Chapter 12: Secondary Storage Structure

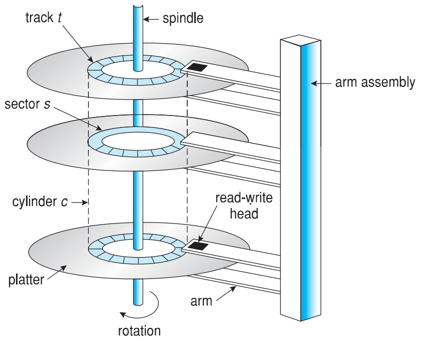
Contents

- Overview of Mass-Storage
- Disk Structure
- Disk Scheduling
- Disk Management
- Swap-Space Management

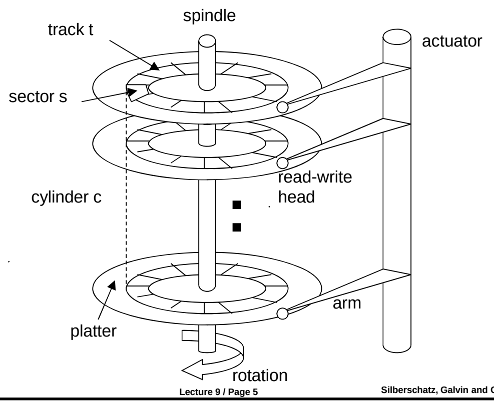
Overview

- Magnetic disks provide bulk of secondary storage of modern computers
  - Drives rotate at 60 to 250 times per second
  - Transfer rate is rate at which data flows between the drive and the computer
  - Positioning time (random-access time) consists of:
    - (1) Seek time - time to move disk arm to desired cylinder
    - (2) Rotational Latency - time for desired sector to rotate under the disk head
  - **Head crash** results from disk head making contact with the disk surface
    - That's bad
- Disks can be removable
- Drive attached to computer by a set of wires called an **I/O bus**
  - Busses vary, including enhanced integrated drive electronics(**EIDE)**, **advanced technology attachment(ATA)**, **serial ATA(SATA)**, **universal serial bus(USB)**, **Fibre Channel**, **small computer systems interface(SCSI)**

Moving-head Disk Mechanism



Hard Disk



Logical Disk Structure

- tracks
  - concentric circles on a platter
  - typically several hundred to several thousand per disk
- cylinder
  - same tracks on different platters
- sectors
  - sections within a track
  - typically 10 to 100 per track
  - can be fixed or variable length
- blocks
  - units of transfer between disk and memory
  - must be compatible with sector sizes

Physical Disk Structure

- mechanical components
  - platters
    - store information on their magnetized surfaces
  - spindle
    - rotational movement of the platters
  - actuator with read/write heads
    - reading and writing of information to the platters
- electronic component (the controller)
  - conversion of analog signals from the disk heads into digital signals
  - arrangement of bits into larger units
    - Bytes, words, blocks

Hard Disk Access Times

- seek time
  - time to move the disk arm to the proper cylinder
  - typically several milliseconds (2-10 ms)
- latency time
  - time until the right sector is under the read/write head
  - depends on the rotational speed
    - 7200 rpm = 120 rps = 1/120 sec/revolution = 8.3 ms/revol.
- transfer time
  - **Transfer rate** is rate at which data flow between drive and computer
  - linear to the amount of information transferred
  - for 100 blocks per track, it is 1/100 of a revolution per block
- average access time
  - av. seek time + av. latency + av. transfer time
= 6 ms + 4 ms + 0.4 ms
≈ 10 ms

Disk Structure

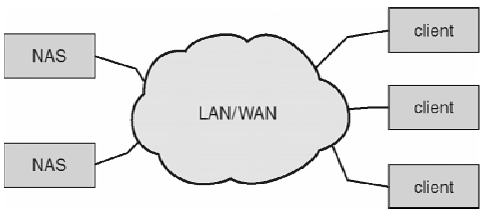
- Disk drives are addressed as large 1-dimensional arrays of **logical blocks**, where the logical block is the smallest unit of transfer
- The 1-dimensional array of logical blocks is mapped into the sectors of the disk sequentially
  - Sector 0 is the first sector of the first track on the outermost 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
- Computer access disk storage in two ways
  - One way is via I/O ports (or host attached storage)
  - Network attached storage

