Dr. Vishwanath Karad
MIT WORLD PEACE UNIVERSITY | PUNE
TECHNOLOGY, RESEARCH, SOCIAL INNOVATION & PARTNERSHIPS

CS223 Operating Systems

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Syllabus

File Management: Overview, File Organization and Access, File Directories, File Sharing, Record Blocking.

I/O Management: I/O Devices, Organization of the I/O Functions, I/O Buffering, Disk Scheduling.

File Management

Overview

- Files are the central element to most applications
- Desirable properties of files:
  - Long-term existence
  - Sharable between processes
  - Structure

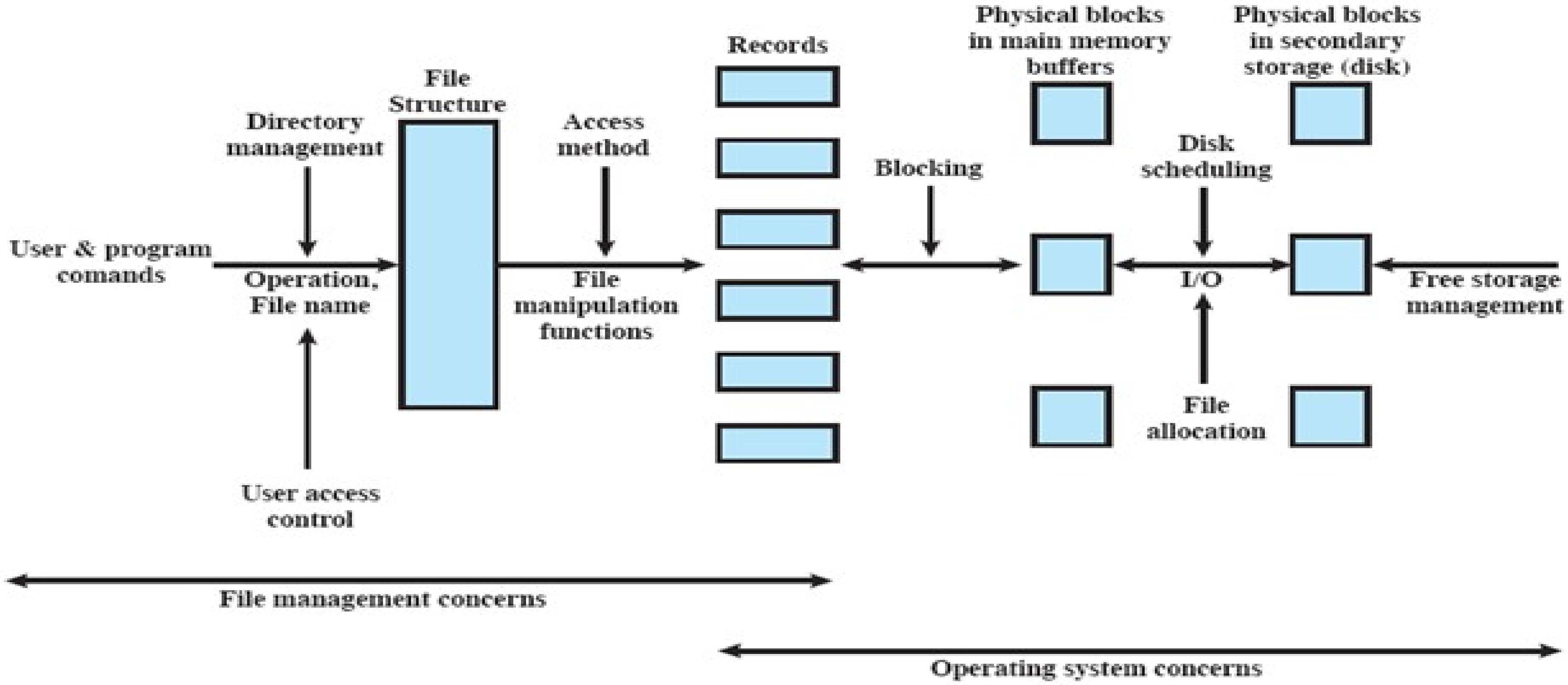
Terms in common use when discussing files

1. Fields
   - Basic element of data
   - Contains a single value
   - Characterized by its length and data type
2. Records
   - Collection of related fields
   - Treated as a unit
3. File
   - Have file names
   - Is a collection of similar records
   - Treated as a single entity
   - May implement access control mechanisms
4. Database
   - Collection of related data
   - Relationships exist among elements
   - Consists of one or more files

File System

- The File System is one of the most important part of the OS to a user
- Concerned with secondary storage
- File systems also provide functions which can be performed on files:
  - Create
  - Delete
  - Open
  - Close
  - Read
  - Write

