



CS & IT ENGINEERING



Operating System

File System (Part-01)

Revision

Lecture No.- 13

By- Dr. Khaleel Khan Sir



Recap of Previous Lecture



Topic

Virtual Memory Performance

Topics to be Covered



Topic

Background

Topic

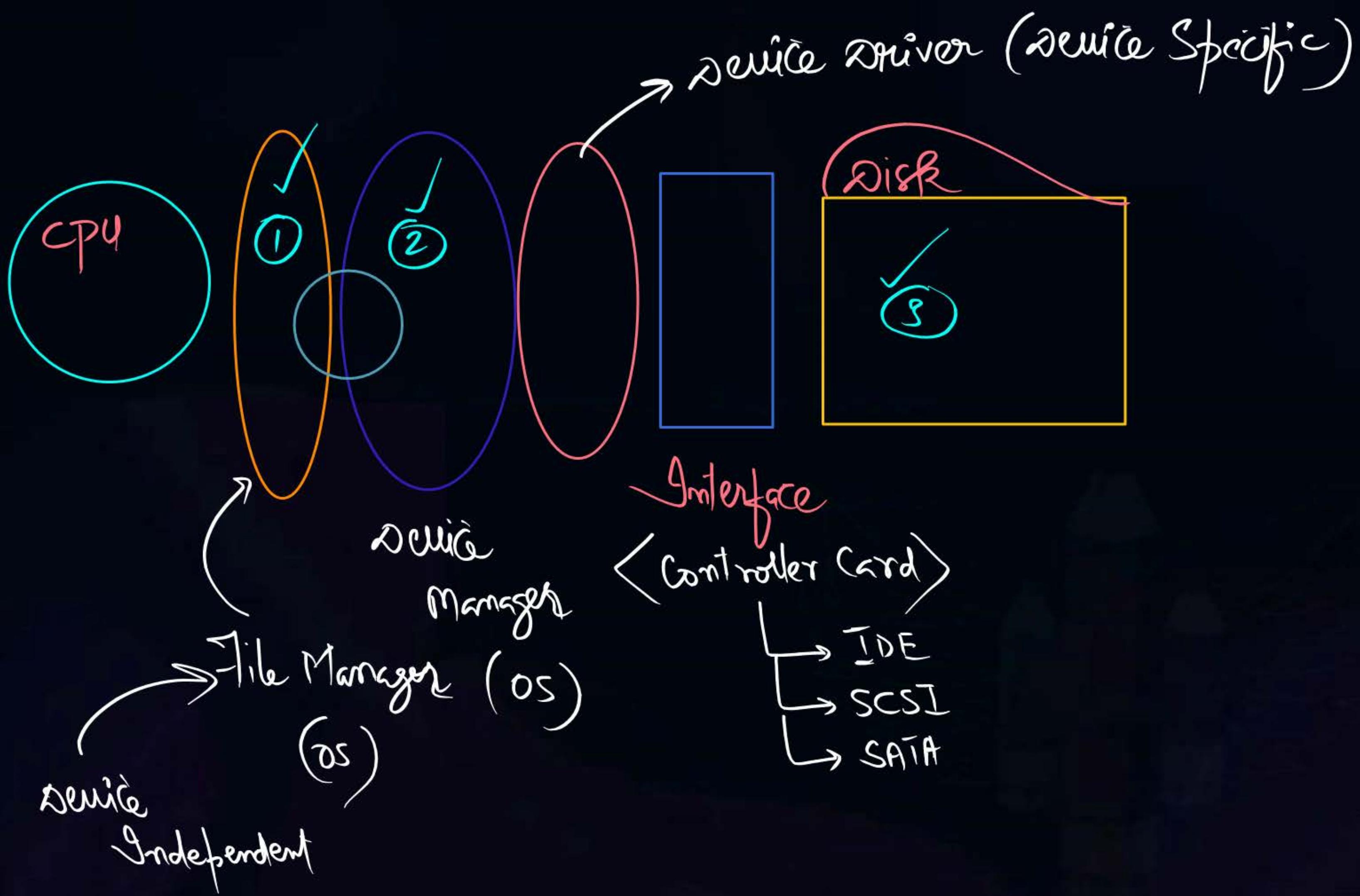
File & Directory Concepts

Topic

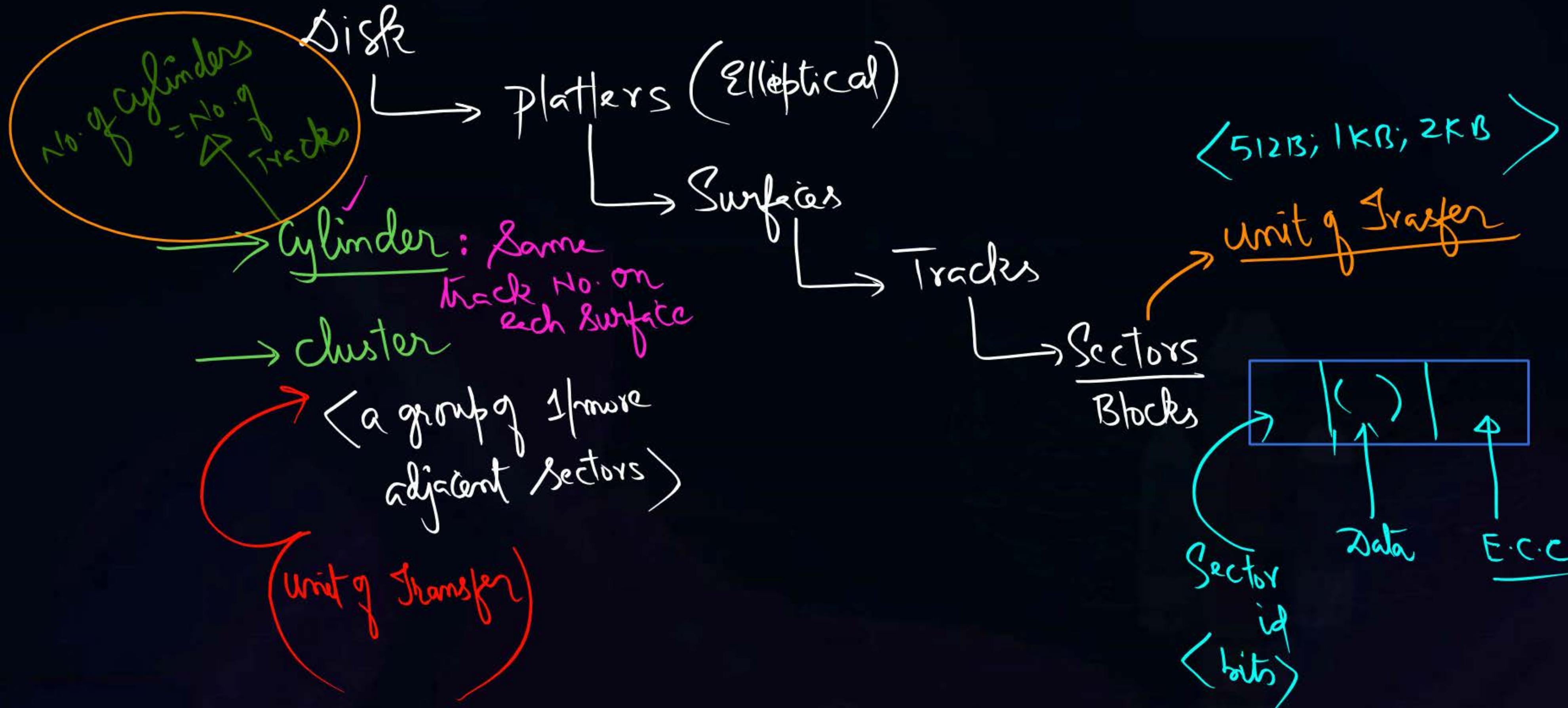
Disk Structure

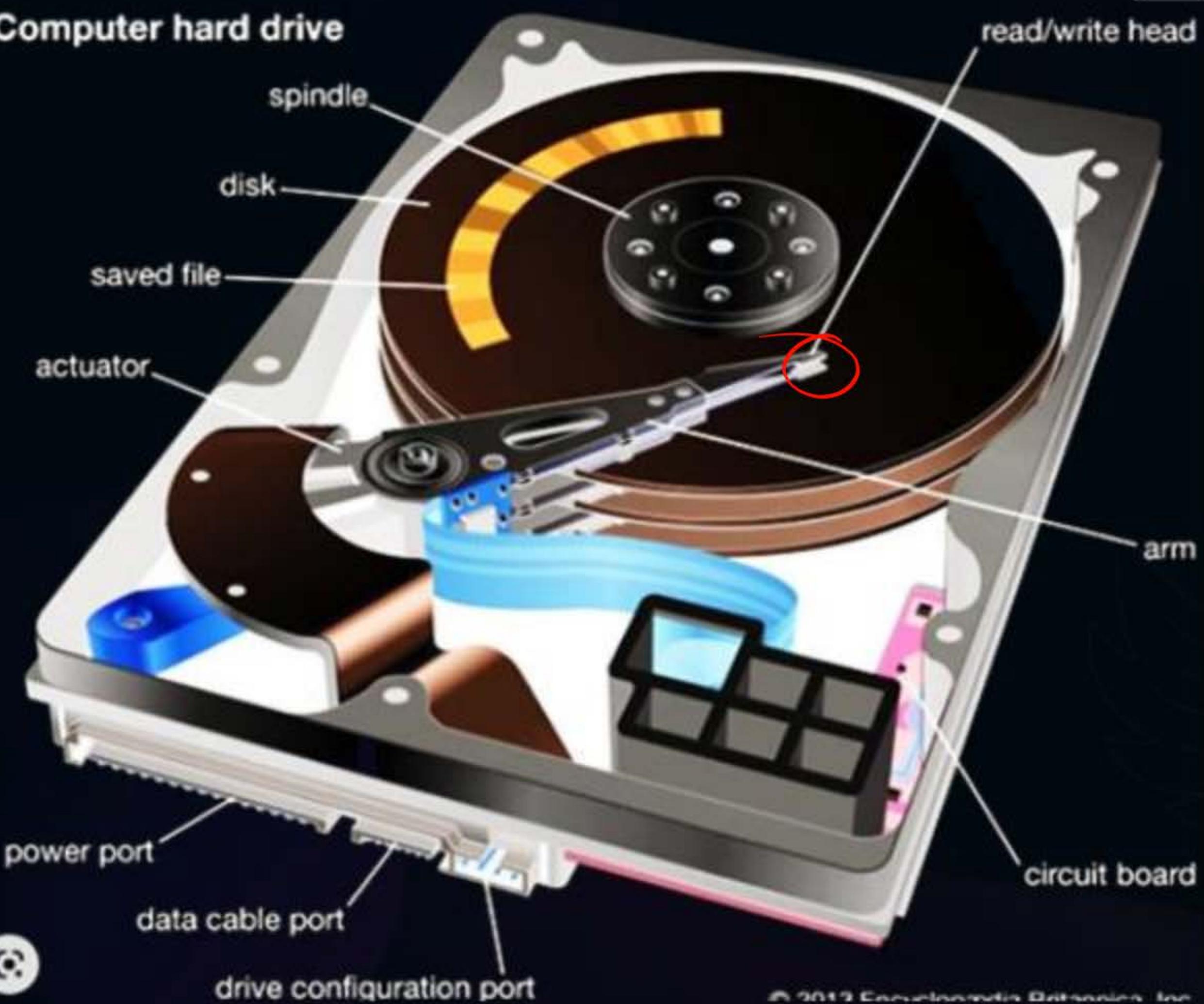
Topic

Interface



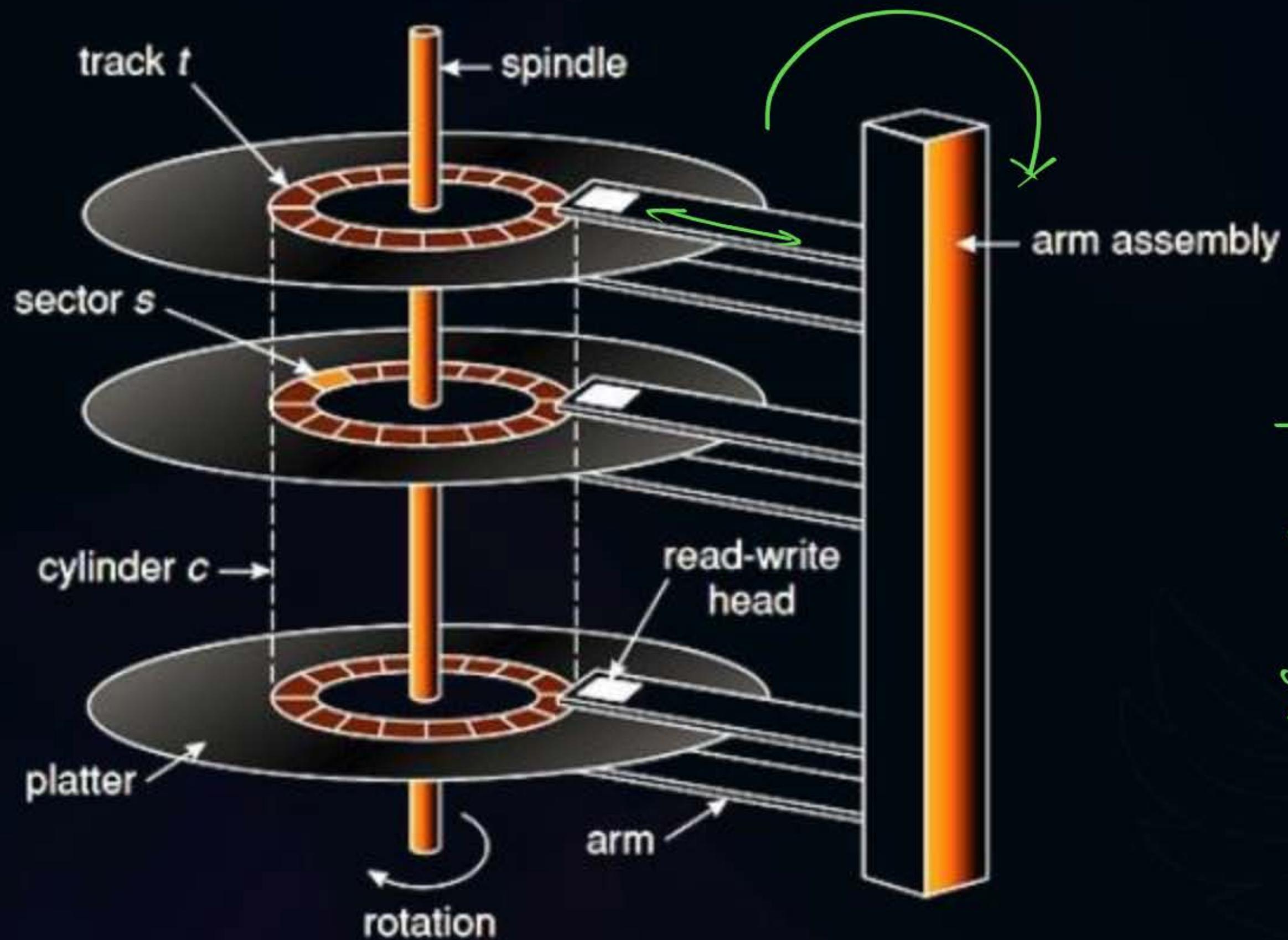
Physical Geometry of Hard-Disk



Computer hard drive

P
W





→ Linear Motion (velocity)
→ Angular " (velocity)
~~Speed - disk (D-T-R)~~

→ Disk Capacity :

→ I_O-Time (R/w : Block/Sector) :

Transfer Time

Track Size: 'X' Bytes

Sec Size: 'S' Bytes ($S < X$)

Rotation Time: R ms

R_{ms} → 'X' Bytes

? → 'S' Bytes

$$T.T = \left(\frac{S.R}{X} \text{ ms} \right)$$

Rotational

Seek Time (ST) + Latency Time +

(LT)

Transfer Time (TT)

$$\xrightarrow{x \rightarrow y} [|x-y| * T.T]$$

$$\left(\frac{R}{2} \right)$$

R = Time for one Rotation (ms)

$$(60\text{ rpm}) \quad 3600\text{ rpm} \quad 3600 \text{ rev - 60 s}$$

$$\left(R = 60 / 3600 \text{ ms} \right) \text{ 1 rev - ?}$$

Date transfer rate (B/S)