Host Attached Storage

- Host-attached storage is storage accessed through local I/O ports
  - The typical PC uses an I/O bus architecture called IDE or ATA
- High-end workstations and servers generally use more sophisticated I/O architectures such as SCSI and FC
  - SCSI itself is a bus, supports up to 16 devices per bus. Generally, the devices include one controller card on the host (the SCSI initiator) and up to 15 storage devices (the SCSI targets)
    - **SCSI initiator** requests operation and **SCSI targets** perform tasks
  - Fiber channel (FC) is high-speed serial architecture
    - Can be switched fabric with 24-bit address space – the basis of **storage area networks (SANs)** in which many hosts attach to many storage units
    - Can be **arbitrated loop (FC-AL)** that can address 126 devices (devices and controllers)

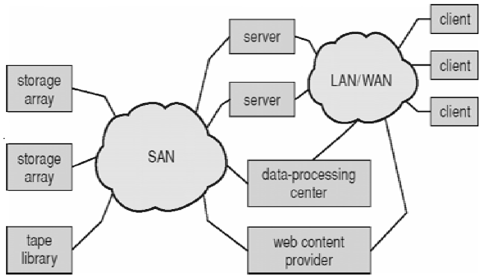
Network-Attached Storage

- Network-attached storage (**NAS**) is storage made available over a network rather than over a local connection (such as a bus)
- NFS and CIFS are common protocols
- Implemented via remote procedure calls (RPCs) between host and storage
- New **iSCSI** protocol uses IP network to carry the SCSI protocol



Storage Area Network (SAN)

- A SAN is a private network (uses storage protocols) connecting servers and storage units
  - Multiple hosts attached to multiple storage arrays - flexible



Disk Scheduling

- The operating system is responsible for using hardware efficiently — for the disk drives, this means having a fast access time (seek time and rotational latency)and large disk bandwidth
  - Want to 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
  - Affected by how efficiently data is read from the disk

Disk Scheduling

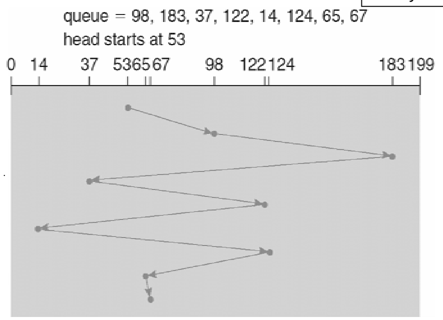
- Each disk I/O request includes the following:
  - Access Mode (i.e. read or write)
  - Disk address for the data transfer
  - A memory address for transfer
  - Number of sectors to transfer
- OS maintains queue of requests, per disk or device
- Several algorithms exist to schedule the servicing of disk I/O requests
- First-Come, First Served (FCFS) scheduling
- Shortest-Seek Time First (SSTF) scheduling
- SCAN / C-SCAN scheduling
- LOOK / C-LOOK scheduling

FCFS

Serve I/O requests in the order that they are received

- Fair, but poor performance

Total head movement of 640 cylinders



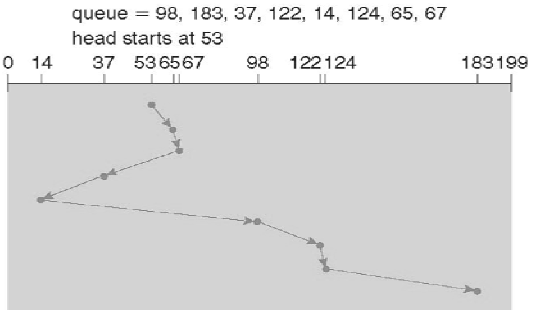
SSTF

Selects the request with the minimum seek time from the current head position

-Reduces overall head movement.

- May lead to *starvation* for some requests.

