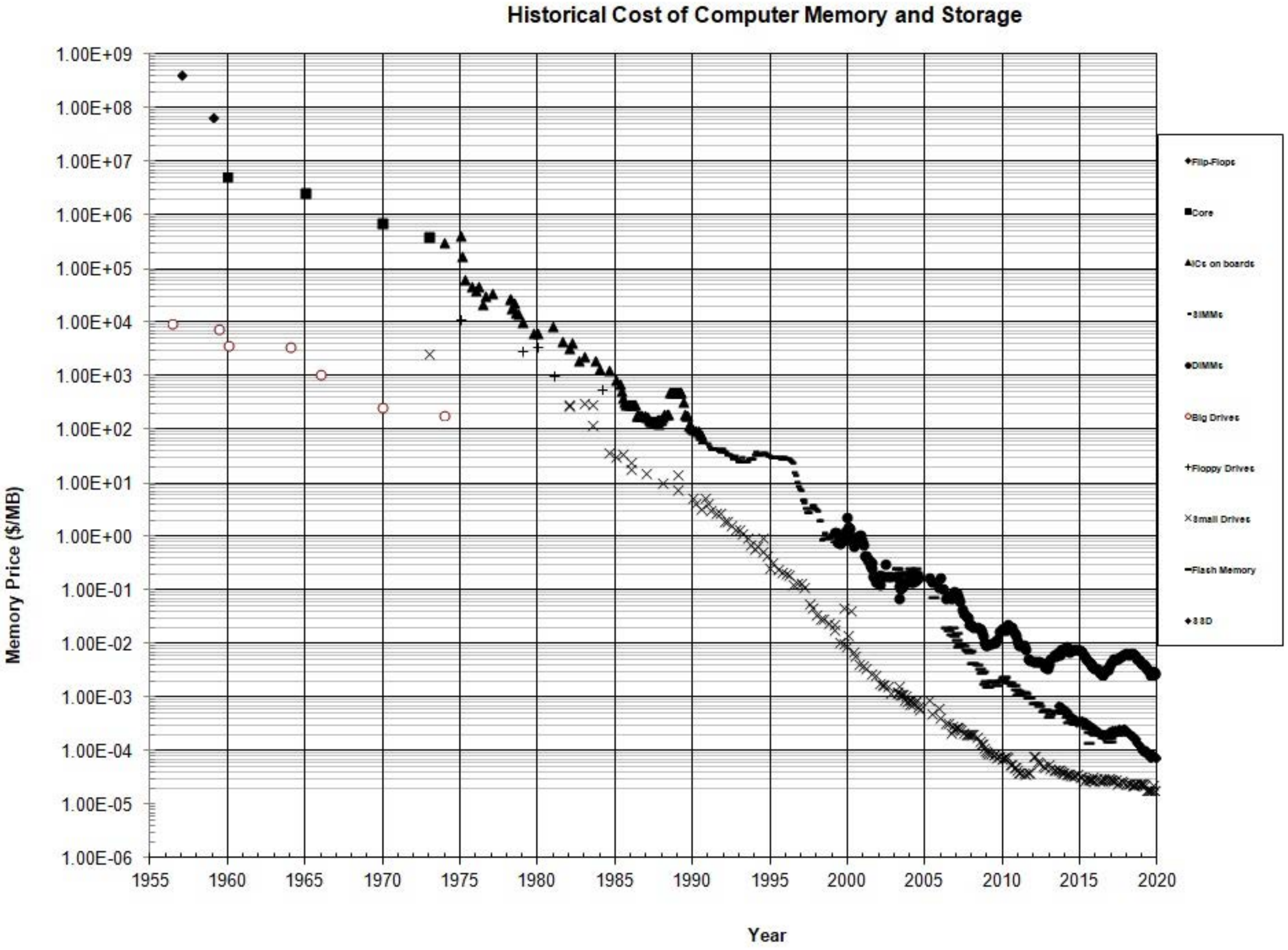
OPERATING SYSTEMS

MASS STORAGE

Memory Prices ($/MB)



https://jcmit.net/

Overview of Mass Storage Structure

- **Magnetic disks** provide bulk of secondary storage of modern computers
  - Drives **rotate** at 5200 to 15000 times per minute
  - **Transfer rate** is the rate at which data flow between drive and computer
  - **Positioning time** is the time to move the disk arm to the desired cylinder (seek time) plus the time for desired sector to rotate under the disk head (rotational latency)
- Disks can be removable
- Drive attached to computer via I/O bus
  - Buses vary, including EIDE, ATA, SATA, USB, Fibre Channel, SCSI, SAS, Firewire

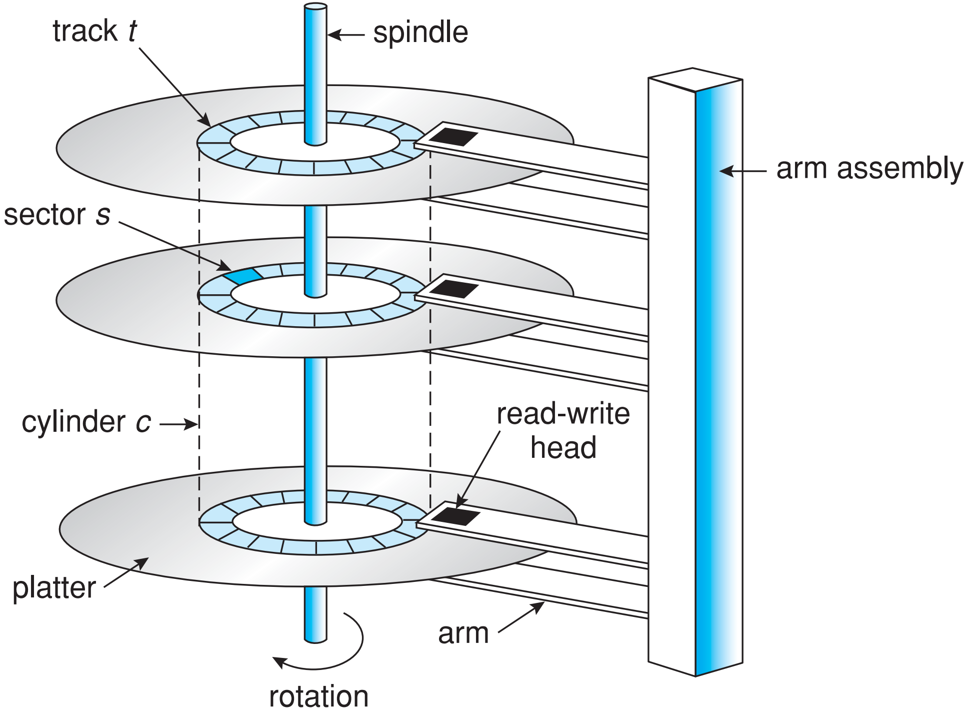
The First Commercial Disk Drive

1956
IBM RAMDAC computer
included the IBM Model 350
disk storage system

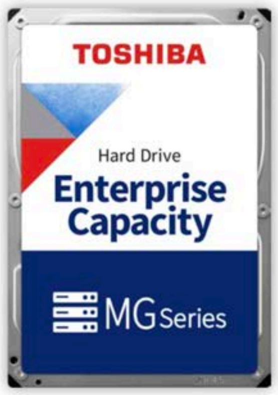
5M (7 bit) characters
50 x 24" platters
Access time = < 1 second

