


Comp 310
Computer Systems and Organization

Lecture #20

File Systems

(Implementation – Part 2)

Prof. Joseph Vybihal



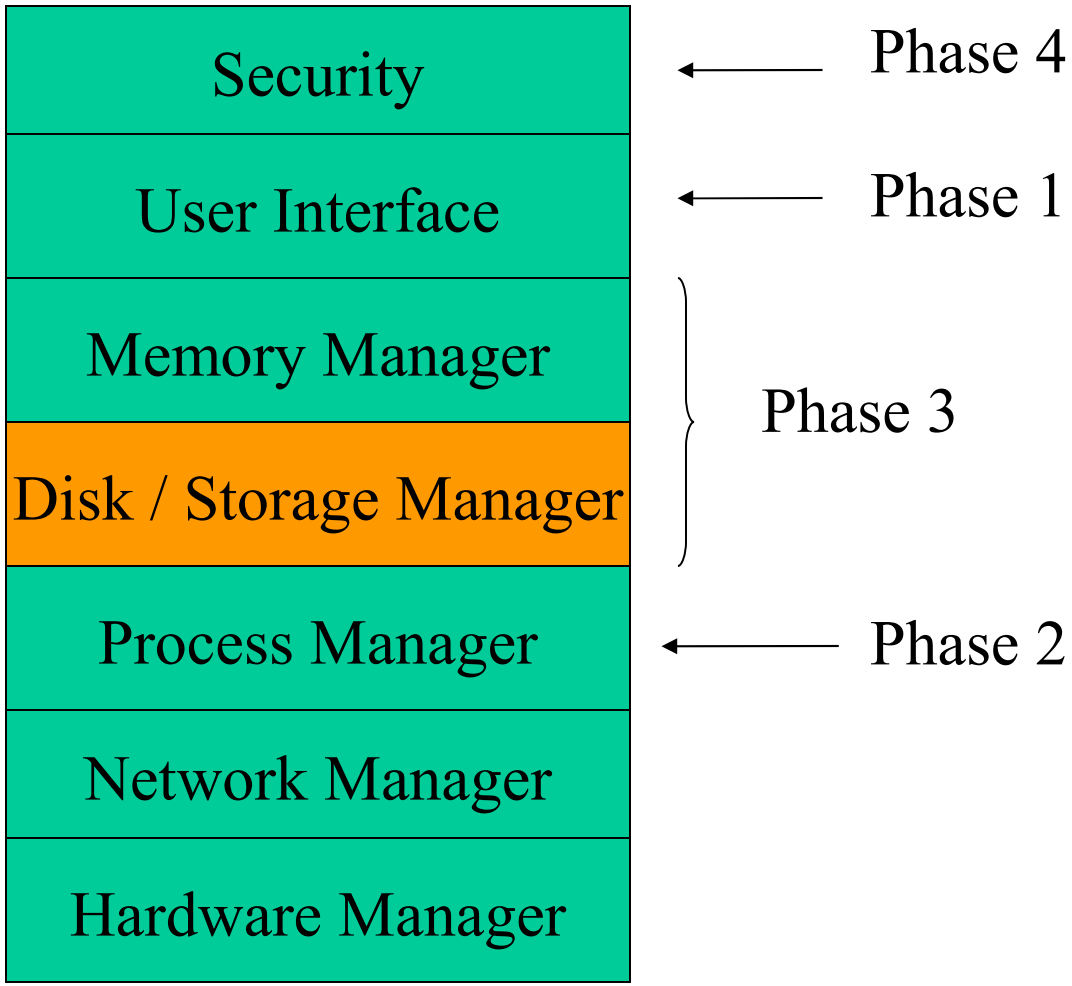

Announcements

- Kelvin teaching next class
  - Nov 18




Basic OS Architecture

(Course Table of Contents)






Part 1

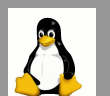
File System Mounting




Mounting (a.k.a Mapping)

- We cannot simply assume a directory is available to us
- Some devices like diskettes and hard disks have their directories available by default
- Other directory trees are not:
  - Private folders
  - Network folders
- To get access to them we need to perform an operation called Mounting or Mapping

