

Chapter 12: Secondary-Storage Systems





Chapter 12: Mass-Storage Systems

- Overview of Mass-Storage Structure
- Disk Structure
- Disk Attachment
- Disk Scheduling
- Disk Management
- Swap-Space Management
- RAID Structure
- Stable-Storage Implementation
- Tertiary-Storage Structure





Objectives

- Describe the physical structure of secondary and tertiary storage devices and the resulting effects on the uses of the devices
- Explain the performance characteristics of mass-storage devices
- Discuss operating-system services provided for mass storage, including RAID and HSM (hierarchical storage management)



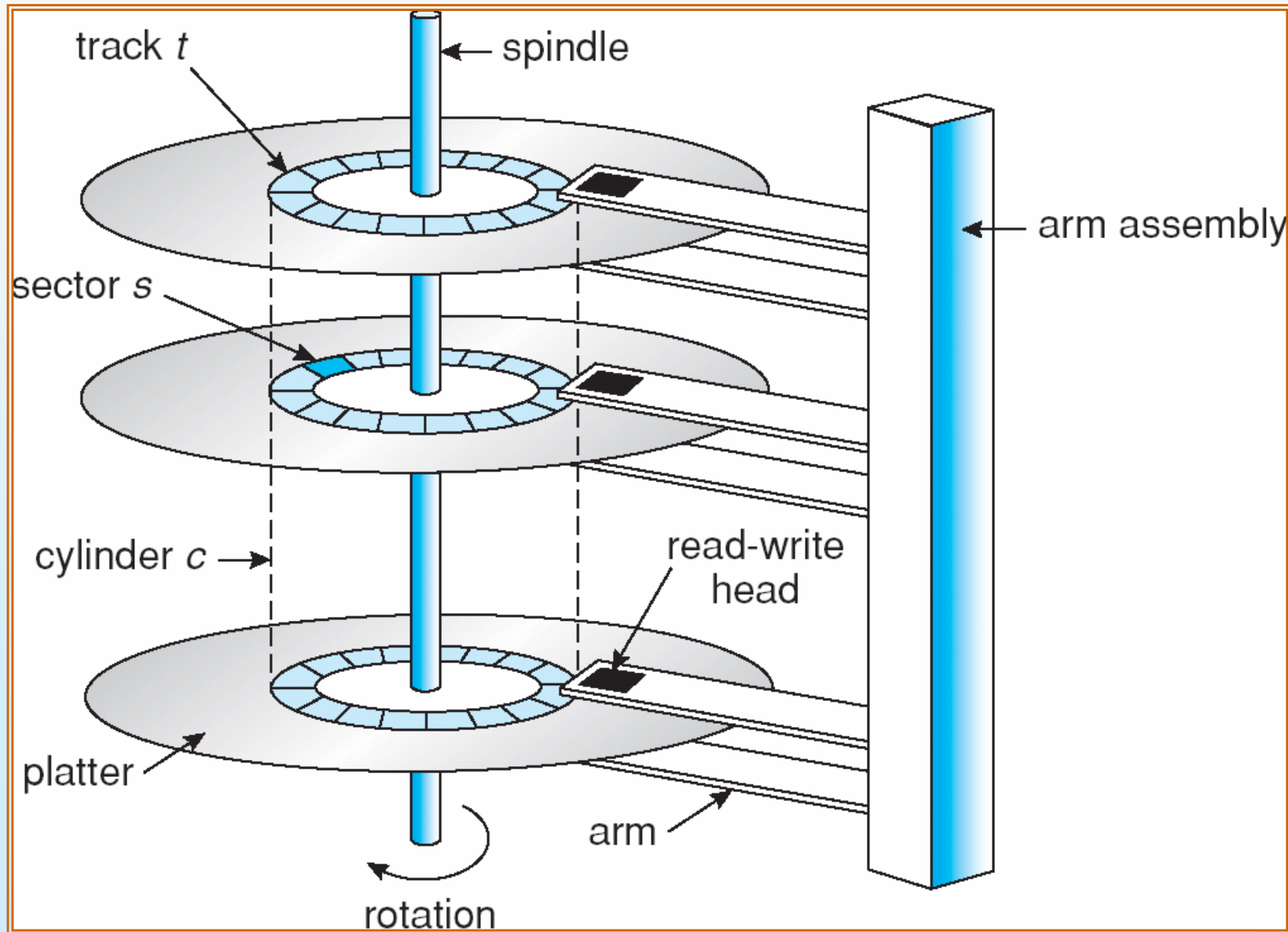


Overview of Mass-Storage Structure

- Magnetic disks provide bulk of secondary storage of modern computers
 - Drives rotate at 60 to 200 times per second
 - **Transfer rate** is rate at which data flow between drive and computer
 - **Positioning time (random-access time)** is time to move disk arm to desired cylinder (**seek time**) and time for desired sector to rotate under the disk head (**rotational latency**)
 - **Head crash** results from disk head making contact with the disk surface
 - ▶ That's bad
- Disks can be removable
- Drive attached to computer via **I/O bus**
 - Busses vary, including **EIDE, ATA, SATA, USB, Fiber Channel, SCSI**
 - **Host controller** in computer uses bus to talk to **disk controller** built into drive or storage array



Moving-head Disk Mechanism





Overview of Mass Storage Structure (Cont.)

- Magnetic tape
 - Was early secondary-storage medium
 - 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
 - 20-200GB typical storage
 - Common technologies are 4mm, 8mm, 19mm, LTO-2 and SDLT





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.





Disk Attachment

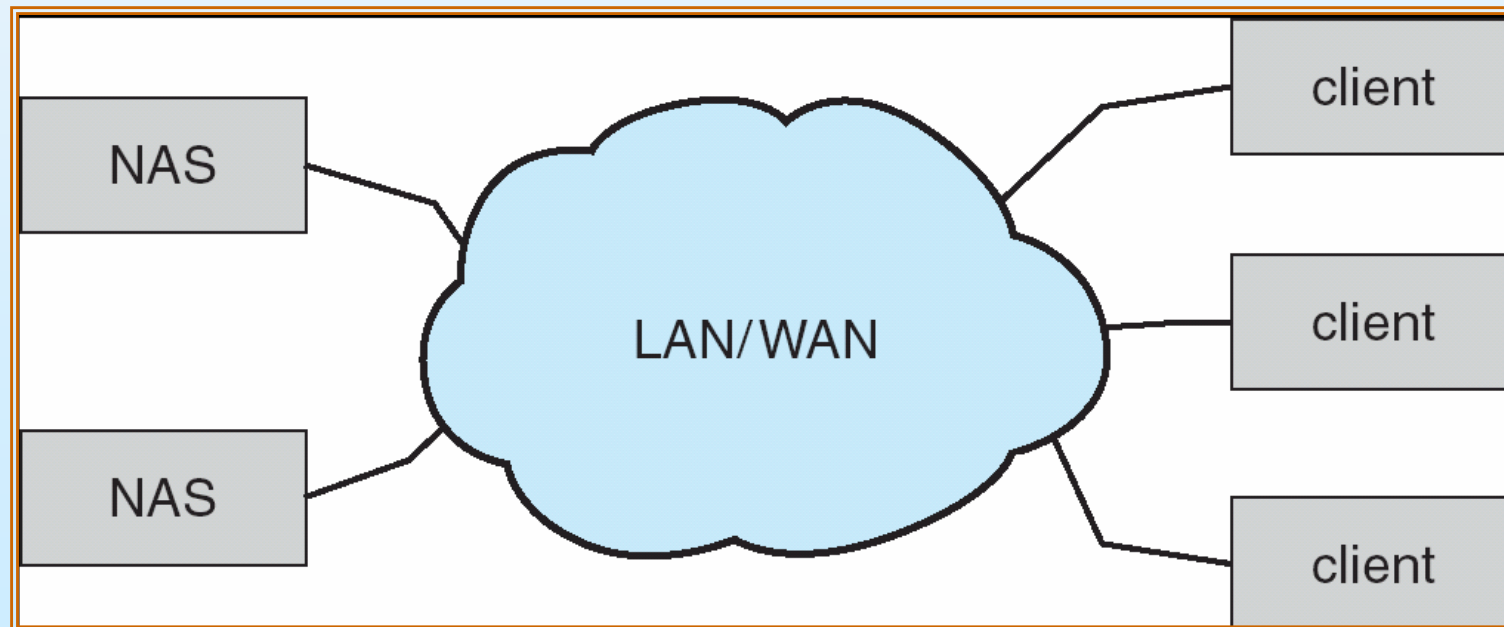
- Host-attached storage accessed through I/O ports talking to I/O busses
- SCSI itself is a bus, up to 16 devices on one cable, **SCSI initiator** requests operation and **SCSI targets** perform tasks
 - Each target can have up to 8 **logical units** (disks attached to device controller)
- FC is high-speed serial architecture
 - Can be switched fabric with 24-bit address space – the basis of **storage area networks (SANs)** in which many hosts attach to many storage units
 - Can be **arbitrated loop (FC-AL)** of 126 devices