Elements of File Management



File Organization

- Refers to the logical structuring of records
- Determined by the ***way*** in which files are accessed
- Important criteria while choosing a file organization:
  - Short access time
  - Ease of update
  - Economy of storage
  - Simple maintenance
  - Reliability

File Organization Types

Many exist, but usually variations of:

- Pile
- Sequential file
- Indexed file
- Direct or hashed file

File Organization

Pile

- Data are collected in the order they arrive
- No structure
- Purpose is to accumulate a mass of data and save it
- Records may have different fields
- Record access is by exhaustive search

Sequential File

- Fixed format used for records
- Records are the same length
- Key field
  - Uniquely identifies the record
  - Records are stored in key sequence

File Organization

Indexed File

- Maintains the key characteristic of the sequential file: records are organized in sequence based on a key field
  - An index is added to the file to support random access
  - Can use multiple indexes for different key fields
  - May contain an exhaustive index that contains one entry for every record in the main file
  - May contain a partial index
  - When a new record is added to the main file, all of the index files must be updated

Direct or Hash File

- Directly access a block at a known address
- Key field required for each record
- But there is no concept of sequential ordering
- Makes use of hashing on the key value

File Directories

- Contains information about files
  - Attributes
  - Location
  - Ownership
- Provides mapping between file names and the files themselves
- A directory system should support a number of operations including:
  - Search
  - Create files
  - Deleting files
  - Listing directory
  - Updating directory

Directory Elements

Basic Information

- **File Name**

-Name as chosen by creator

-Must be unique within a specific directory

Address Information

- **Volume** -Indicates device on which file is stored
- **Starting Address**
- **Size Used -**Current size of the file in bytes, words, or blocks
- **Size Allocated -** Maximum size of the file

Access Control Information

- **Owner**

  -Owner has control of this file & able to grant/deny access to other users and to change these privileges.

- **Access Permission**

-Specifies read, write & execute permissions on the file for owner, group & others

Directory Elements: Usage Information

- Date Created
- Identity of Creator
- Date Last Read Access
- Identity of Last Reader
- Date Last Modified
- Identity of Last Modifier
- Date of Last Backup
- Current Usage

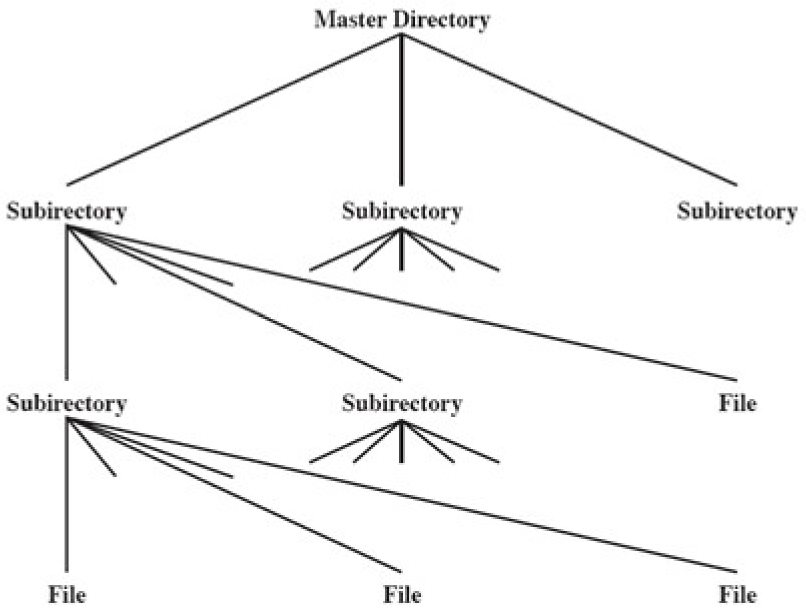
Hierarchical or Tree-Structured Directory

Master directory with user directories underneath it

Each user directory may have subdirectories and files as entries

Naming

- Users need to be able to refer to a file by name
- Files need to be named uniquely
- Tree structure allows users to find a file by following the directory path
- Duplicate filenames are possible if they have different pathnames



File Sharing

- In multiuser system, allow files to be shared among users
- Two issues
  - Access rights
  - Management of simultaneous access