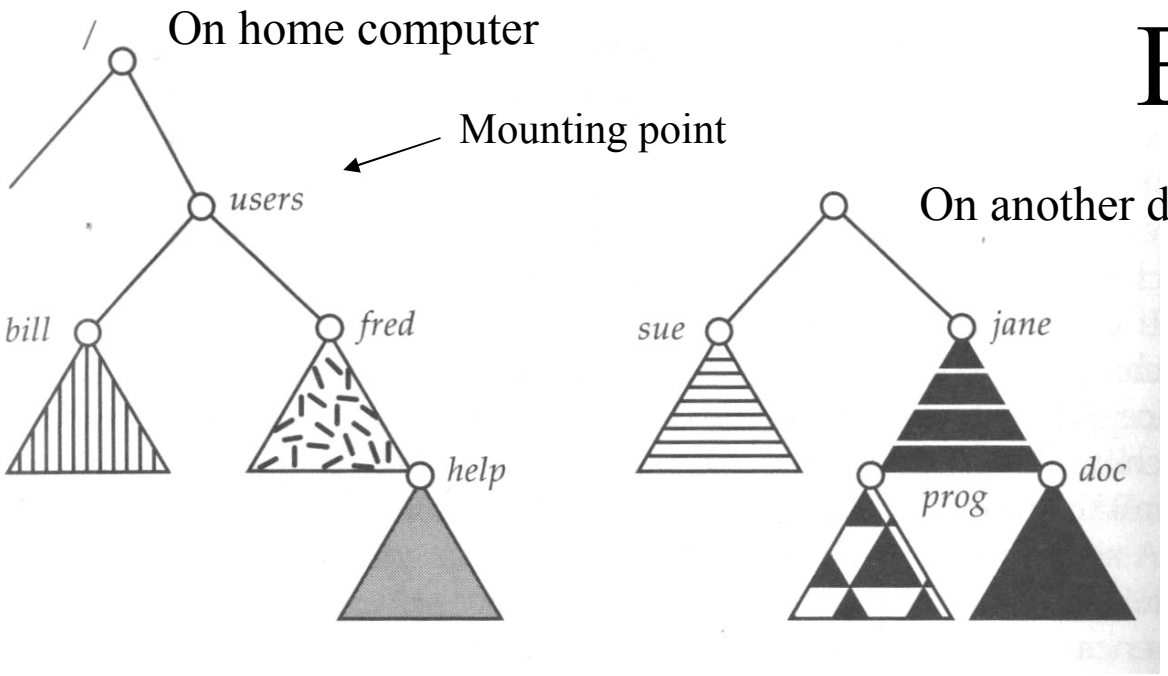





MSDOS Example

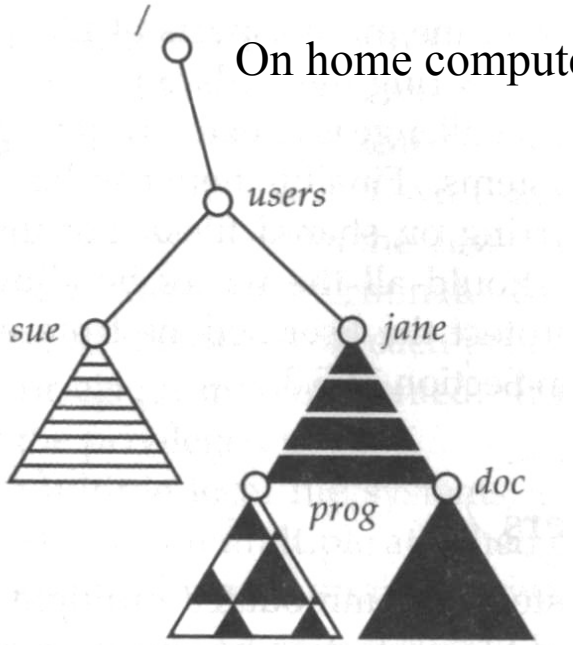
```
c:\>

c:\> map d: //mimi.cs.mcgill.ca/~billsmith/    ← mounting

c:\> d:

d:billsmith/>

c:\> map d:                                    ← un-mount
```

Example



Before

Need to create a folder name and that becomes the mounting point



After
(replacement)




Unix Example

$ mkdir /media/floppy0 ← a mount point

$ mkfs deviceName options

mkfs -t msdos /dev/fd0 ← making a file system

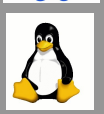
$ mount existingDeviceName existingDirectory

mount /dev/fd0 /media/floppy0

$ cat /etc/mnttab ← see the mounted directories

$ unmount f10

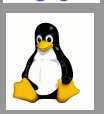





How does mounting work?

- Two concepts
  - Directory Trees and FATs
  - Getting directories






FATs vs Directory Trees

- FAT
  - A structure on disk that records the folders and files on disk.
  - A semi-static data structure
- Directory Tree
  - A data structure in RAM (table or linked)
  - Generated at login time
    - System tree (can be hidden)
    - User tree (can limit the view of the hard disk)