Network-Attached Storage

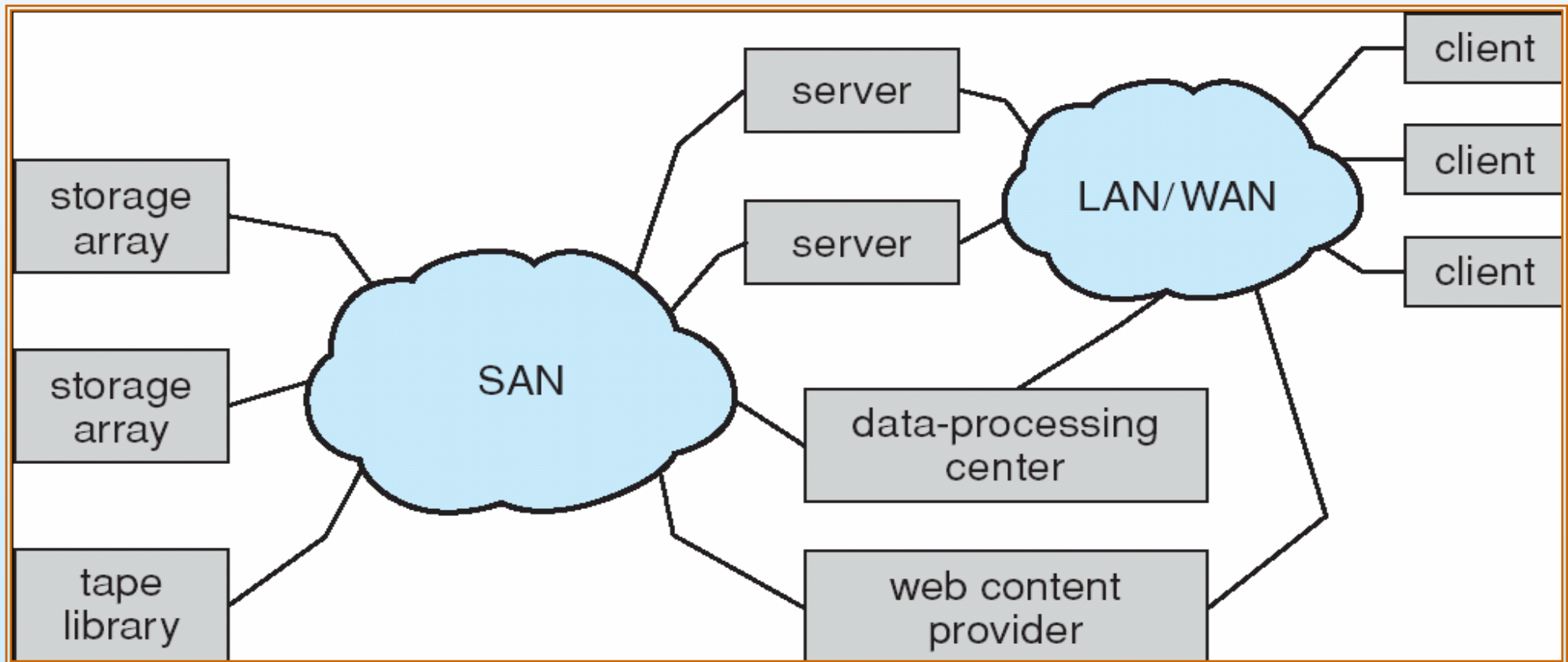
- Network-attached storage (**NAS**) is storage made available over a network rather than over a local connection (such as a bus)
- NFS and CIFS are common protocols
- Implemented via remote procedure calls (RPCs) between host and storage
- New iSCSI protocol uses IP network to carry the SCSI protocol





Storage-Area Network

- Common in large storage environments (and becoming more common)
- 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 and disk bandwidth.
- Access time has two major components
 - *Seek time* is the time for the disk are 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 \approx 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

