

Disk Systems

Jonathan Windle

University of East Anglia

J.Windle@uea.ac.uk

June 10, 2017

1 Secondary Storage

2 Hard Drives

- Issues
- Typical Disk Access Operations
- Read-Write Latency

3 Disk Scheduling

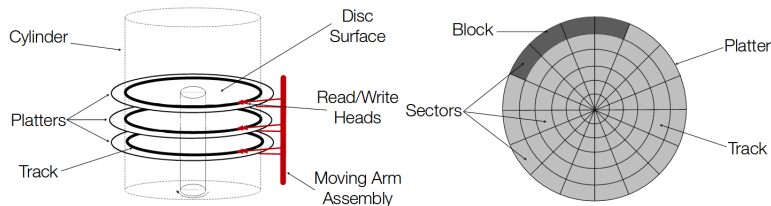
- First come first served
- Shortest-seek-time-first
- Elevator Algorithm (SCAN)
- Circular-SCAN (C-SCAN)
- Freeze-SCAN (F-SCAN)
- Where its performed
- Other performance factors

Secondary Storage

- Most secondary storage devices utilise magnetic media
- A read-write head is moved across the media
- Early tape systems accessed data sequentially, data is streamed in order from first bit to last sequential bit, access not practical for interactive systems
- More modern devices use magnetic disks, data can be accessed in any order (random access)
- Early magnetic disk system used large removable disks (platters)
- Over time these have gotten progressively smaller
- Floppy Disk removable storage media now largely replaced by semiconductor memory

Hard Drives

- Platters are formed from billions of ferrite grains
- Limiting factors affecting actual storage density:
 - Interference at boundaries between logic 0 and 1 (opposing magnetic domains)
 - Heat.



- Main (semiconductor) memory provides almost uniform access time
- Can read/write any location equally quickly
- But access time for disk store depends on position of bits on actual disk relative to read/write head
 - Requires mechanical movements
 - Relevant regions of disc must be located
 - Aim to minimise amount of time spent searching

Typical Disk Access Operations

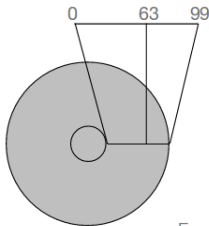
- OS specifies location of the data (head/cylinder/sector)
- Moving arm assembly moves disk arm to cylinder
- Time taken to move known as seek time
- Disk must be rotated until sector is under read/write head
- Time delay referred to as rotational latency
- Data is read/written as sector moves past head
- Referred to as transmission time

Read-Write Latency

- Disk access time is a function of the three delays
 - Seek time + rotational latency + transmission time
 - These all vary depending on the relative position of the read-write head and sector of interest.
- Typically in the order of milliseconds
- In this time the CPU may have executed millions of instructions
- Many processes are becoming increasingly I/O bound:
 - CPU/Memory speeds are increasing rapidly (due to advances in semiconductor fabrication) while disk access has not increased proportionally.

Disk Scheduling

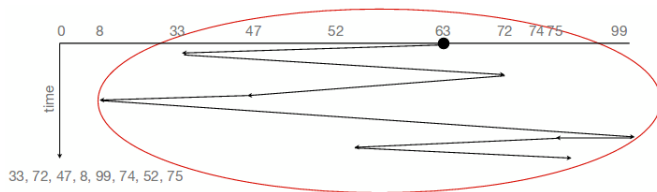
- Disk access requires scheduling:
 - Many processes can request disk access much faster than they can be serviced
 - Both absolute and relative locations of data are important. CPU/memory speeds are increasing rapidly, disk access has not.
- Scheduling disk access imposes additional delay.
- For examples, assume cylinder requests are in order:
 - 33,72,47,8,99,74,42,75
- Assume read-write head is initially positioned at cylinder 63:



From Deitel et al.