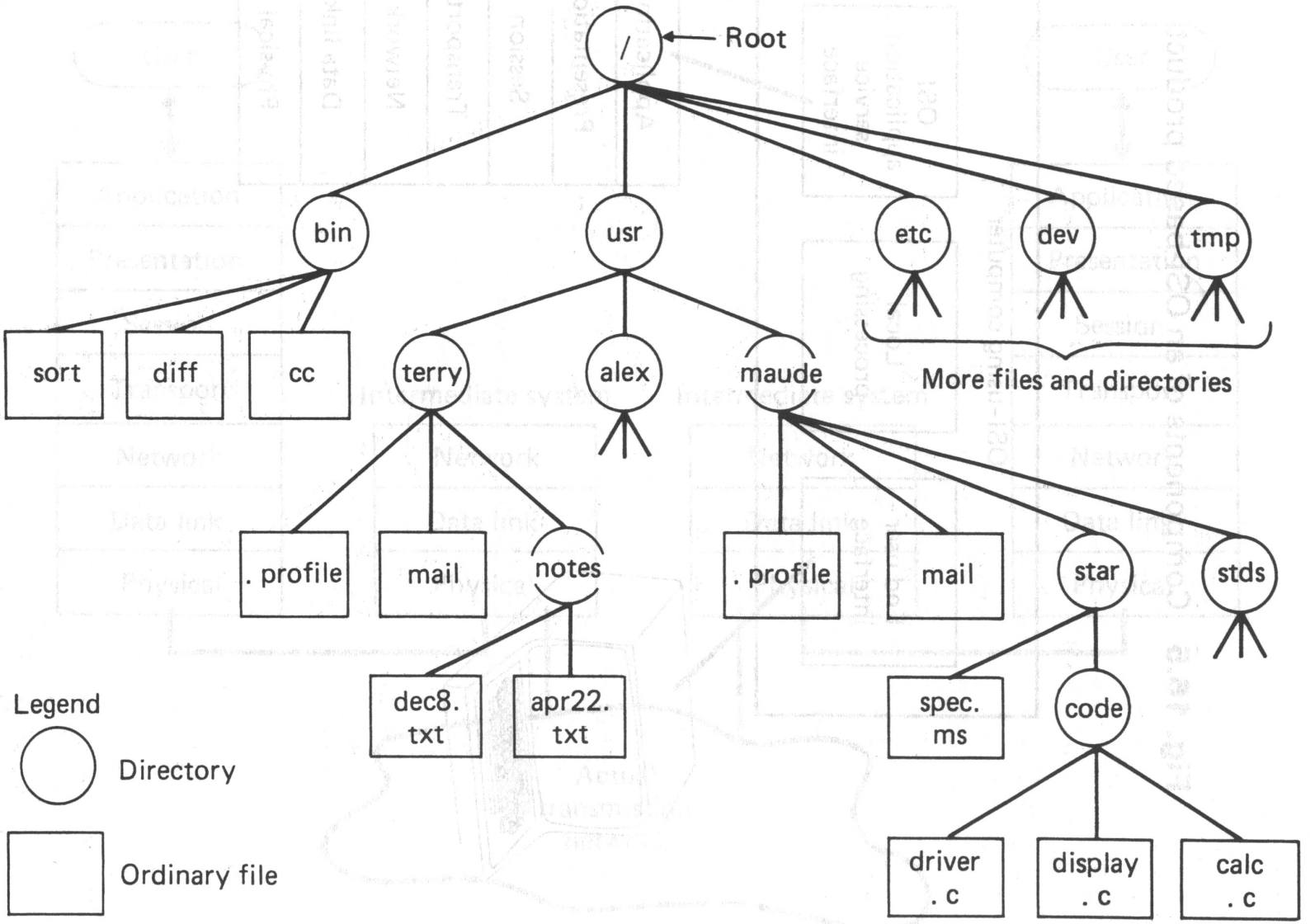



Getting Directories

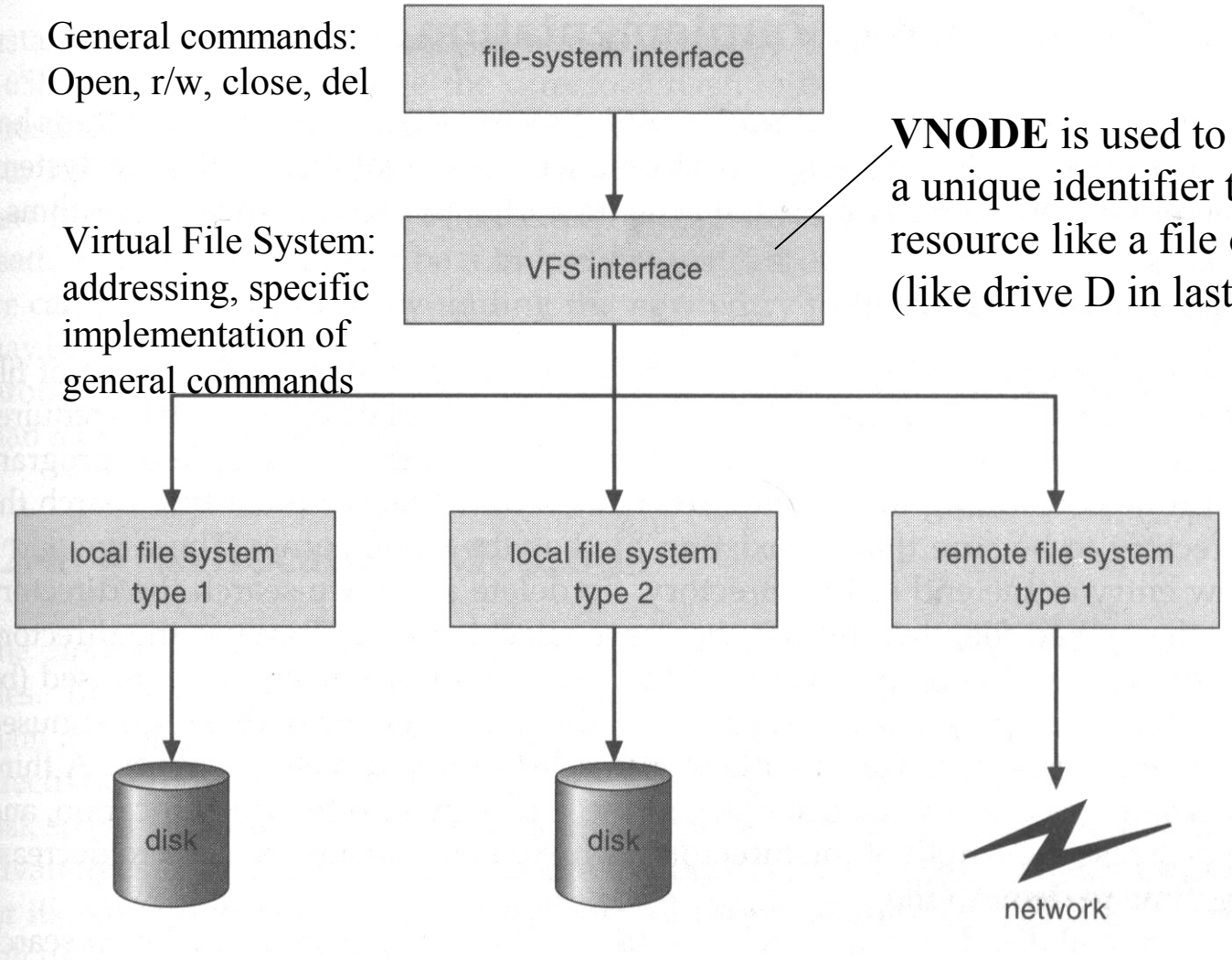
- If the directory tree is implemented as a linked list (tree)
  - Mounting is adding a node to the linked list
  - It then appears as an entry when 'ls' and 'cd' used
- If the directory tree is implemented through the FAT then the mounting point requires a new row entry, and that would look like a file.

Examples?

The Unix File System Example



Virtual File Systems



You can 'ls' Vista, Linux or Mac and it all works the same from the top.




Remote Mounting Methods

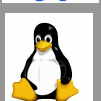
- Distributed File System (DFS)
  - Remote directories are made visible though software
  - Must login with user name and password
- Network File System (NFS)
  - Network directory mounted on a drive letter after login validation and verification that user's ID is logged as a valid member/user of the directory.
    - Local directory x → network sever address : / usr
- Domain Name System (DNS)
  - Host machine name mapped to internet address (local):
    - Local directory x → mimi.cs.mcgill.ca : /usr
  - Then it functions much like NFS
- Network Information Service (NIS) ~Sun
  - Unsecured centralize database of all servers, users, printers and directories.
  - Uses internet IP addresses like DNS




File Security Methods

- File system records the following for every file and folder:
  - Owner,
  - Read, Write, Execute privileges by:
    - Owner
    - Friends (general or can set groups)
    - Public
  - Mountable if you have access right
    - User name and password on server of directory
    - You have r/w, or x privilege

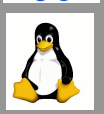





Part 2

File System Implementation



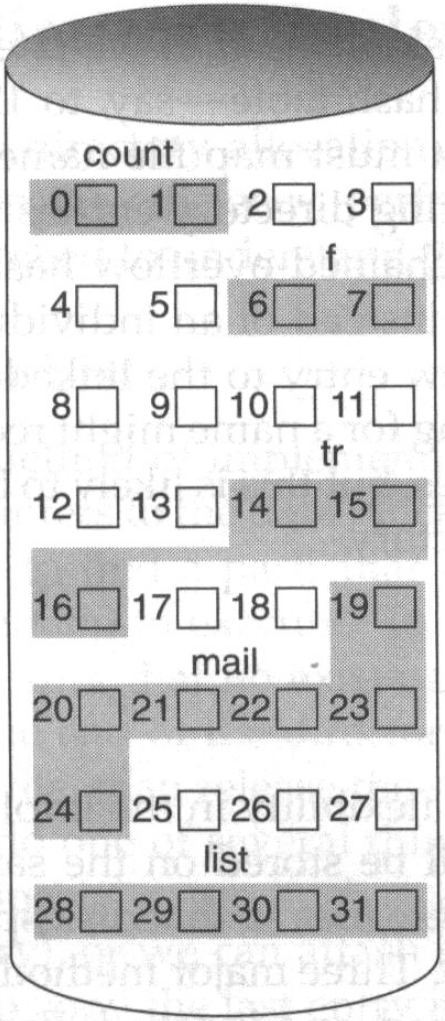



RAM Directory Implementations

- Tabular
  - A linear list of available files
  - One table for each folder
  - Each entry is a file or pointer to another table
  - Locate files using a linear search O(n) search, O(1) insert
- Hash Tables
  - Similar to tabular but files are located using a hash function to index to the file
  - Hash collisions are handled by linear searches from the point of the crash down the list. O(1) or O(n)
- B-Tree
  - Directory stored as a Tree
  - O(log n) search, O(n log n) insertion

