Unit 5
I/O management and Introduction to Linux

Types of I/O devices:

- **Input Devices**

Input devices are used to provide data and control signals to a computer. Some common input devices include:

Keyboard: Used for entering text and commands.

Mouse: A pointing device for navigating and interacting with the computer interface.

Joystick: Often used for gaming, it allows the user to control movement within a program.

Scanner: Converts physical documents into digital format.

Microphone: Captures audio and converts it into digital data.

Webcam: Captures video and images.

Types of I/O devices:

- **Output Devices**
- Output devices receive data from the computer and present it to the user. Common output devices are:
- **Monitor**: Displays visual output from the computer.
- **Printer**: Produces hard copies of digital documents.
- **Speakers**: Output sound from the computer.
- **Headphones**: Personal audio output device.

- **Storage Devices**
- Output devices receive data from the computer and present it to the user. Common output devices are:
- **Monitor**: Displays visual output from the computer.
- **Printer**: Produces hard copies of digital documents.
- **Speakers**: Output sound from the computer.
- **Headphones**: Personal audio output device.
- These devices store data processed by the computer:
- **Hard Disk Drives (HDD)**: Store large amounts of data persistently.
- **Solid State Drives (SSD)**: Faster storage alternative to HDDs.
- **USB Flash Drives**: Portable storage devices.