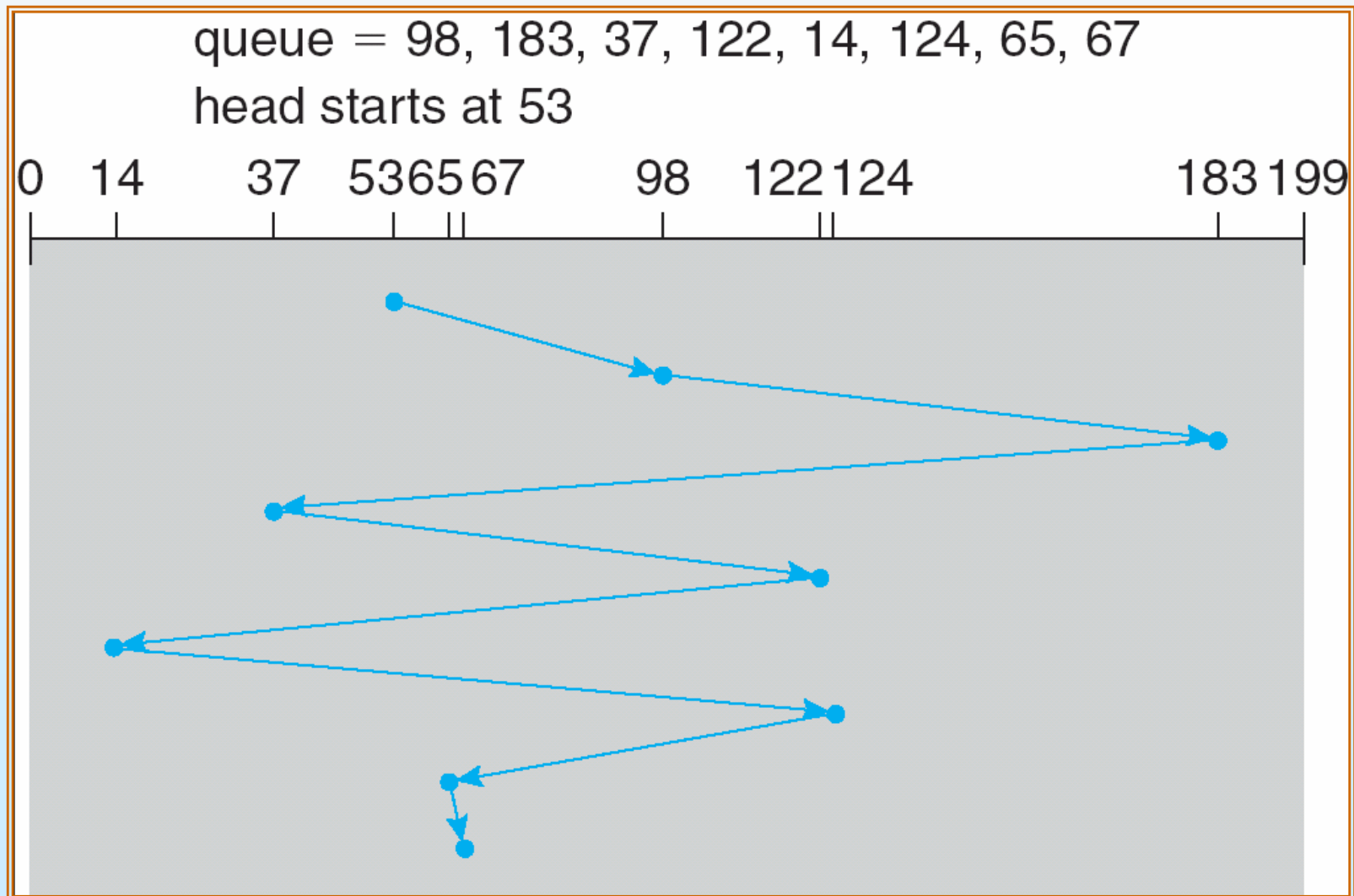
Head pointer 53





FCFS Disk Scheduling

Illustration shows total head movement of 640 cylinders.





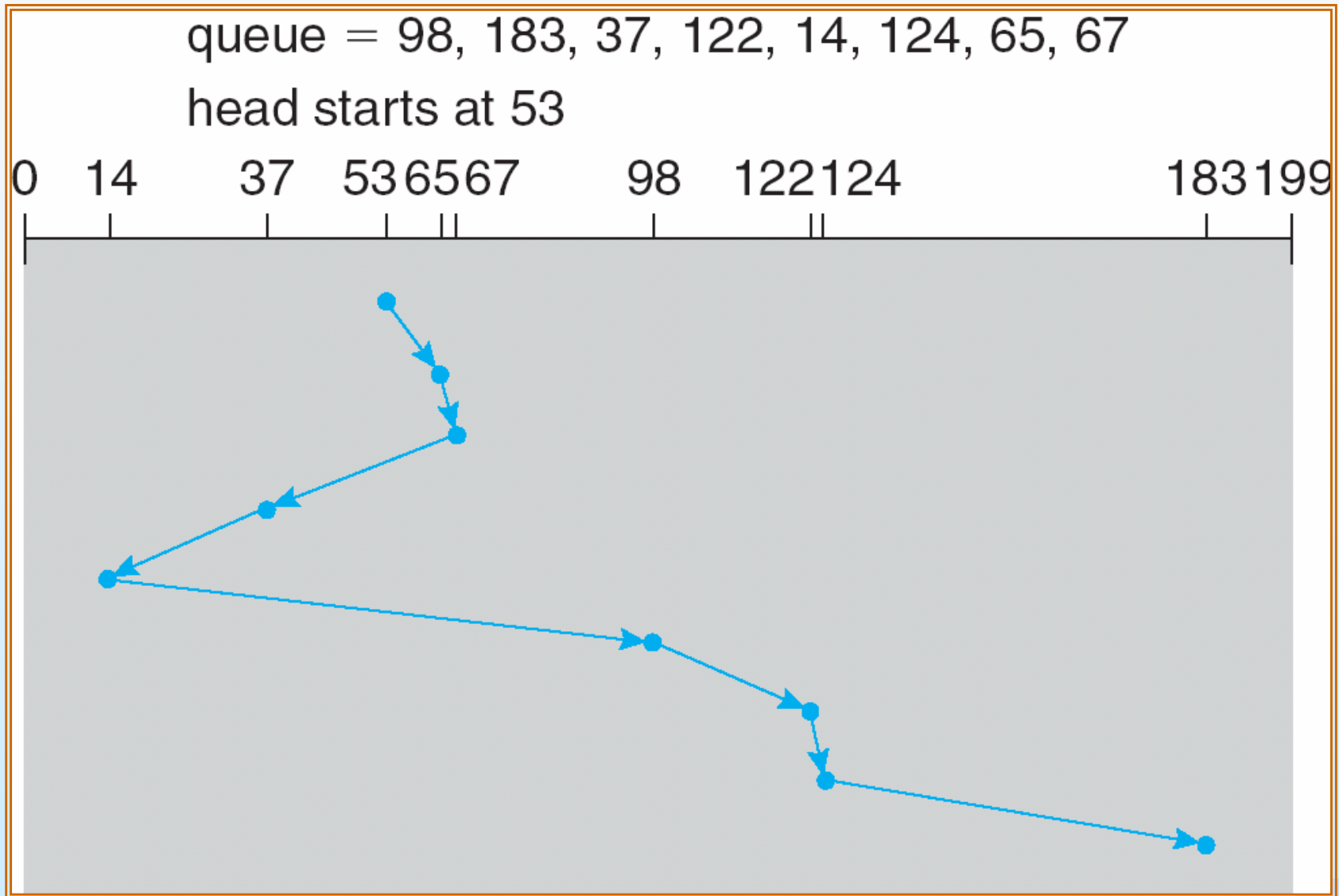
SSTF Scheduling

- 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.





SSTF Disk Scheduling



SSTF Scheduling

- Although SSTF works like SJF
 - Not optimal
- 53 -> 37, 14, 65, 67, 98, 122, 124, 183
 - total head movement: 208 cylinders