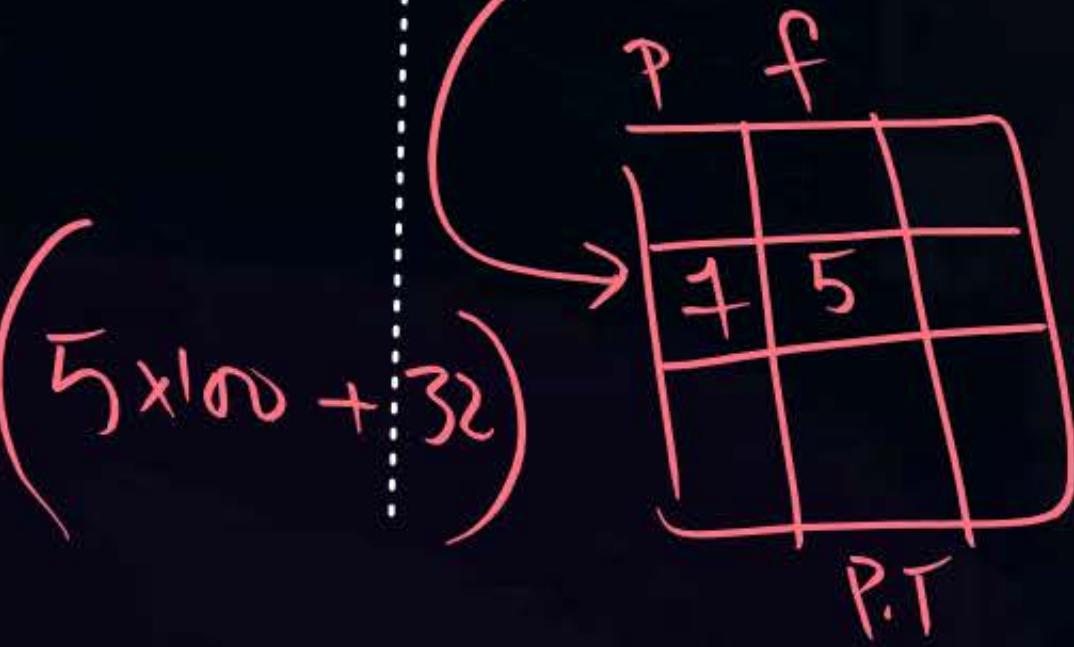
\times Bytes — R_{ms}

$$DTR = \left(\frac{\times}{R} \times 10^3 \right) B/S$$

? $\leftarrow 1S$

$$DTR = \left(\frac{\times}{R} \times 10^3 \right) B/S$$

$$(5 \times 100 + 32)$$



Flat Address

V.A: 732 P7

P.S = 100W

$$P = 7 \quad (732/100)$$

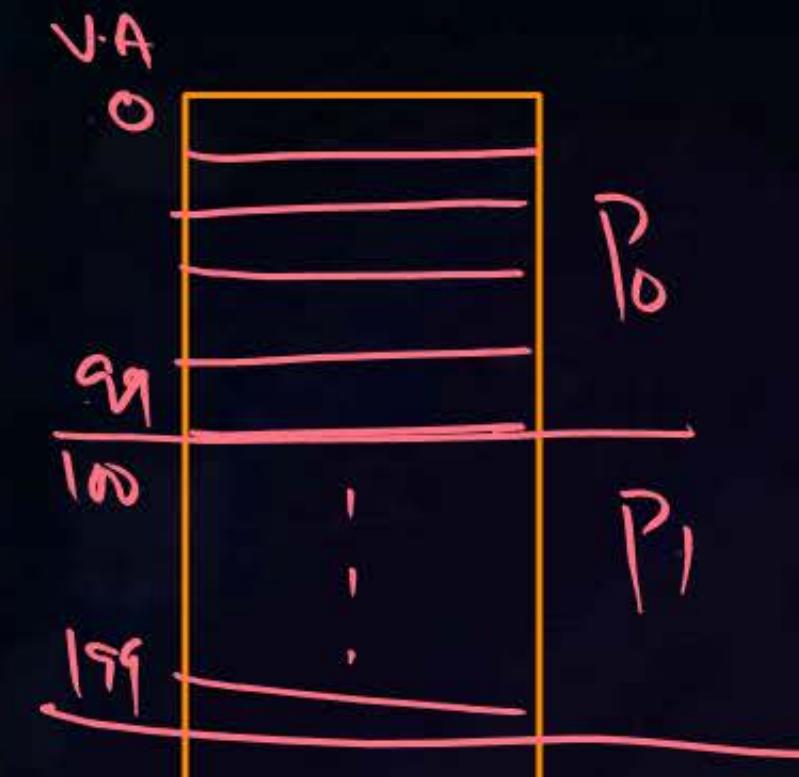
$$d = 32 \quad (732 \% 100)$$

V.A:

7	32
---	----

 < Hierarchical >

V.A: 7, 32



①

32 platters

↳ Surface has 2K Tracks → 512 Sectors
 ↳ 4096 B Sector;

$$\frac{6000 - 605}{1n - ?} = \frac{60}{6000} = \frac{1}{100} s = 0.01 s \\ = 0.01 \times 10^{-3} \\ = 10^3 \times 0.01 = 10 \text{ ms}$$

P
W

$$\rightarrow \text{Av. ST} = 30 \text{ ms}$$

$$\text{RPM} = 6000$$

$$\text{① Capacity : } [32 * 2 * 2K * 4KB] = 2^5 * 2 * 2 * 2 * 2 * 2^{38} = 2^{56} \text{ GB}$$

$$\text{② Tr Time/Sector : } \left(30 \text{ ms} + \frac{10}{2} \text{ ms} + x \right)$$

$$\text{Tr-Size} = 512 \times 4 \text{ KB}$$

$$\text{③ DTR : } 2 \text{ MB} \rightarrow 10 \times 10^{-3} \text{ s}$$

$$? \leftarrow 1 \text{ s}$$

$$\frac{2 \text{ MB} \times 10^3}{10 \times} = 0.5 \text{ GB/s}$$

$$= 2 \text{ MB}$$

$$2 \text{ MB} - 10 \text{ ms}$$

$$\left(\frac{4 \text{ KB} \times 10}{2 \text{ MB}} \right) = ?$$

$$4 \text{ KB} - ?$$



Topic : File Concept

- ❑ Contiguous logical address space
- ❑ Types:
 - Data
 - ➔ Numeric
 - ➔ Character
 - ➔ Binary
 - Program
- ❑ Contents defined by file's creator
 - Many types
 - ➔ text file,
 - ➔ source file,
 - ➔ executable file

File System Interface

(i) File vs Directories





Topic : File Attributes

~~x File is an A.D.T x~~

P
W

- Name - only information kept in human-readable form
- Identifier - unique tag (number) identifies file within file system
- Type - needed for systems that support different types
- Location - pointer to file location on device
- Size - current file size
- Protection - controls who can do reading, writing, executing <UNIX|UNUX>
- Time, date, and user identification - data for protection, security, and usage monitoring
- Information about files are kept in the directory structure, which is maintained on the disk
- Many variations, including extended file attributes such as file checksum
- Information kept in the directory structure