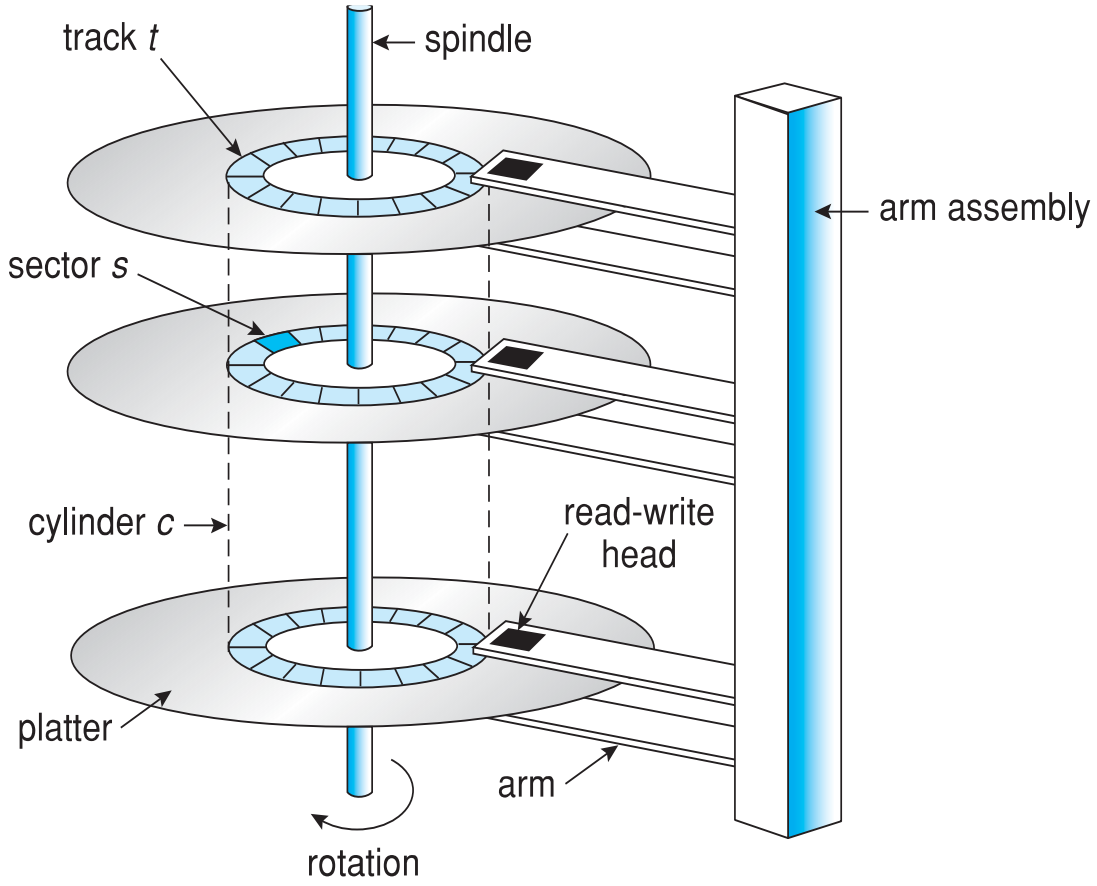
I/O Management

The First Commercial Disk Drive

1956
IBM RAMDAC computer included the IBM Model 350 disk storage system

5M (7 bit) characters
50 x 24” platters
Access time = < 1 second

Moving-head Disk Mechanism



Overview of Mass Storage Structure

- **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 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**)
- Disks can be removable
- Drive attached to computer via **I/O bus**
  - Busses vary, including **EIDE**, **ATA**, **SATA**, **USB**, **Fibre Channel**, **SCSI, SAS, Firewire**
  - **Host controller** in computer uses bus to talk to **disk controller** built into drive or storage array

Disk Scheduling

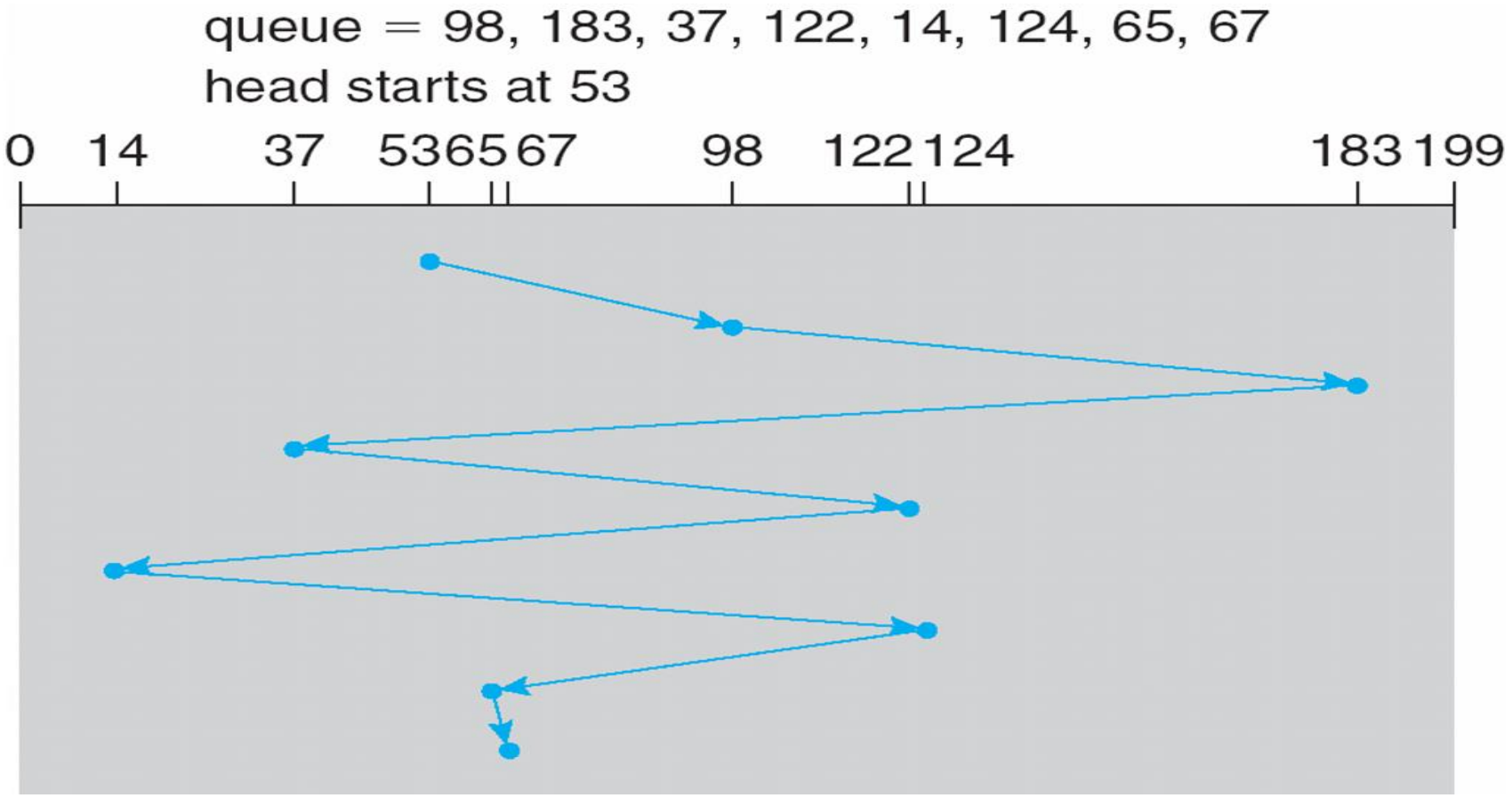
- The operating system is responsible for using hardware efficiently — for the disk drives, this means having a fast access time and disk bandwidth
- Minimize seek time
- Seek time ≈ seek distance
- Disk **bandwidth** is the total number of bytes transferred, divided by the total time between the first request for service and the completion of the last transfer