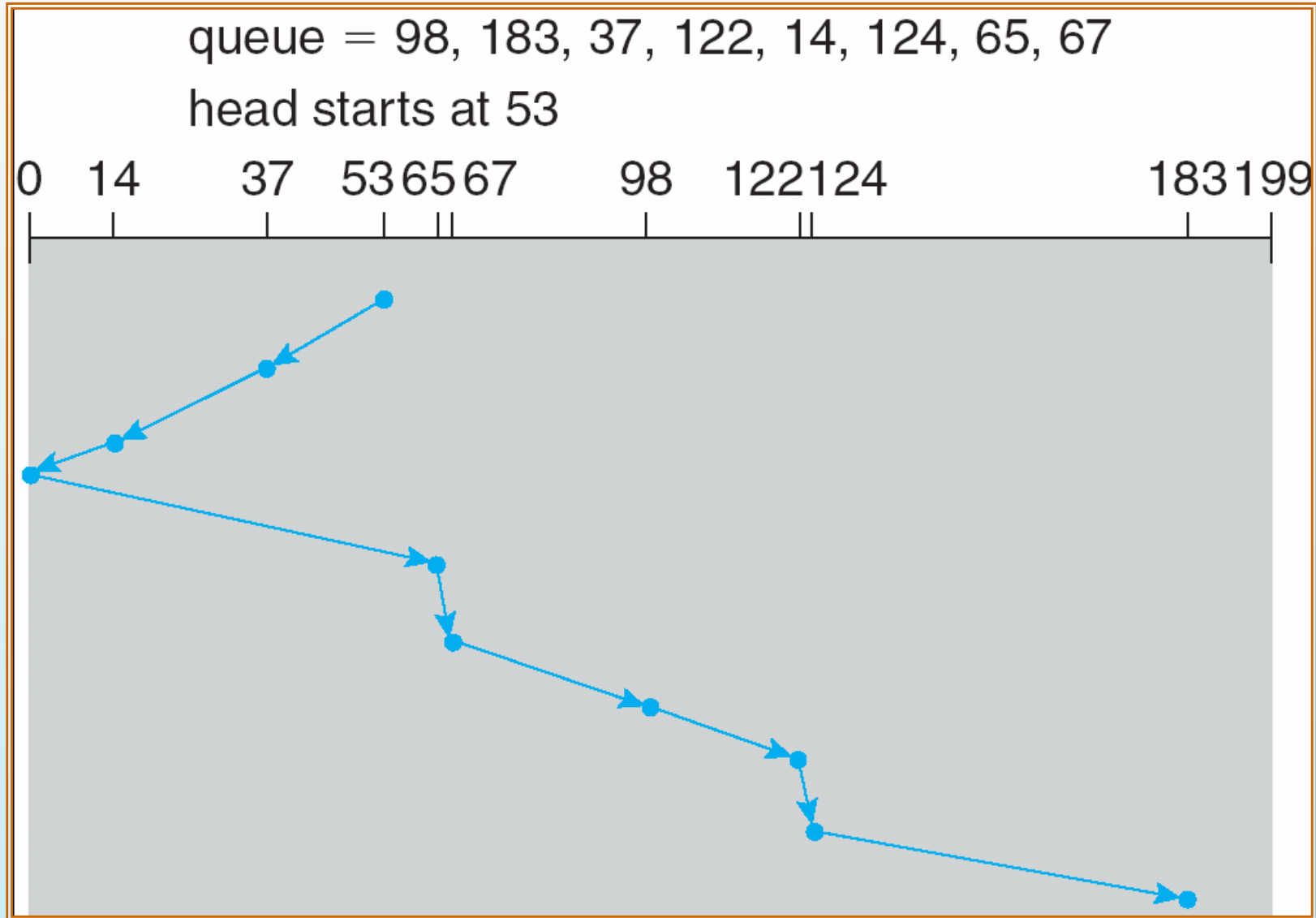
SCAN Scheduling

- 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 Disk Scheduling





C-SCAN Scheduling

- 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.

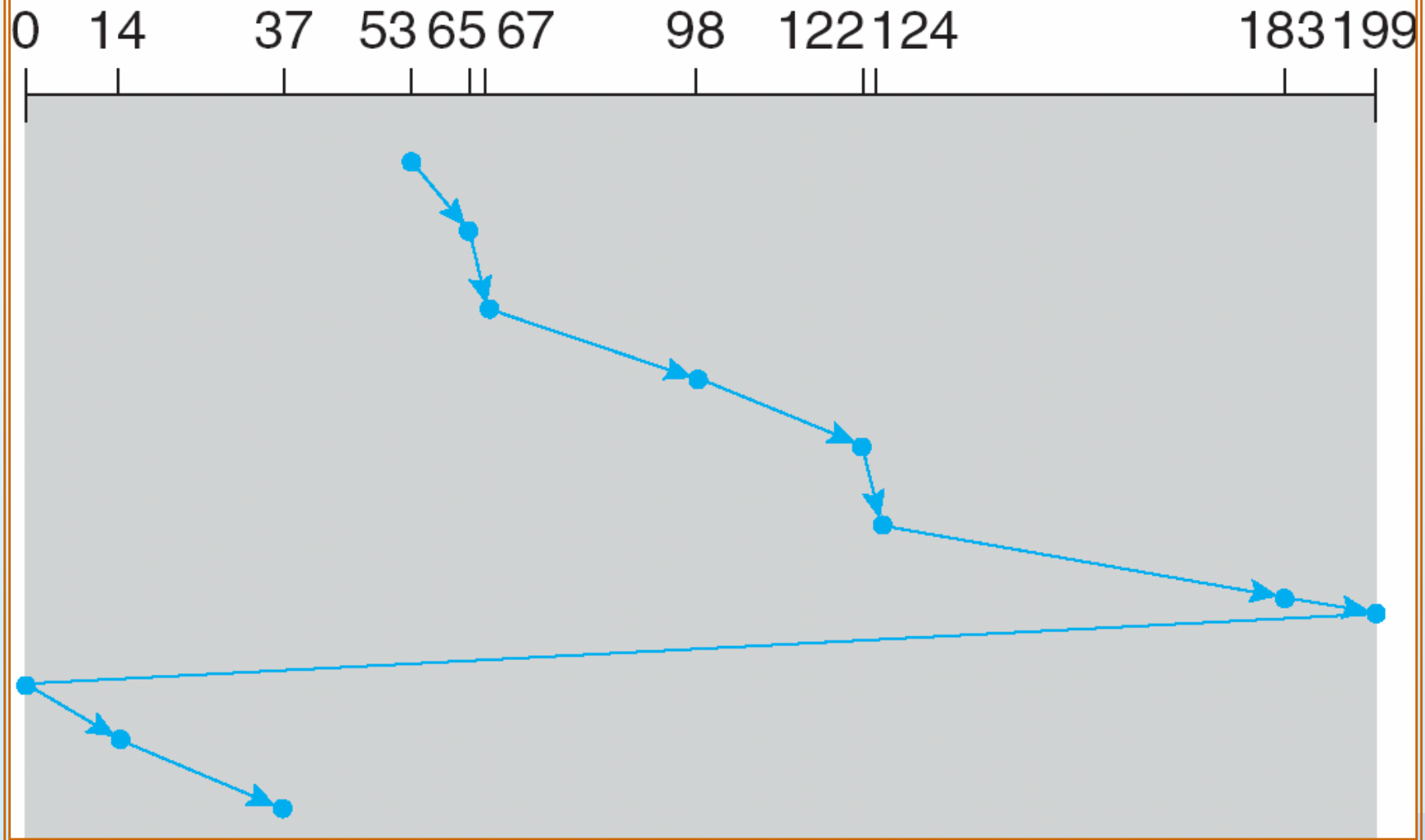




C-SCAN Disk Scheduling

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

head starts at 53





C-LOOK

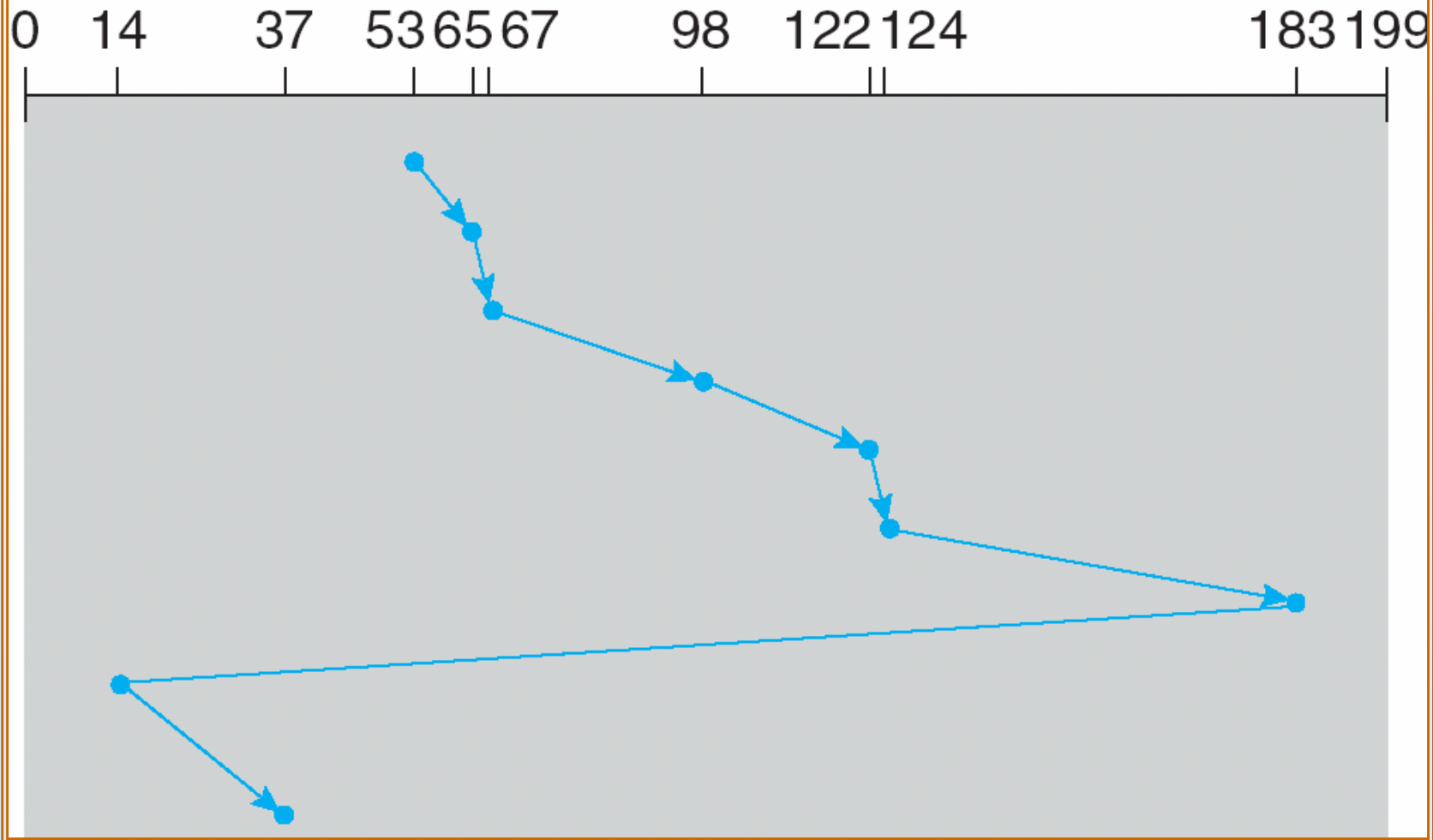
- Version of C-SCAN
- Arm only goes as far as the last request in each direction, then reverses direction immediately, without first going all the way to the end of the disk.