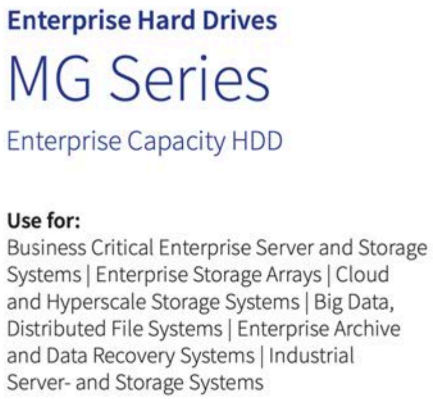
Mechanical Hard Disks

Moving-head Disk Mechanism



Example



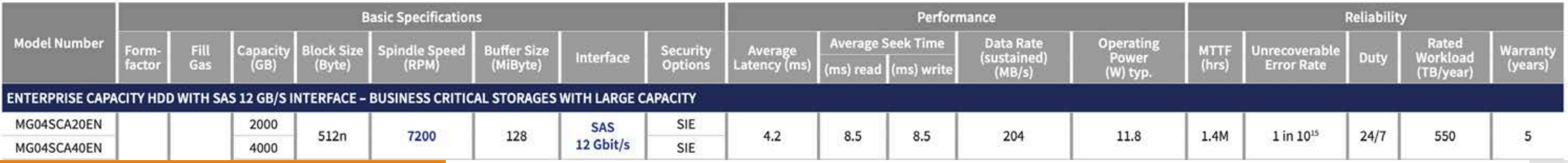


Enterprise Hard Drives

MG Series

Enterprise Capacity HDD

Use for:
Business Critical Enterprise Server and Storage Systems | Enterprise Storage Arrays | Cloud and Hyperscale Storage Systems | Big Data, Distributed File Systems | Enterprise Archive and Data Recovery Systems | Industrial Server- and Storage Systems



Model Number	Basic Specifications								Performance					Reliability				
	Form-factor	Fill Gas	Capacity (GB)	Block Size (Byte)	Spindle Speed (RPM)	Buffer Size (MiByte)	Interface	Security Options	Average Latency (ms)	Average Seek Time		Data Rate (sustained) (MB/s)	Operating Power (W) typ.	MTTF (hrs)	Unrecoverable Error Rate	Duty	Rated Workload (TB/year)	Warranty (years)
										(ms) read	(ms) write							
ENTERPRISE CAPACITY HDD WITH SAS 12 GB/S INTERFACE – BUSINESS CRITICAL STORAGES WITH LARGE CAPACITY																		
MG04SCA20EN			2000	512n	7200	128	SAS 12 Gbit/s	SIE	4.2	8.5	8.5	204	11.8	1.4M	1 in 10^{15}	24/7	550	5
MG04SCA40EN			4000					SIE										

Hard Disk Performance

- Access Latency = Average access time = average seek time + average latency
  - For fastest disk 3ms + 2ms = 5ms
  - For slow disk 9ms + 5.56ms = 14.56ms
- Average I/O time = average access time + (amount to transfer / transfer rate) + controller overhead
- For example to transfer a 4KB block on a 7200 RPM disk with a 5ms average seek time, 1Gb/sec transfer rate with a .1ms controller overhead =
  - 5ms + 4.17ms + 0.1ms + transfer time =
  - Transfer time = 4KB / 1Gb/s * 8Gb / GB * 1GB / 10242KB = 32 / (10242) = 0.031 ms
  - Average I/O time for 4KB block = 9.27ms + .031ms = 9.301ms

Solid-State Storage

- Nonvolatile memory used like a hard drive
  - Many technology variations
- No moving parts, so no seek time or rotational latency
  - Faster access than magnetic disks
- More expensive per MB
- Less capacity
- Maybe have shorter life span

Magnetic Tape

- Was early secondary-storage medium
  - Evolved from open spools to cartridges
- Relatively permanent and holds large quantities of data
- Access time slow
- Random access ~1000 times slower than disk
- Mainly used for backup, storage of infrequently-used data, transfer medium between systems
- Kept in spool and wound or rewound past read-write head
- Once data under head, transfer rates comparable to disk
  - 140MB/sec and greater
- 200GB to 1.5TB typical storage
- Common technologies are LTO-{3,4,5} and T10000

Disks

Disk Structure

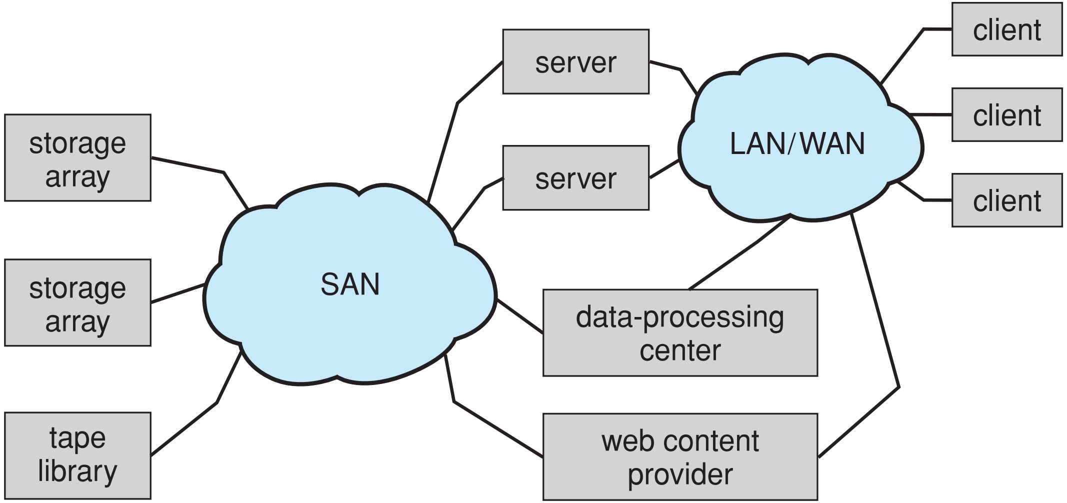
- Disk drives are **addressed as large 1-dimensional arrays of logical blocks**
- The logical block is the smallest unit of transfer
  - low-level formatting creates logical blocks on physical media
- The 1-dimensional array of logical blocks is mapped into the sectors of the disk sequentially
  - Sector 0 is the first sector of the first track on the outermost cylinder
  - Mapping proceeds in order through that track, then the rest of the tracks in that cylinder, and then through the rest of the cylinders from outermost to innermost
  - Logical to physical address should be easy
    - Except for bad sectors
    - Non-constant # of sectors per track via constant angular velocity