Disk Scheduling (Cont.)

- There are many sources of disk I/O request
  - OS
  - System processes
  - Users processes
- I/O request includes input or output mode, disk address, memory address, number of sectors to transfer
- OS maintains queue of requests, per disk or device
- Idle disk can immediately work on I/O request, busy disk means work must queue

FCFS

Illustration shows total head movement of 640 cylinders

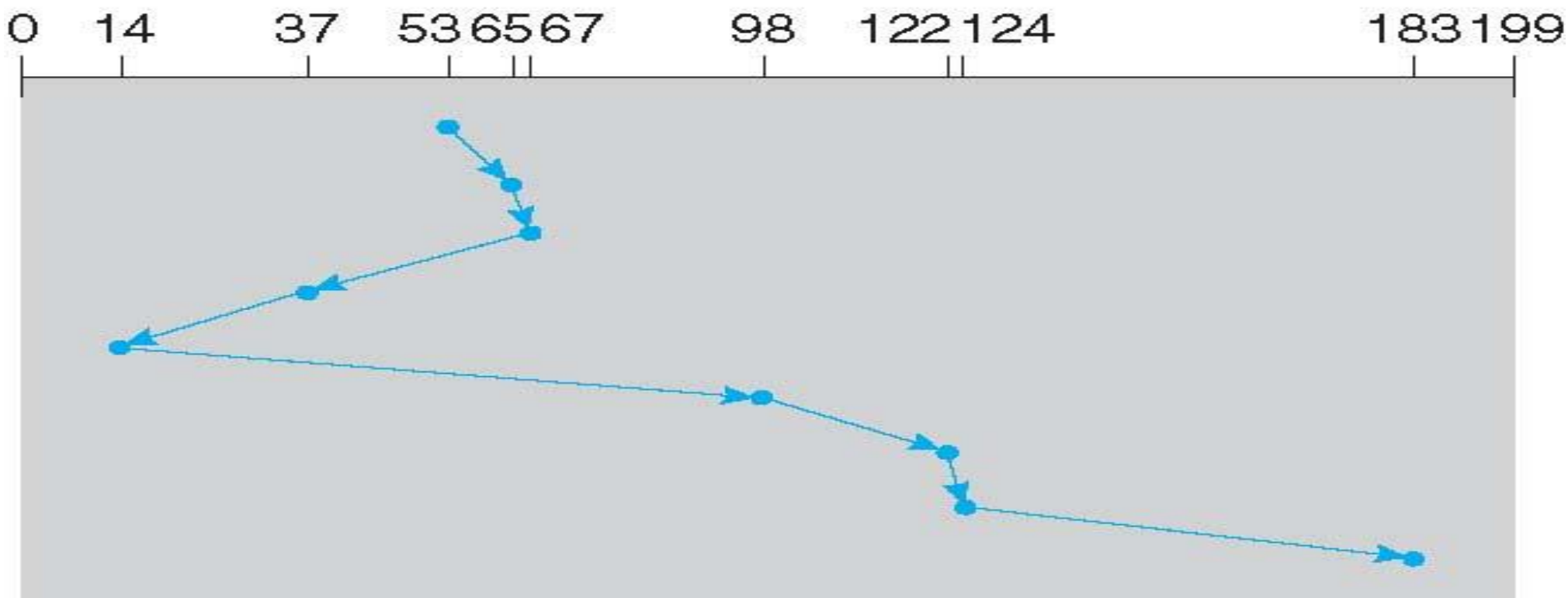


SSTF

- Shortest Seek Time First selects the request with the minimum seek time from the current head position
- SSTF scheduling is a form of SJF scheduling
- Illustration shows total head movement of 236 cylinders