C-LOOK DISK Scheduling

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

head starts at 53





Selection of 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.





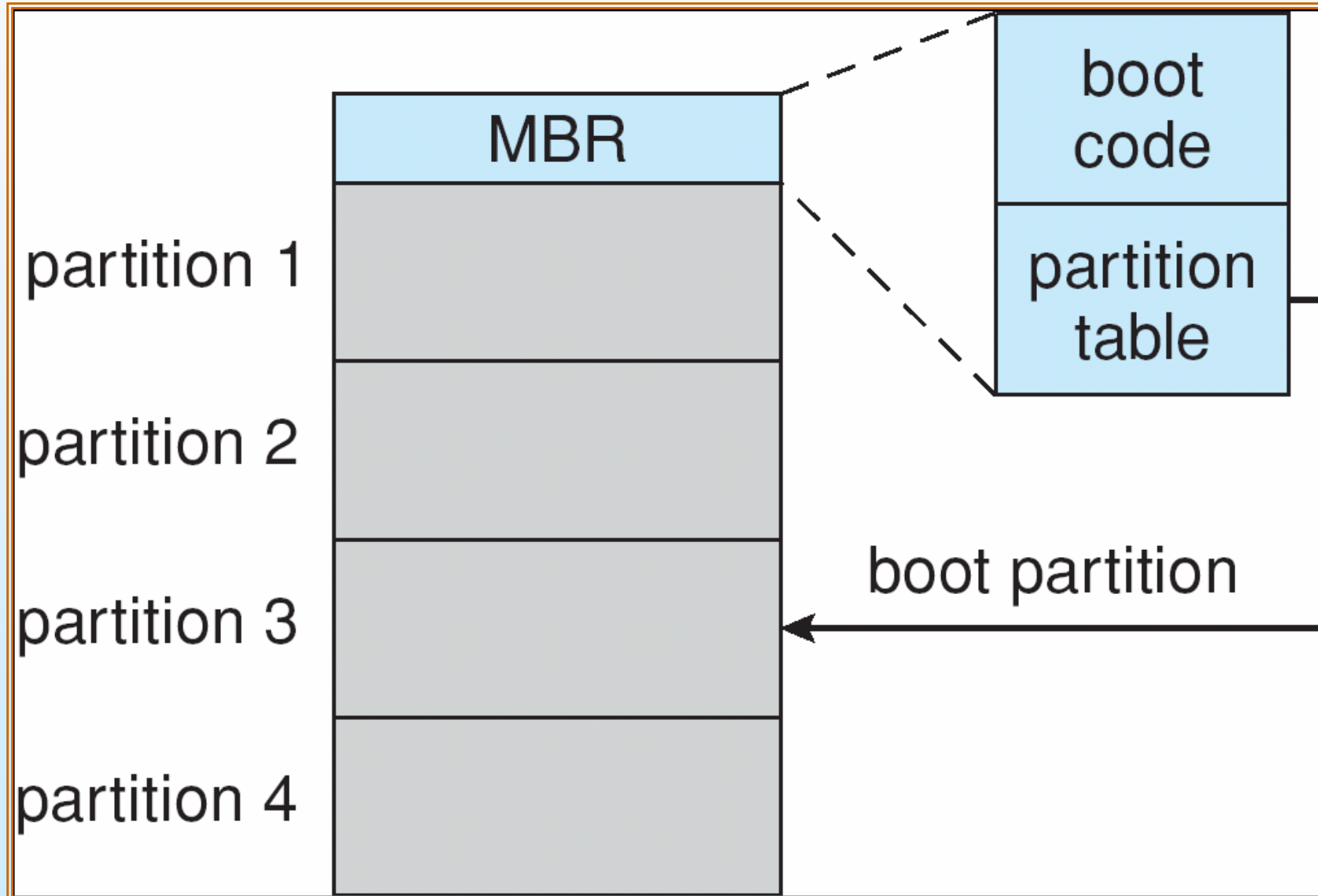
Disk Management

- *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.
 - Low-level formatting sets aside spare sectors not visible to OS
 - The controller can be told to replace each bad sector logically with a spare





Booting from Disk in Windows 2000





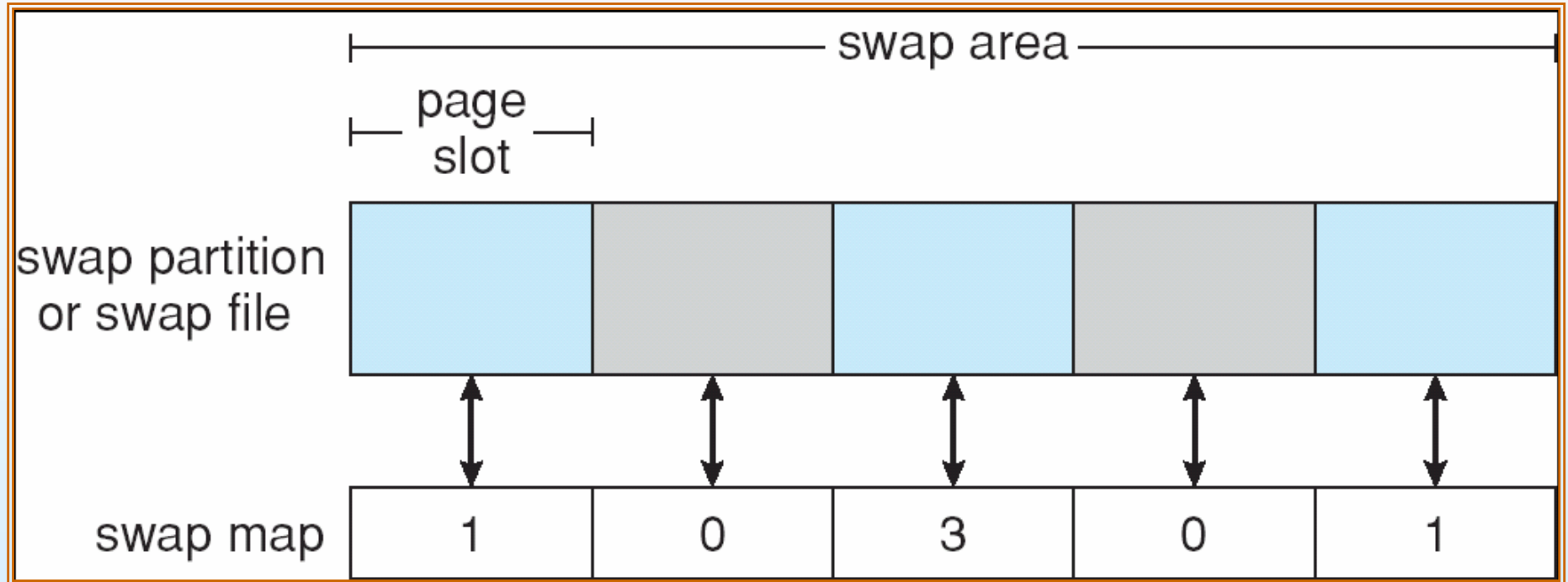
Swap-Space Management

- 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.
- 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.





