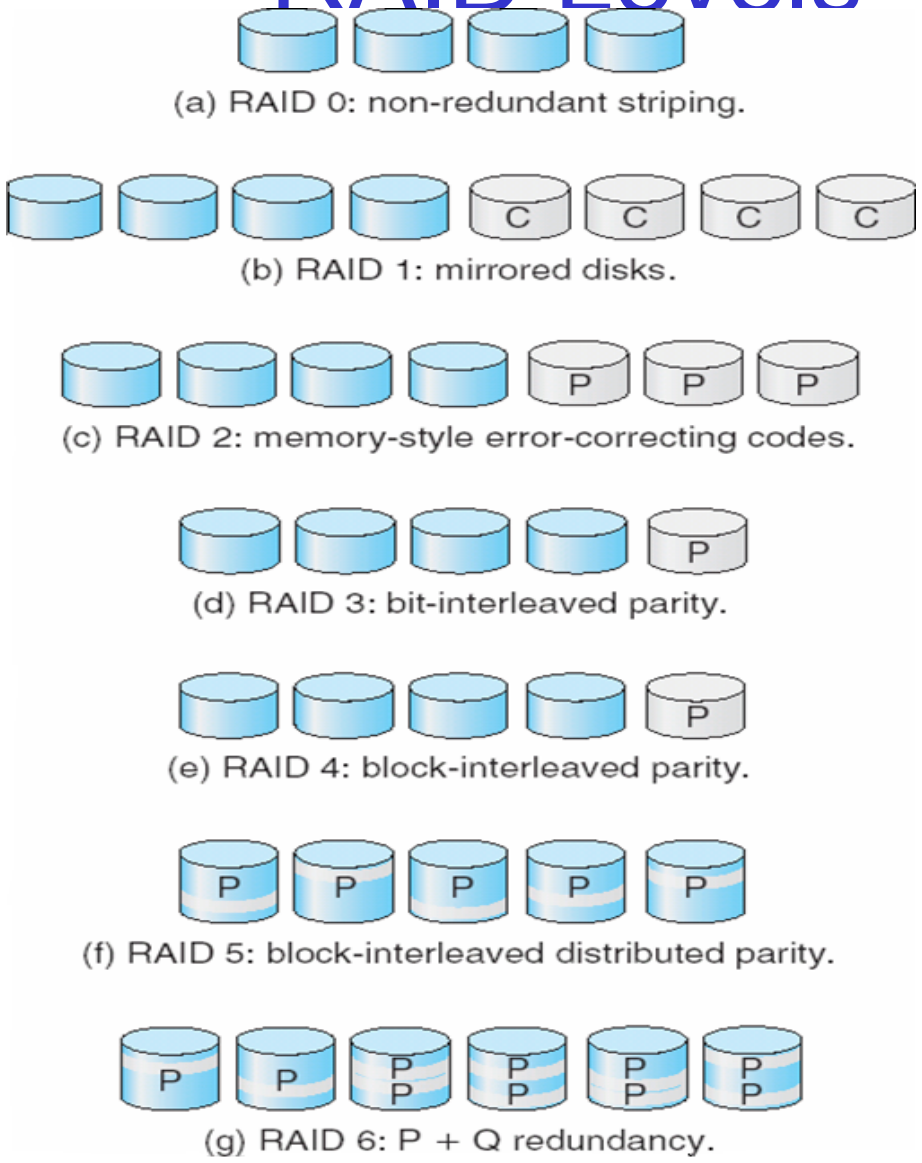
Mirroring

- Keep two copies of data on two separate disks
- Gives good error recovery
  - if some data is lost, get it from the other source
- Expensive
  - requires twice as many disks
- Write performance can be slow
  - have to write data to two different spots
- Read performance is enhanced
  - can read data from file in parallel

RAID Structure

- Numerous schemes to provide redundancy at low cost and high performance have been proposed
- These schemes are classified into RAID levels

RAID Levels



(a) RAID 0: non-redundant striping.

(b) RAID 1: mirrored disks.

(c) RAID 2: memory-style error-correcting codes.

(d) RAID 3: bit-interleaved parity.

(e) RAID 4: block-interleaved parity.

