Block Driver 2 Day -:

==================

#disk on ram -> loaded driver -> disk created on the ram -> unloaded -> ram was released

#loaded – malloc(512 KB) ------>> read/write on ram

path -: BlockFSDriver/Stage2Ramdisk

apis.txt

Makefile

partition.c -: partition layout

partition.h

ram\_block.c -: Block driver (Verticals)

ram\_device.c -:

ram\_device.h steps.txt

files -:

ram\_block.c

Todo 1 -:

in char driver

char = regter char driver + register file operation + create device file

block = allocate gendisk + add disk

Having device file without file opertaion no need

block = allocate gendisk+ rgister fops + add disk

file -: include/linux/blkdev.h

endif /\* CONFIG\_BLK\_DEV\_INTEGRITY \*/

struct block\_device\_operations {

int (\*open) (struct block\_device \*, fmode\_t);

void (\*release) (struct gendisk \*, fmode\_t);

int (\*rw\_page)(struct block\_device \*, sector\_t, struct page \*, unsigned int);

int (\*ioctl) (struct block\_device \*, fmode\_t, unsigned, unsigned long);

int (\*compat\_ioctl) (struct block\_device \*, fmode\_t, unsigned, unsigned long);

unsigned int (\*check\_events) (struct gendisk \*disk,

unsigned int clearing);

/\* ->media\_changed() is DEPRECATED, use ->check\_events() instead \*/

int (\*media\_changed) (struct gendisk \*);

void (\*unlock\_native\_capacity) (struct gendisk \*);

int (\*revalidate\_disk) (struct gendisk \*);

int (\*getgeo)(struct block\_device \*, struct hd\_geometry \*);

/\* this callback is with swap\_lock and sometimes page table lock held \*/

void (\*swap\_slot\_free\_notify) (struct block\_device \*, unsigned long);

struct module \*owner;

const struct pr\_ops \*pr\_ops;

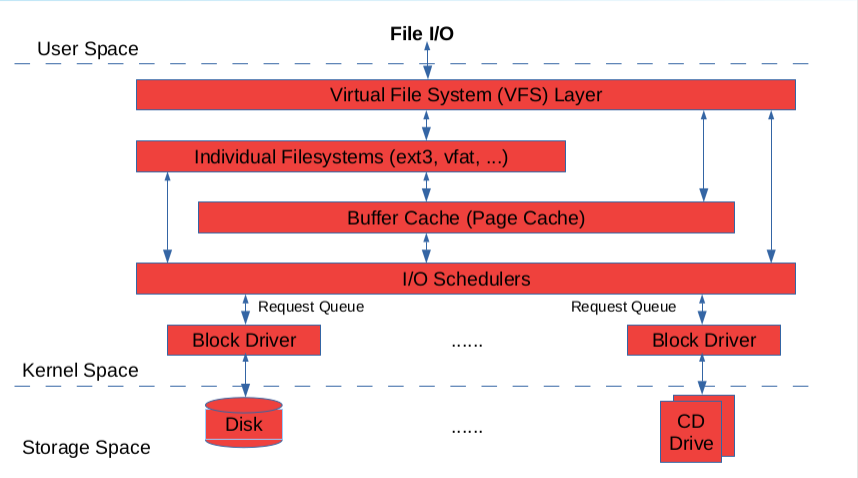
};

In files there are not read or write operation.

From userspace we are not exposing read/write system call.

Why?

block input /output



input /output

VFS = system driver +block driver

VFS knows which file sytem driver works

when we do read/ write operation it will store in some memory (Buffer cache) for frequesntly data.

When unmount or save file. Like all operation flush on the disk. So Buffer cache send request to Disk. So there are some request.

In char driver like direct read or write.

So i need a queue , so the block driver has queue. So block driver serve keep on processing request que.

bloack driver doesnt have directly interaction with disk(userspace). No direct dealing with read/write operation.

w.r.t char driver buffer cache + additional layer not be there

Lets Assume -> request comes

# sector 2 to sector N

# Sector 10---->>> Sector 5

So 1st serve request no 10 and then request Sector no 5 -: This is inproper

Another senariao

Sector 10 -->> Sector 9-->> Sector 7 -->>Sector 22 -->>Sector 4-->>Sector 8 -->>

So we need like elevator .