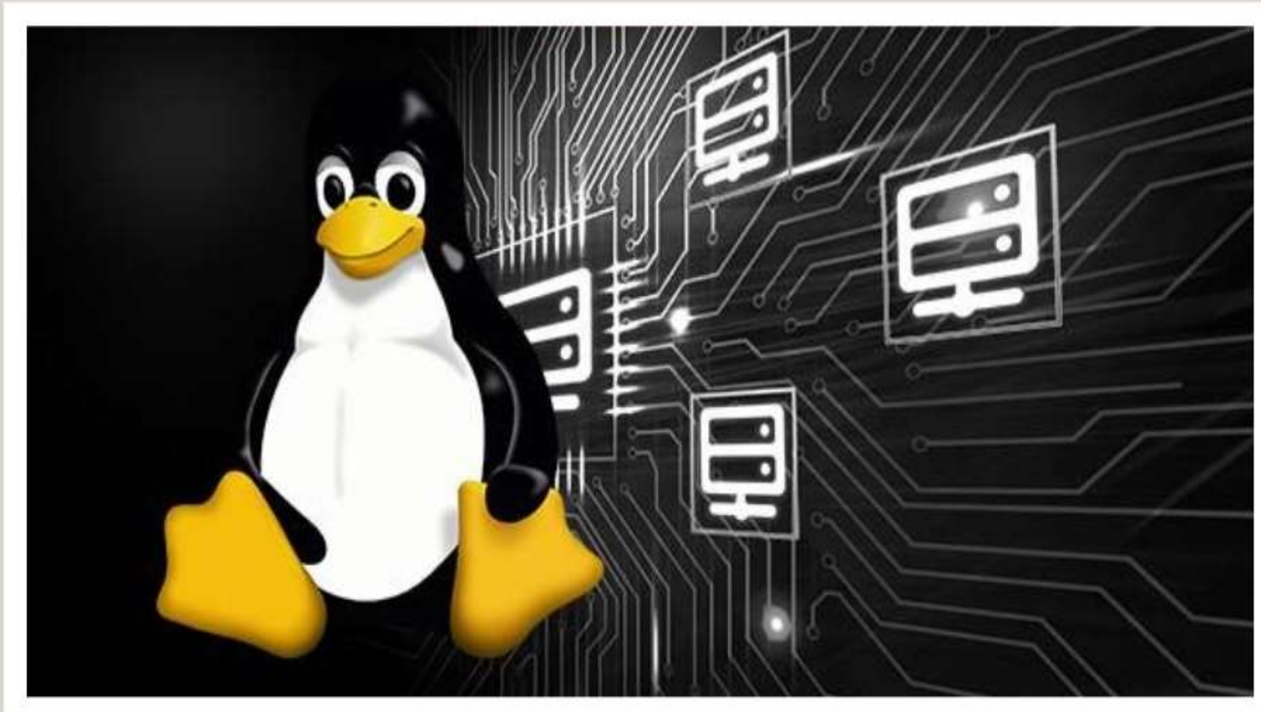


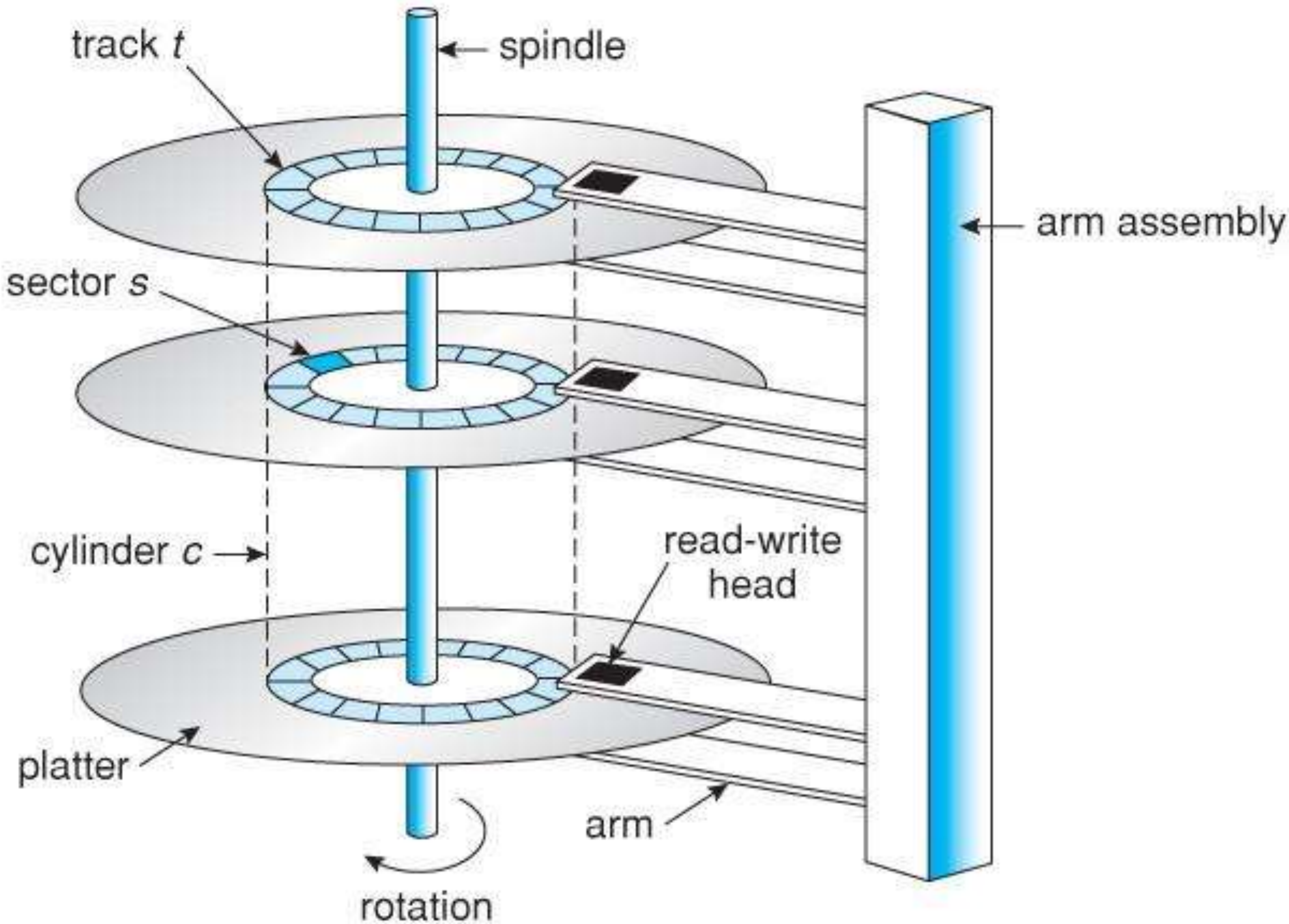
Operating systems

By
I Ravindra kumar, B.Tech, M.Tech,(Ph.D.)
Assistant professor,
Dept of CSE, VNR VJIET