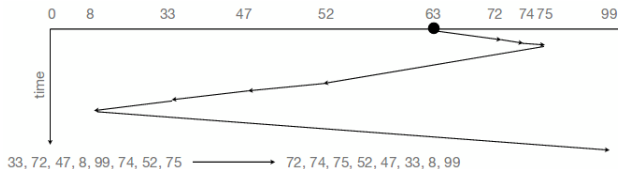
First come first served

- Requests are serviced in order they come.
- A fair policy
- Results in long waiting times if load is high
- Tendency to switch between tracks/sectors



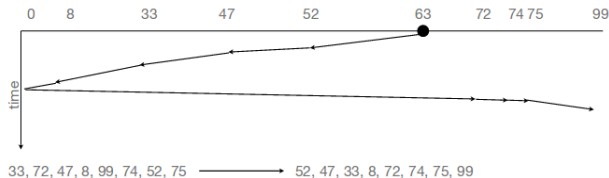
Shortest-seek-time-first

- Schedule request closest to current location of r/w head
- Achieves high throughput
- Does not ensure fairness
- Relatively high variance in response time



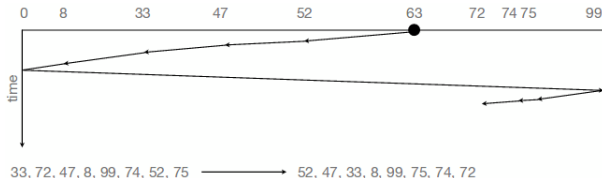
Elevator Algorithm (SCAN)

- Choose shortest seek time in preferred direction
 - At inner/outer cylinder switch direction
- Offers high throughput, low mean response time, and lower variance of response time
- Could suffer indefinite postponement (but unlikely)
- Bias towards middle cylinders
- Unnecessary seek operations are performed to scan to the inner/outer most cylinder



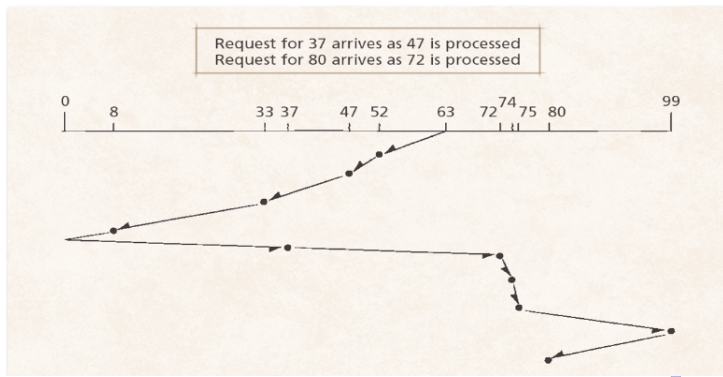
Circular-SCAN (C-SCAN)

- Modification of SCAN strategy
- Choose shortest seek time in inward direction
 - At inner most cylinder, jump back to outermost
- Further reduces variance of response time
- Removes bias towards middle cylinders
- Performs unnecessary seek operations
- Could suffer from indefinite postponement (but unlikely)



Freeze-SCAN (F-SCAN)

- Modification of SCAN
- Choose shortest seek time in inwards direction
 - Service only requests present at the beginning of sweep
 - Order incoming requests during sweep for optimum service
 - Process waiting requests on return
- Solves problem of indefinite postponement



Where it's performed

- The OS could filter requests before sending to the disk controller.
 - OS has knowledge of the overall system load requirements
- Disk controllers themselves are becoming increasingly intelligent
 - Takes burden off OS
- Likely that both perform scheduling

Other performance factors

- Better placement of files on disk:
 - Ensure disk is defragmented to minimise seek time
 - Scheduling overhead can degrade performance
 - Consider transferring blocks rather than sectors
- Utilise data compression/decompression
 - Less data needs read/written for same information storage
 - Assume decompression time is less than inflated read/write time
- Use disk cache buffer
 - Memory to store data to be written
 - Only write during periods of relatively light load
 - Give preference to read operations
 - Need to worry about consistency
- Store multiple copies of frequently accessed data:
 - Read from copy that is closest to r/w head
 - Need to worry about consistency
 - Reduces effective disk capacity
- Utilise RAID (Redundant Array of Independent Disks)
 - Data stored on multiple drives for fast access.

Summary

- Disk access is slow (several milliseconds)
 - Function of 3 effective delays
- Physical layout of bits on drive are addresses as:
 - Surface
 - Cylinder
 - Sector
- OS works at block level
 - Typically many sectors per block

The End