Total head movement of 236 cylinders



Not Optimal

- If the queue was serviced in a slightly different order:
  - 53, **37, 14**, 65, 67, 98, 122, 124, 183
  - then the total head movement would be less: *208* cylinders

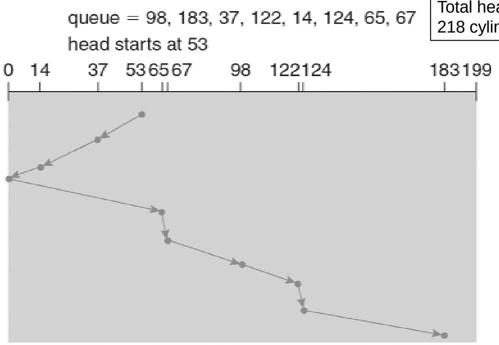
SCAN (Elevator Algorithm)

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

- Scan back and forth across the disk

- Requests at far end of disk may have to wait long time

Total head movement of 218 cylinders

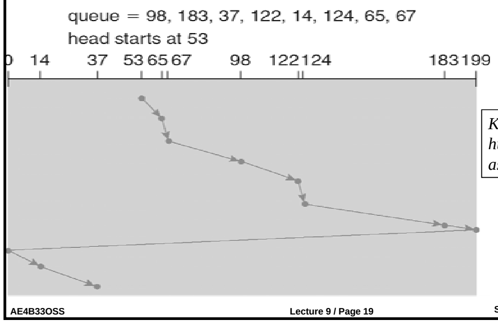


C-SCAN (Circular-SCAN)

Disk head moves from one end of the disk to the other, servicing requests as it goes. When it reaches the other end, it immediately returns to the beginning of the disk, without servicing any requests on the return trip

- Provides a more uniform wait time than SCAN

Total head movement of 183 cylinders



Keep in mind that the huge jump doesn't count as a head movement

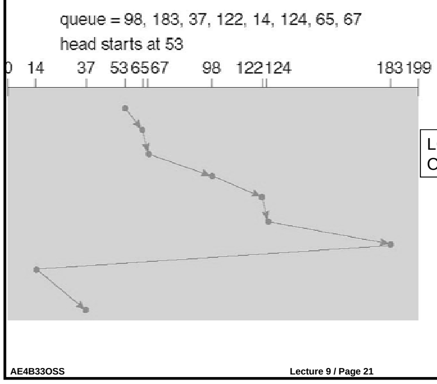
LOOK Scheduling

- SCAN (C-SCAN) moves the head across the full width of the disk.
- LOOK (C-LOOK) moves the head only as far as the last request in each direction, and then reverses (wraps around).

C-LOOK

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
- Eliminates unnecessarily traversing to the edge of the disk before turning around

Total head movement of 153 cylinders



LOOK is a version of SCAN
C-LOOK a version of C-SCAN

Which Disk Scheduling Algorithm to Select?

- SSTF - commonly used
- SCAN/C-SCAN
  - good for heavily loaded disks since they avoid starvation
- Choice depends on number/type of requests
  - e.g. file I/O will generate sequences of requests to adjacent blocks
- Requests for disk service can be influenced by the file-allocation method

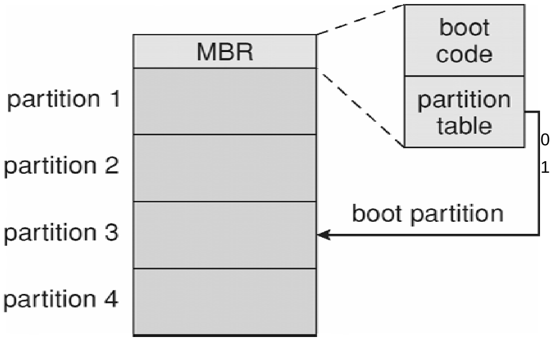
Disk Formatting

- Before a disk can store data, it must be divided into sectors that disk controller can read and write
- The disk controller uses *low-level formatting* (physical formatting)
  - a data structure per each sector
  - consists of
    - header, data area (512 bytes), trailer
    - The header and trailer contain information used by the disk controller, such as a sector number and an error-correcting code (ECC)
- The OS adds its own data structures:
  - partitions the disk into groups of cylinders
  - *logical formatting* (make the file system)
    - includes the directory/file structure, and lists of free and allocated pages

continued

Boot Blocks

- Most boot blocks hold a simple bootstrap loader whose only job is to load and execute a full bootstrap program from disk
  - the program is located in a fixed place on disk (*a boot partition*)



Bad Blocks

- A bad block is a defective sector.
- The OS can mark bad blocks and then skip them, or maintain a list of bad blocks.