    -User may lock entire file when it is to be updated

    -User may lock the individual records during the update

    -Mutual exclusion and deadlock are issues for shared access

User Classes

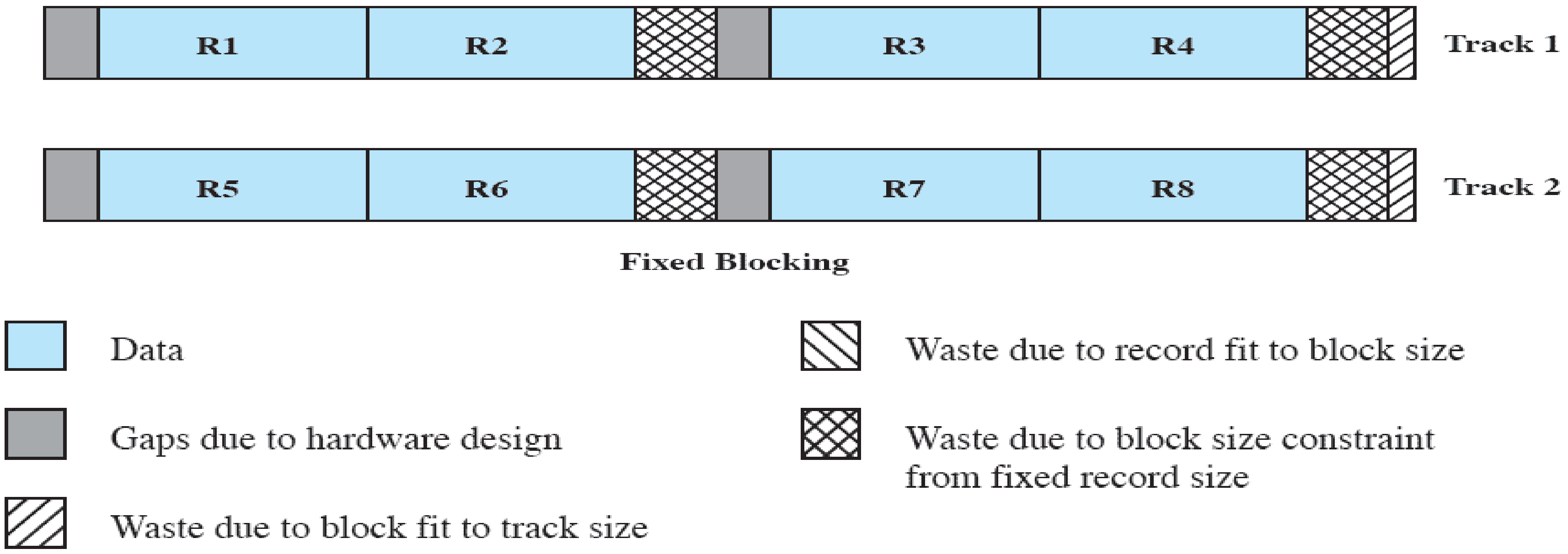
- Owner - Usually the files creator, has full rights
- User Groups - A set of users identified as a group
- Others

Record Blocking

- Records are the logical unit of access of a structured file
  - But blocks are the unit for I/O with secondary storage
- Three approaches are common
  - Fixed length blocking
  - Variable length spanned blocking
  - Variable-length unspanned blocking

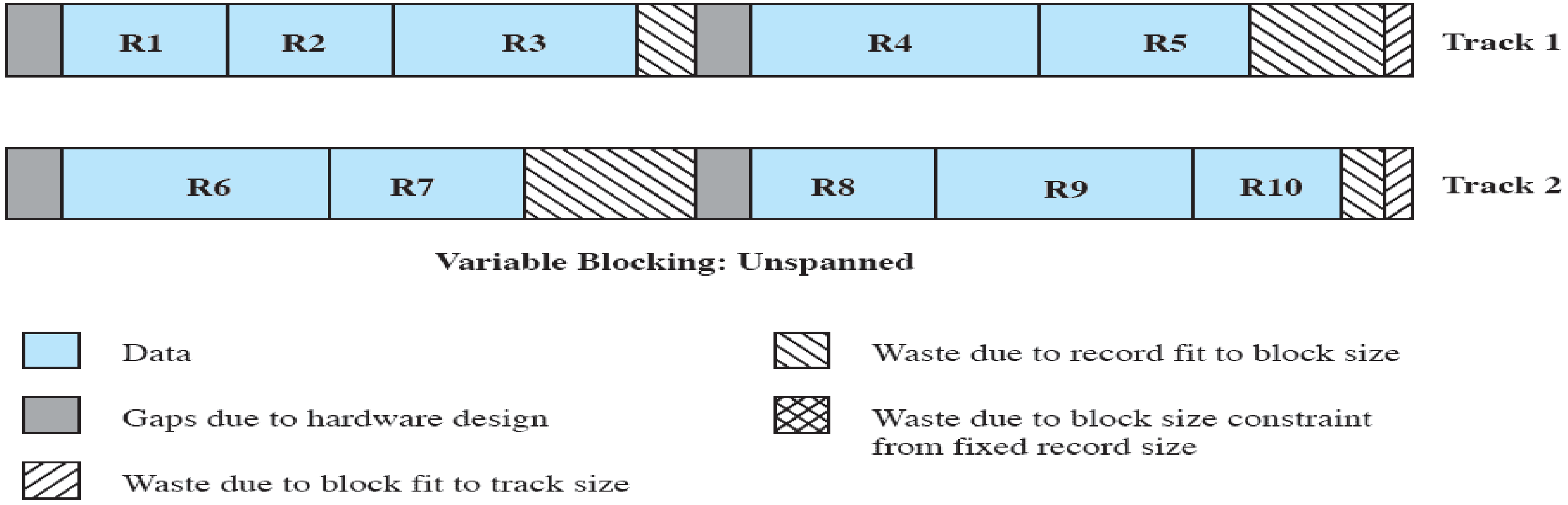
Fixed Blocking

- Fixed-length records are used
- Unused space at the end of a block is ***internal fragmentation***



