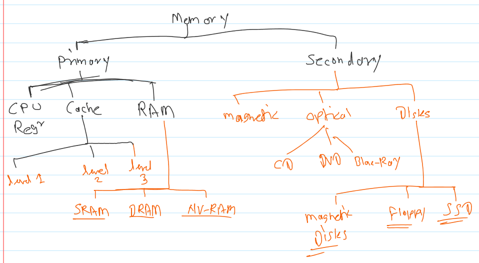
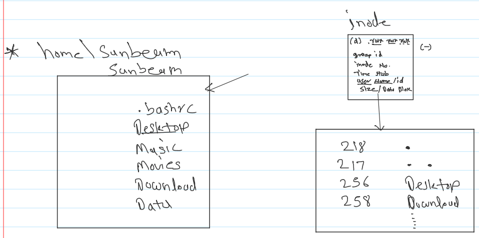
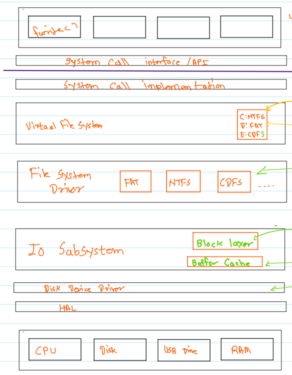
[[[

22 November 2023 08:38

① memory and Storage







user program

open(), close(), read(), write()

Sys_open(), Sys_close(), Sys_read(), Sys_write()

These is mount table it has partition info (File System) // To check $ mount | grep "sda"

VFS redirect FS required to appropriate FS Manager.
Define std structs for all FS & use function pointer
e.g. inode, file, file_struct, file_operation

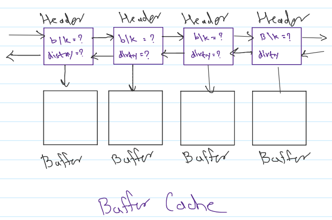
File system layout on disk partition.
✓ superblock, inode list, data blocks ....(Info about partition
✓ which operation is done like Read / write on perticuler sector

- maintains waiting queue for rd/wr operation on disk.
- use some scheduling Algo. to schedule them.

It depen on os if
Windows = FAT, NTFS, CDFS
Linux = FAT, EXT 3/4, CDFS

Handle speedmismatch between RAM & Disk.

Read / write sector



Buffer Cache

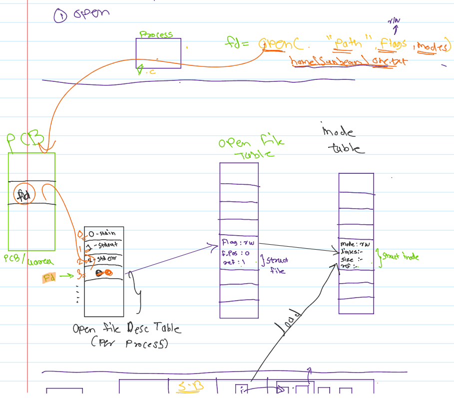
- Buffer cache kept in RAM
- Internally it is link list (Doubly link list)
- Every buffer has header (size 4kb)
- Header kept info like buffer Block where in Disk
  └ This Buffer dirty buffer or Not

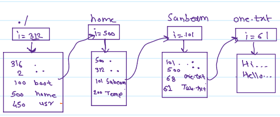
Dirty Buffer : If we read data from buffer and after we can modify Disk content of that particular buffer..... means file has
- diff data on RAM and disk is known as Dirty Buffer
- If copy is both place same it not be Dirty Buffer.
-

$ free -mh

* Unix File Sys Call

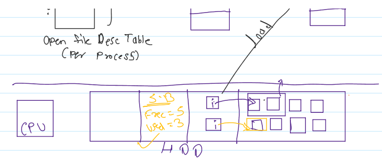
① open





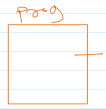
⊛ open() sys call

① FilePath -> inode number (namei)
② load inode into in-memory inode table / cache
③ - make an entry into open file table
④ add a pointer into open file descriptor table
⑤ return index to open File Table entry i.e. file descriptor



③ make an entry into open file table
④ add a pointer into open file descriptor table
⑤ return index to open file table entry i.e. file descriptor

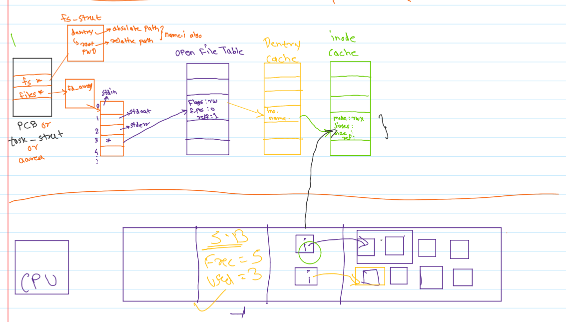
② Linux



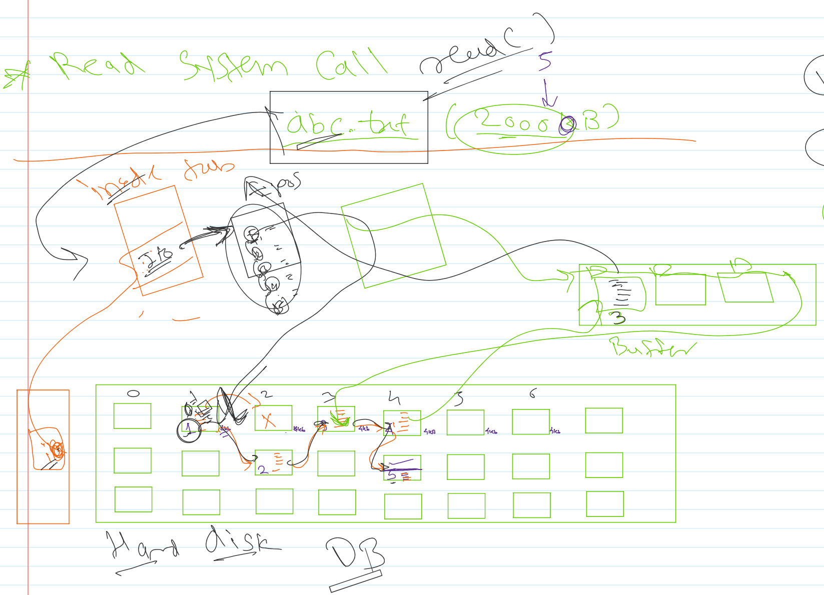
rd | wr

fd = open("file path", flags, mode);

home / sunbeam / one.txt

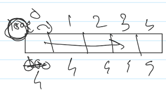


* Read System Call read( )



① To set file position
② Check inode table data entry
③ entry block id
- → Check in buffer
  - └ Data Return
- → Device driver it goes on Hard disk
  - → Read data
  - → Make Entry on buffer
    - └ Data Return

* Contiguous space algo →



② Link space algo →

③ Index →