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

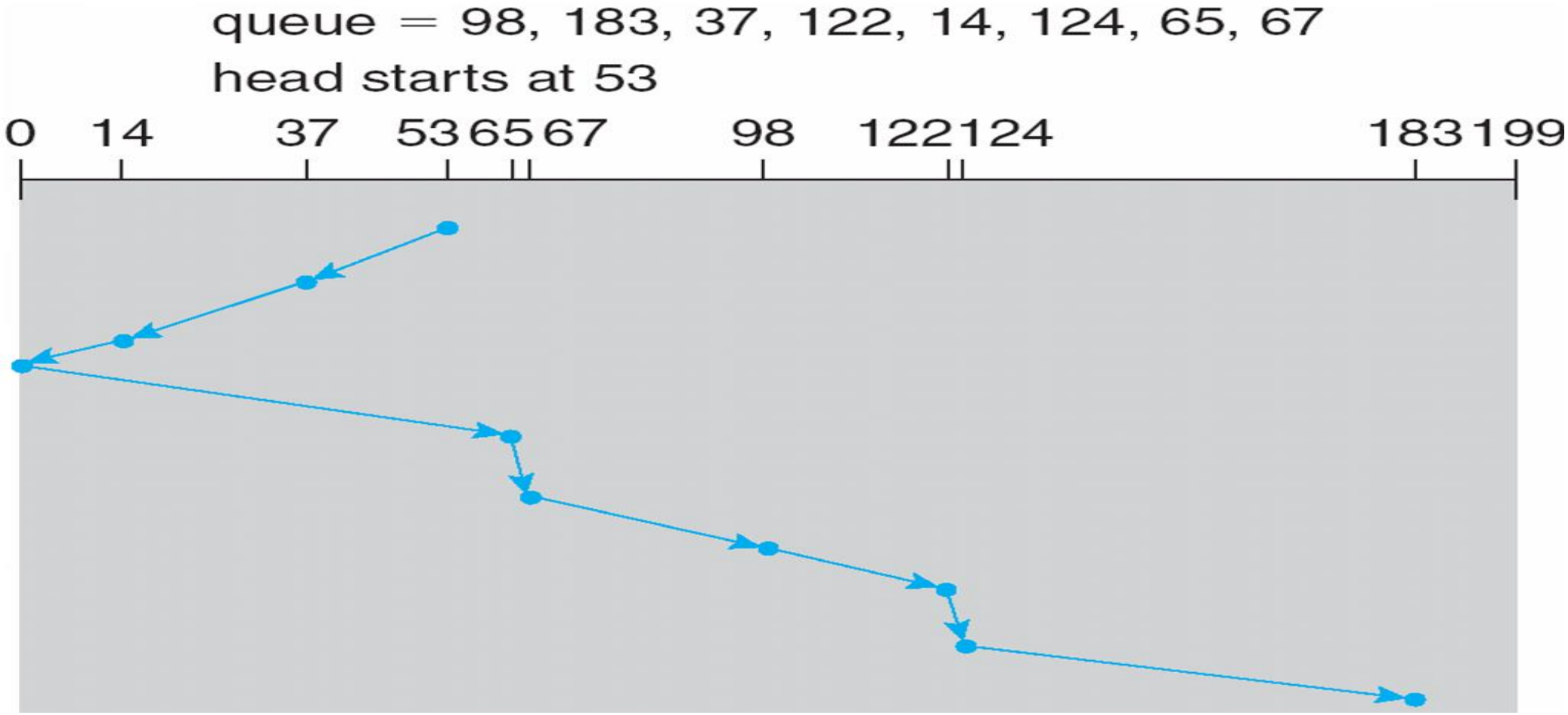
head starts at 53



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.
- **SCAN algorithm** Sometimes called the **elevator algorithm**
- Illustration shows total head movement of 236 cylinders