Variable-length un-spanned blocking

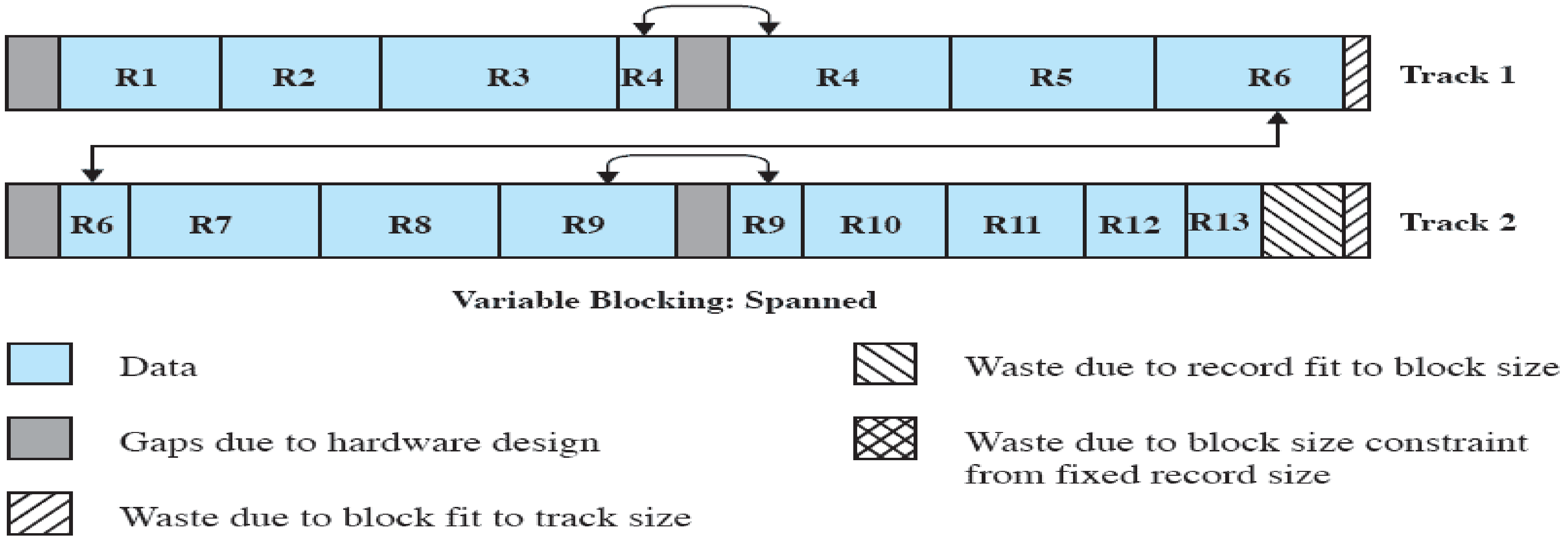
- Uses variable length records without spanning
- Wasted space in most blocks because of the inability to use the remainder of a block if the next record is larger than the remaining unused space



Variable Blocking: Unspanned

Variable Length Spanned Blocking

- Variable-length records are used
- Some records may span multiple blocks
- Continuation is indicated by a pointer to the successor block



Variable Blocking: Spanned

I/O Management

- I/O Devices
- Organization of the I/O Function
- I/O Buffering
- Disk Scheduling

Categories of I/O Devices

Three Categories:

1. Human readable

- Devices used to communicate with the user
  - Video display, Keyboard, Mouse, Printer, etc

2. Machine readable

- Used to communicate with electronic equipment
  - Disk drives, Sensors, Controllers, Actuators, etc

3. Communications

- Used to communicate with remote devices
  - Modems, Digital line drivers, etc

Differences in I/O Devices

- Devices differ in a number of areas
  - Data Rate - Massive difference between the data transfer rates of devices
  - Application
  - Complexity of Control - Complexity of the I/O module that controls the device. Eg. Disk is much more complex as compared to printer
  - Unit of Transfer - Data may be transferred as a stream of bytes or characters (e.g., terminal I/O) or in larger blocks (e.g., disk I/O).
  - Data Representation - Different data encoding schemes are used by different devices
  - Error Conditions - nature of errors differ widely from one device to another