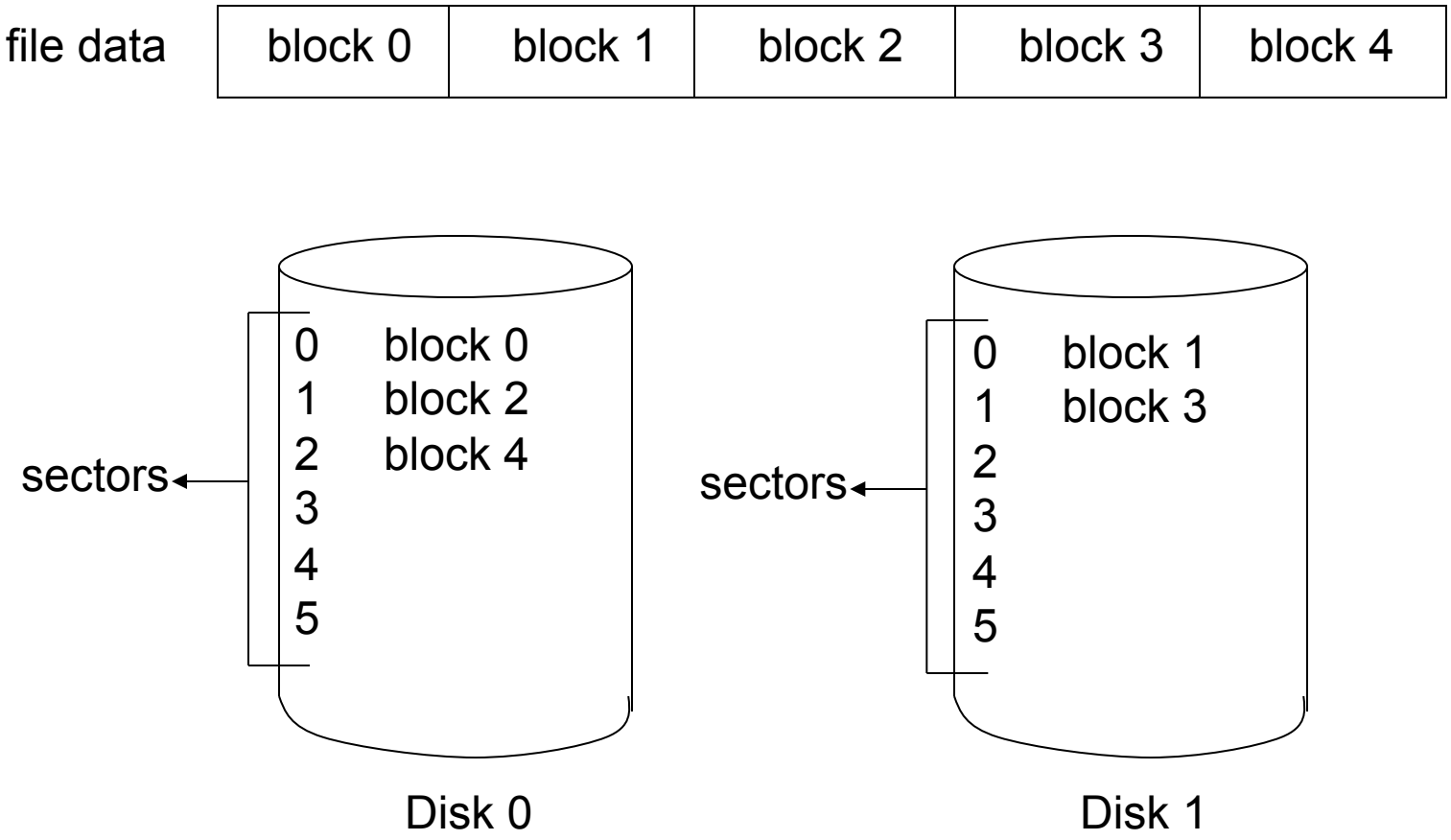
(f) RAID 5: block-interleaved distributed parity.

(g) RAID 6: P + Q redundancy.

RAID Level-0

- Break a file into blocks of data
- Stripe the blocks across disks in the system
- Provides no redundancy or error detection
- Uses
  - Some gaming systems where fast reads are required but minimal data integrity is needed

RAID Level-0


file data
block 0
block 1
block 2
block 3
block 4
sectors
0
1
2
3
4
5
block 0
block 2
block 4
Disk 0
sectors
0
1
2
3
4
5
block 1
block 3
Disk 1