Attributes of
File stored in
F.C.B

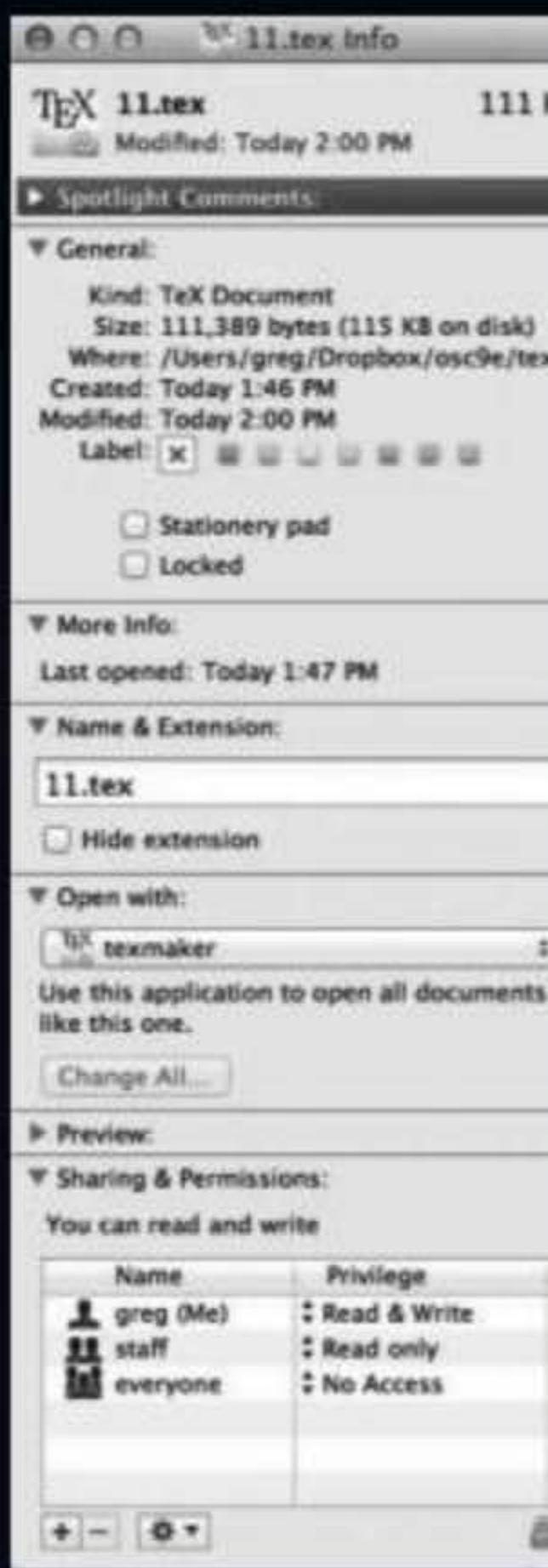
File Control Block

J-Node

Dir-Entry



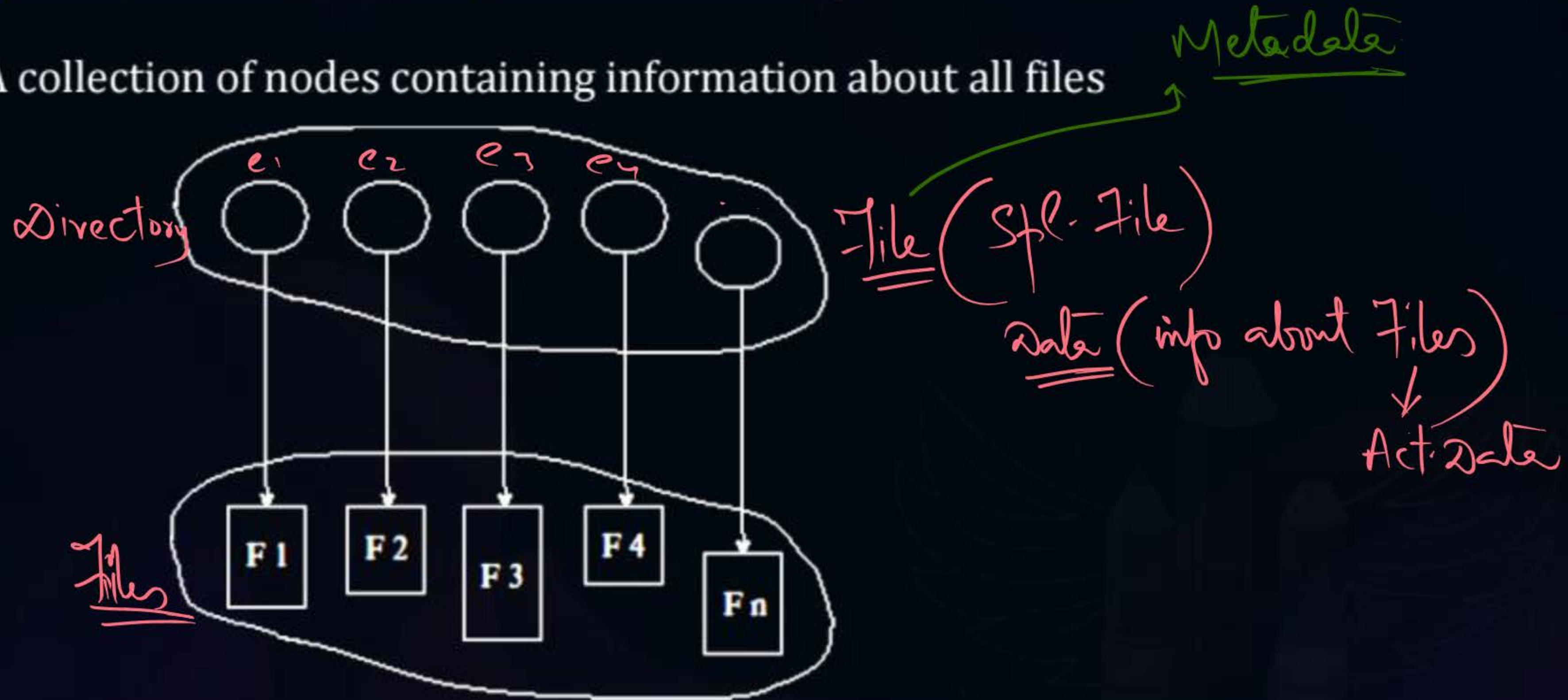
Topic : File info Window on Mac OS X





Topic : Directory Structure

- A collection of nodes containing information about all files

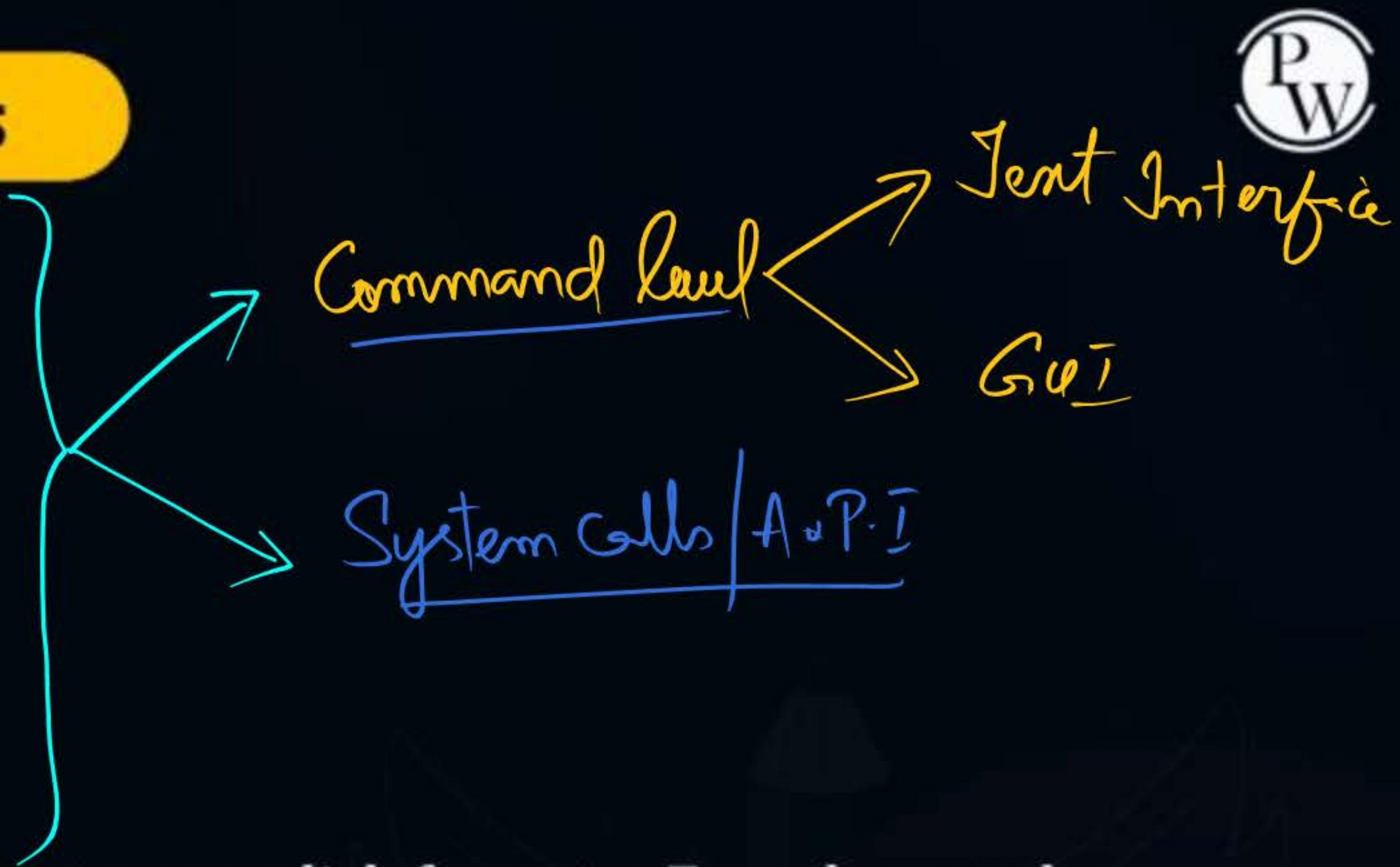


- Both the directory structure and the files reside on disk



Topic : File Operations

- Create
- Write - at **write pointer** location
- Read - at **read pointer** location
- Reposition within file - **seek**
- Delete
- Truncate
- Open (F_i) - search the directory structure on disk for entry F_i , and move the content of entry to memory
- Close (F_i) - move the content of entry F_i in memory to directory structure on disk





Topic : Open Files



- Several pieces of data are needed to manage open files:
 - Open-file table: tracks open files
 - File pointer: pointer to last read/write location, per process that has the file open
 - File-open count: counter of number of times a file is open - to allow removal of data from open-file table when last processes closes it
 - Disk location of the file: cache of data access information
 - Access rights: per-process access mode information



Topic : File Locking



- Provided by some operating systems and file systems
 - Similar to reader-writer locks
 - **Shared lock** similar to reader lock - several processes can acquire concurrently
 - **Exclusive lock** similar to writer lock
- Mediates access to a file
- Mandatory or advisory:
 - **Mandatory** - access is denied depending on locks held and requested
 - **Advisory** - processes can find status of locks and decide what to do



Topic : File Types – Name, Extension



File type	usual extension	Function
Executable	exe, com, bin or none	ready-to-run machine language program
Object	Obj, o	Compiled, machine language, not linked
Source code	c, cc, java, pas, asm, a	source code in various languages
batch	bat, sh	commands to the command interpreter
Text	txt, doc	textual data, documents
word processor	wp, tex, rtf, doc	various word-processor formats
library	lib, a, so, dll	libraries of routines for programmers
print or view	ps, pdf, jpg	ASCII or binary file in a format for printing or viewing
archive	arc, zip, tar	related files grouped into one file, sometimes compressed, for archiving or storage
multimedia	mpeg, mov, rm, mp3, avi	binary file containing audio or A/V information



Topic : File Structure



- None - sequence of words, bytes
- Simple record structure
 - Lines
 - Fixed length
 - Variable length
- Complex Structures
 - Formatted document
 - Relocatable load file
- Can simulate last two with first method by inserting appropriate control characters
- Who decides:
 - Operating system
 - Program