Organization of the I/O Function

Techniques for performing I/O

- Programmed I/O
- Interrupt-driven I/O
- Direct memory access (DMA)

	No Interrupts	**Use of Interrupts**
I/O-to-memory transfer through processor	Programmed I/O	Interrupt-driven I/O
Direct I/O-to-memory transfer		Direct memory access (DMA)

Direct Memory Address

- Processor delegates I/O operation to the DMA module
- DMA module transfers data directly to or form memory
- When complete, DMA module sends an interrupt signal to the processor

I/O Buffering

- Processes must wait for I/O to complete before proceeding
- It may be more efficient to perform input transfers in advance of requests being made and to perform output transfers some time after the request is made.

Block-oriented Buffering

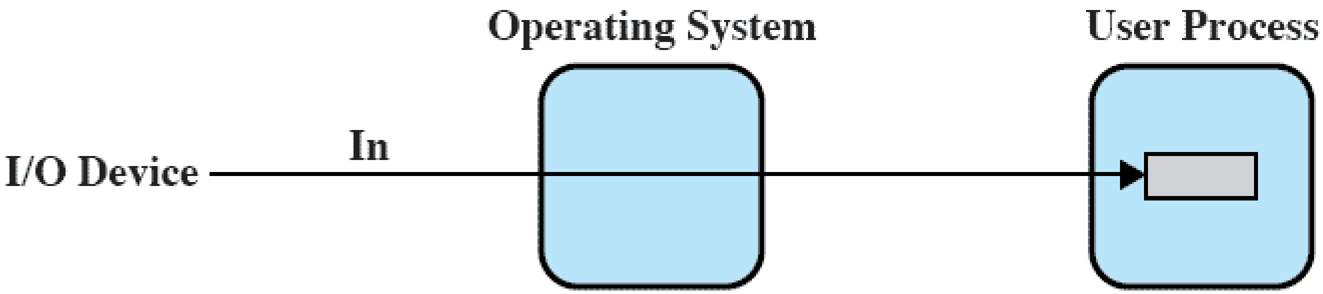
- Information is stored in fixed sized blocks
- Transfers are made a block at a time Eg. Used for disks

Stream-Oriented Buffering

- Transfer information as a stream of bytes
- Used for terminals, printers, communication ports, mouse, etc.

No Buffer

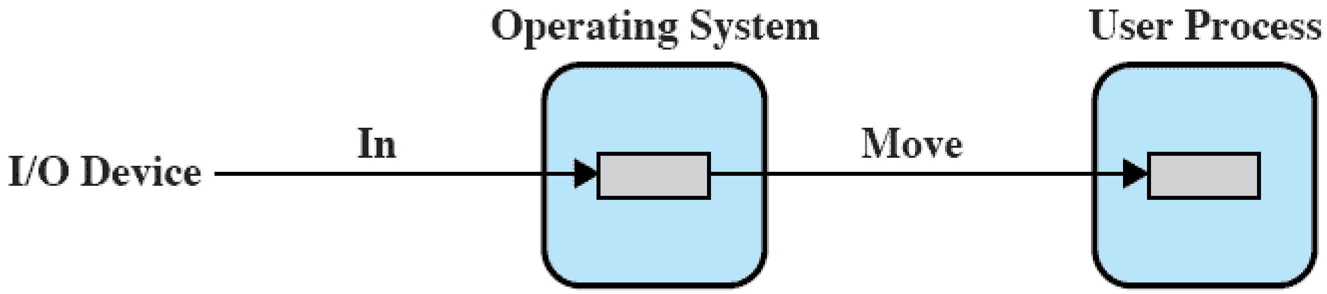
Without a buffer, the OS directly access the device as and when it needs



(a) No buffering

Single Buffer

Operating system assigns a buffer in main memory for an I/O request

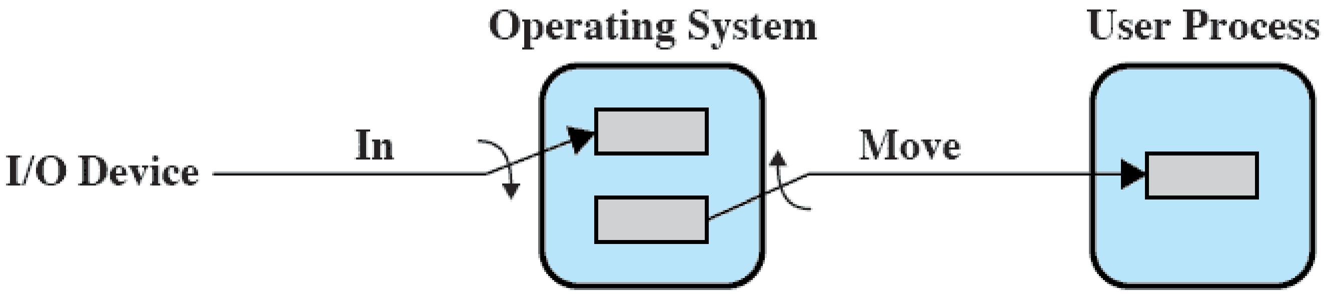


(b) Single buffering

Double Buffer

Use two system buffers instead of one

A process can transfer data to or from one buffer while the operating system empties or fills the other buffer