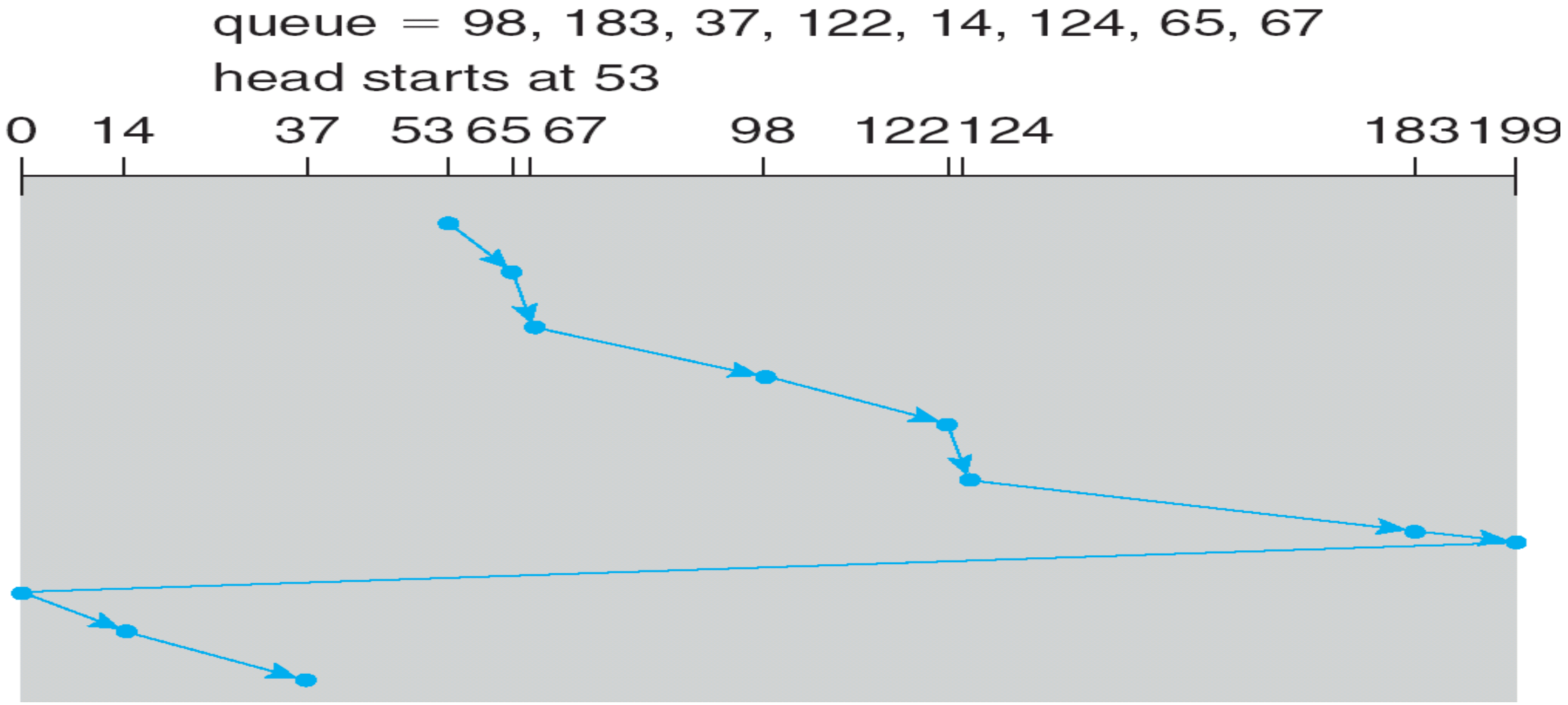
SCAN (Cont.)


queue = 98, 183, 37, 122, 14, 124, 65, 67
head starts at 53
0
14
37
53
65
67
98
122
124
183
199

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

C-SCAN (Cont.)


queue = 98, 183, 37, 122, 14, 124, 65, 67
head starts at 53
0
14
37
53
65
67
98
122
124
183
199

Introduction to Linux

Linux File Systems

- To the user, Linux's file system appears as a hierarchical directory tree obeying UNIX semantics
- Internally, the kernel hides implementation details and manages the multiple different file systems via an abstraction layer, that is, the virtual file system (VFS)
- The Linux VFS is designed around object-oriented principles and is composed of four components:
  - A set of definitions that define what a file object is allowed to look like
    - The **inode object** structure represent an individual file
    - The **file object** represents an open file
    - The **superblock object** represents an entire file system
    - A **dentry object** represents an individual directory entry

File Systems (Cont.)