Topic : Access Methods

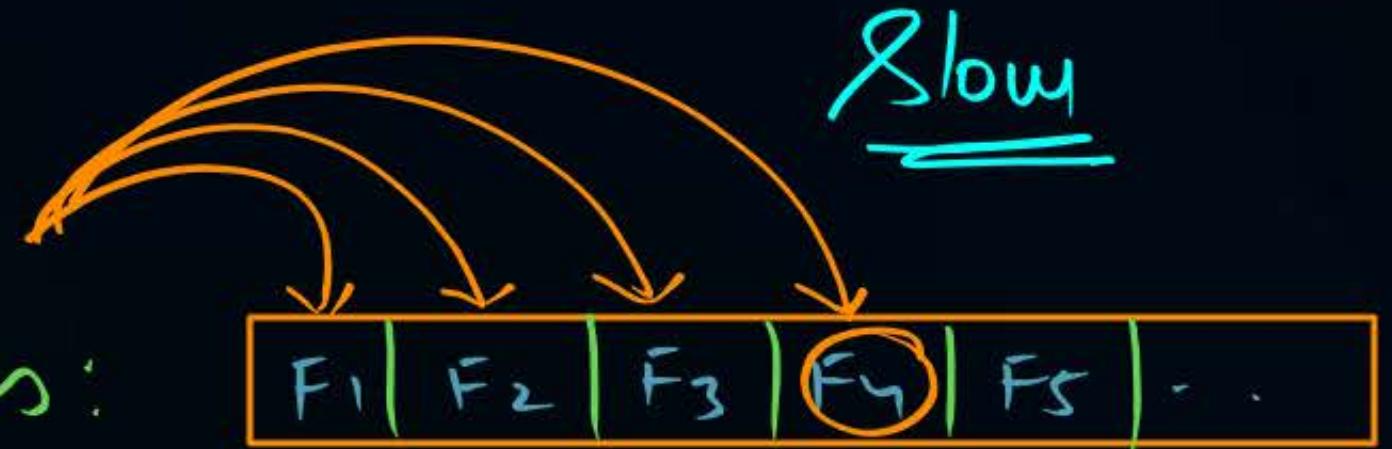
- A file is fixed length logical records

- Sequential Access

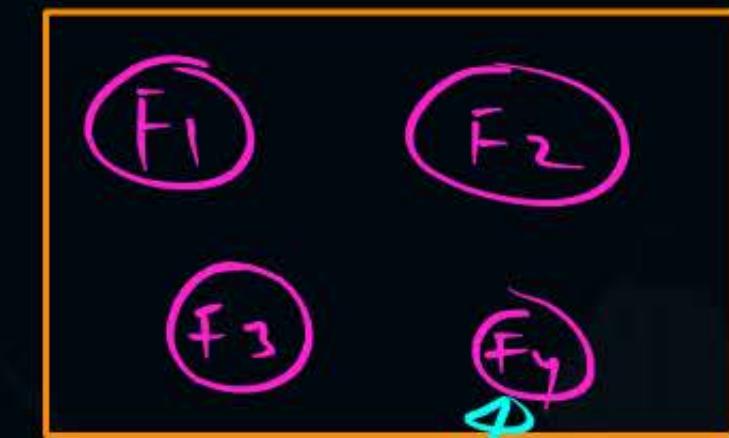
- Direct Access

- Other Access Methods

Magnetic Tapes:



Disk



< Faster >



Topic : Sequential Access

- Operations
 - read next
 - write next
 - Reset
 - no read after last write (rewrite)





Topic : Direct Access

- Operations
 - read n
 - write n
 - position to n
 - read next
 - write next
 - rewrite n
- n = relative block number
- Relative block numbers allow OS to decide where file should be placed



Topic : Simulation of Sequential Access on Direct-access File



sequential access	implementation for direct access
<i>reset</i>	$cp = 0;$
<i>read next</i>	<i>read cp;</i> $cp = cp + 1;$
<i>write next</i>	<i>write cp;</i> $cp = cp + 1;$



Topic Other Access Methods



- Can be other access methods built on top of base methods
- General involve creation of an **index** for the file
- Keep index in memory for fast determination of location of data to be operated on (consider Universal Produce Code (UPC code) plus record of data about that item)
- If the index is too large, create an in-memory index, which an index of a disk index
- IBM indexed sequential-access method (ISAM)
 - Small master index, points to disk blocks of secondary index
 - File kept sorted on a defined key
- All done by the OS
- VMS operating system provides index and relative files as another example (see next slide)