The Data Structures for Swapping on Linux Systems





RAID Structure

- **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 uses a group of disks as one storage unit.
- RAID schemes improve performance and improve the reliability of the storage system by storing redundant data.
 - *Mirroring or shadowing* keeps duplicate of each disk.
 - *Block interleaved parity* uses much less redundancy.





RAID Level 0 and 1

- Level 0
 - Striping at the level of blocks but without any redundancy such as mirroring or parity bits
- Level 1
 - Disk mirroring



RAID Level 2

- Memory-style error-correcting code (ECC)
 - A parity bit for detecting all single-bit errors
 - Store two or more extra bits and can reconstruct the data if a single bit is damaged
- ECC can be used in disk arrays via striping
- If one of disks fails, the remaining bits of the byte and the associated error-correction bits can be read from other disks to reconstruct the damaged data
- RAID level 2 requires three disks' overhead for four disks of data
 - RAID level 1 requires four disks' overhead





RAID Level 3

- Known as bit-interleaved parity organization
- ECC + the fact that disk controllers can detect whether a sector can be read correctly
 - A single parity bit can be used for error detection and correction
- Idea
 - If one of sector is damaged, we know exactly which sector it is.
 - It is possible to know any bit on the sector is a 1 or 0 by computing the parity of the corresponding bits from sectors in the other disks.
 - ▶ If the parity of the remaining bits is equal to the stored parity, the missing bit is 0
 - ▶ Otherwise it is 1
- Less overhead in the expense of computing and writing the parity





RAID Level 4

- Known as block-interleaved parity organization
- Block-level striping as RAID 0 + a parity block on separate block
 - If one of disks fails, the parity block can be used with the corresponding blocks from the other disks to restore the blocks of the failed disk
- A block read accesses only one block, allowing other requests to be processed by the other disks
 - Data-transfer rate for each access is slower
 - Higher overall I/O rate because multiple read access can proceed in parallel
- Allow disks to added to a RAID seamlessly
 - If the added disks are initialized with blocks containing all zeros, the parity value does not change





RAID Level 5

- Known as block-interleaved distributed parity
- Spreading data and parity among all $N+1$ disks, rather than storing data in N disks and parity in one disk
- Avoid the potential overuse of a single parity disk that can occur with RAID 4
 - Most common parity RAID system



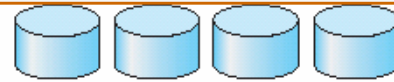


RAID Level 6

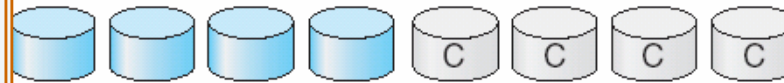
- Known as P+Q redundancy scheme
- Similar to RAID 5 but storing more extra information to guard against multiple disk failures
- For example, 2 bits of redundant data can tolerate two disk failures



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.

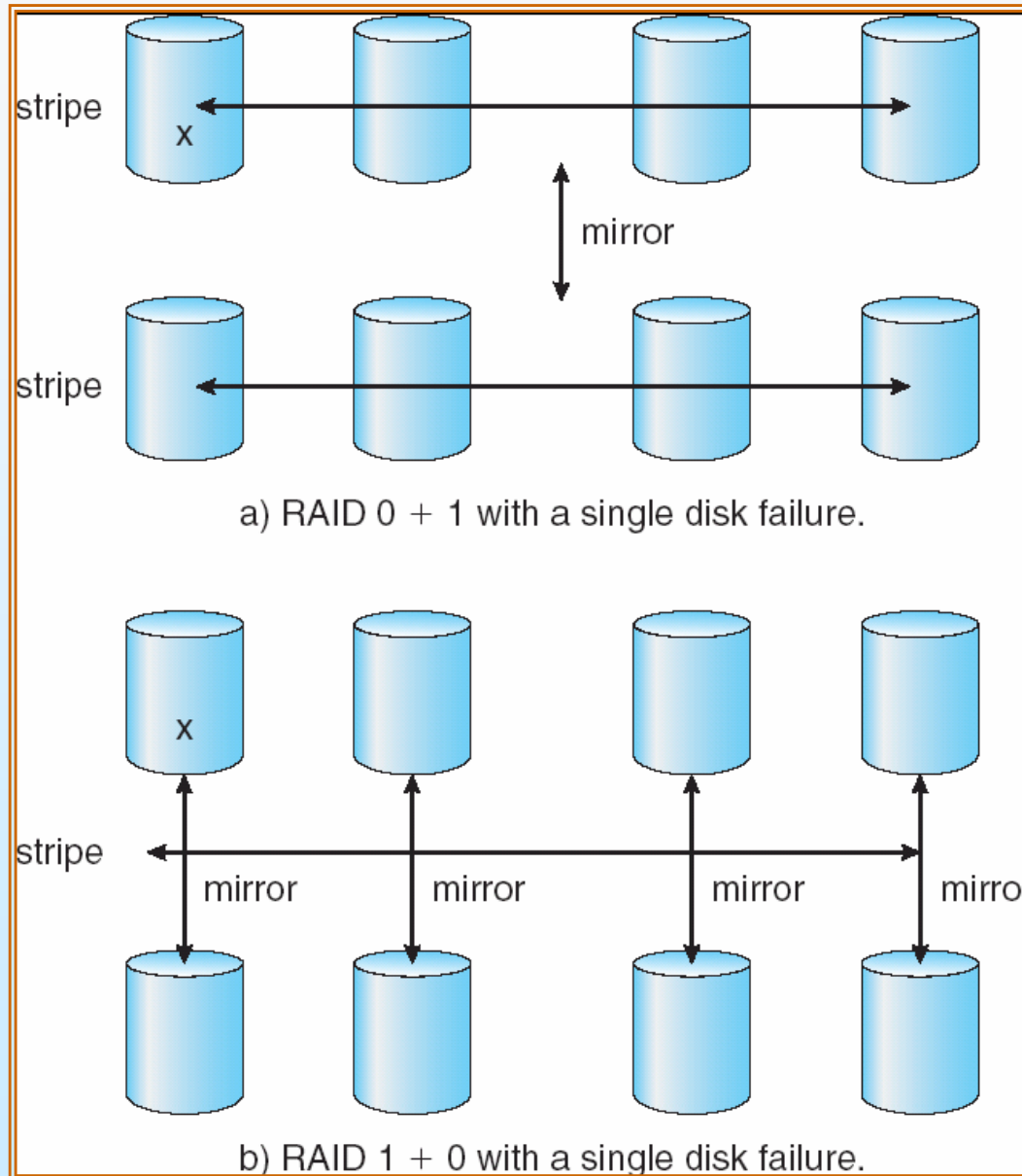


(g) RAID 6: P + Q redundancy.





RAID 0 + 1 and 1 + 0





Stable-Storage Implementation

- Write-ahead log scheme requires stable storage.
- To implement stable storage:
 - Replicate information on more than one nonvolatile storage media with independent failure modes.
 - Update information in a controlled manner to ensure that we can recover the stable data after any failure during data transfer or recovery.