Storage Array

- Can just attach disks, or arrays of disks
- Storage Array has controller(s), provides features to attached host(s)
  - Ports to connect hosts to array
  - Memory, controlling software (sometimes NVRAM, etc)
  - A few to thousands of disks
  - RAID, hot spares, hot swap (discussed later)
  - Shared storage -> more efficiency
  - Features found in some file systems
    - Snaphots, clones, thin provisioning, replication, deduplication, etc

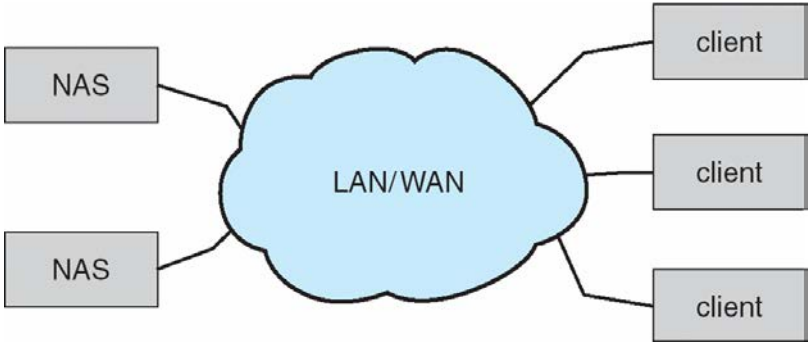
Storage Area Network (SAN)

- Common in large storage environments
- Multiple hosts attached to multiple storage arrays - flexible



Network-Attached Storage (NAS)

- Storage made available over a network rather than over a local connection (such as a bus)
  - Remotely attaching to file systems
- NFS and CIFS are common protocols
- Implemented via remote procedure calls (RPCs) between host and storage over typically TCP or UDP on IP network
- iSCSI protocol uses IP network to carry the SCSI protocol



Disk Scheduling

- The operating system is responsible for using hardware efficiently — for the disk drives, this means having a fast access time and disk bandwidth
- Minimize seek time
- Seek time ≈ seek distance
- Disk bandwidth is the total number of bytes transferred, divided by the total time between the first request for service and the completion of the last transfer

Disk Scheduling

- There are many sources of disk I/O request
  - OS
  - System processes
  - Users processes
- I/O request includes input or output mode, disk address, memory address, number of sectors to transfer
- OS maintains queue of requests, per disk or device
- Idle disk can immediately work on I/O request, busy disk means work must queue
  - Optimization algorithms only make sense when a queue exists

Disk Scheduling

- Note that drive controllers have small buffers and can manage a queue of I/O requests
- **Several algorithms** exist to schedule the servicing of disk I/O requests
- We illustrate scheduling algorithms with a request queue (0-199)

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

Head pointer 53

FCFS

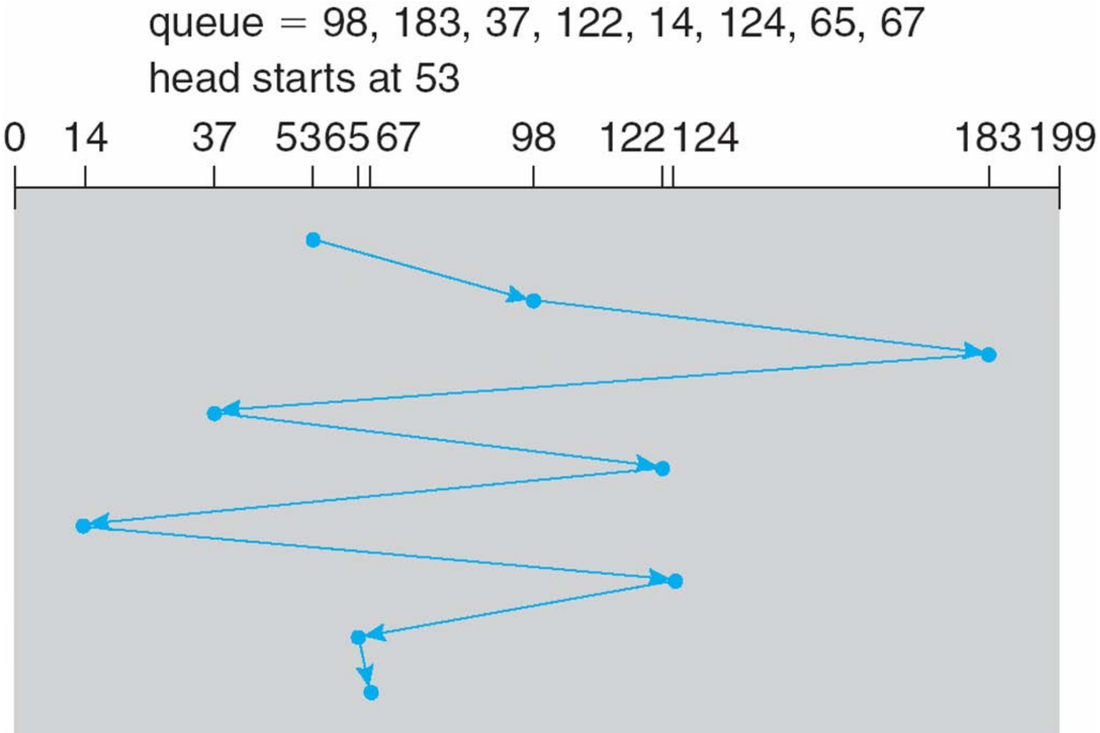
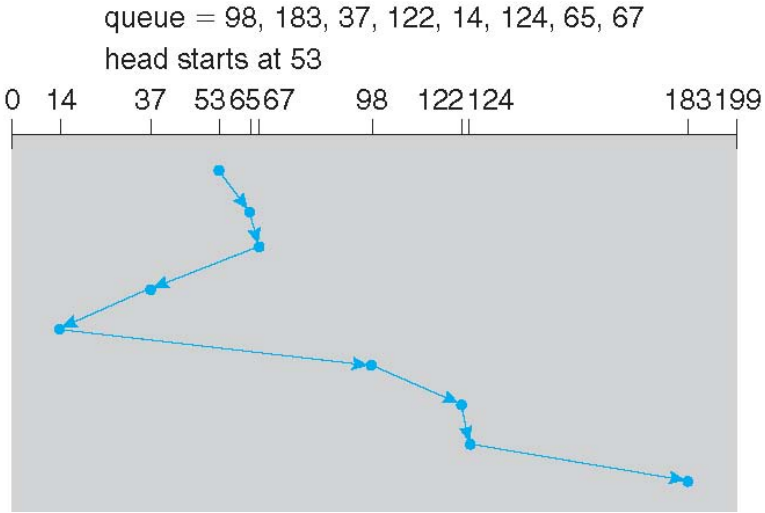


Illustration shows total head movement of 640 cylinders

SSTF

- **Shortest Seek Time First** selects the request with the minimum seek time from the current head position
- SSTF scheduling is a form of SPN scheduling; may cause starvation of some requests