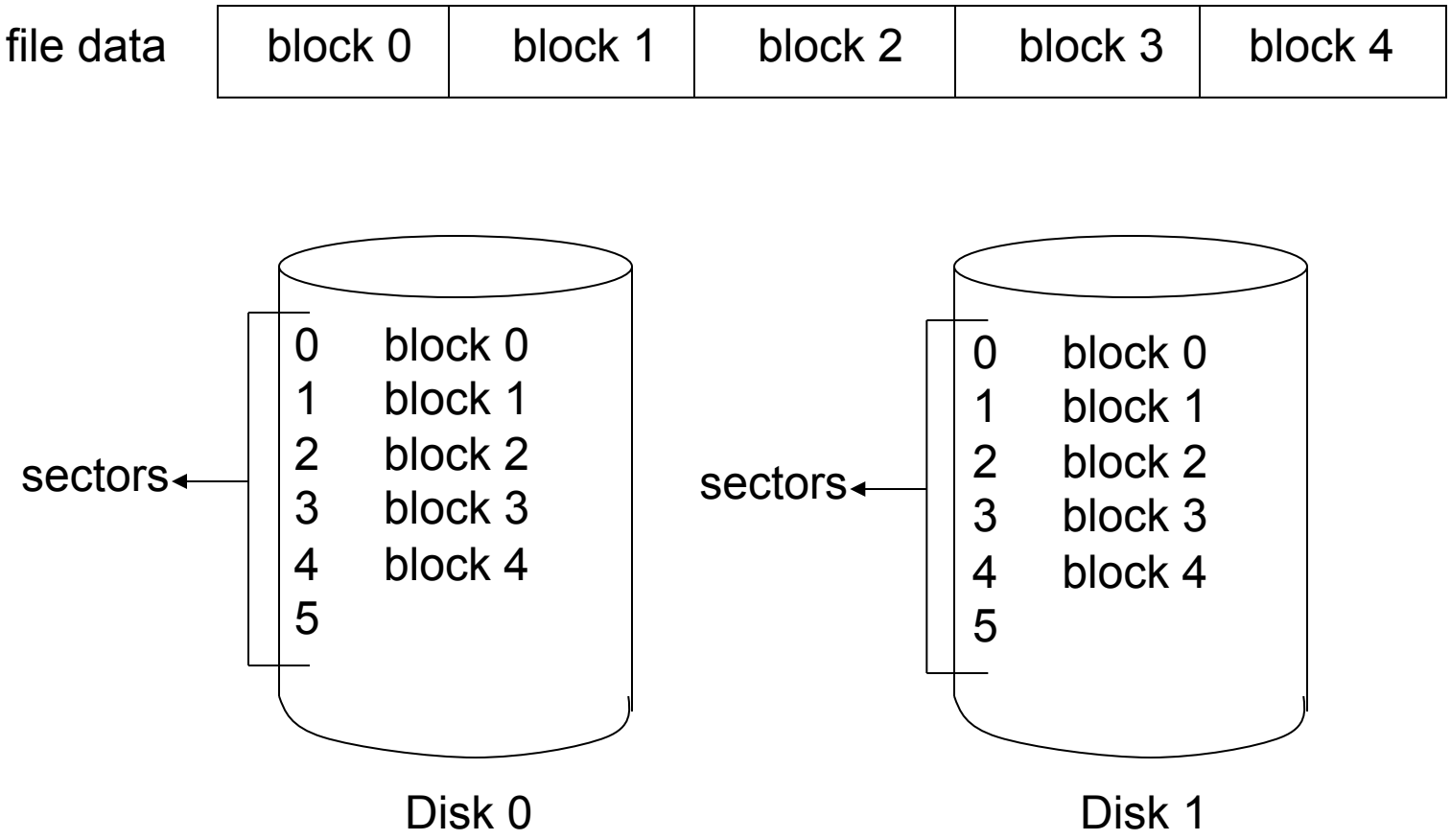
RAID Level 0

- No redundancy
  - No reliability
  - Loss of one disk means all is lost
- Can be very fast since data is being accessed in parallel

RAID Level-1

- A complete file is stored on a single disk
- A second disk contains an exact copy of the file
- Provides complete redundancy of data
- Lose one disk you are ok
- Write performance suffers
  - Must write the data out twice
- Most expensive RAID implementation
  - requires twice as much storage space

RAID Level-1


file data
block 0
block 1
block 2
block 3
block 4
sectors
0
1
2
3
4
5
block 0
block 1
block 2
block 3
block 4
Disk 0
sectors
0
1
2
3
4
5
block 0
block 1
block 2
block 3
block 4
Disk 1

RAID Level-1

- Read performance improves
- The redundancy is good but it does come up at a high cost
- Mission-critical missions use this level

RAID Level 2

- The basic idea is to add check bits to the information bits such that errors in some bits can be detected, and if possible corrected.
- The process of adding check bits to information bits is called encoding.
- The reverse process of extracting information from the encoded data is called decoding.