Tertiary-Storage Structure

- 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.





Removable Disks (Cont.)

- A magneto-optic disk records data on a rigid platter coated with magnetic material.
 - Laser heat is used to amplify a large, weak magnetic field to record a bit.
 - Laser light is also used to read data (Kerr effect).
 - The magneto-optic head flies much farther from the disk surface than a magnetic disk head, and the magnetic material is covered with a protective layer of plastic or glass; resistant to head crashes.

- Optical disks do not use magnetism; they employ special materials that are altered by laser light.





Tapes

- Compared to a disk, a tape is less expensive and holds more data, but random access is much slower.
- Tape is an economical medium for purposes that do not require fast random access, e.g., backup copies of disk data, holding huge volumes of data.
- Large tape installations typically use robotic tape changers that move tapes between tape drives and storage slots in a tape library.
 - stacker – library that holds a few tapes
 - silo – library that holds thousands of tapes
- A disk-resident file can be *archived* to tape for low cost storage; the computer can *stage* it back into disk storage for active use.





Operating-System Support

- Major OS jobs are to manage physical devices and to present a virtual machine abstraction to applications

- For hard disks, the OS provides two abstraction:
 - Raw device – an array of data blocks.
 - File system – the OS queues and schedules the interleaved requests from several applications.





Application Interface

- Most OSs handle removable disks almost exactly like fixed disks — a new cartridge is formatted and an empty file system is generated on the disk.
- Tapes are presented as a raw storage medium, i.e., and application does not open a file on the tape, it opens the whole tape drive as a raw device.
- Usually the tape drive is reserved for the exclusive use of that application.
- Since the OS does not provide file system services, the application must decide how to use the array of blocks.
- Since every application makes up its own rules for how to organize a tape, a tape full of data can generally only be used by the program that created it.





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.





Hierarchical Storage Management

- A hierarchical storage system extends the storage hierarchy beyond primary memory and secondary storage to incorporate tertiary storage — usually implemented as a jukebox of tapes or removable disks.
- Usually incorporate tertiary storage by extending the file system.
 - Small and frequently used files remain on disk.
 - Large, old, inactive files are archived to the jukebox.
 - In some file-archiving systems, the directory entry for the file continues to exist, but the contents of the file no longer occupy space in secondary storage
 - ▶ If applications try to open the file, the `open()` system call is suspended until the file contents can be staged in from tertiary storage
- HSM is usually found in supercomputing centers and other large installations that have enormous volumes of data.





Speed

- Two aspects of speed in tertiary storage are bandwidth and latency.

- Bandwidth is measured in bytes per second.
 - Sustained bandwidth – average data rate during a large transfer; # of bytes/transfer time.
Data rate when the data stream is actually flowing.
 - Effective bandwidth – average over the entire I/O time, including **seek** or **locate**, and cartridge switching.
Drive's overall data rate.





Speed (Cont.)

- Access latency – amount of time needed to locate data.
 - Access time for a disk – move the arm to the selected cylinder and wait for the rotational latency; < 35 milliseconds.
 - Access on tape requires winding the tape reels until the selected block reaches the tape head; tens or hundreds of seconds.
 - Generally say that random access within a tape cartridge is about a thousand times slower than random access on disk.
- The low cost of tertiary storage is a result of having many cheap cartridges share a few expensive drives.
- A removable library is best devoted to the storage of infrequently used data, because the library can only satisfy a relatively small number of I/O requests per hour.





Reliability

- A fixed disk drive is likely to be more reliable than a removable disk or tape drive.
- An optical cartridge is likely to be more reliable than a magnetic disk or tape.
- A head crash in a fixed hard disk generally destroys the data, whereas the failure of a tape drive or optical disk drive often leaves the data cartridge unharmed.





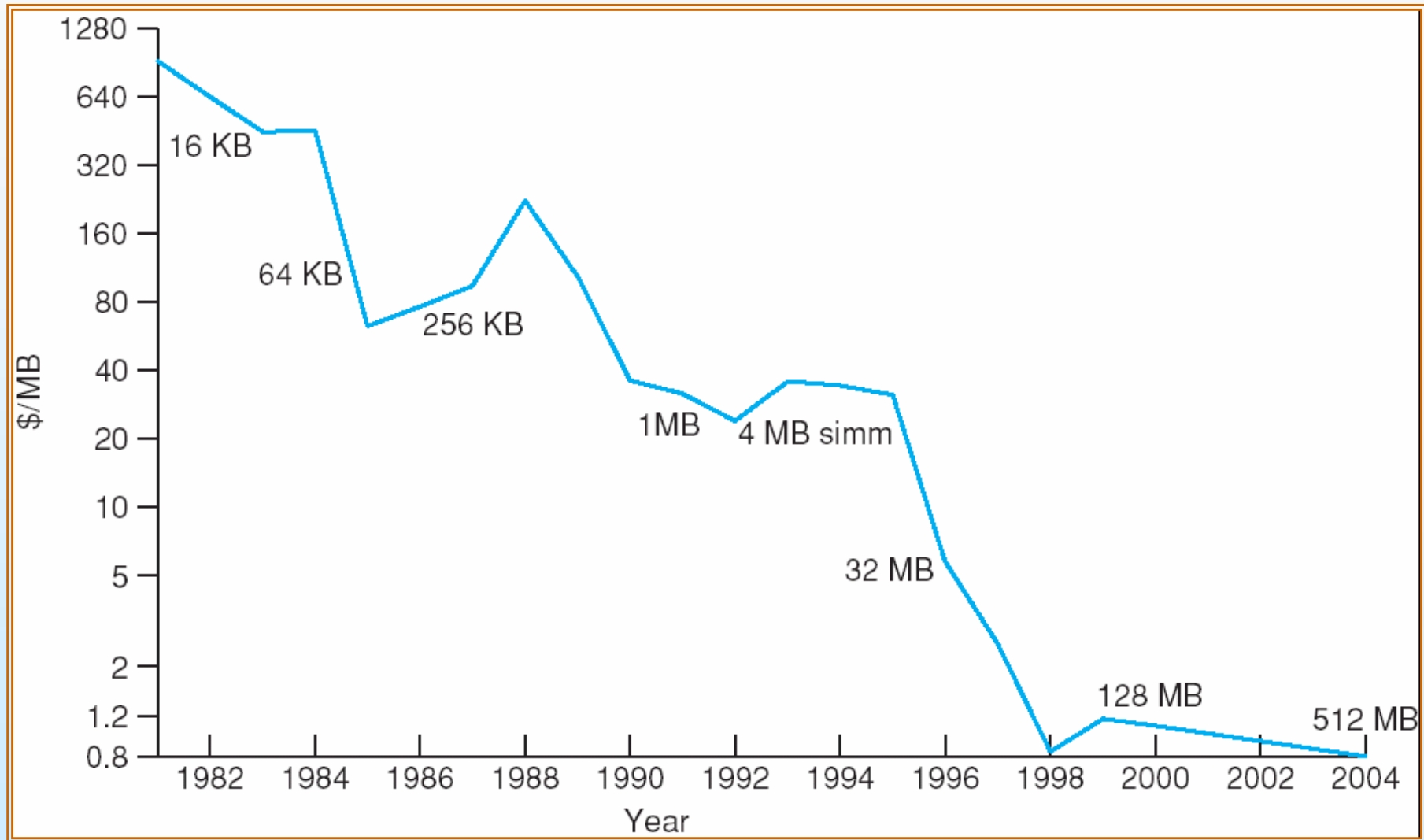
Cost

- 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.
- Tertiary storage gives a cost savings only when the number of cartridges is considerably larger than the number of drives.