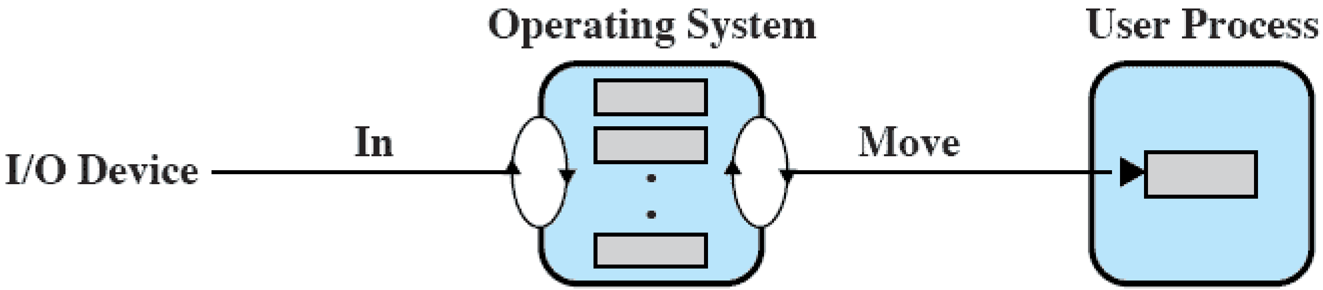
(c) Double buffering

Circular Buffer

More than two buffers are used

Each individual buffer is one unit in a circular buffer

Used when I/O operation must keep up with process



(d) Circular buffering

Buffer Limitations

- Buffering smoothens out peaks in I/O demand
- But with enough demand eventually all buffers become full and their advantage is lost
- Buffering can increase the efficiency of the OS and the performance of individual processes.

Disk Scheduling

- When the disk drive is operating, the disk is rotating at constant speed
- Positioning the Read/Write Head
- Track selection involves moving the head to a specific track
- **Disk Performance Parameters**

  -***Access Time*** is the sum of:
  - ***Seek time:*** Time it takes to position the head at the desired track
  - ***Rotational delay*** or ***rotational latency:*** The time its takes for the beginning of the sector to reach the head

  -***Transfer Time*** is the time taken to transfer the data.

Disk Scheduling Policies

To compare various schemes, consider a disk head is initially located at track 100.

- assume a disk with 200 tracks and that the disk request queue has random requests in it

The requested tracks, in the order received by the disk scheduler, are

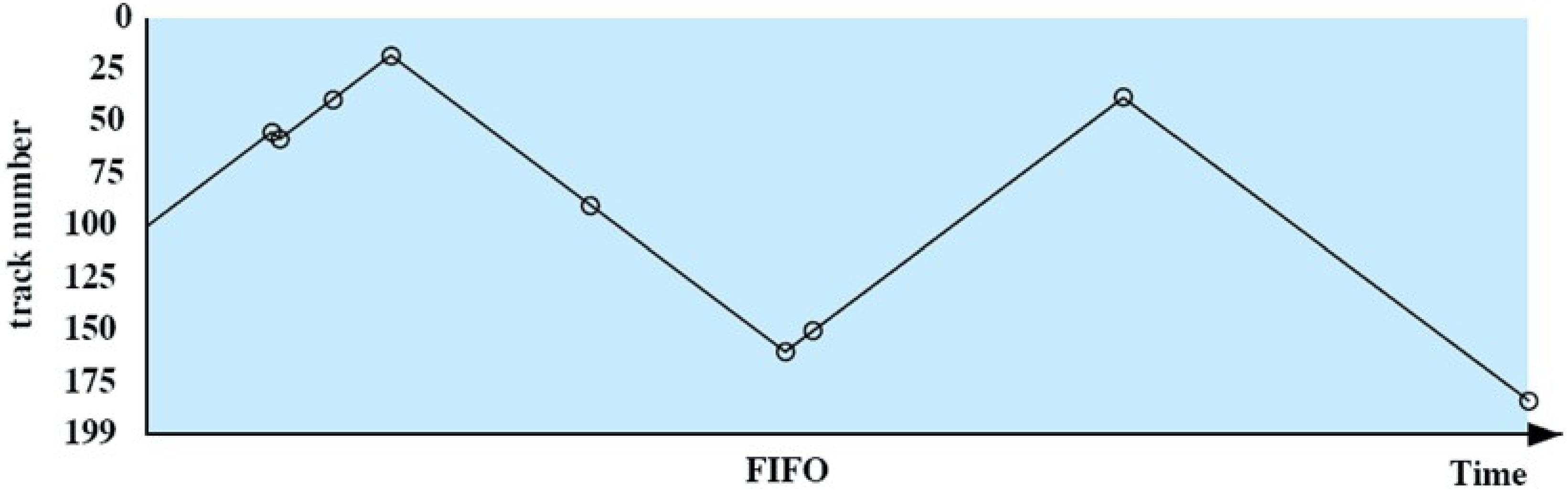
- 55, 58, 39, 18, 90, 160, 150, 38, 184.