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

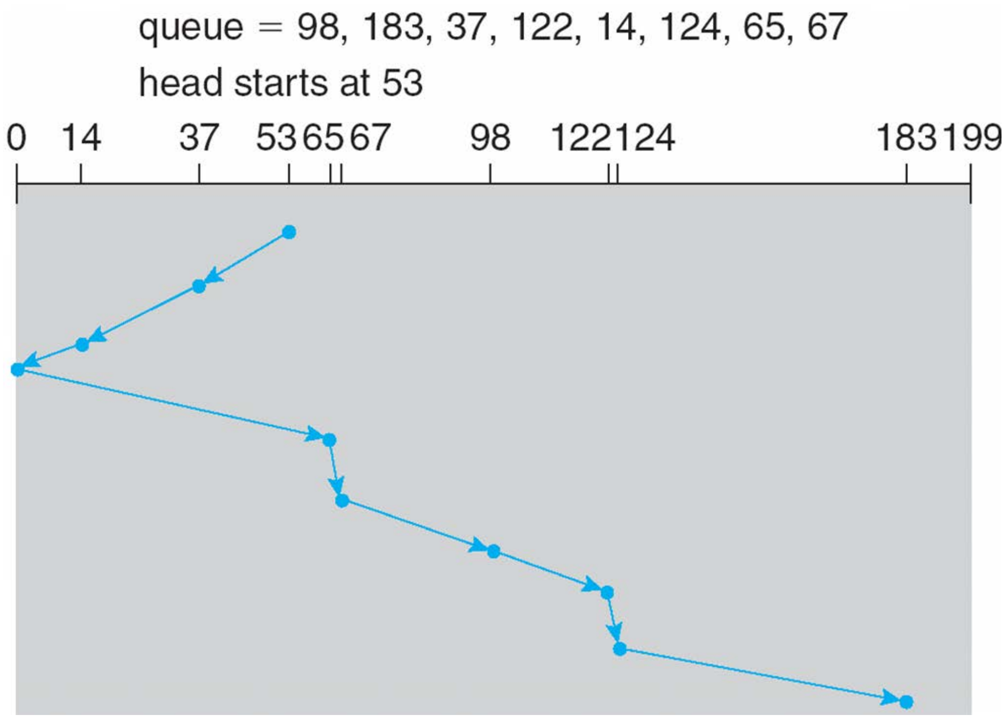
head starts at 53

- Illustration shows total head movement of 236 cylinders

SCAN

- The disk arm starts at one end of the disk, and moves toward the other end, servicing requests until it gets to the other end of the disk, where the head movement is reversed and servicing continues.
- Sometimes called *the elevator algorithm*
- Note that if requests are uniformly dense, largest density at other end of disk and those wait the longest

SCAN



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

head starts at 53

Illustration shows total head movement of 236 cylinders

C-SCAN

- Provides a more uniform wait time than SCAN
- The head moves from one end of the disk to the other, servicing requests as it goes
  - When it reaches the other end, however, it immediately returns to the beginning of the disk, without servicing any requests on the return trip
- Treats the cylinders as a circular list that wraps around from the last cylinder to the first one

C-SCAN

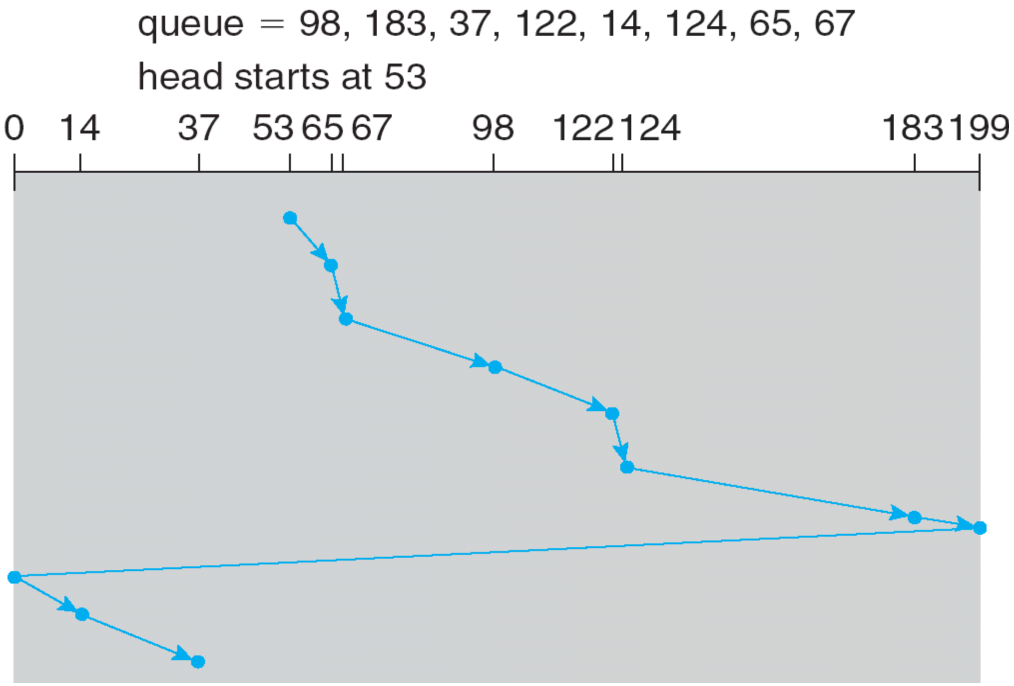


Illustration shows total head movement of 183 cylinders

C-LOOK

- LOOK a version of SCAN, C-LOOK a version of C-SCAN
- Arm only goes as far as the last request in each direction, then reverses direction immediately, without first going all the way to the end of the disk