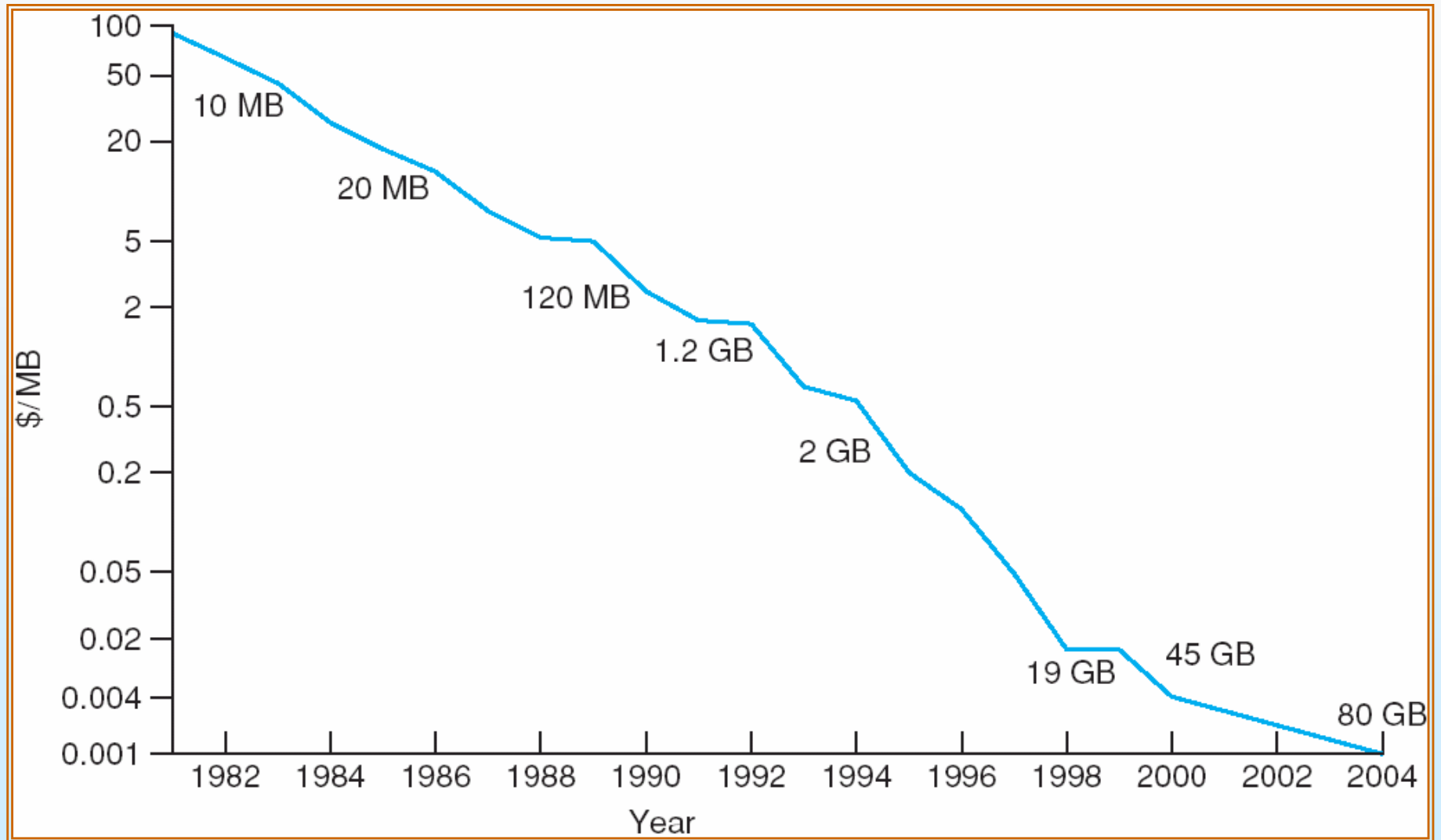


Price per Megabyte of DRAM, From 1981 to 2004



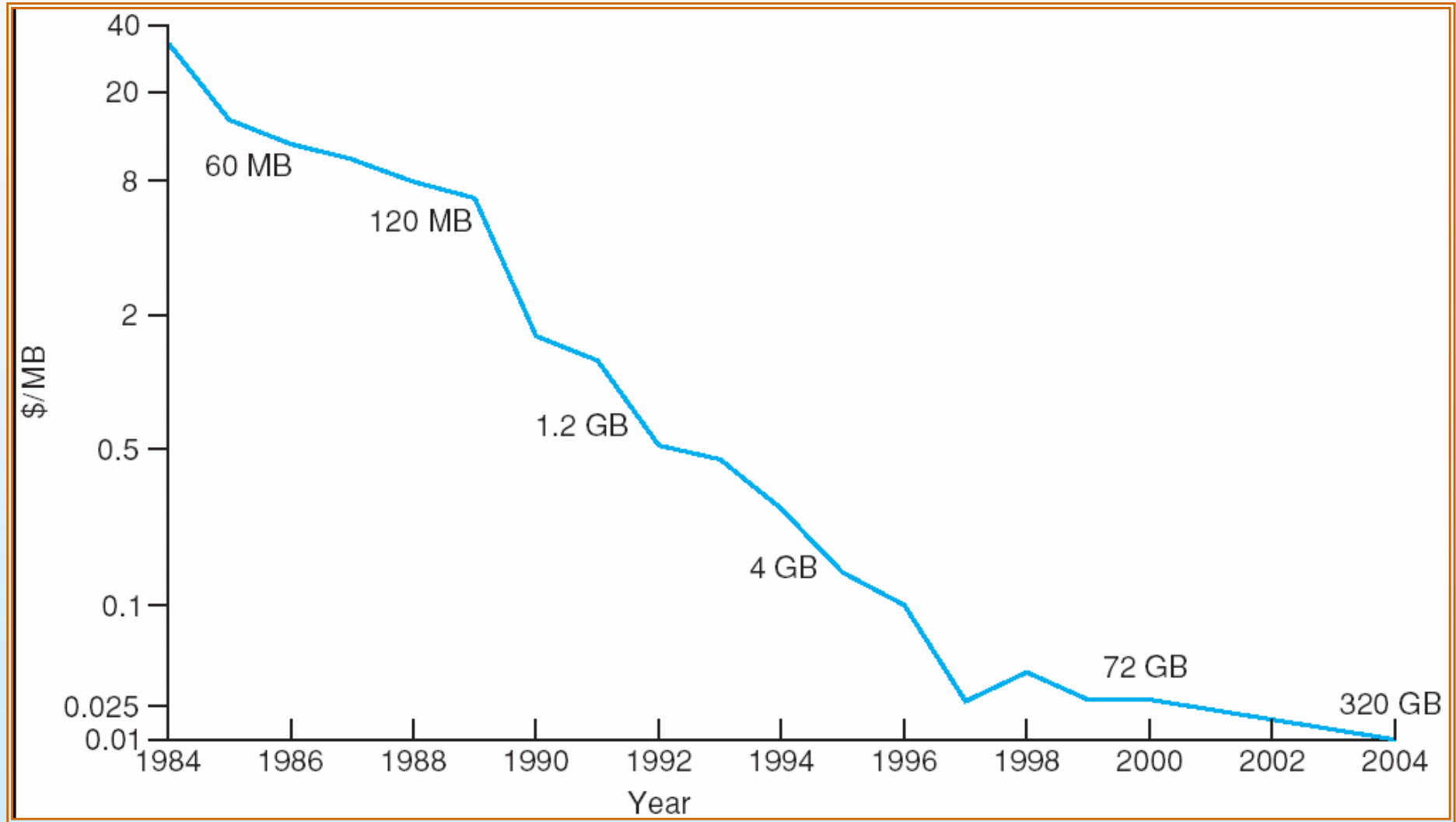


Price per Megabyte of Magnetic Hard Disk, From 1981 to 2004





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



End of Chapter 12