Secondary Storage

Magnetic Disks

- One or more ***platters*** in the form of disks covered with magnetic media.
- Each platter has two working ***surfaces.***
- Each working surface is divided into a number of concentric rings called ***tracks.***
- The collection of all tracks that are the same distance from the edge of the platter is called a ***cylinder***.
- Each track is further divided into ***sectors,*** traditionally containing 512 bytes of data each, although some modern disks occasionally use larger sector sizes
- The data on a hard drive is read by read-write ***heads.***
- The standard configuration (shown below) uses one head per surface, each on a separate ***arm***, and controlled by a common ***arm assembly*** which moves all heads simultaneously from one cylinder to another.
- On operation the disk rotates at high speed, such as 7200 rpm (120 revolutions per second.)
- Disk heads "fly" over the surface on a very thin cushion of air.
- If they should accidentally contact the disk, then a head crash occurs,
  - which may or may not permanently damage the disk or even destroy it completely.


track t
spindle
arm assembly
sector s
cylinder c
read-write
head
platter
arm
rotation

- The rate at which data can be transferred from the disk to the computer is composed of several steps:
  - The ***positioning time***, a.k.a. the ***seek time*** or ***random access time*** is the time required to move the heads from one cylinder to another, and for the heads to settle down after the move.
  - The ***rotational latency*** is the amount of time required for the desired sector to rotate around and come under the read- write
  - The ***transfer rate***, which is the time required to move the data electronically from the disk to the computer
- Floppy disks are normally ***removable***. Hard drives can also be removable, and some are even ***hot-swappable***,
- Disk drives are connected to the computer via a cable known as the ***I/O Bus.***
- Some of the common interface formats include
  - Enhanced Integrated Drive Electronics, EIDE;
  - Advanced Technology Attachment, ATA; Serial ATA, SATA,
  - Universal Serial Bus, USB;
  - Fiber Channel, FC, and
  - Small Computer Systems Interface, SCSI.
- The ***host controller*** is at the computer end of the I/O bus, and the ***disk controller*** is built into the disk itself.

Disk Structure:

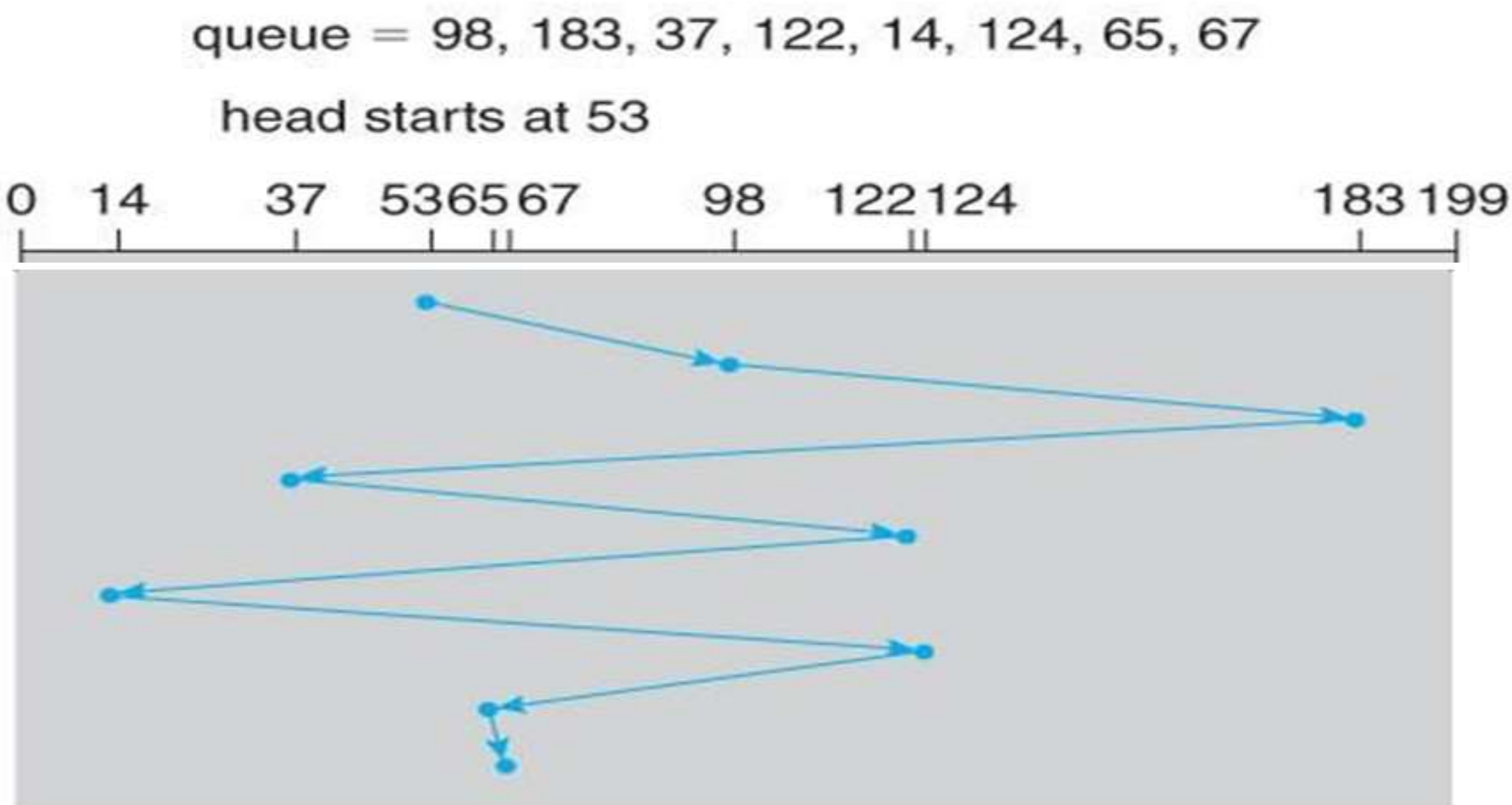
- Modern disk drives are addressed as large one-dimensional arrays of logical blocks, where the logical block is the smallest unit of transfer.
- The size of a logical block is usually 512 bytes, although some disks can be low-level formatted to choose a different logical block size, such as 1,024 bytes.
- The one-dimensional array of logical blocks is mapped onto the sectors of the disk sequentially.
- Sector 0 is the first sector of the first track on the outermost cylinder.
- The mapping proceeds in order through that track, then through the rest of the tracks in that cylinder, and then through the rest of the cylinders from outermost to innermost.
- convert a logical block number into an old-style disk address that consists of a cylinder number, a track number within that cylinder, and a sector number within that track.