- *Sector sparing* (forwarding)
  - spare blocks replace the defective ones

- *Sector slipping*
  - move related blocks to be contiguous if they contain a bad block
    - Example: Suppose that logical block 17 becomes defective and the first available spare follows sector 202.
    - Then, sector slipping remaps all the sectors front 17 to 202, moving them all down one spot.
    - That is, sector 202 is copied into the spare, then sector 201 into 202, then 200 into 201, and so on, until sector 18 is copied into sector 19.
    - Slipping the sectors in this way frees up the space of sector 18, so sector 17 can be mapped to it.

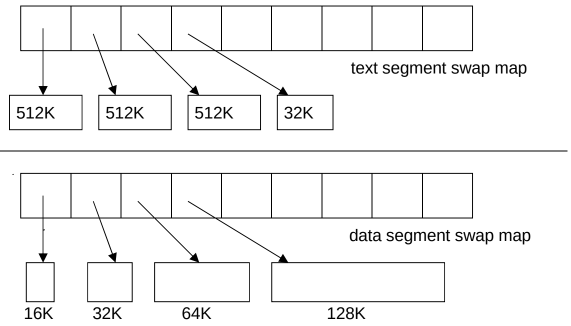
Bad-sector transaction

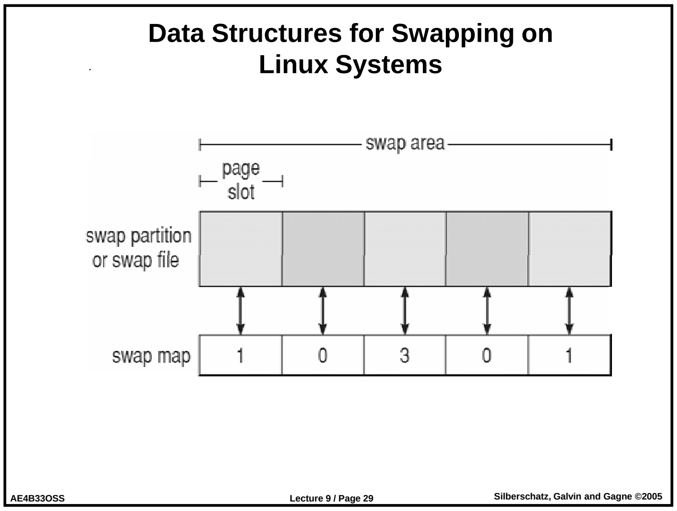
- A typical bad-sector transaction might be as follows:
  - The operating system tries to read logical block 87.
  - The controller calculates the ECC and finds that the sector is bad. It reports this finding to the operating system.
  - The next time the system is rebooted, a special command is run to tell the SCSI controller to replace the bad sector with a spare.
  - After that, whenever the system requests logical block 87, the request is translated into the replacement sector's address by the controller

Swap-Space Management

- Swap-space
  - Virtual memory (VM) uses disk space as an extension of main memory
- Swap-Space Location
  - A swap space can reside in one of two places :
    - Swap-space can be carved out of the normal file system, or
    - it can be in a separate disk partition
- Swap-space management (UNIX swap space management)
  - when process starts, each process is assigned a swap space which holds:
    - *text segments* (for pages of the program)
    - *data segments* (for runtime data)
  - The kernel uses two *swap maps* to track swap space usage in a process.
- Solaris 2 allocates swap space only when a page is forced out of physical memory, not when the virtual memory page is first created

Swap Maps





Data Structures for Swapping on Linux Systems

AE4B33OSS Lecture 9 / Page 29 Silberschatz, Galvin and Gagne ©2005