File Allocation Methods

Contiguous Allocation



directory

file	start	length
count	0	2
tr	14	3
mail	19	6
list	28	4
f	6	2

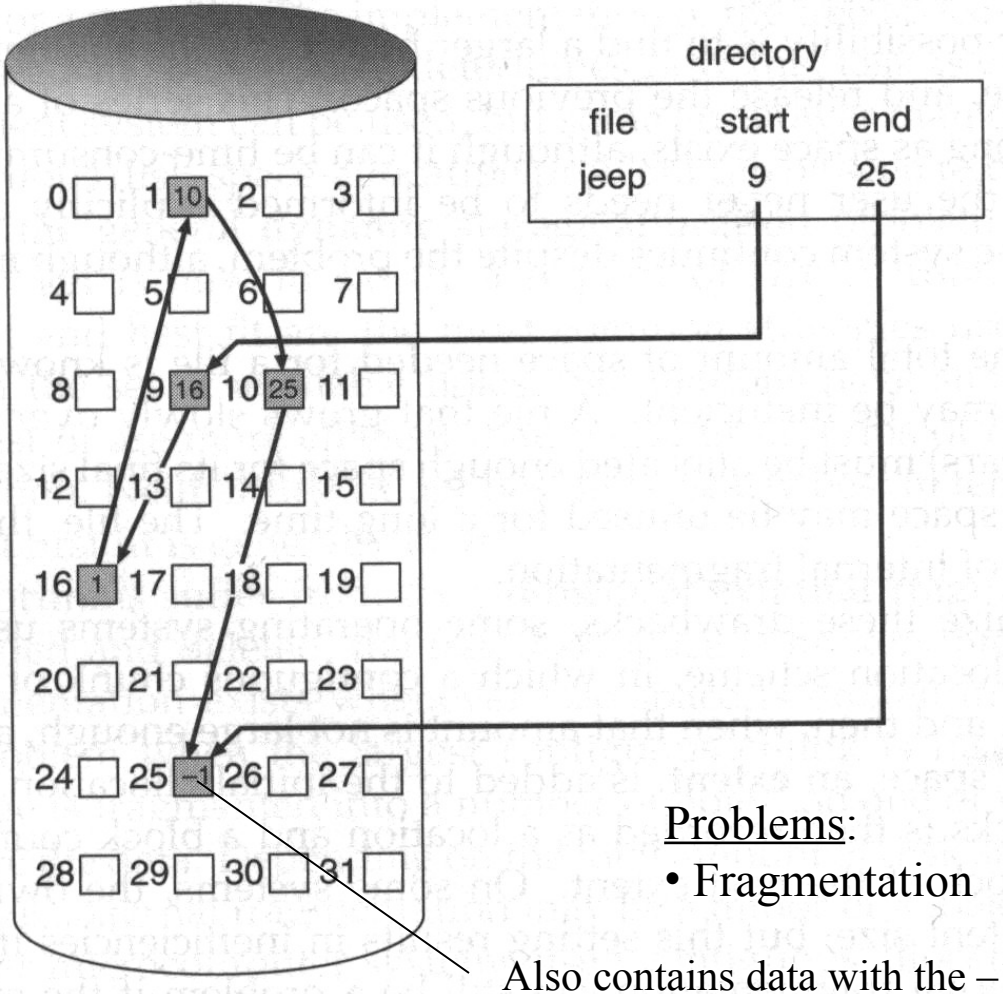
No EOB pointer needed (EOF used anyway)

Problems:

- Fragmentation
- Enough continuous space
- Multiple file handles (how much space to leave?)

File Allocation Methods

Linked Allocation



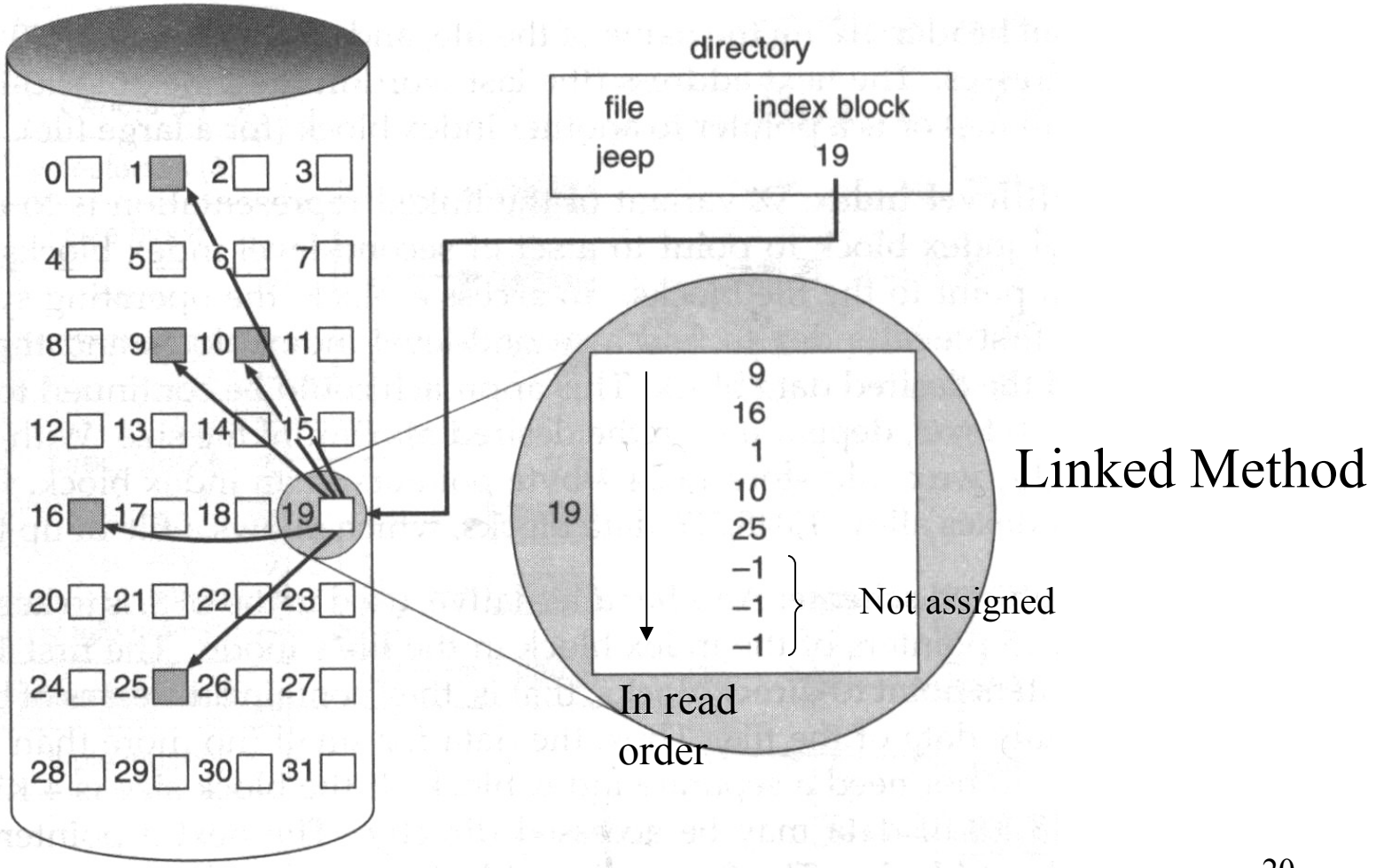
Problems:

- Fragmentation

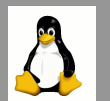
Also contains data with the –1 at end for EOB

File Allocation Methods

Indexed Allocation







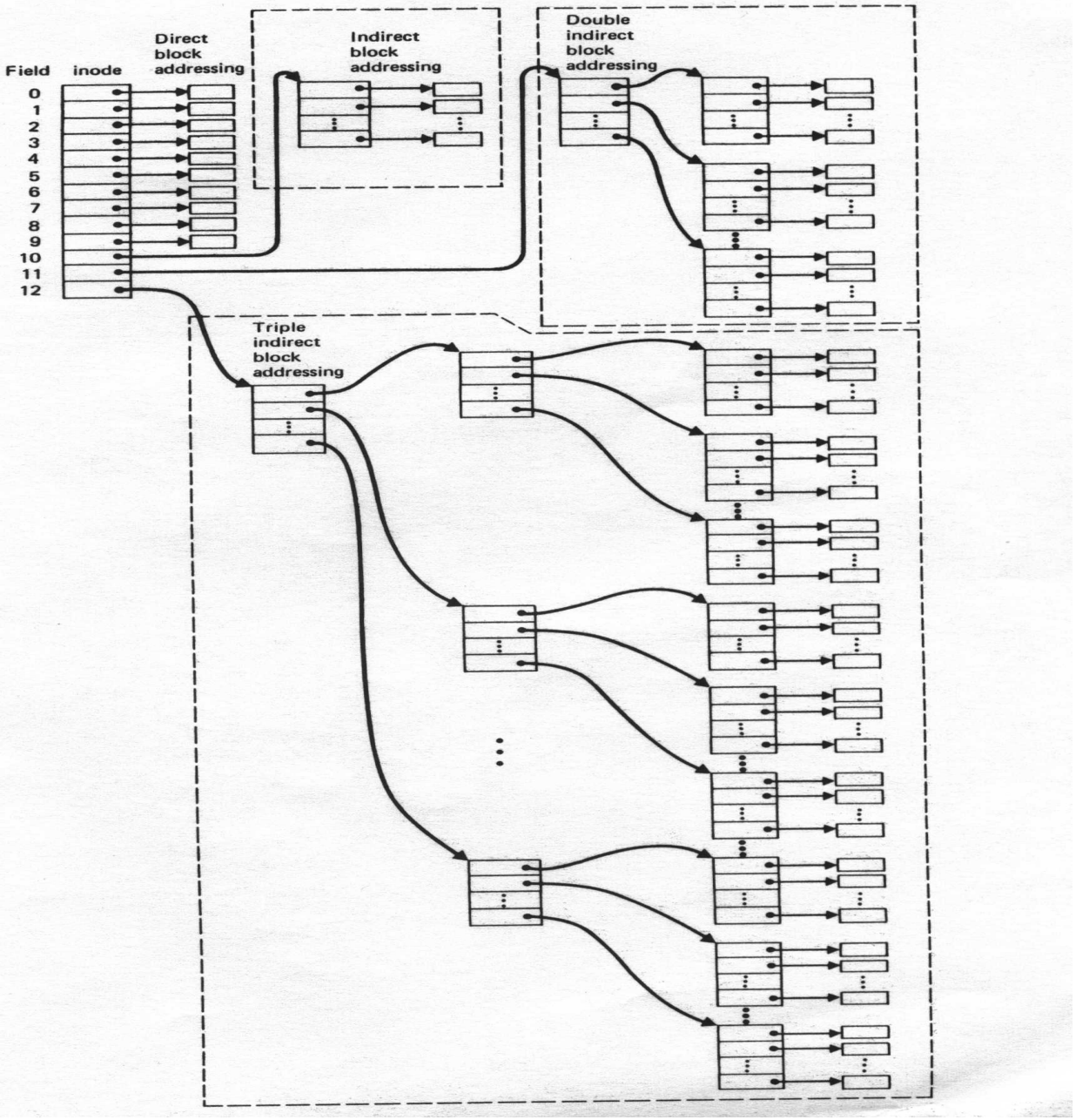

Other Indexed Allocation Methods

- Multilevel Index
  - First index block is a pointer to a maximum of n simple Linked method blocks that point to files.

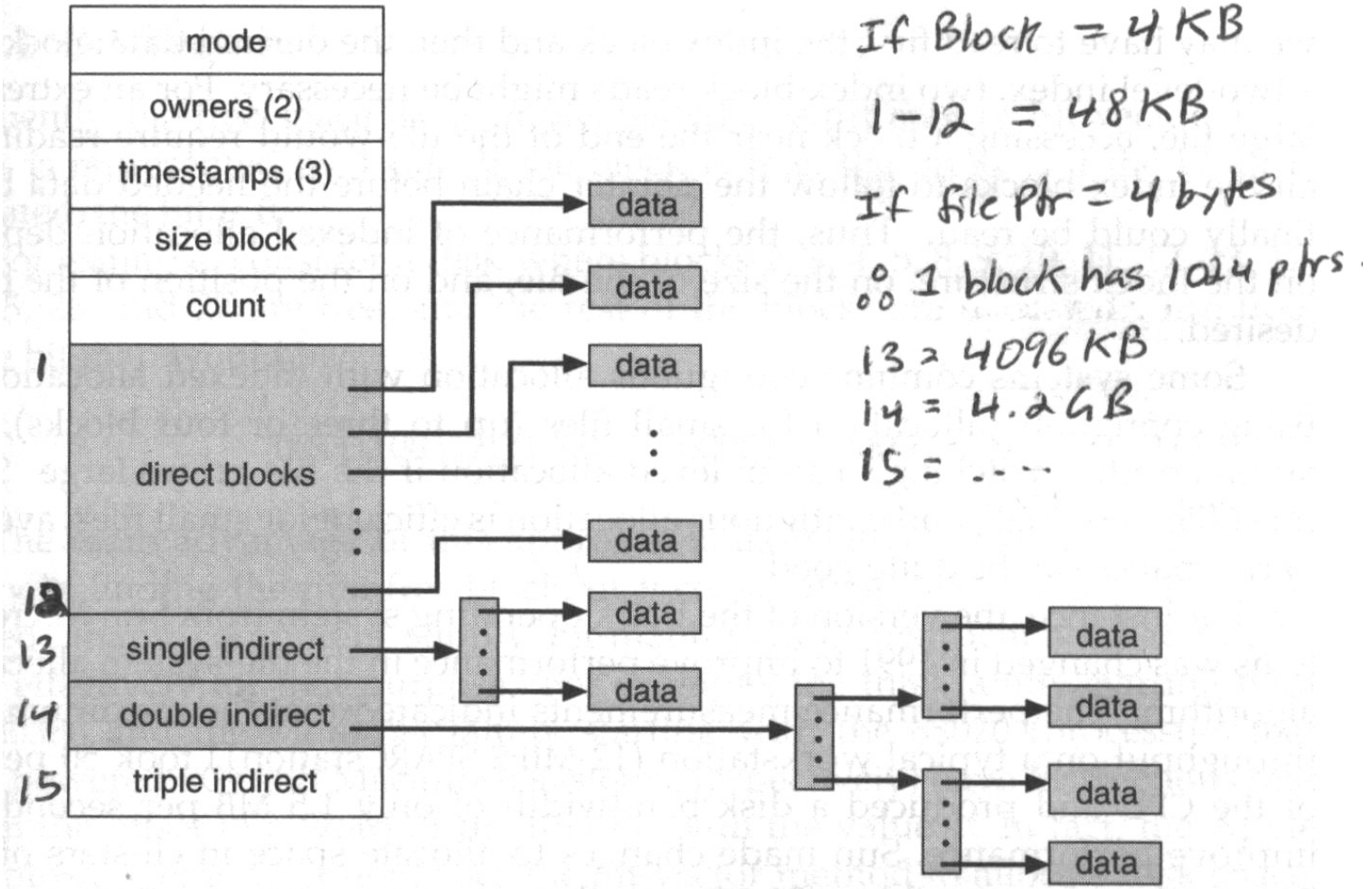
- Combined Index
  - For example the Unix File System does this:
    - 15 pointers to files
      - 12 directly to file data blocks
      - 1 points to a Linked block
      - 1 point to a Multilevel index block (2 levels)
      - 1 pointer to a Multilevel index block (3 levels)