C-LOOK

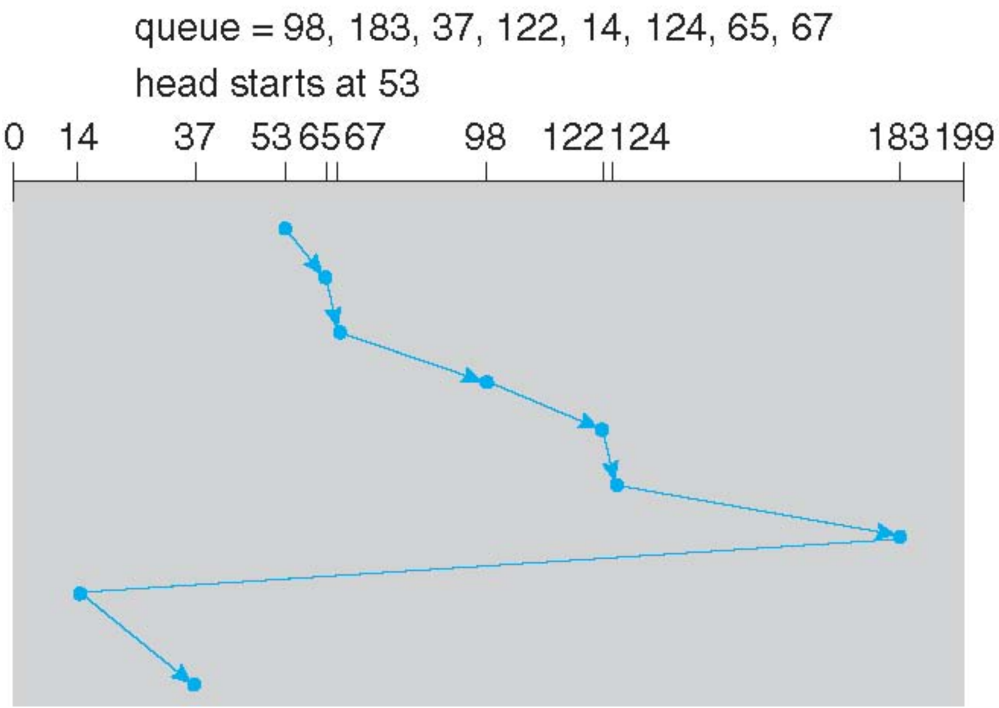


Illustration shows total head movement of 153 cylinders

Selecting a Disk-Scheduling Algorithm

- SSTF is common and has a natural appeal
- SCAN and C-SCAN perform better for systems that place a heavy load on the disk
  - Less starvation
- Performance depends on the number and types of requests
- Requests for disk service can be influenced by the file-allocation method
- The disk-scheduling algorithm should be written as a separate module of the operating system, allowing it to be replaced with a different algorithm if necessary
- Either SSTF or LOOK is a reasonable choice for the default algorithm

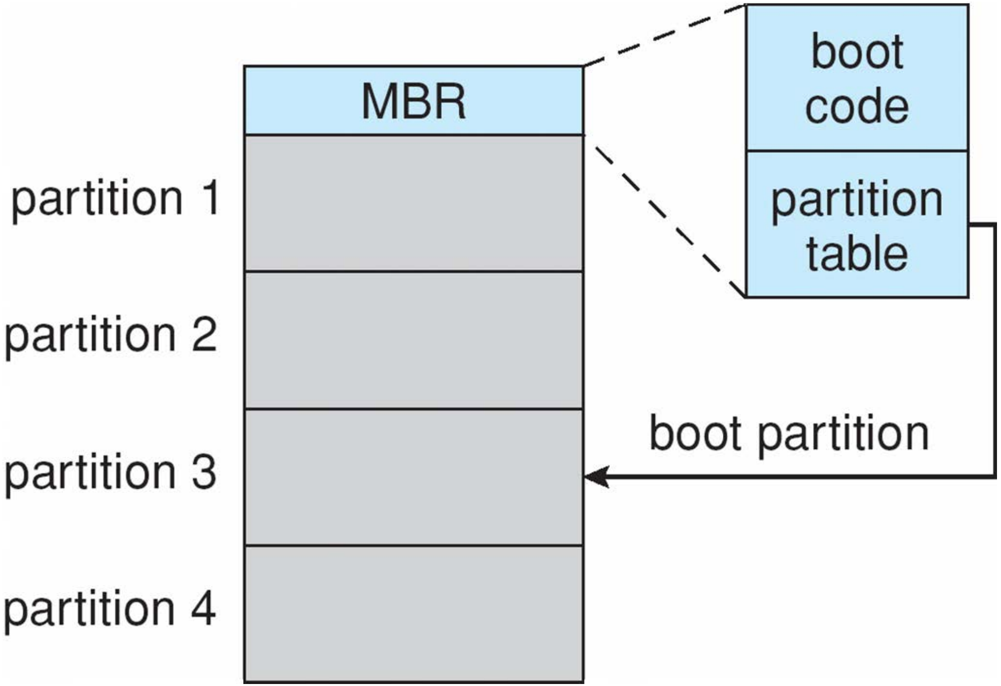
Disk Management

- Low-level formatting, or physical formatting
  Dividing a disk into sectors that the disk controller can read and write
  - Each sector can hold header information, plus data, plus error correction code (ECC)
  - Usually 512 bytes of data but can be selectable
- To use a disk to hold files, the operating system still needs to record its own data structures on the disk
  - Partition the disk into one or more groups of cylinders, each treated as a logical disk
  - Logical formatting or *making a file system*

Disk Management

- Raw disk access for apps that want to do their own block management, keep OS out of the way (databases for example)
- Boot block initializes system
  - The bootstrap is stored in ROM
  - Bootstrap loader program stored in boot blocks of boot partition
- Methods such as sector sparing used to handle bad blocks