Topic : Example of Index and Relative Files



logical record
last name number

last name	number
Adams	
Arthur	
Asher	
⋮	
Smith	

index file

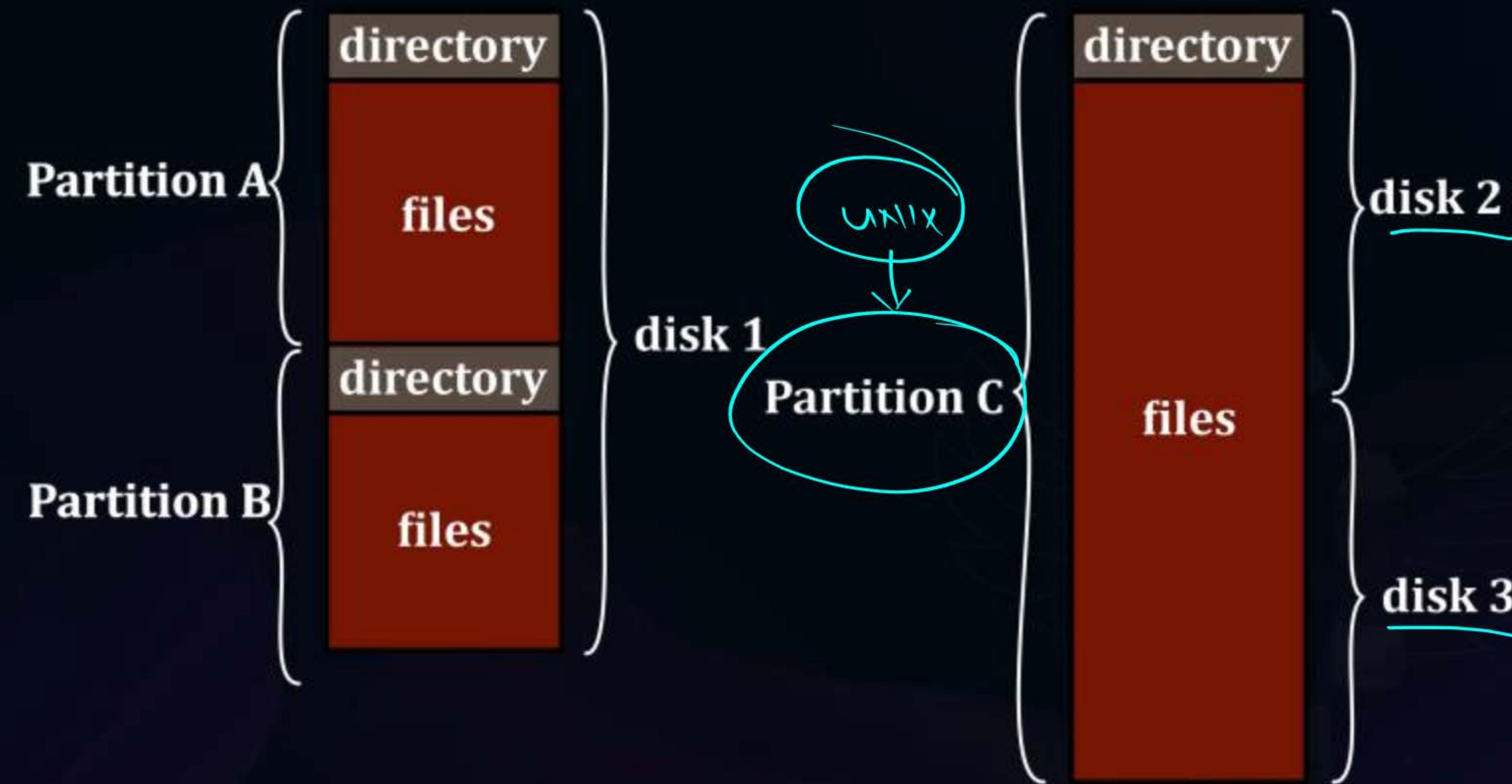
smith, john	social-security	age
⋮	⋮	⋮

relative file





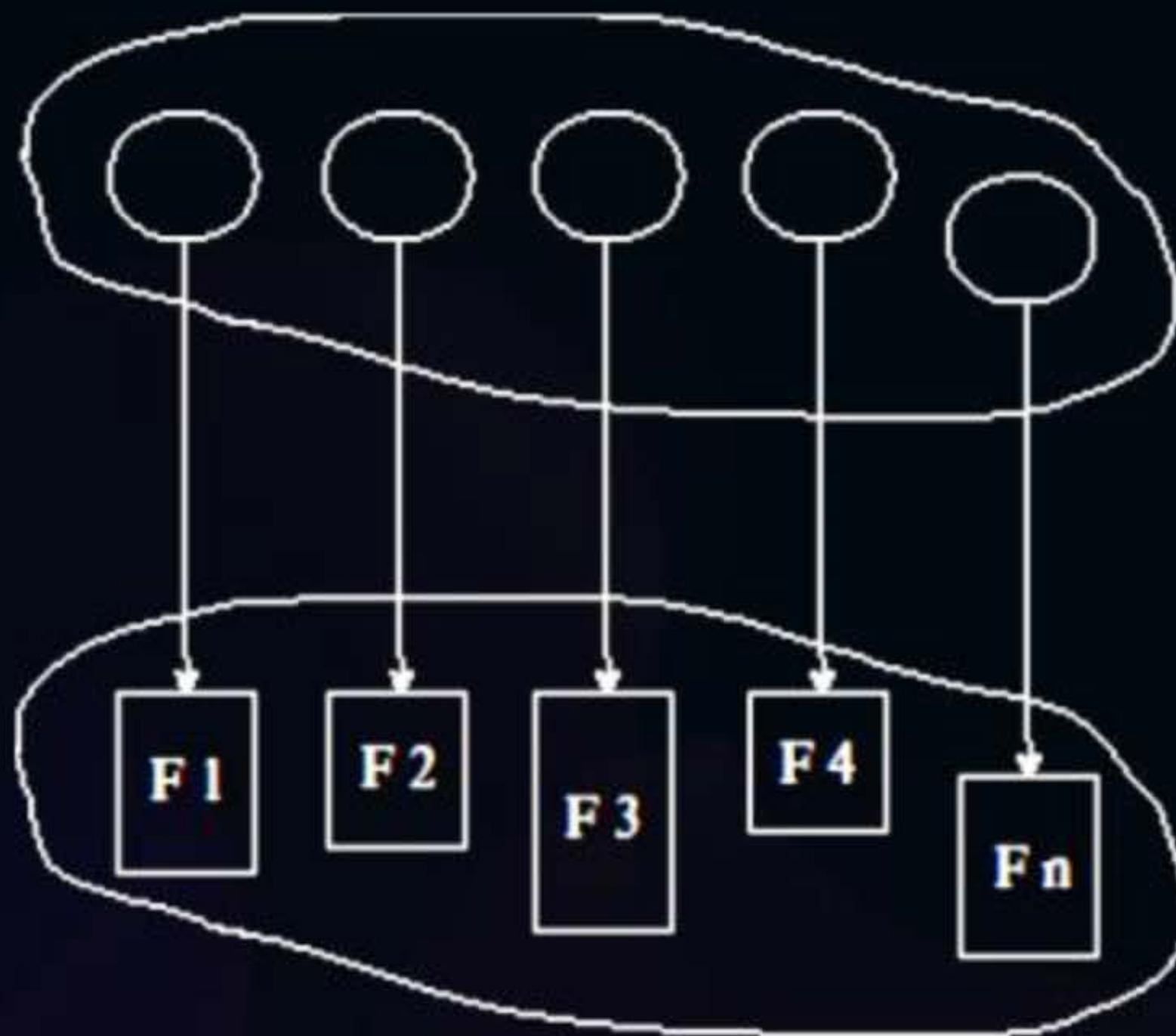
Topic : A Typical File-system Organization





Topic : Directory Structure

- A collection of nodes containing information about all files



- Both the directory structure and the files reside on disk



Topic : Operations Performed on Directory



- Search for a file ✓
- Create a file ✓
- Delete a file ✓
- List a directory ✓
- Rename a file ✓
- Traverse the file system ✓



Topic : Directory Organization



The directory is organized logically to obtain

- **Efficiency** - locating a file quickly
- **Naming** - convenient to users
 - Two users can have same name for different files
 - The same file can have several different names
- **Grouping** - logical grouping of files by properties, (e.g., all Java programs, all games, ...)



Topic : Single-Level Directory

- A single directory for all users

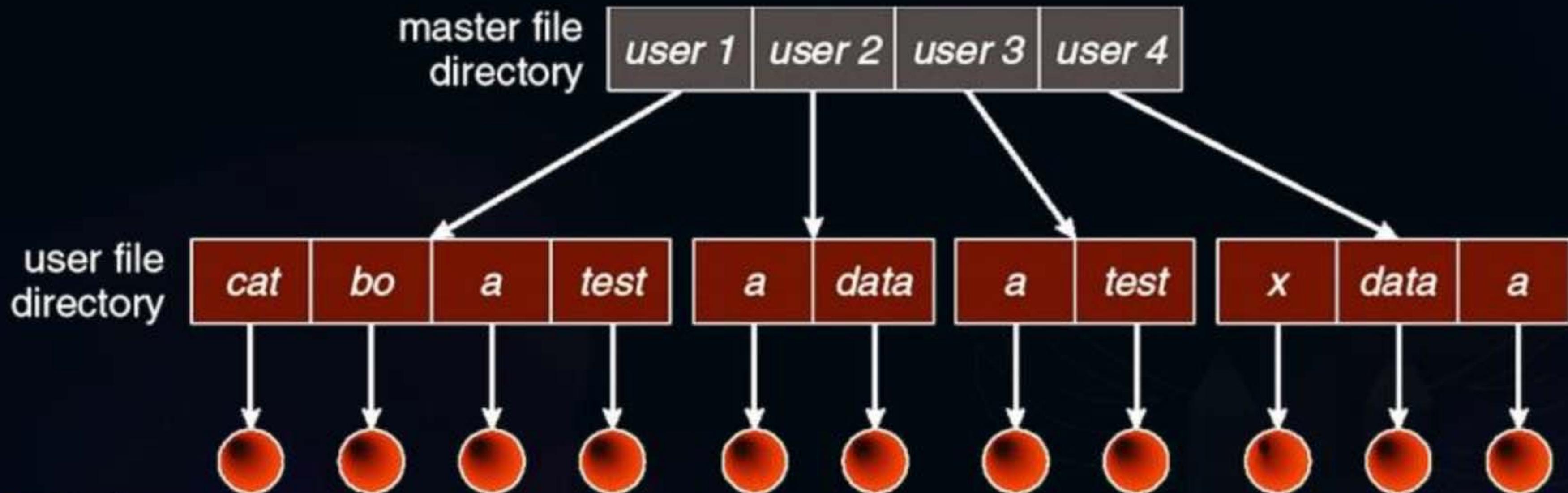




Topic : Two-Level Directory



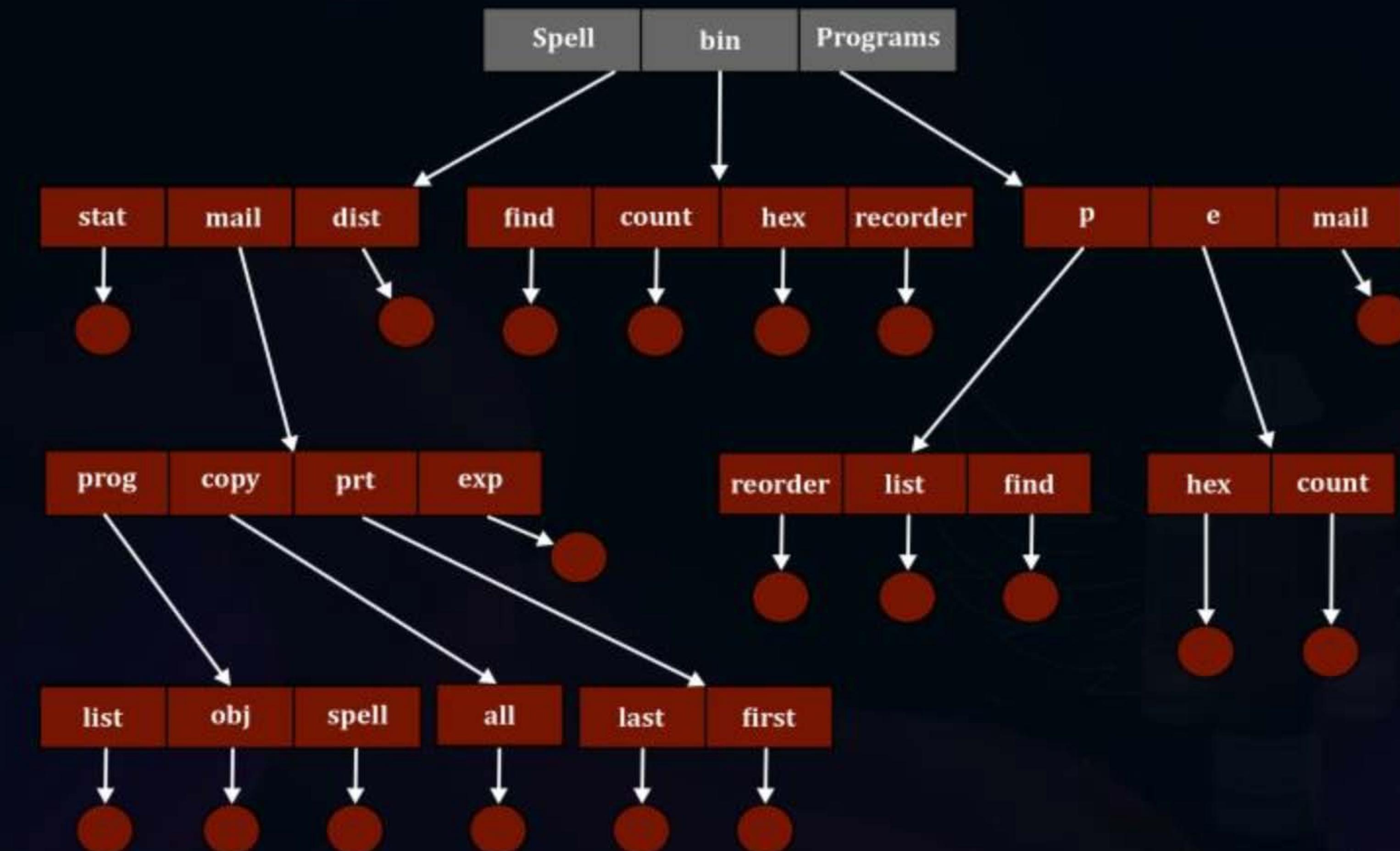
- Separate directory for each user



- Path name
- Can have the same file name for different user
- Efficient searching
- No grouping capability