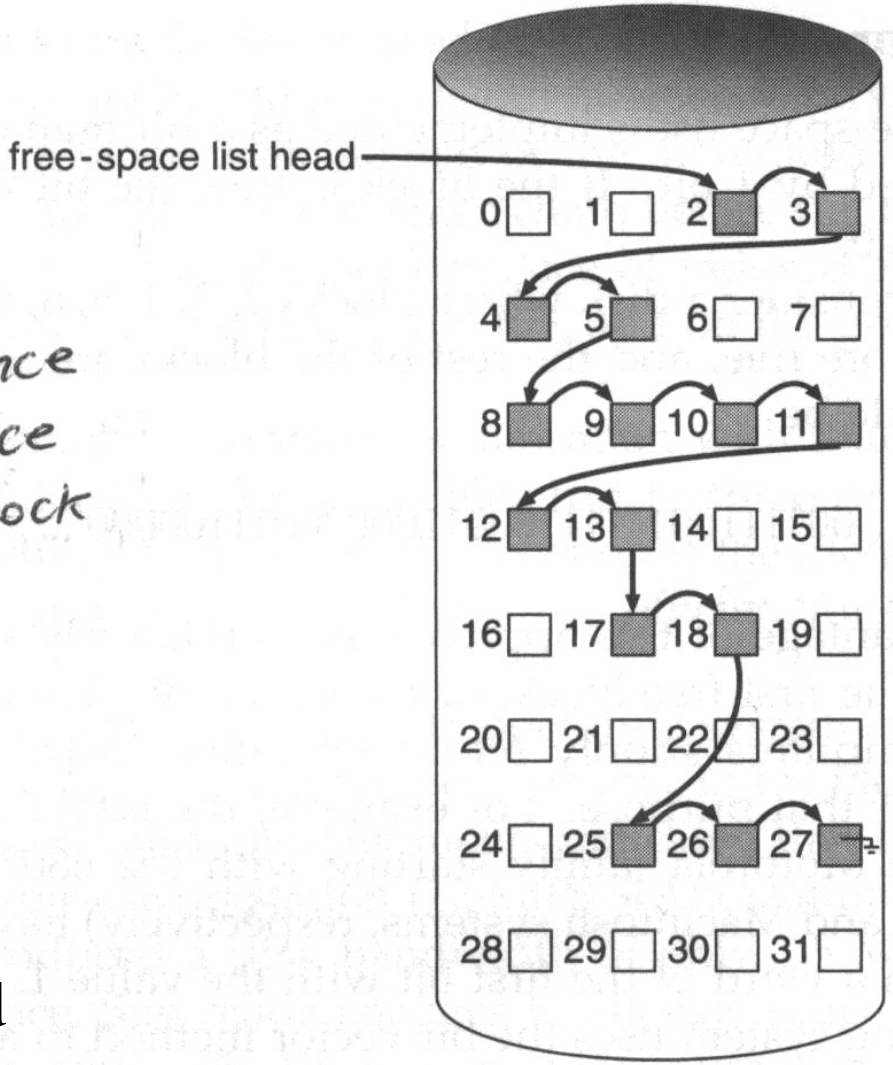

Multiple indirect block addressing used in inode to locate every block in a file.



Example: Unix inode Method



Managing Free Disk Space



Not O(n) since when free space needed, 1st block removed only.

Pointer Method

Formatting?




Managing Free Disk Space

The Bit Vector Method

- Stored on disk (but loaded in RAM to improve performance)
- 1 bit = 1 block (where 1 means free / 0 means used)

block No = (bits in word) X (No. of 0 words) + offset to 1 bit

00011111000000111110000 0 etc.

Eg/ The size of a Bit vector table?
1.3 GB HDD with 512 byte blocks = about 332KB
(fast only if stored in RAM)(used by MAC)