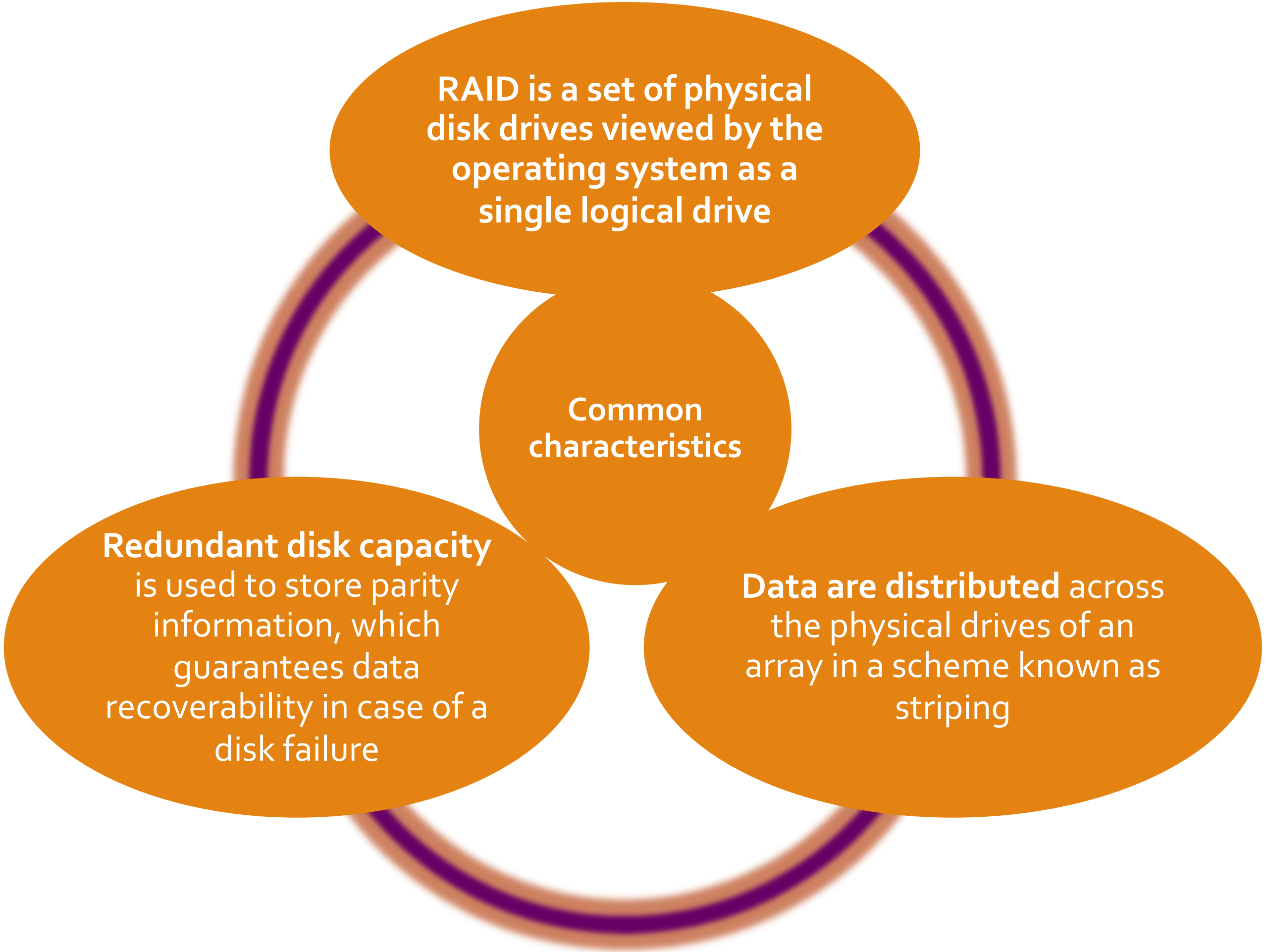
Booting from a Disk in Windows



RAID

Redundant Arrays of Independent Disks

RAID



Common characteristics

RAID is a set of physical disk drives viewed by the operating system as a single logical drive

Redundant disk capacity is used to store parity information, which guarantees data recoverability in case of a disk failure

Data are distributed across the physical drives of an array in a scheme known as striping

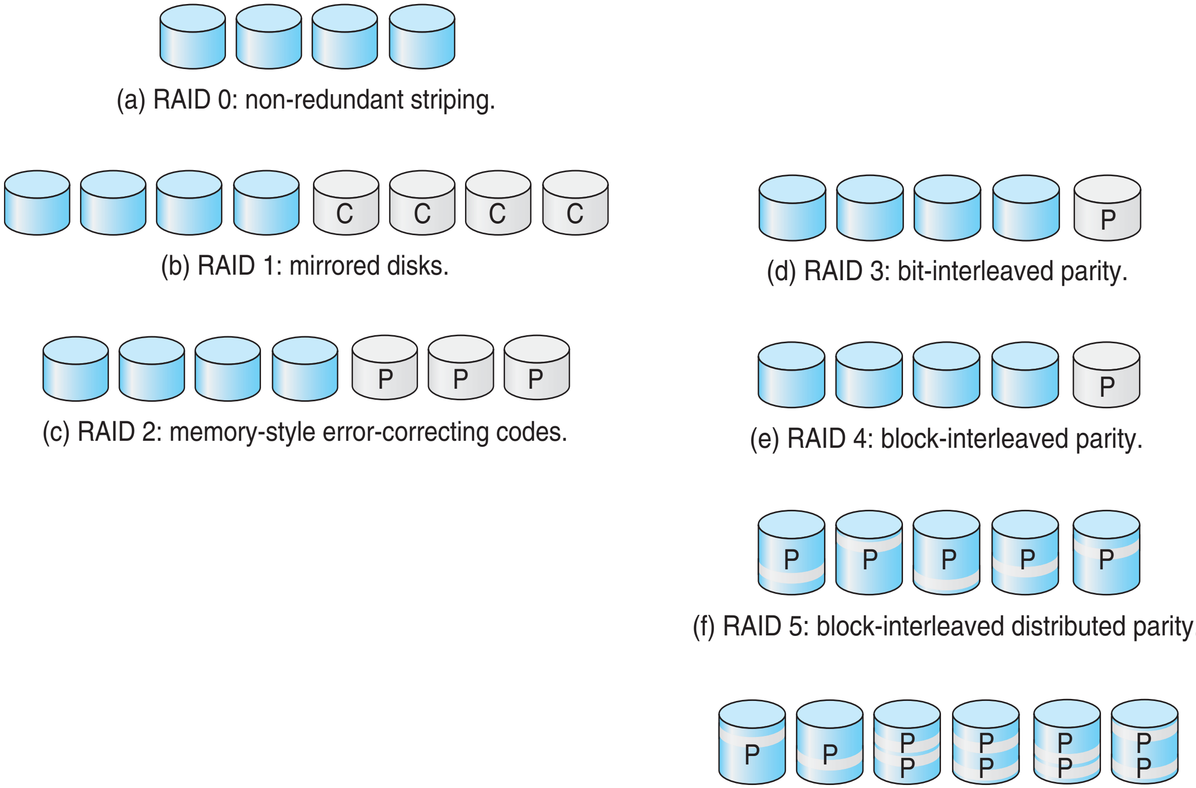
RAID

- Multiple disk drives provides reliability via redundancy
  - Increases the mean time to failure
- The term was originally coined in a paper by a group of researchers at the University of California at Berkeley
  - The paper outlined various configurations and applications and introduced the definitions of the RAID levels
- Multiple disk drives and data distributed in such a way as to enable simultaneous access to data from multiple drives
  - Improves I/O performance and allows easier incremental increases in capacity
- Makes use of stored parity information that enables the recovery of data lost due to a disk failure

RAID

- RAID is arranged into six different levels
- RAID schemes improve performance and improve the reliability of the storage system by storing redundant data
  - Mirroring or shadowing (RAID 1) keeps duplicate of each disk
  - Striped mirrors (RAID 1+0) or mirrored stripes (RAID 0+1) provides high performance and high reliability
  - Block interleaved parity (RAID 4, 5, 6) uses much less redundancy
- RAID within a storage array can still fail if the array fails, so automatic replication of the data between arrays is common
- Frequently, a small number of hot-spare disks are left unallocated, automatically replacing a failed disk and having data rebuilt onto them

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.

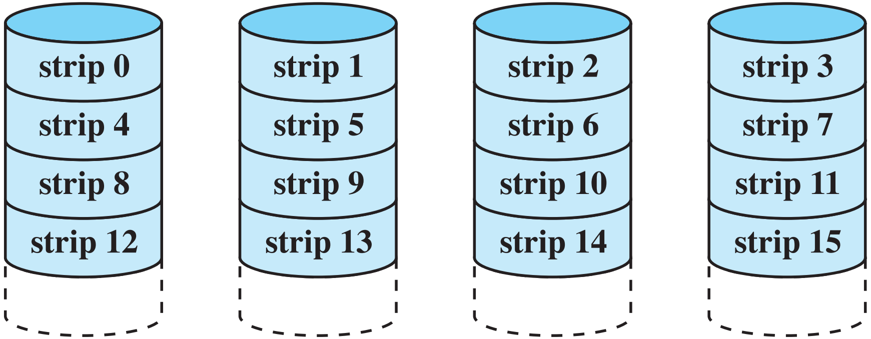
(e) RAID 4: block-interleaved parity.

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

(g) RAID 6: P + Q redundancy.