First-in, first-out (FIFO)

The requested tracks, in the order received by the disk scheduler, are

- 55, 58, 39, 18, 90, 160, 150, 38, 184.

- Processes requests sequentially in the order received
- Fair to all processes
- Approaches random scheduling in performance

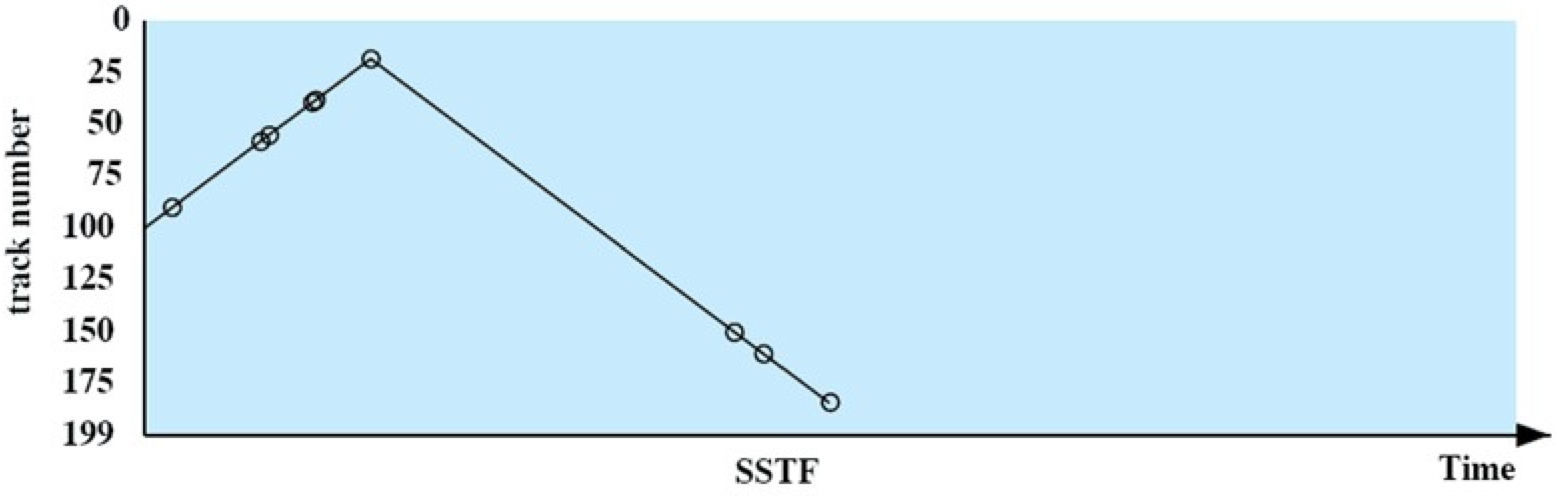


Shortest Service Time First

The requested tracks, in the order received by the disk scheduler, are

- 55, 58, 39, 18, 90, 160, 150, 38, 184.

- Select the disk I/O request that requires the least movement of the disk arm from its current position
- Always choose the minimum seek time