- In practice, it is difficult to perform this translation, for two reasons.
  - First, most disks have some defective sectors, but the mapping hides this by substituting spare sectors from elsewhere on the disk.
  - Second, the number of sectors per track is not a constant on some drives.
- On media that use constant linear velocity (CLV), the density of bits per track is uniform.
- The farther a track is from the center of the disk, the greater its length, so the more sectors it can hold.
- As we move from outer zones to inner zones, the number of sectors per track decreases.
- Tracks in the outermost zone typically hold 40 percent more sectors than do tracks in the innermost zone.
- The drive increases its rotation speed as the head moves from the outer to the inner tracks to keep the same rate of data moving under the head.
  - This method is used in CD-ROM and DVD-ROM drives.
- Alternatively, the disk rotation speed can stay constant, and the density of bits decreases from inner tracks to outer tracks to keep the data rate constant.
- This method is used in hard disks and is known as constant angular velocity (CAV).
- The number of sectors per track has been increasing as disk technology improves, and the outer zone of a disk usually has several hundred sectors per track.
- Similarly, the number of cylinders per disk has been increasing; large disks have tens of thousands of cylinders.

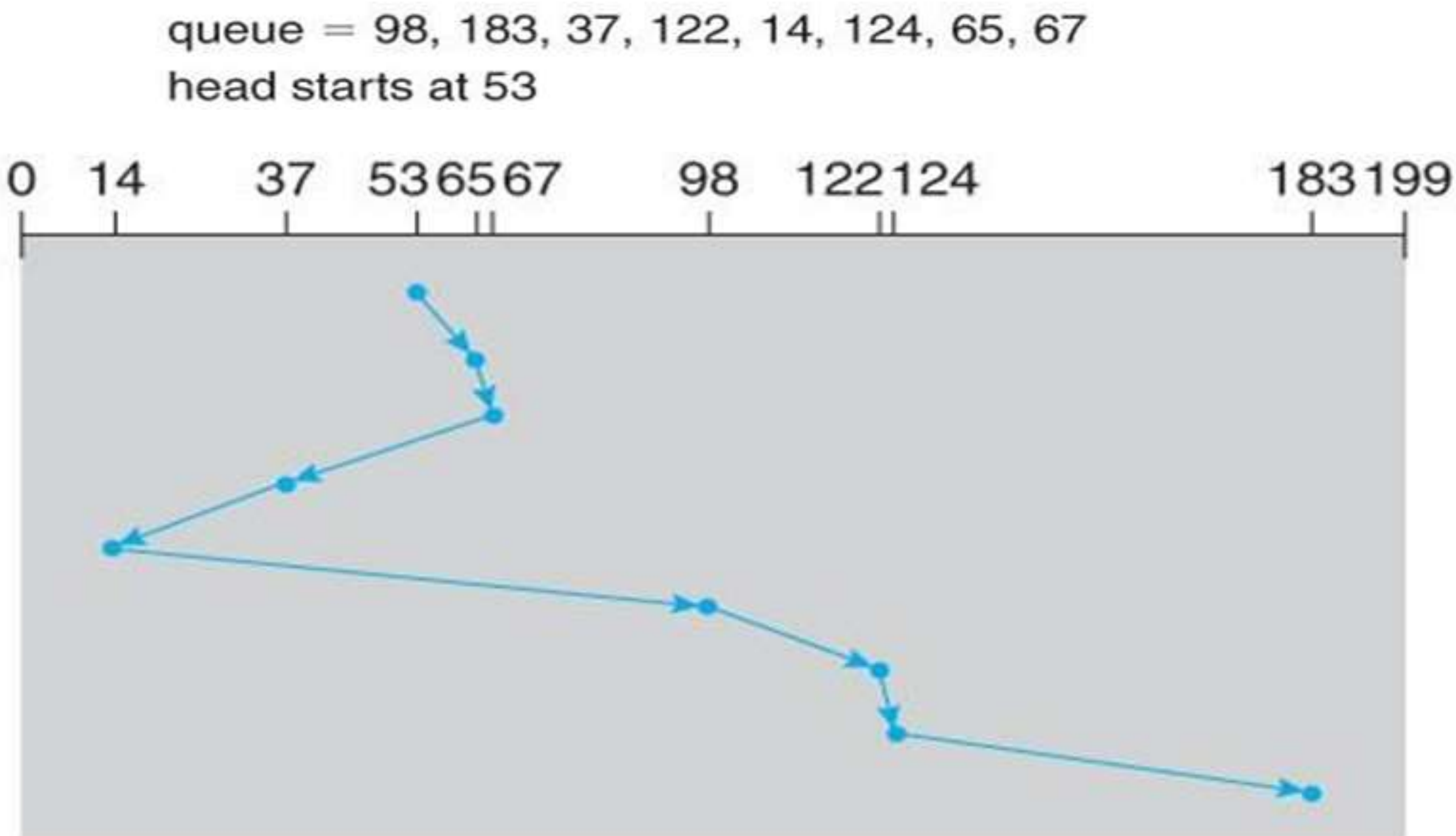
Disk Scheduling :

- The access time has two major components
  - The seek time
  - The rotational latency
- The 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.
- We can improve both the access time and the bandwidth by scheduling the servicing of disk I/O requests in a good order
- If the desired disk drive and controller are available, the request can be serviced immediately.
- If the drive or controller is busy, any new requests for service will be placed on the queue of pending requests for that drive.
- For a multiprogramming system with many processes, the disk queue may often have several pending requests.
- Thus, when one request is completed, the operating system chooses which pending request to service next.