Raid Level 2

- Each block may have a parity bit associated with it
- The parity bit is selected in one of the following two ways:
  - Odd-Parity: The total number of 1's in the data is odd (indicating that the byte has an even number of ones)
  - Even Parity: The total number of 1's in the data is even (indicating that the byte has an even number of zeros)
- Use Striping
  - Parity bits may be on different disks

Raid Level 2

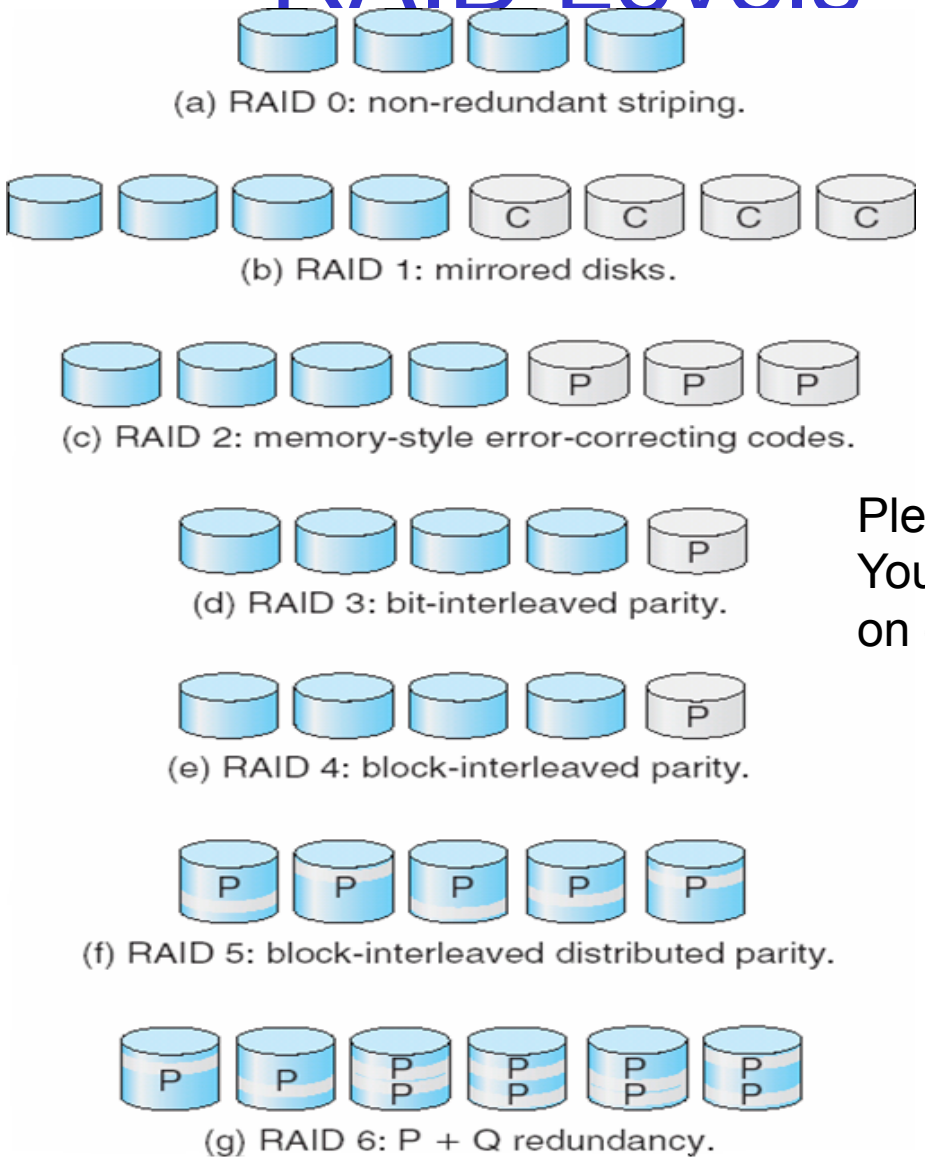
Simple Parity Bits

- Example (assume odd-parity):
  - Information is 000; the parity bit is 1
  - Information is 001; the parity bit is 0
  - Information is 010; the parity bit is 0
- Transformed code words
  - 0001 0010 0100
  - One bit errors can be detected

RAID Level 2

- Error Checking
  - When you write you compute the parity bit and it to the data you are writing
  - When you read you compute it again and see if it is the same parity as you wrote.
- Single parity bits are limited – can detect errors but not correct
  - Multiple parity bits are needed for correction

RAID Levels



Please note that for RAID 2 You would have different parities on different disks

RAID Level 5

- Data is striped so that each sequential block is on a different disk
- Parity is calculated per block
- Data and parity are spread over the disks
- The parity for a block is not on the same disk as the block
- This is the most commonly used RAID system

Summary

- We have examined the lowest level of the file system
  - Disk
  - RAID