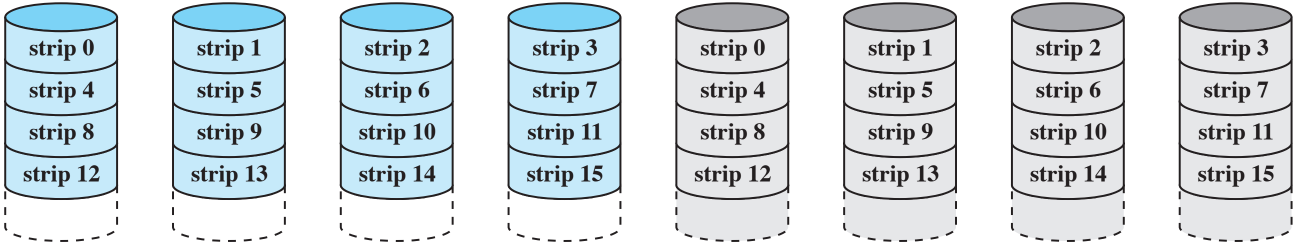
RAID Level 0

- Not a true RAID
  it does not include redundancy to improve performance or provide data protection
- User and system data are distributed across all of the disks in the array
- Logical disk is divided into strips

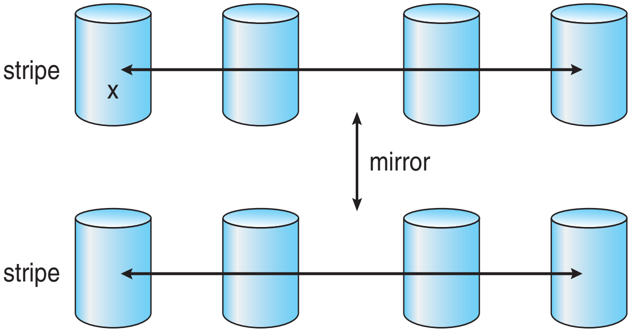


RAID Level 1

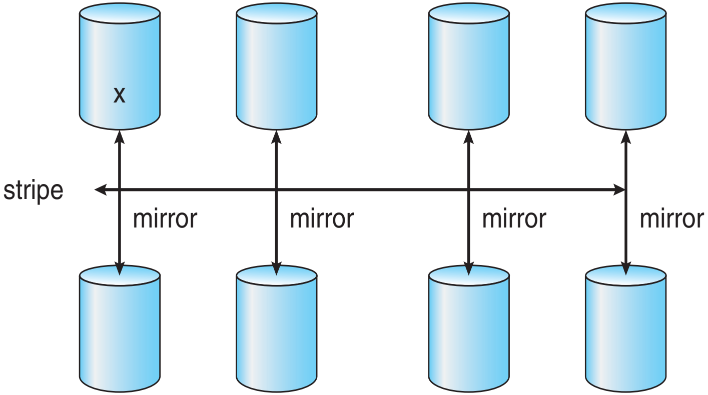
- Redundancy is achieved by the simple expedient of duplicating all the data
- There is no “write penalty”
- When a drive fails the data may still be accessed from the second drive
- Principal disadvantage is the cost



RAID 01 and 10



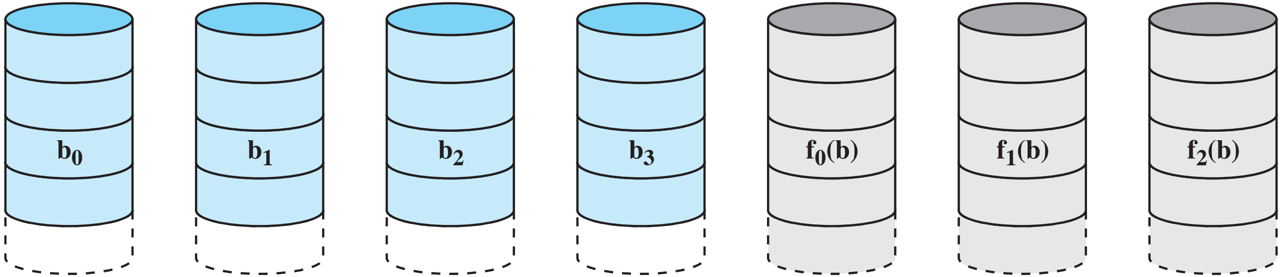
a) RAID 0 + 1 with a single disk failure.



b) RAID 1 + 0 with a single disk failure.

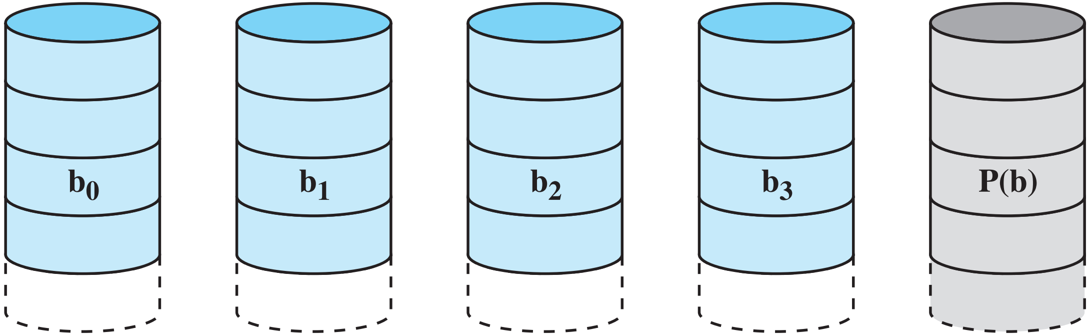
RAID Level 2

- Parallel access technique
- Data striping
- Hamming code is used
- Effective choice only for environments in which many disk errors occur



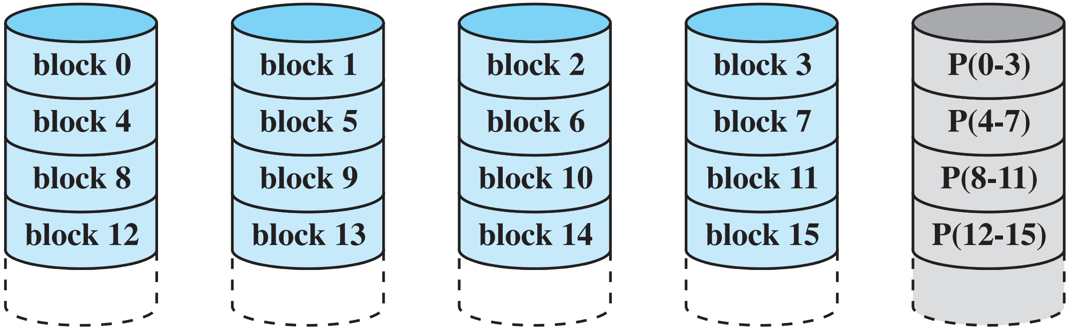
RAID Level 3

- Requires only a single redundant disk, no matter how large the disk array
- Employs parallel access, with data distributed in small strips
  - Slow writes, as they have to queue on the parity disk
- Can achieve very high data transfer rates