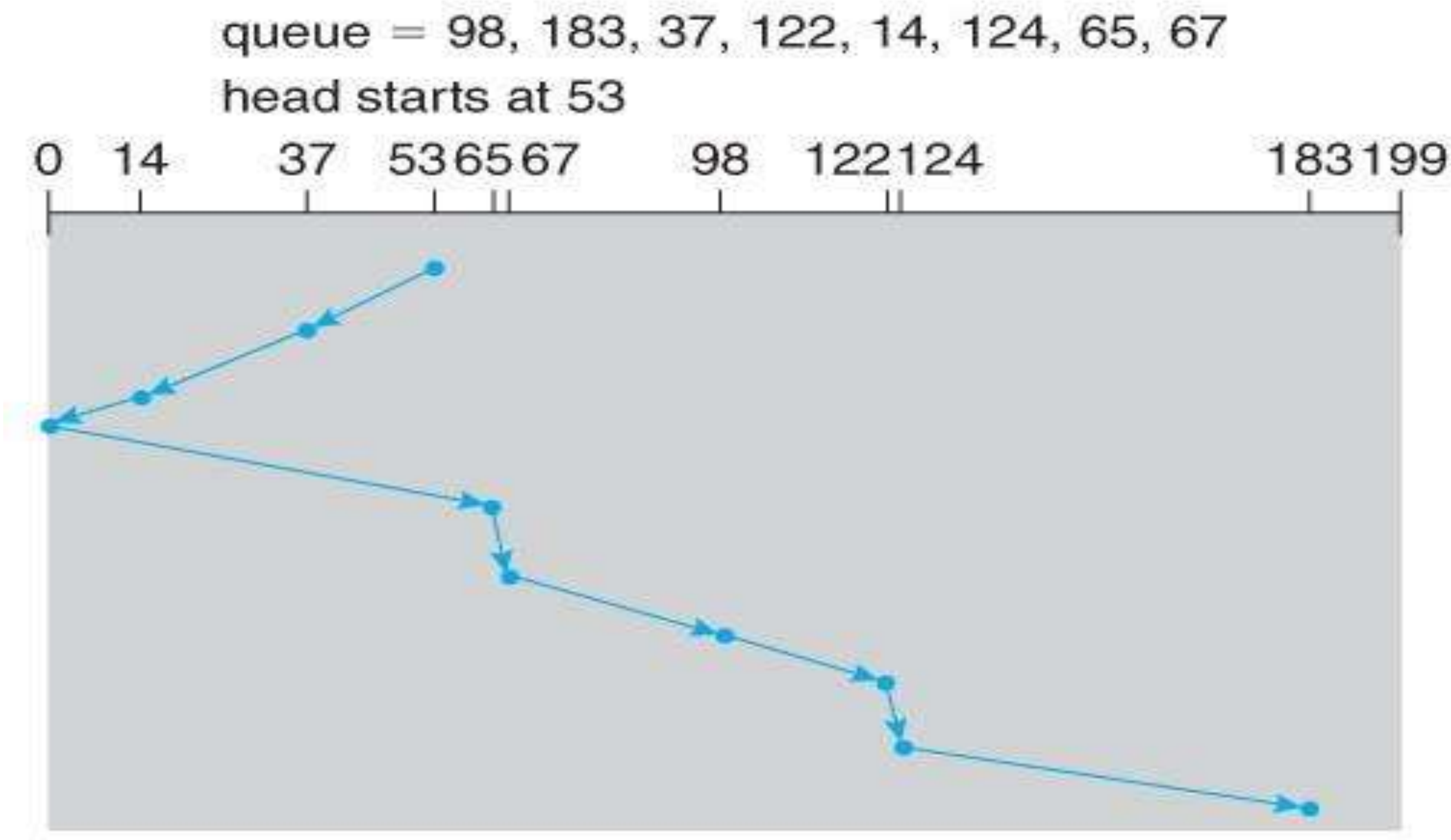
- FCFS Scheduling :
- the first-come, first-served (FCFS) algorithm.
- Consider, for example, a disk queue with requests for 1/0 to blocks on cylinders



- **SSTF Scheduling:**
- ***Shortest Seek Time First*** scheduling is more efficient, but may lead to starvation if a constant stream of requests arrives for the same general area of the disk.



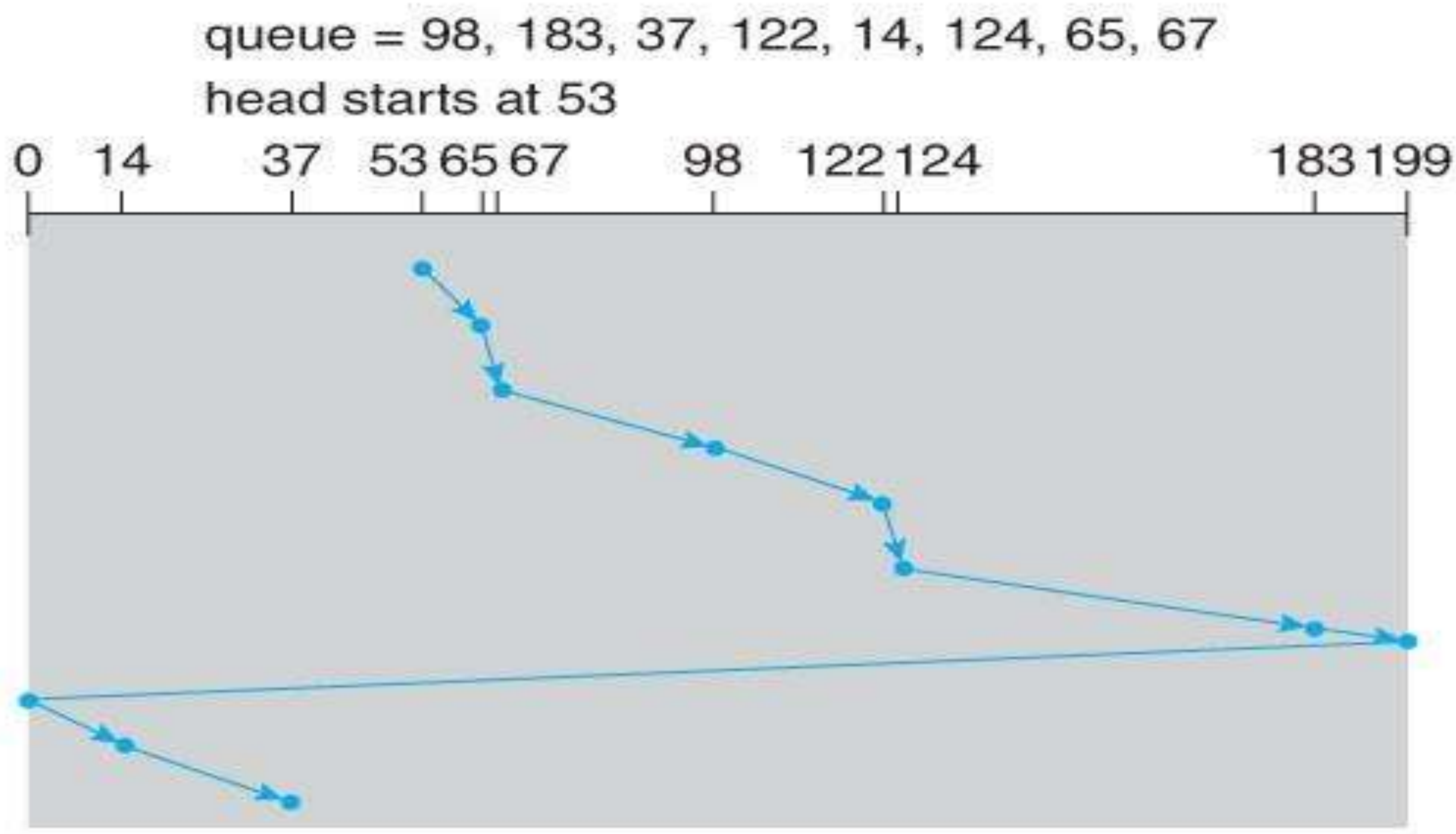
- **SCAN Scheduling:**
- The ***SCAN*** algorithm, a.k.a. the ***elevator*** algorithm moves back and forth from one end of the disk to the other, similarly to an elevator processing requests in a tall building.