Topic : Tree-Structured Directories

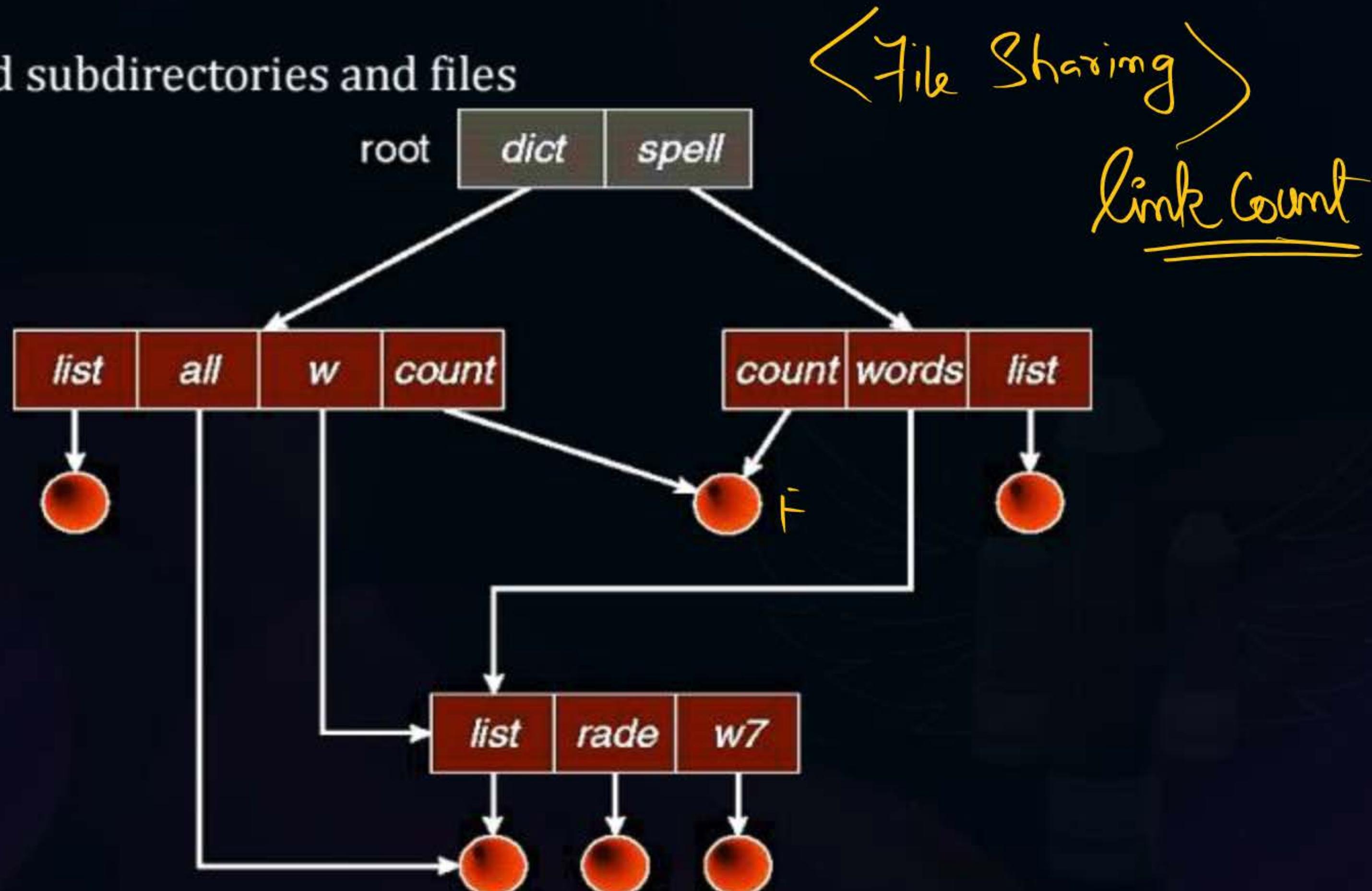




Topic : Acyclic-Graph Directories



- Have shared subdirectories and files
- Example





Topic : Acyclic-Graph Directories (Cont.)



- Two different names (aliasing)
- If dict deletes w/list \Rightarrow dangling pointer

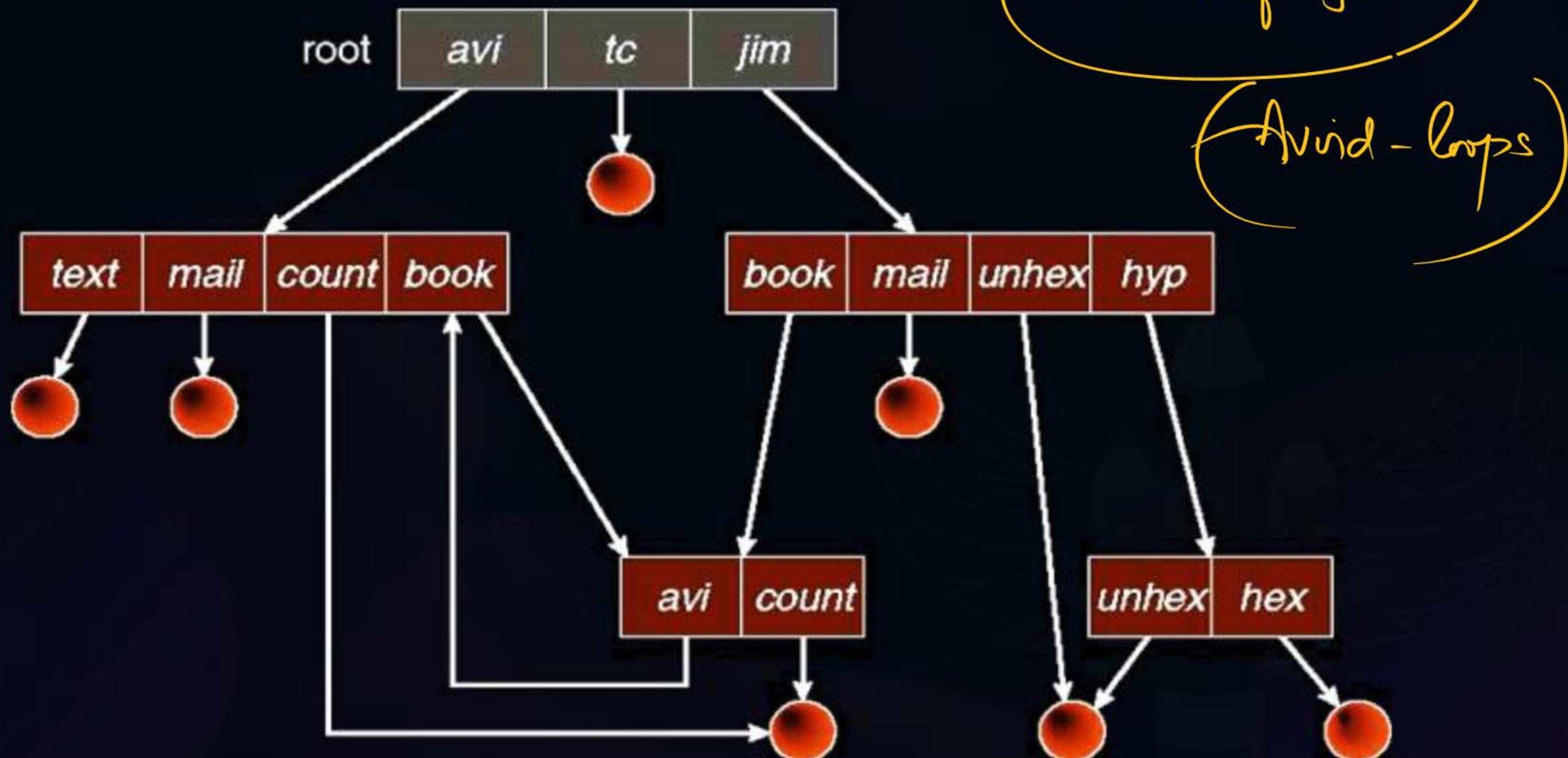
Solutions:

- Backpointers, so we can delete all pointers.
 - Variable size records a problem
- Backpointers using a daisy chain organization
- Entry-hold-count solution
- New directory entry type
 - **Link** - another name (pointer) to an existing file
 - **Resolve the link** - follow pointer to locate the file



Topic : General Graph Directory

P
W





Topic : General Graph Directory (Cont.)