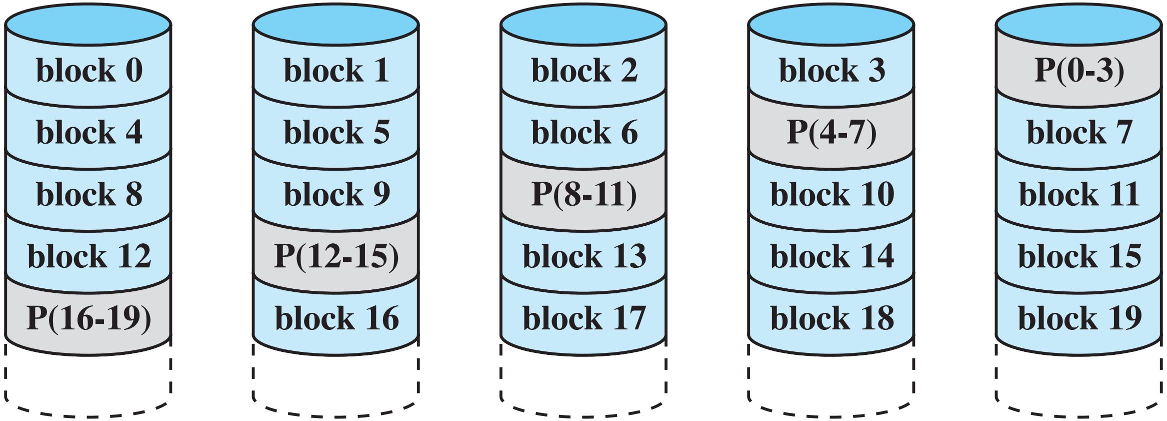
RAID Level 4

- Makes use of an independent access technique
- A bit-by-bit parity strip is calculated across corresponding strips on each data disk, and the parity bits are stored in the corresponding strip on the parity disk
- Involves a write penalty when an I/O write request of small size is performed



RAID Level 5

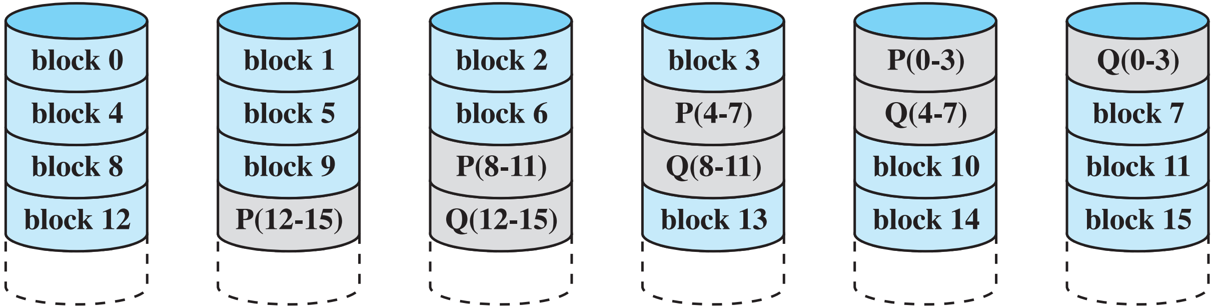
- Similar to RAID-4 but distributes the parity bits across all disks
- Typical allocation is a round-robin scheme
- Has the characteristic that the loss of any one disk does not result in data loss



block 0	block 1	block 2	block 3	P(0-3)
block 4	block 5	block 6	P(4-7)	block 7
block 8	block 9	P(8-11)	block 10	block 11
block 12	P(12-15)	block 13	block 14	block 15
P(16-19)	block 16	block 17	block 18	block 19

RAID Level 6

- Two different parity calculations are carried out and stored in separate blocks on different disks
- Provides extremely high data availability
- Each write affects two parity blocks



Overview of RAID Levels

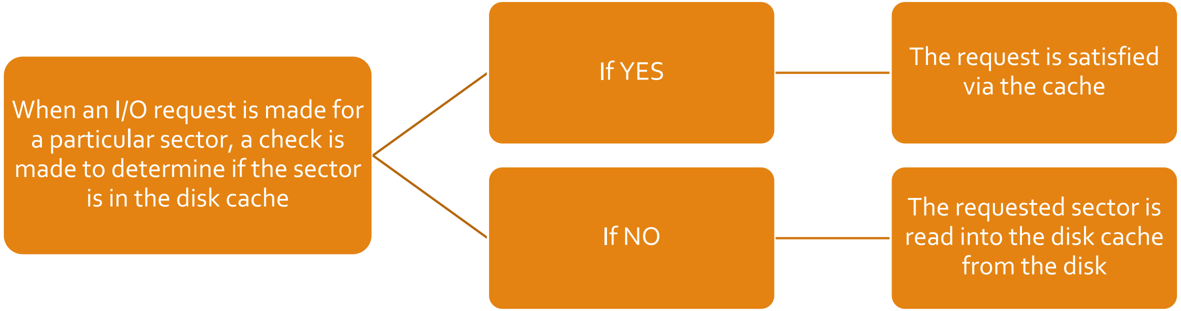
Category	Level	Description	Disks required	Data availability	Large I/O data transfer capacity	Small I/O request rate
Striping	0	Nonredundant	N	Lower than single disk	Very high	Very high for both read and write
Mirroring	1	Mirrored	2N	Higher than RAID 2, 3, 4, or 5; lower than RAID 6	Higher than single disk for read; similar to single disk for write	Up to twice that of a single disk for read; similar to single disk for write
Parallel access	2	Redundant via Hamming code	N + m	Much higher than single disk; comparable to RAID 3, 4, or 5	Highest of all listed alternatives	Approximately twice that of a single disk
	3	Bit-interleaved parity	N + 1	Much higher than single disk; comparable to RAID 2, 4, or 5	Highest of all listed alternatives	Approximately twice that of a single disk
Independent access	4	Block-interleaved parity	N + 1	Much higher than single disk; comparable to RAID 2, 3, or 5	Similar to RAID 0 for read; significantly lower than single disk for write	Similar to RAID 0 for read; significantly lower than single disk for write
	5	Block-interleaved distributed parity	N + 1	Much higher than single disk; comparable to RAID 2, 3, or 4	Similar to RAID 0 for read; lower than single disk for write	Similar to RAID 0 for read; generally lower than single disk for write
	6	Block-interleaved dual distributed parity	N + 2	Highest of all listed alternatives	Similar to RAID 0 for read; lower than RAID 5 for write	Similar to RAID 0 for read; significantly lower than RAID 5 for write

Other Features

- Regardless of where RAID implemented, other useful features can be added
- Snapshot is a view of file system before a set of changes take place (i.e. at a point in time)
- Replication is automatic duplication of writes between separate sites
  - For redundancy and disaster recovery
  - Can be synchronous or asynchronous
- Hot spare disk is unused, automatically used by RAID production if a disk fails to replace the failed disk and rebuild the RAID set if possible
  - Decreases mean time to repair

Disk Cache

- Cache memory is used to apply to a memory that is smaller and faster than main memory and that is interposed between main memory and the processor
- Reduces average memory access time by exploiting the principle of locality
- Disk cache is a buffer in main memory for disk sectors
- Contains a copy of some of the sectors on the disk



When an I/O request is made for a particular sector, a check is made to determine if the sector is in the disk cache

If YES

The request is satisfied via the cache

If NO

The requested sector is read into the disk cache from the disk