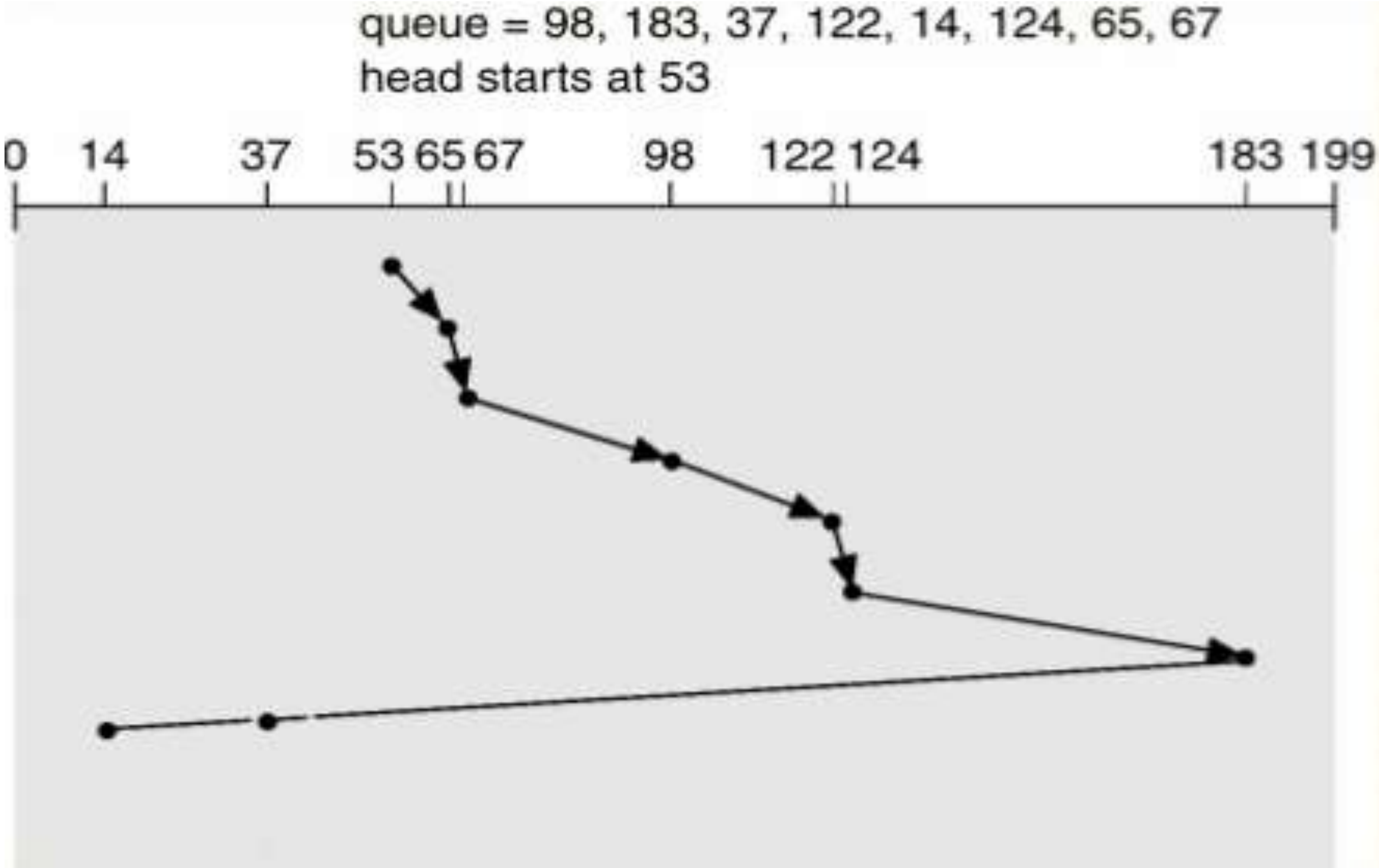
- Under the SCAN algorithm, If a request arrives just ahead of the moving head then it will be processed right away,
  - but if it arrives just after the head has passed, then it will have to wait for the head to pass going the other way on the return trip.
- This leads to a fairly wide variation in access times which can be improved upon.
- Consider, for example, when the head reaches the high end of the disk:
  - Requests with high cylinder numbers just missed the passing head, which means they are all fairly recent requests,
  - whereas requests with low numbers may have been waiting for a much longer time.
- Making the return scan from high to low then ends up accessing recent requests first and making older requests wait that much longer.

- **C-SCAN Scheduling**
- The ***Circular-SCAN*** algorithm improves upon SCAN by treating all requests in a circular queue fashion

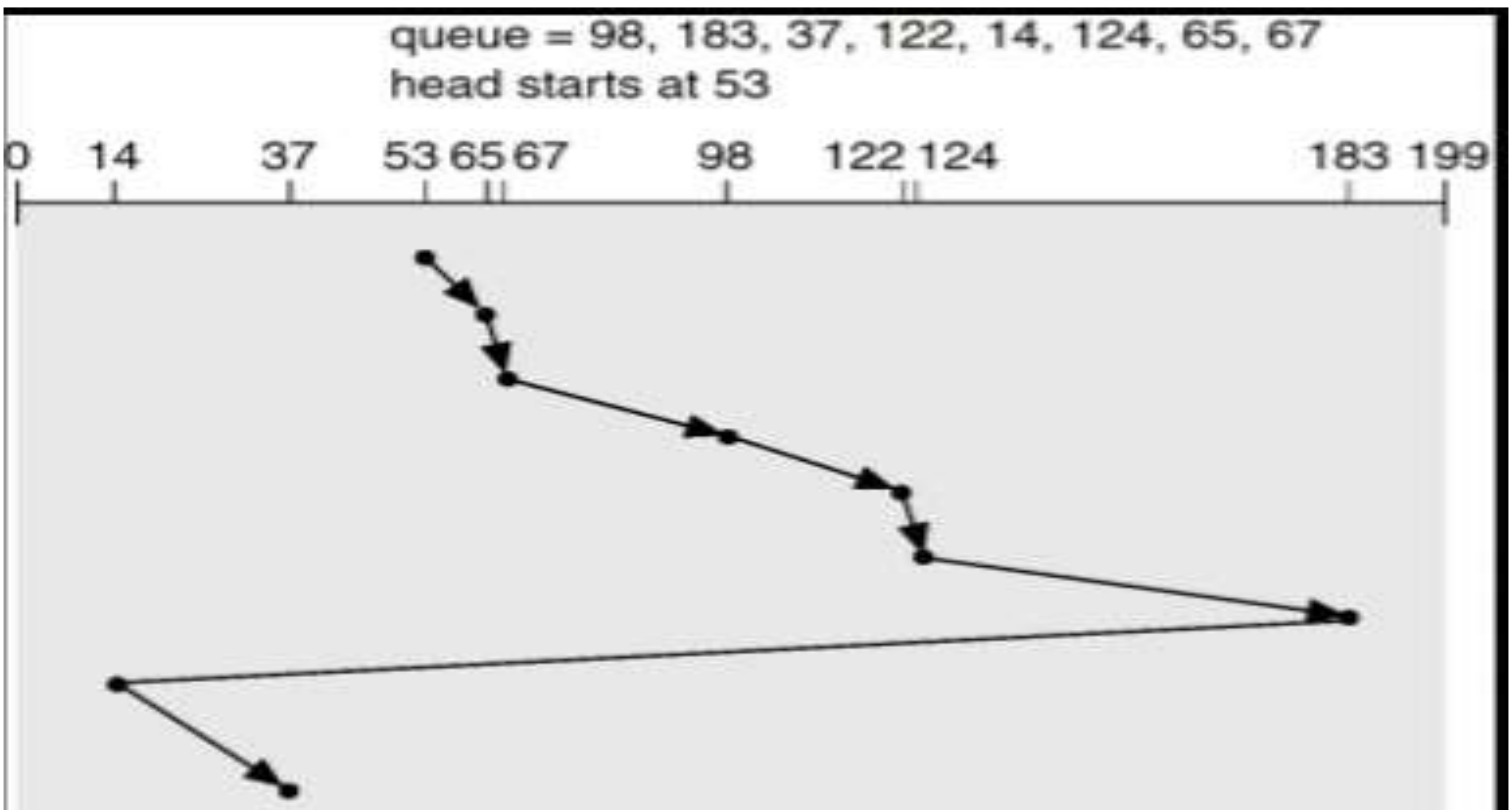
  - Once the head reaches the end of the disk, it returns to the other end without processing any requests, and then starts again from the beginning of the disk:



- **LOOK Scheduling:**
- ***LOOK*** scheduling improves upon SCAN by looking ahead at the queue of pending requests, and not moving the heads any farther towards the end of the disk than is necessary



- **CLOOK:**
- As LOOK is similar to SCAN algorithm, in similar way, CLOOK is similar to CSCAN disk scheduling algorithm.
- In CLOOK, the disk arm inspite of going to the end goes only to the last request to be serviced in front of the head and then from there goes to the other end's last request.
- Thus, it also prevents the extra delay which occurred due to unnecessary traversal to the end of the disk.



Swap-Space Management:

- Swap-space management is another low-level task of the operating system.
- Virtual memory uses disk space as an extension of main memory.
- Since disk access is much slower than memory access, using swap space significantly decreases system performance.
- The main goal for the design and implementation of swap space is to provide the best throughput for the virtual-memory system.
- **Swap-Space Use :**
- For instance, systems that implement swapping may use swap space to hold the entire process image, including the code and data segments.
- Paging systems may simply store pages that have been pushed out of main memory.
- The amount of swap space needed on a system can therefore vary depending on the amount of physical memory, the amount of virtual memory it is backing, and the way in which the virtual memory is used.
- It can range from a few megabytes of disk space to gigabytes.
- Note that it is safer to overestimate than to underestimate swap space, because if a system runs out of swap space it may be forced to abort processes or may crash entirely.
- Overestimation wastes disk space that could otherwise be used for files, but does no other harm.

- **Swap-Space Location :**
- A swap space can reside in two places:
- Swap space can be carved out of the normal file system, or it can be in a separate disk partition.
- If the swap space is simply a large file within the file system, normal file-system routines can be used to create it, name it, and allocate its space.
- This approach, though easy to implement, is also inefficient.
- Navigating the directory structure and the disk-allocation data structures takes time and (potentially) extra disk accesses.
- External fragmentation can greatly increase swapping times by forcing multiple seeks during reading or writing of a process image.
- We can improve performance by caching the block location information in physical memory,
  - and by using special tools to allocate physically contiguous blocks for the swap file, but the cost of traversing the file-system data structures still remains.

- swap space can be created in a separate disk partition.
- No file system or directory structure is placed on this space.
- Rather, a separate swap-space storage manager is used to allocate and deallocate the blocks.
- This manager uses algorithms optimized for speed, rather than for storage efficiency.
- Internal fragmentation may increase,
  - but this tradeoff is acceptable because data in the swap space generally live for much shorter amounts of time than do files in the file system, and the swap area may be accessed much more frequently.
- This approach creates a fixed amount of swap space during disk partitioning.
- Adding more swap space can be done only via repartitioning of the disk (which involves moving or destroying and restoring the other filesystem partitions from backup), or via adding another swap space elsewhere.

- **RAID Structure:**
- Disk drives have continued to get smaller and cheaper,
  - so it is now economically feasible to attach a large number of disks to a computer system
- failure of one disk does not lead to loss of data.
- A variety of disk-organization techniques, collectively called redundant arrays of inexpensive disks (RAID), are commonly used to address the performance and reliability issues
- RAIDS are used for their higher reliability and higher data-transfer rate, rather than for economic reasons.
- Hence, RAID stands for "independent", instead of ‘Inexcepensive’

- **Improvement of Reliability via Redundancy**
- The chance that some disk out of a set of N disks will fail is much higher than the chance that a specific single disk will fail.
- Suppose that the mean time to failure of a single disk is 100,000 hours.
- Then, the mean time to failure of some disk in an array of 100 disks will be 100,000/100 = 1,000 hours, or 41.66 days,
- If we store only one copy of the data, then each disk failure will result in loss of a significant amount of data-such a high rate of data loss is unacceptable.
- The solution to the problem of reliability is to introduce redundancy
- The simplest (but most expensive) approach to introducing redundancy is to duplicate every disk. This technique is called mirroring (or shadowing).
- A logical disk then consists of two physical disks, and every write is carried out on both disks.
- If one of the disks fails, the data can be read from the other
- Suppose that the failures of the two disks are independent; that is, the failure of one disk is not connected to the failure of the other.
- Then, if the mean time to failure of a single disk is 100,000 hours and the mean time to repair is 10 hours,
- then the mean time to data loss of a mirrored disk system is 100, 000^2/(2 * 10) = 500 * 10^6 hours, or 57,000 years!

Improvement in Performance via Parallelism:

- With multiple disks, we can improve the transfer rate as well (or instead) by striping data across multiple disks.
- In its simplest form, data striping consists of splitting the bits of each byte across multiple disks; such striping is called bit-level striping.
- For example, if we have an array of eight disks, we write bit i of each byte to disk i.
- The array of eight disks can be treated as a single disk with sectors that are eight times the normal size, and, more important, that have eight times the access rate.
- Bit-level striping can be generalized to a number of disks that either is a multiple of 8 or divides 8.
- For example, if we use an array of four disks, bits i and 4+i of each byte go to disk i.
- Further, striping does not need to be at the level of bits of a byte:
- For example, in block-level striping, blocks of a file are striped across multiple disks; with n disks, block i of a file goes to disk (i mod n) + 1.
- there are two main goals of parallelism in a disk system:

  1. Increase the throughput of multiple small accesses (that is, page accesses) by load balancing.

  2. Reduce the response time of large accesses.

- RAID Levels:
- Mirroring provides high reliability, but it is expensive.
- Striping provides high data-transfer rates, but it does not improve reliability.
- Numerous schemes to provide redundancy at lower cost by using the idea of disk striping combined with "parity" bits (which we describe next) have been proposed.
- These schemes have different cost-performance tradeoffs and are classified into levels called 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