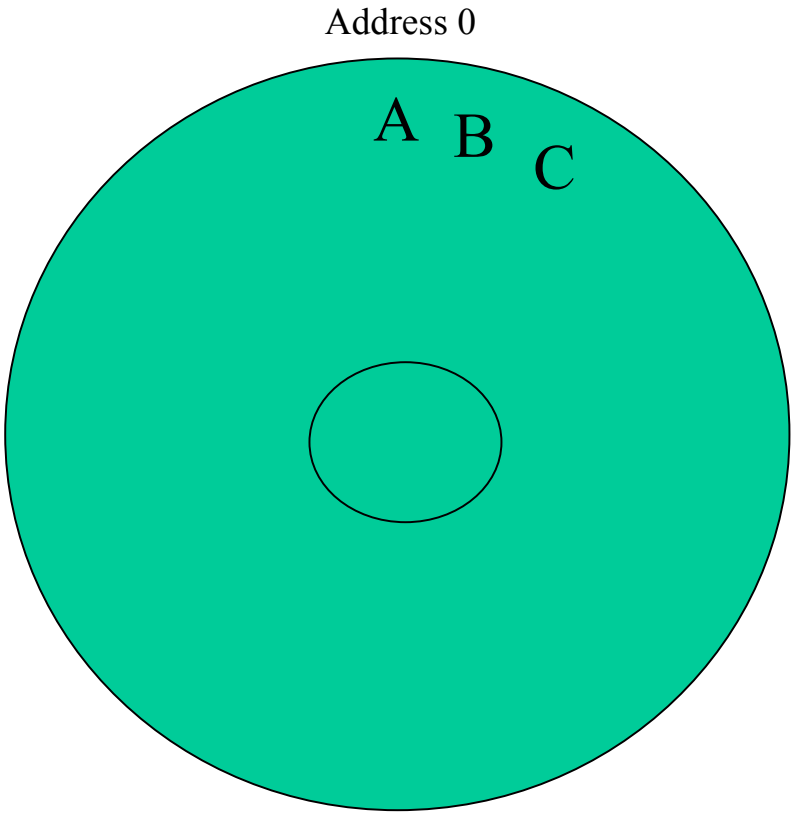



Part 3

OS Management of File Systems




Basic Disk Layout

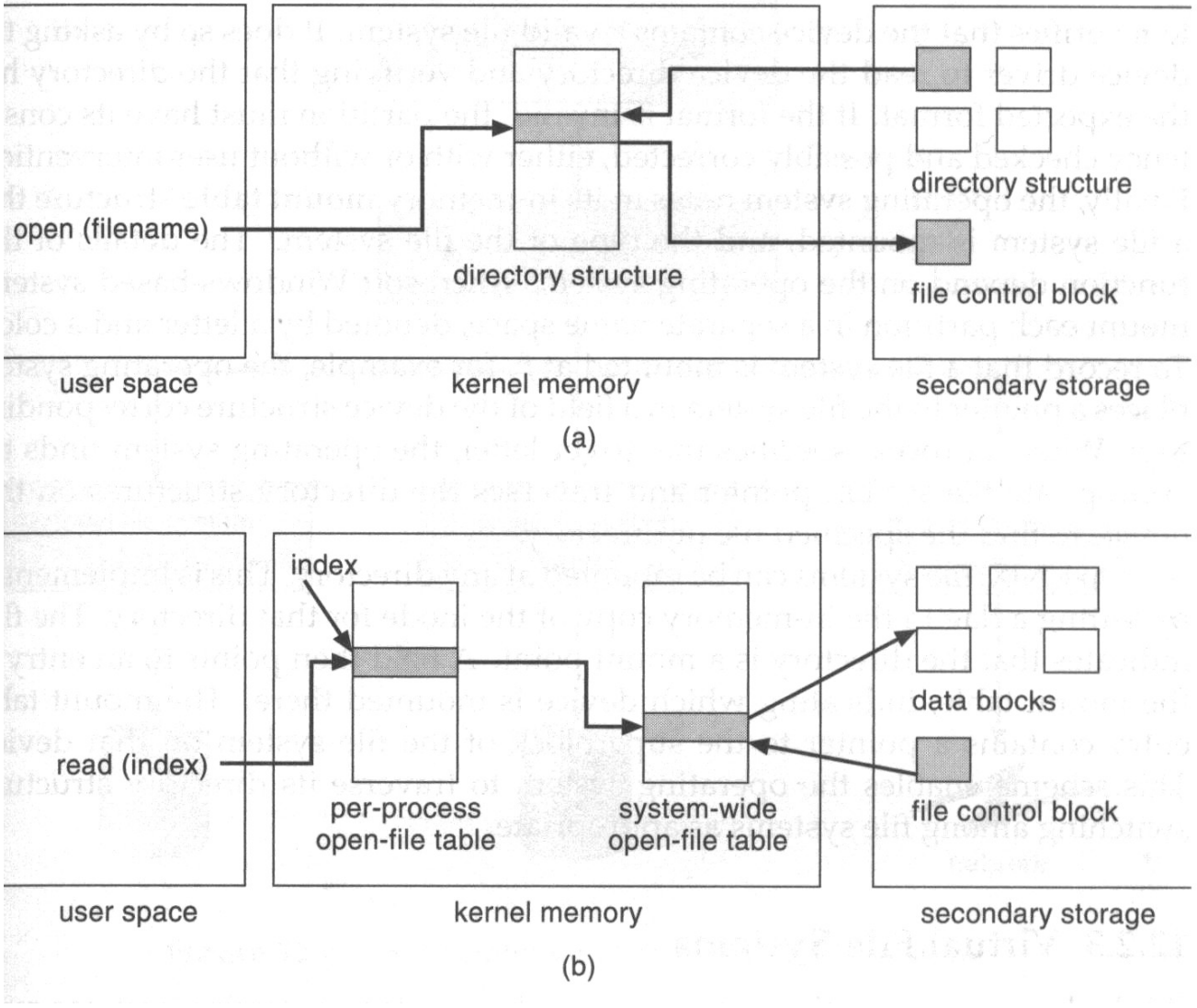


A = Boot Control Block
B = Partition Control Block
C = Directory Structure
D = File Control Block (FCB)
(either in C or external)

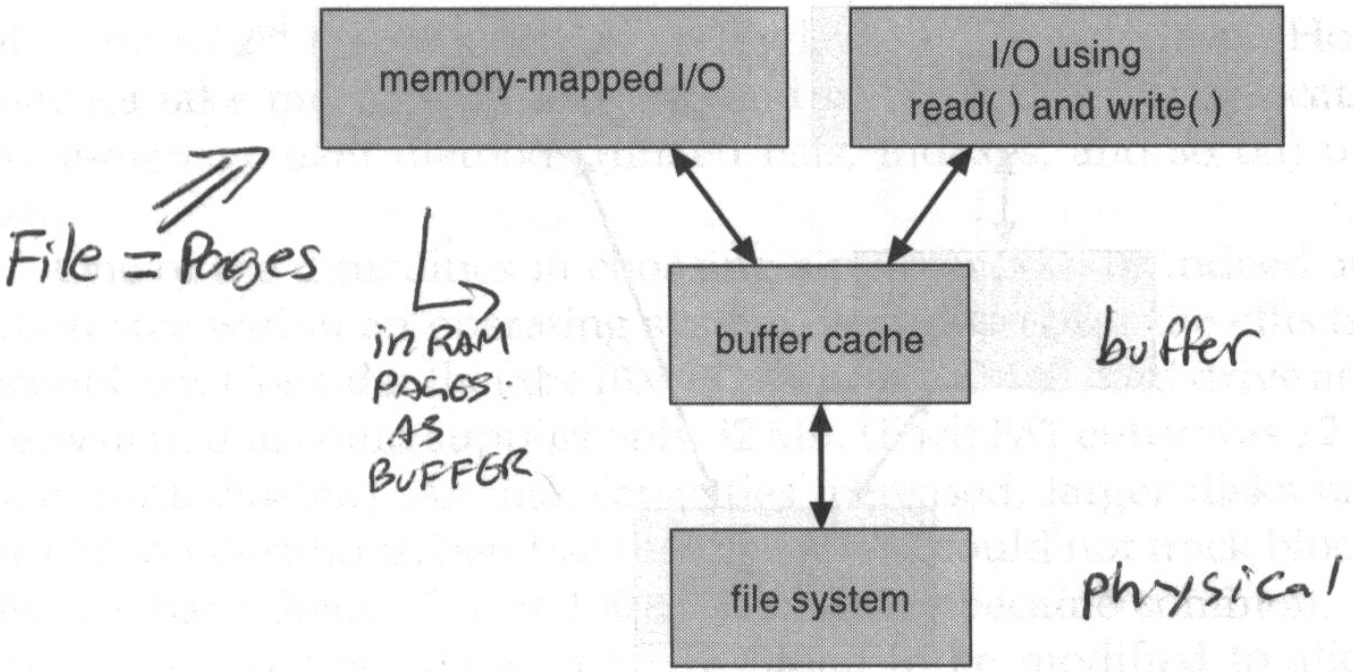
Block = track/sector intersection or fixed byte size