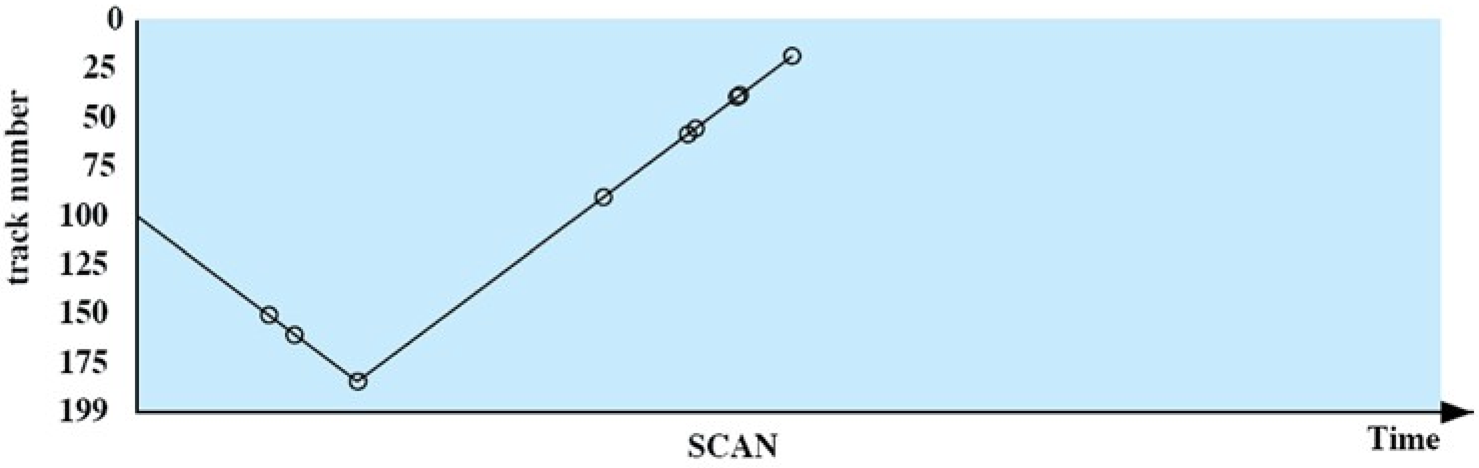


SCAN

The requested tracks, in the order received by the disk scheduler, are

- 55, 58, 39, 18, 90, 160, 150, 38, 184.

- Arm moves in one direction only, processing all outstanding requests until it reaches the last track in that direction & then the direction is reversed

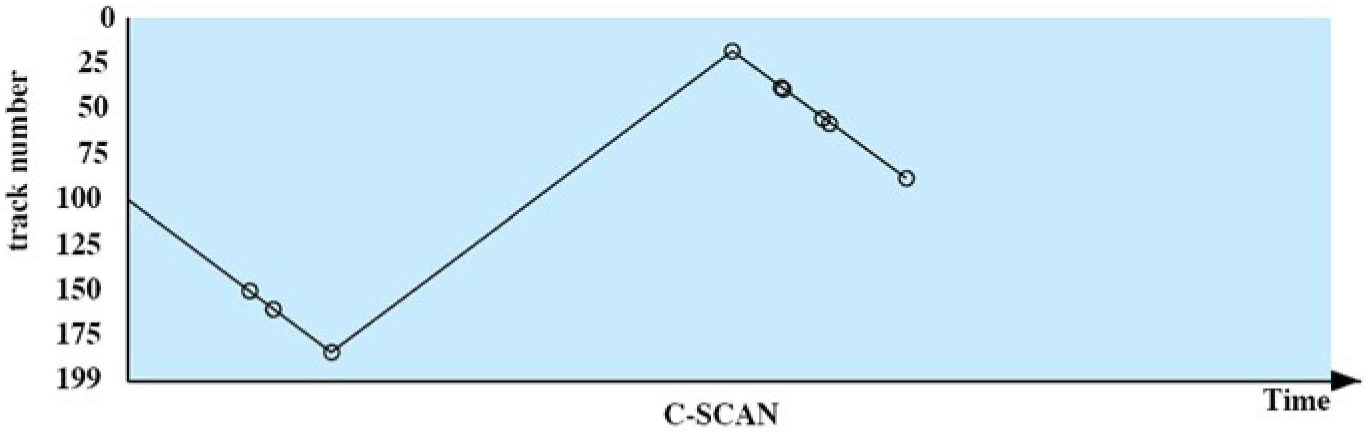


C-SCAN

The requested tracks, in the order received by the disk scheduler, are

- 55, 58, 39, 18, 90, 160, 150, 38, 184.

- Restricts scanning to one direction only
- When the last track has been visited in one direction, the arm is returned to the opposite end of the disk and the scan begins again



Performance Compared

Comparison of Disk Scheduling Algorithms

(a) FIFO (starting at track 100)		(b) SSTF (starting at track 100)		(c) SCAN (starting at track 100, in the direction of increasing track number)		(d) C-SCAN (starting at track 100, in the direction of increasing track number)	
Next track accessed	**Number of tracks traversed**	**Next track accessed**	**Number of tracks traversed**	**Next track accessed**	**Number of tracks traversed**	**Next track accessed**	**Number of tracks traversed**
55	45	90	10	150	50	150	50
58	3	58	32	160	10	160	10
39	19	55	3	184	24	184	24
18	21	39	16	90	94	18	166
90	72	38	1	58	32	38	20
160	70	18	20	55	3	39	1
150	10	150	132	39	16	55	16
38	112	160	10	38	1	58	3
184	146	184	24	18	20	90	32
Average seek length	55.3	**Average seek length**	27.5	**Average seek length**	27.8	**Average seek length**	35.8