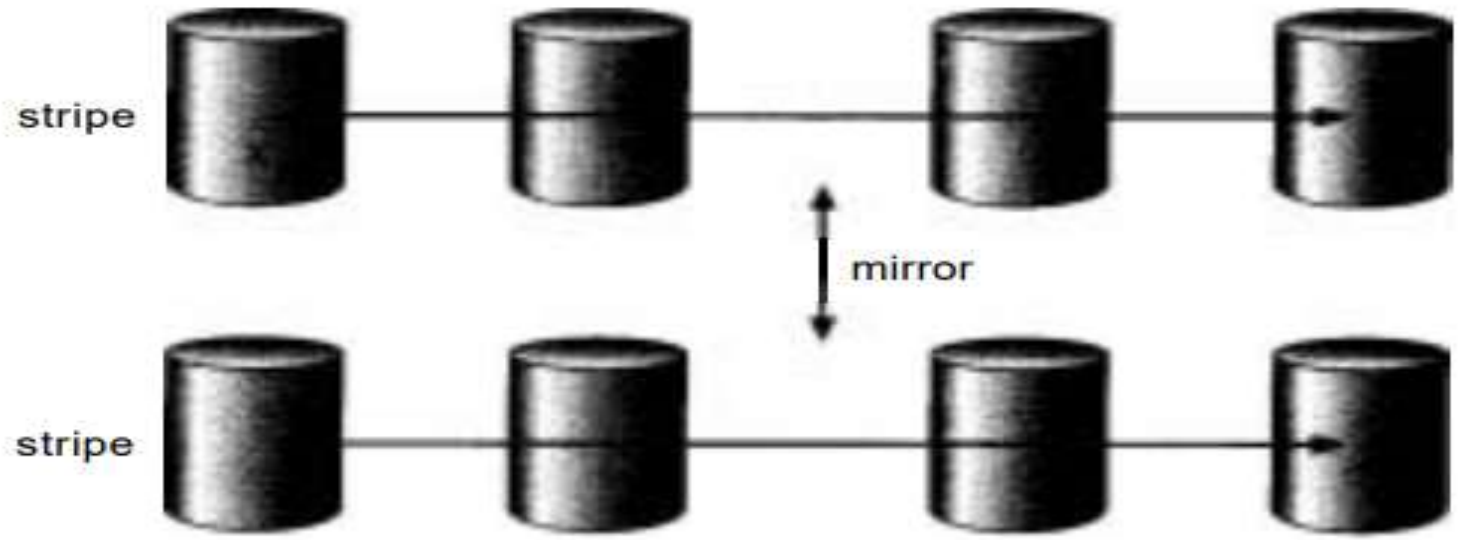
(e) RAID 4: block-interleaved parity

(f) RAID 5: block-Interleaved distributed parity

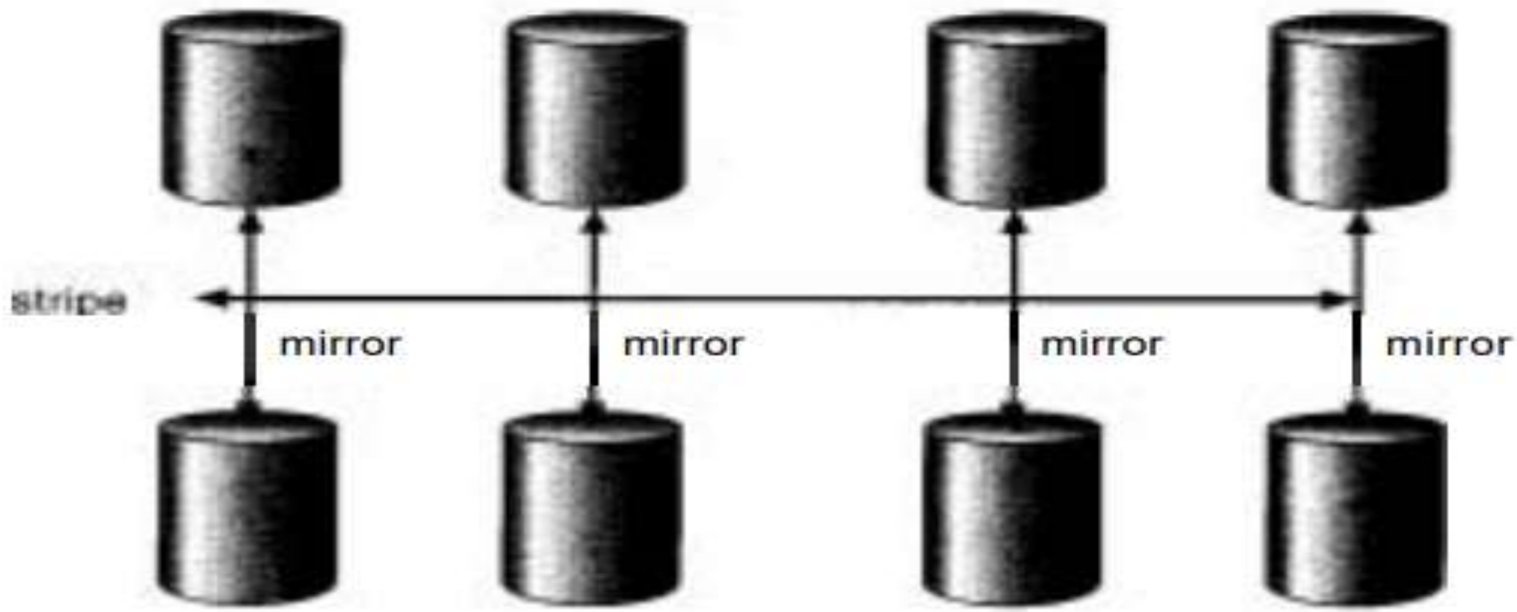
(g) RAID 6: P + Q redundancy

- RAID Level 0: RAID level 0 refers to disk arrays with striping at the level of blocks, but without any redundancy
- RAID Level 1: RAID level 1 refers to disk mirroring.
- RAID Level 2: RAID level 2 is also known as memory-style error-correctingcode (ECC) organization.
  - Memory systems have long implemented error detection using parity bits.
- RAID level 3: RAID level 3, or bit-interleaved parity organization, improves on level 2 by noting that,
  - unlike memory systems, disk controllers can detect whether a sector has been read correctly,
  - so a single parity bit can be used for error correction, as well as for detection
- RAID Level 4: RAID level 4, or block-interleaved parity organization, uses block-level striping,
  - as in RAID 0, and in addition keeps a parity block on a separate disk for corresponding blocks from N other disks.

