Chapter 13: Mass-Storage Systems

- Disk Structure
- Disk Scheduling
- Disk Management
- Swap-Space Management
- RAID Structure
- Disk Attachment
- Stable-Storage Implementation
- Tertiary Storage Devices
- Operating System Issues
- Performance Issues

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.
Moving-Head Disk Mechanism

- Disk platters 1.8-5"
- Set of tracks form cylinder
- 1000s of cylinders in a disk
- Each track may contain 100s of sectors
- Outer tracks contain more sectors
- 60-150 rotations per second
- Move to desired cylinder “seek time”
- Rotate to the right sector “rotational time”

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.
- Access time has two major components
  - Seek time is the time for the disk to move the heads to the cylinder containing the desired sector.
  - Rotational latency is the additional time waiting for the disk to rotate the desired sector to the disk head.
- 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 (Cont.)

- Several algorithms exist to schedule the servicing of disk I/O requests.
- We illustrate them 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

- Selects the request with the minimum seek time from the current head position.
- SSTF scheduling is a form of SJF scheduling; may cause starvation of some requests.
- 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*.
- Illustration shows total head movement of 208 cylinders.

SCAN (Cont.)

![Diagram showing head movement across 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 (Cont.)

queue = 98, 183, 37, 122, 14, 124, 65, 67
head starts at 53

![Diagram of C-SCAN sequence with cylinders and requests]
C-LOOK

- 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 (Cont.)

queue = 98, 183, 37, 122, 14, 124, 65, 67
head starts at 53
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.
- 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.
- 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.
  - Logical formatting or “making a file system”.
- Boot block initializes system.
  - The bootstrap is stored in ROM.
  - Bootstrap loader program.
- Methods such as sector sparing used to handle bad blocks.
  - SCSI for example maintains a list of bad blocks
  - Replacement sector addresses (e.g., new sector is associated for
    say logical block 67)
  - If a new bad block is detected, the OS is notified by controller and
    after reboot a new spare is assigned to that sector (bad blocks are
    first detected during low level formatting at the factory and updated
    during the life time of the disk)
MS-DOS Disk Layout

| sector 0       | boot block |
|               | FAT        |
| sector 1      | root directory |
|               | data blocks |
|               | (subdirectories) |

Swap-Space Management

- **Swap-space** —
  - Virtual memory uses disk space as an extension of main memory.
- Swap-space can be carved out of the normal file system, or, more commonly, it can be in a separate disk partition (more efficient).
- **Swap-space management**
  - 4.3BSD allocates swap space when process starts; holds text segment (the program) and data segment.
  - Kernel uses swap maps to track swap-space use.
  - Solaris 2 allocates swap space only when a page is forced out of physical memory, not when the virtual memory page is first created.
### 4.3 BSD Text-Segment Swap Map

- **Swap-space management**
  - 4.3BSD allocates swap space when process starts; holds text segment (the program) and data segment.
  - Pre-allocates all the needed space
    - Text segment is fixed size so swap space is allocated in chunks of 512 KB and the remainder in 1K
  - Kernel uses swap maps (contains swap addresses) to track swap-space use.
  - Two per process maps (text & data)

### 4.3 BSD Data-Segment Swap Map

- Data segment map contains swap addresses for blocks of varying size (2 to the power of i x16K increments)
- When run out of space because of growing a twice larger area is assigned
RAID Structure

- Disks used to be the least reliable components of a system
- RAID – multiple disk drives provides reliability via redundancy.
- RAID is arranged into six different levels.

RAID (cont)

- Several improvements in disk-use techniques involve the use of multiple disks working cooperatively.
- Disk striping (interleaving) uses a group of disks as one storage unit.
  - Each data block is broken into sub-blocks stored on different disk
  - Sub-block transfer is in parallel. Improves speed!
- RAID schemes improve performance and improve the reliability of the storage system by storing redundant data.
  - Mirroring or shadowing keeps duplicate of each disk.
    - Keeps copy of each disk
  - Block interleaved parity uses much less redundancy.
    - A small fraction of the disk space holds parity blocks
    - E.g., for nine disks one could be holding parity for every eight blocks
Intro to RAID Levels

- RAID 0 – only striping
- RAID 1 – mirrored copies
- RAID 2 – ECC codes for words
- RAID 3 – striping combined by parity
- RAID 4 – parity across blocks of same rank on disks
- RAID 5 – distributed parity
- RAID 6 – two parity schemes

RAID2

Calculates ECC code for each data word and can correct single disk errors on the fly for reads
RAID 3

The data block is subdivided ("striped") and written on the data disks. Stripe parity is generated on Writes, recorded on the parity disk and checked on Reads. One disk failure can handled by recalculating bits from parity disk and functioning disks

RAID 4 similar except parity is calculated between blocks of same rank

RAID 4

RAID 4 similar to RAID 3 except parity is calculated between blocks of same rank
RAID 5 would distribute the parity block instead of having it on one block
RAID 6 is extension to RAID 5, having an additional fault-tolerance by second distributed parity scheme added (expensive but can handle multiple faults)
Disk Attachment

- Disks may be attached one of two ways:

1. Host attached via an I/O port
2. Network attached via a network connection

Network-Attached Storage
Tertiary Storage Devices

- Low cost is the defining characteristic of tertiary storage.
- Generally, tertiary storage is built using removable media.
- Common examples of removable media are floppy disks and CD-ROMs; other types are available.
Removable Disks

- Floppy disk — thin flexible disk coated with magnetic material, enclosed in a protective plastic case.
  - Most floppies hold about 1 MB; similar technology is used for removable disks that hold more than 1 GB.
  - Removable magnetic disks can be nearly as fast as hard disks, but they are at a greater risk of damage from exposure.

File Naming

- The issue of naming files on removable media is especially difficult when we want to write data on a removable cartridge on one computer, and then use the cartridge in another computer.
- Contemporary OSs generally leave the name space problem unsolved for removable media, and depend on applications and users to figure out how to access and interpret the data.
- Some kinds of removable media (e.g., CDs) are so well standardized that all computers use them the same way.
Cost

- Main memory is much more expensive than disk storage.
- The cost per megabyte of hard disk storage is competitive with magnetic tape if only one tape is used per drive.
- The cheapest tape drives and the cheapest disk drives have had about the same storage capacity over the years.

Price per Megabyte of DRAM, From 1981 to 2000

![Price per Megabyte of DRAM, From 1981 to 2000](image)
Price per Megabyte of Magnetic Hard Disk, From 1981 to 2000

Price per Megabyte of a Tape Drive, From 1984-2000