- To the user, Linux's file system appears as a hierarchical directory tree obeying UNIX semantics
- Internally, the kernel hides implementation details and manages the multiple different file systems via an abstraction layer, that is, the virtual file system (VFS)
- The Linux VFS is designed around object-oriented principles and layer of software to manipulate those objects with a set of operations on the objects
  - For example for the file object operations include (from struct file_operations in /usr/include/linux/fs.h

    int open(. . .) — Open a file

    ssize t read(. . .) — Read from a file

    ssize t write(. . .) — Write to a file

    int mmap(. . .) — Memory-map a file

Types of Linux File System

1) ext (Extended File System):

Implemented in 1992, it is the first file system specifically designed for Linux. It is the first member of the ext family of file systems.

2) ext2:

The second ext was developed in 1993. It is a non-journaling file system that is preferred to be used with flash drives and SSDs. It solved the problems of separate timestamp for access, inode modification and data modification. Due to not being journaled, it is slow to load at boot time.

3) Xiafs:

Also developed in 1993, this file system was less powerful and functional than ext2 and is no longer in use anywhere.

4) ext3:

The third ext developed in 1999 is a journaling file system. It is reliable and unlike ext2, it prevents long delays at system boot if the file system is in an inconsistent state after an unclean shutdown. Other factors that make it better and different than ext2 are online file system growth and HTree indexing for large directories.

5) JFS (Journaled File System):

First created by IBM in 1990, the original JFS was taken to open source to be implemented for Linux in 1999. JFS performs well under different kinds of load but is not commonly used anymore due to the release of ext4 in 2006 which gives better performance.

6) ReiserFS:

It is a journal file system developed in 2001. Despite its earlier issues, it has tail packing as a scheme to reduce internal fragmentation. It uses a B+ Tree that gives less than linear time in directory lookups and updates. It was the default file system in SUSE Linux till version 6.4, until switching to ext3 in 2006 for version 10.2.

7) XFS:

XFS is a 64-bit journaling file system and was ported to Linux in 2001. It now acts as the default file system for many Linux distributions. It provides features like snapshots, online defragmentation, sparse files, variable block sizes, and excellent capacity. It also excels at parallel I/O operations.

8) SquashFS:

Developed in 2002, this file system is read-only and is used only with embedded systems where low overhead is needed.

9) Reiser4:

It is an incremental model to ReiserFS. It was developed in 2004. However, it is not widely adapted or supported on many Linux distributions.

10) ext4:

The fourth ext developed in 2006, is a journaling file system. It has backward compatibility with ext3 and ext2 and it provides several other features, some of which are persistent pre-allocation, unlimited number of subdirectories, metadata checksumming and large file size. ext4 is the default file system for many Linux distributions and also has compatibility with Windows and Macintosh.

11) btrfs (Better/Butter/B-tree FS):

It was developed in 2007. It provides many features such as snapshotting, drive pooling, data scrubbing, self-healing and online defragmentation. It is the default file system for Fedora Workstation.

12) bcachefs:

This is a copy-on-write file system that was first announced in 2015 with the goal of performing better than btrfs and ext4. Its features include full filesystem encryption, native compression, snapshots, and 64-bit check summing.

Features of Linux operating System

- Open Source: It is an open source operating system. It is easily available for everyone and user can use Linux free without license.
- Free: It is free to use this makes it available to everyone regardless of the financial limitations.
- Security: It is well known for its security. It is free from various virus attacks.
- Light weighted infrastructure: It does not required a lots of system resource it required 700 Mhz, 512mb RAM and 5gb storage.
- Multi- User and multiprogramming OS: Multiple user can access system resources like memory, ram application programs at the same time also multiple application can run at the same time.

- Stability: It provide high stability.
- Graphical User Interface: Linux is the command line Interface OS but it also provide interactive graphical user interface likes windows .
- Performance : The performance of the Linux system is much higher than other operating systems.
- Portability: Linux is designed to be highly portable and can run on a wide range of hardware including desktop, computer ,servers, supercomputer and embedded devices.
- Software Update: The software update in Linux are easy and frequent.

- **Communication Devices**
- These facilitate data transmission between computers:
- **Network Adapters**: Connect computers to a network.
- **Modems**: Enable communication over telephone lines.
- **Bluetooth Adapters/Dongles**: Allow wireless communication with other devices.