- RAID level 5: RAID level 5, or block-interleaved distributed parity, differs from level 4 by spreading data and parity among all N + 1 disks,
  - rather than storing data in N disks and parity in one disk.
- RAID Level 6: RAID level 6, also called the P+Q redundancy scheme,
  - is much like RAID level 5,
  - but stores extra redundant information to guard against multiple disk failures.
- RAID level 0 + 1: RAID level 0 + 1 refers to a combination of RAID levels 0 and 1.
  - RAID 0 provides the performance,
  - while RAID 1 provides the reliability.



a) RAID 0 + 1 with a single disk failure



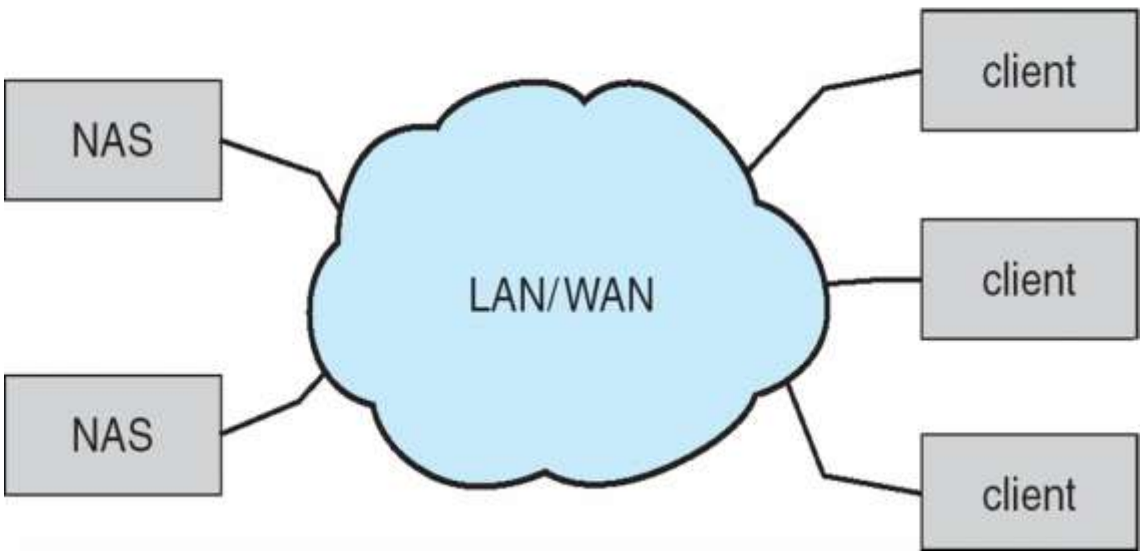
b) RAID 1 + 0 with a single disk failure

Disk Attachment:

- Host-attached storage accessed through I/O ports talking to I/O busses
- SCSI itself is a bus, up to 16 devices on one cable, **SCSI initiator** requests operation and **SCSI targets** perform tasks
  - Each target can have up to 8 **logical units** (disks attached to device controller)
- FC is high-speed serial architecture
  - Can be switched fabric with 24-bit address space – the basis of **storage area networks (SAN**s**)** in which many hosts attach to many storage units
- I/O directed to bus ID, device ID, logical unit (LUN)
- **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
  - Shared storage -> more efficiency
  - Features found in some file systems
    - Snaphots, clones, thin provisioning, replication, deduplication, etc

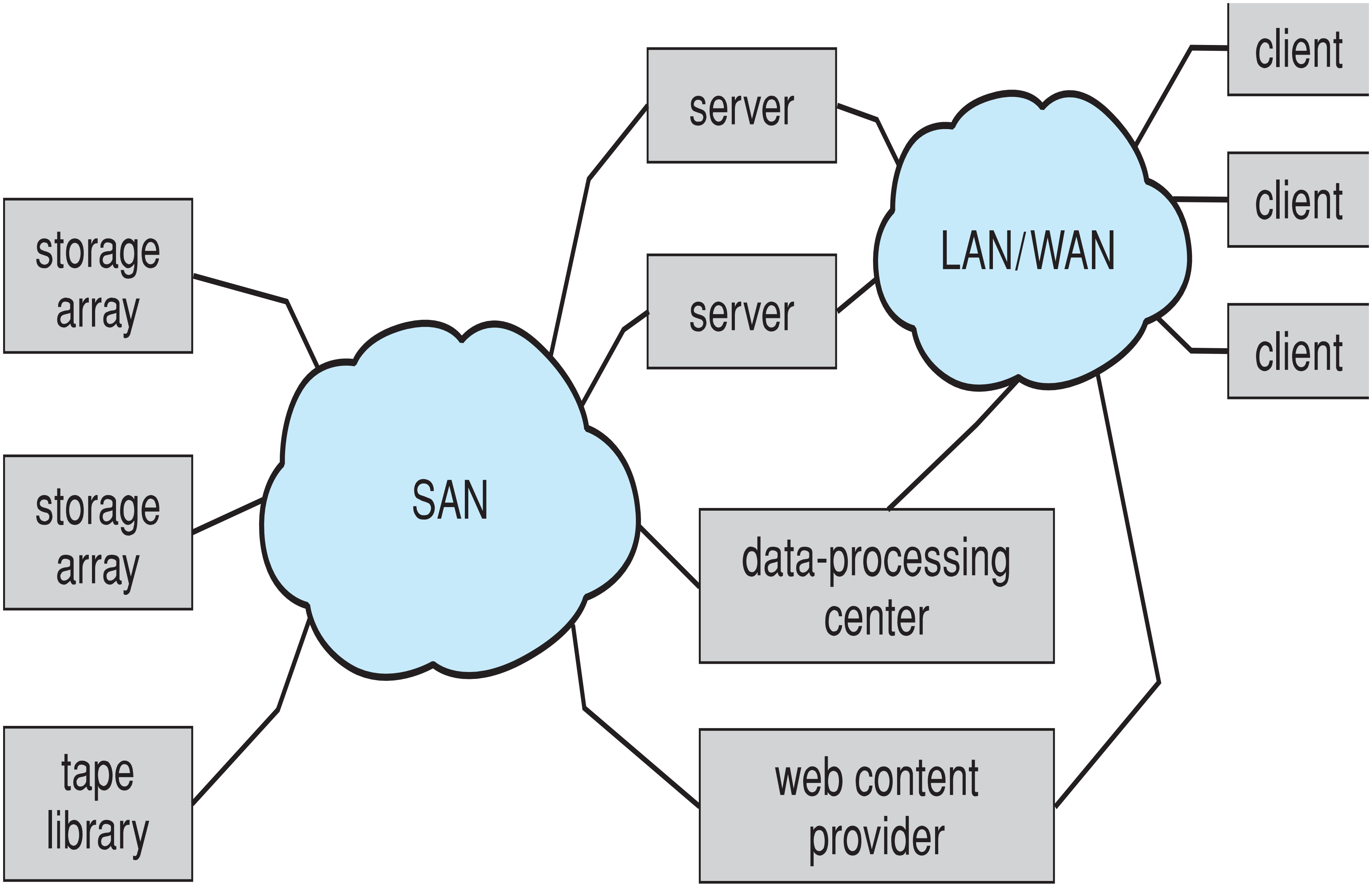
Network-Attached Storage:

- Network-attached storage (**NAS**) is 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
  - Remotely attaching to devices (blocks)



Storage Area Network:

- Common in large storage environments
- Multiple hosts attached to multiple storage arrays – flexible
- SAN is one or more storage arrays
  - Connected to one or more Fibre Channel switches
- Hosts also attach to the switches
- Storage made available via **LUN Masking** from specific arrays to specific servers
- Easy to add or remove storage, add new host and allocate it storage
  - Over low-latency Fibre Channel fabric


storage array
storage array
tape library
SAN
server
server
LAN/WAN
client
client
client
data-processing center
web content provider