- How do we guarantee no cycles?
 - Allow only links to files not subdirectories
 - **Garbage collection**
 - Every time a new link is added use a cycle detection algorithm to determine whether it is OK





Topic : Current Directory



- Can designate one of the directories as the current (working) directory
 - cd /spell/mail/prog
 - type list
- Creating and deleting a file is done in current directory
 - Example of creating a new file
 - If in current directory is /mail
 - The command

`mkdir <dir-name>`

- Results in:
 - prog
 - copy
 - prt
 - exp
 - count
- 
- ```
graph TD; mail[mail] --- prog[prog]; mail --- copy[copy]; mail --- prt[prt]; mail --- exp[exp]; mail --- count[count]
```
- A diagram illustrating a file tree. At the top is a box labeled "mail". Five lines descend from it to five separate boxes at the bottom labeled "prog", "copy", "prt", "exp", and "count".
- Deleting “mail” ⇒ deleting the entire subtree rooted by “mail”



## Topic : Protection



- File owner/creator should be able to control:
  - What can be done
  - By whom
- Types of access
  - Read
  - Write
  - Execute
  - Append
  - Delete
  - List



# Topic : Access Lists and Groups in Unix



- Mode of access: read, write, execute
- Three classes of users on Unix / Linux

(a) owner access

7  $\Rightarrow$  RWX  
1 1 1  
RWX

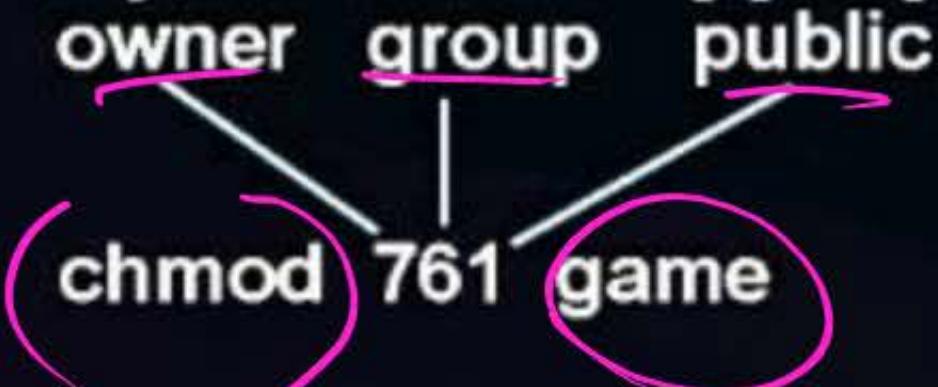
(b) group access

6  $\Rightarrow$  1 1 0  
RWX

(c) public access

1  $\Rightarrow$  0 0 1

- Ask manager to create a group (unique name), say G, and add some users to the group.
- For a file (say game) or subdirectory, define an appropriate access.
- Attach a group to a file



`chgrp G game`



## Topic : A Sample UNIX Directory Listing



\$ ls -al

|              |   |     |         |       |              |               |
|--------------|---|-----|---------|-------|--------------|---------------|
| -rw-rw-r--   | 1 | pbg | staff   | 31200 | Sep 3 08:30  | intro.ps      |
| drwx-----    | 5 | pbg | staff   | 512   | Jul 8 09.33  | private/      |
| drwxrwxr-x   | 2 | pbg | staff   | 512   | Jul 8 09:35  | doc/          |
| drwxrwx---   | 2 | pbg | student | 512   | Aug 3 14:13  | student-proj/ |
| -rw-r -- r-- | 1 | pbg | staff   | 9423  | Feb 24 2003  | prog ram. c   |
| -rwxr-xr-x   | 1 | pbg | staff   | 20471 | Feb 24 2003  | program       |
| drwx- -x- -x | 4 | pbg | faculty | 512   | Jul 31 10:31 | lib/          |
| drwx---      | 3 | pbg | staff   | 1024  | Aug 29 06:52 | mail/         |
| drwxrwxrwx   | 3 | pbg | staff   | 512   | Jul 8 09:35  | test/         |



# Topic : Windows 7 Access-Control List Management



ListPanel.java Properties

General Security Details Previous Versions

Object name: H:\DATA\Patterns Material\Src\ListPanel.java

Group or user names:

- SYSTEM
- Gregory G. Gagne (ggagne@wcusers.int)
- Guest (WCUSERS\Guest)**
- FileAdmins (WCUSERS\FileAdmins)
- Administrators (FILES\Administrators)

To change permissions, click Edit.

Edit...

| Permissions for Guest | Allow | Deny |
|-----------------------|-------|------|
| Full control          | ✓     |      |
| Modify                | ✓     |      |
| Read & execute        | ✓     |      |
| Read                  | ✓     |      |
| Write                 | ✓     |      |
| Special permissions   |       |      |

For special permissions or advanced settings, click Advanced.

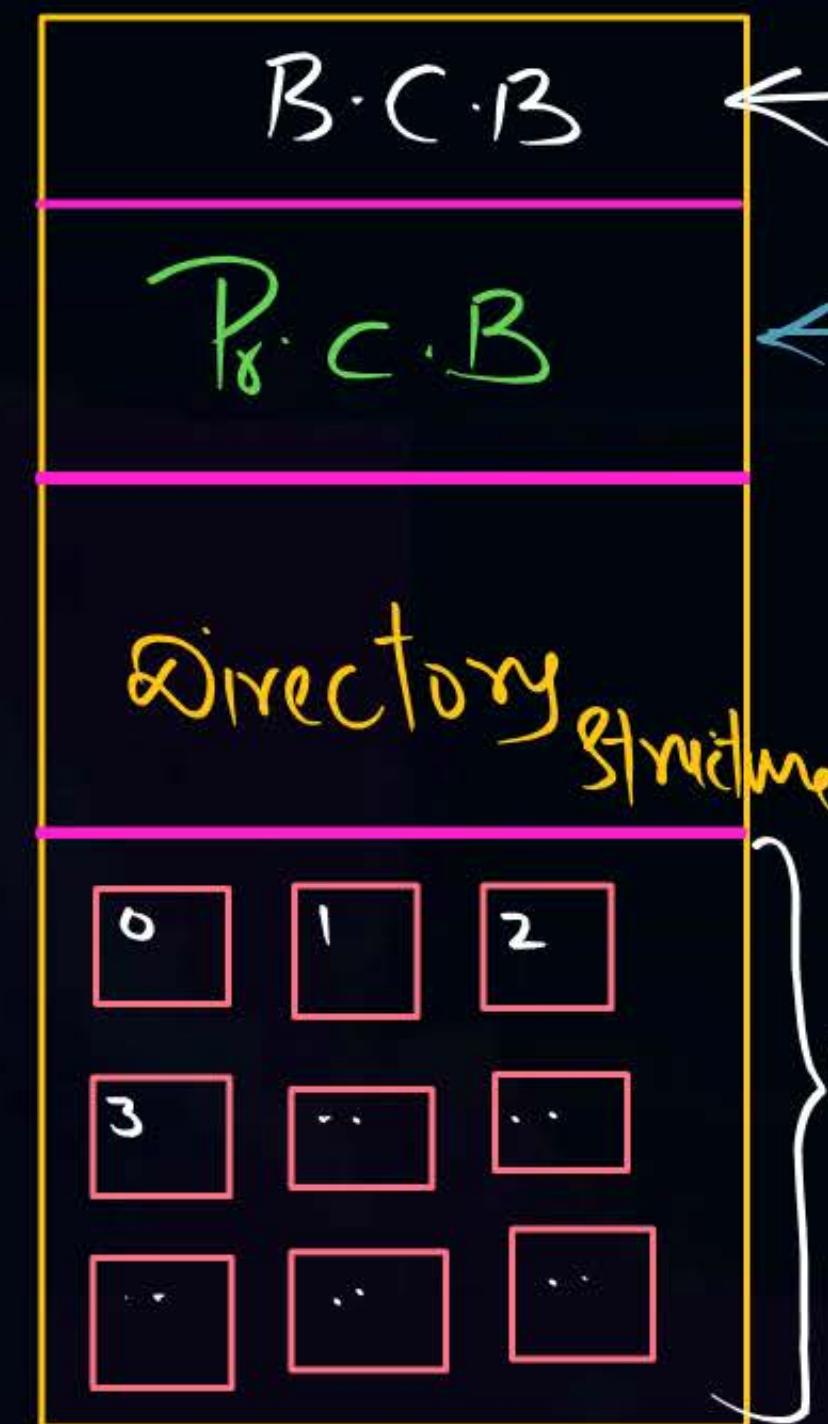
Advanced

Learn about access control and permissions

OK Cancel Apply

Logical Structure of Disk

Partition



<Formatting Process>

Boot Control Block

Partition Control Block

Directory Structure

Data Blks (Managing Disk Blks)

Part-Boot Sector

Boot Blk

SuperBlock

F.A.T

Master fileTable



## 2 mins Summary



**Topic** One

**Topic** Two

**Topic** Three

**Topic** Four

38

**Topic** Five



THANK - YOU