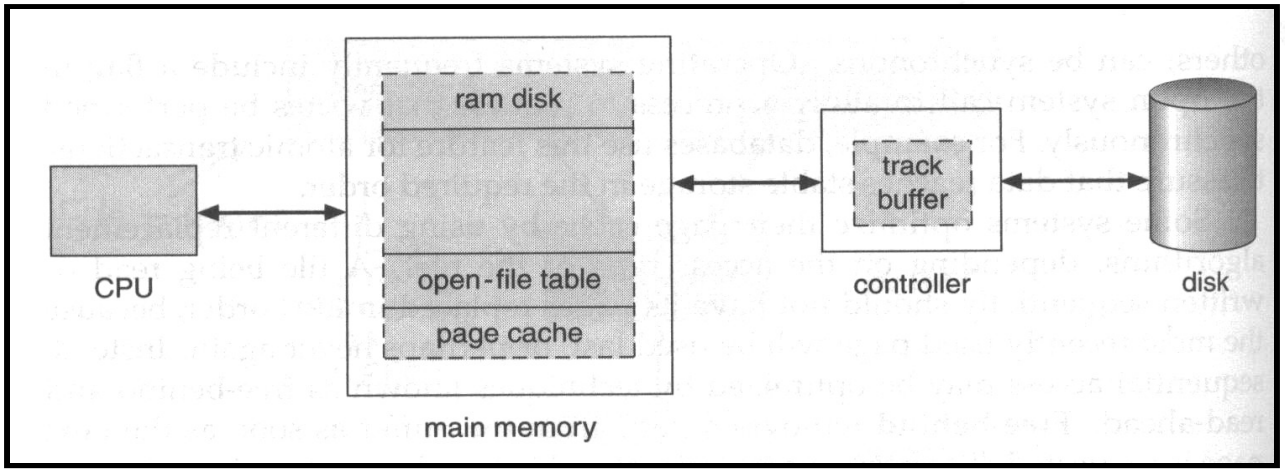
a.k.a FAT

OS Management of Directories

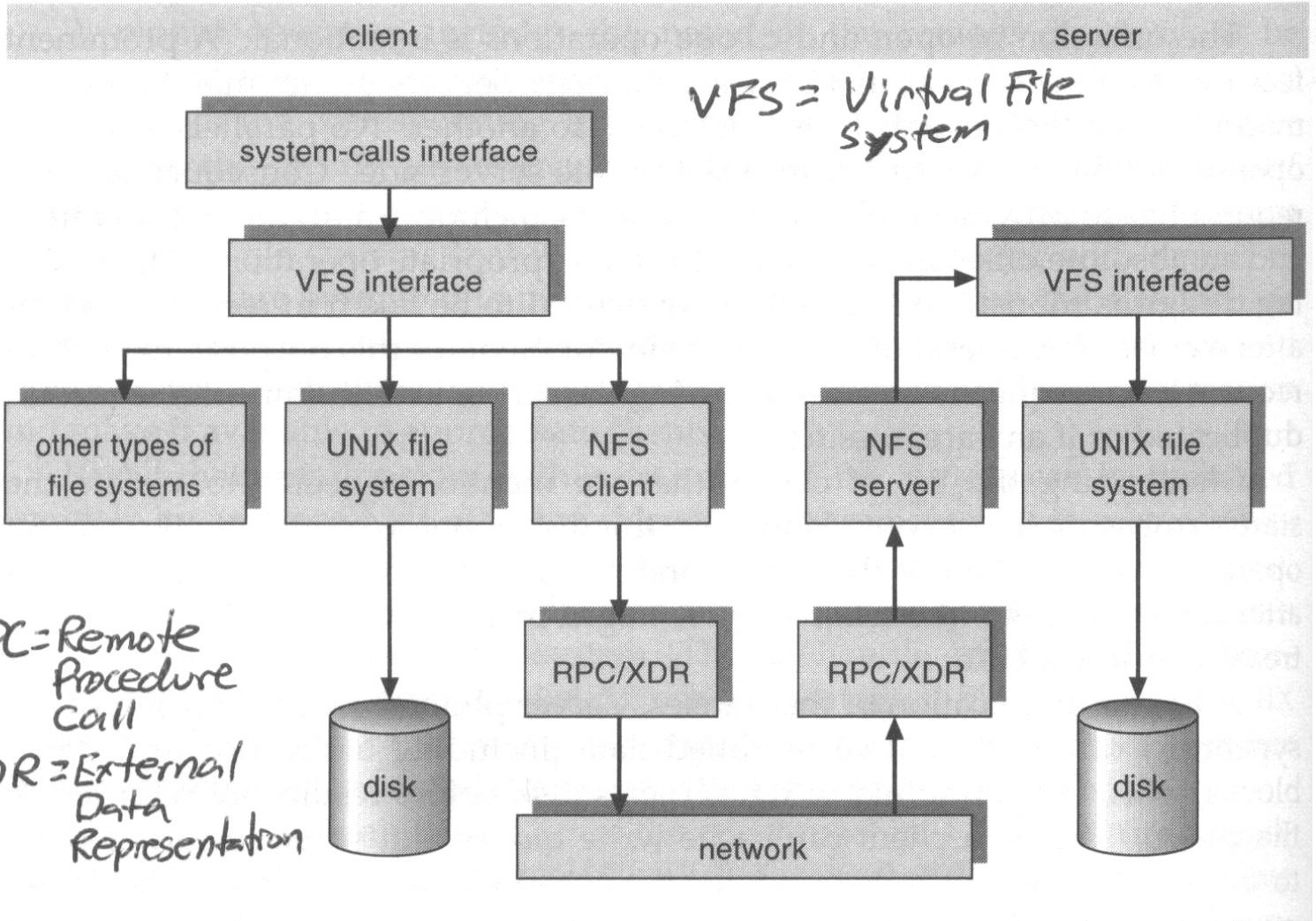


Buffers & More Buffers…

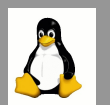




Example NFS OS Architecture








Question

- How could we implement social networking through directory structures?




Part 4

At Home




Things to try out

1. Using the command line prompt mount a drive letter to an external network directory using Unix
2. Now do this using the window from the command-line interface and the windowed interface.
3. Web Resources:
   - http://www.beginningtoseethelight.org/fat16/
   - http://www.dewassoc.com/support/win98/fat.htm
   - http://home.no.net/tkos/info/fat.html