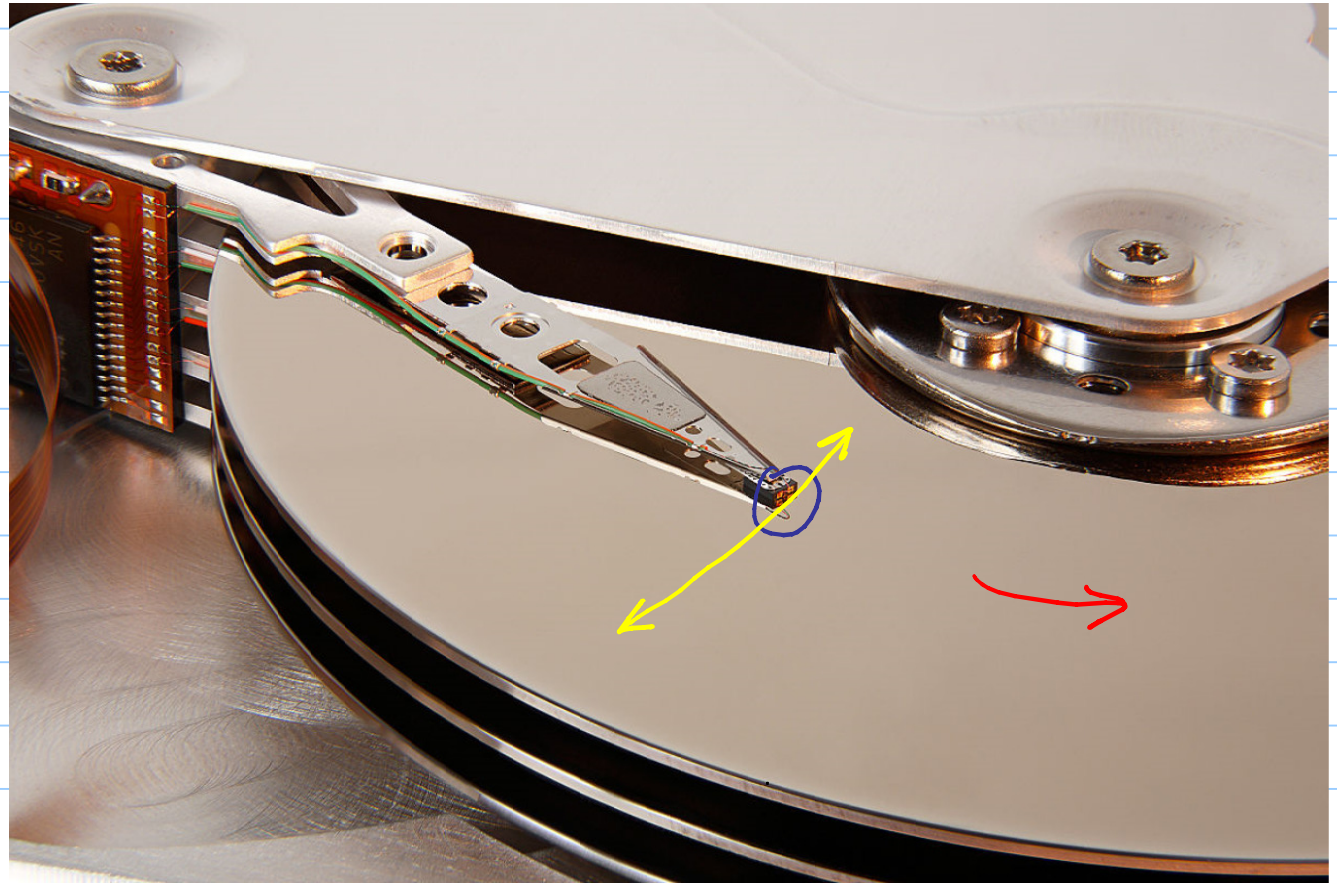
CSE 330: Operating Systems

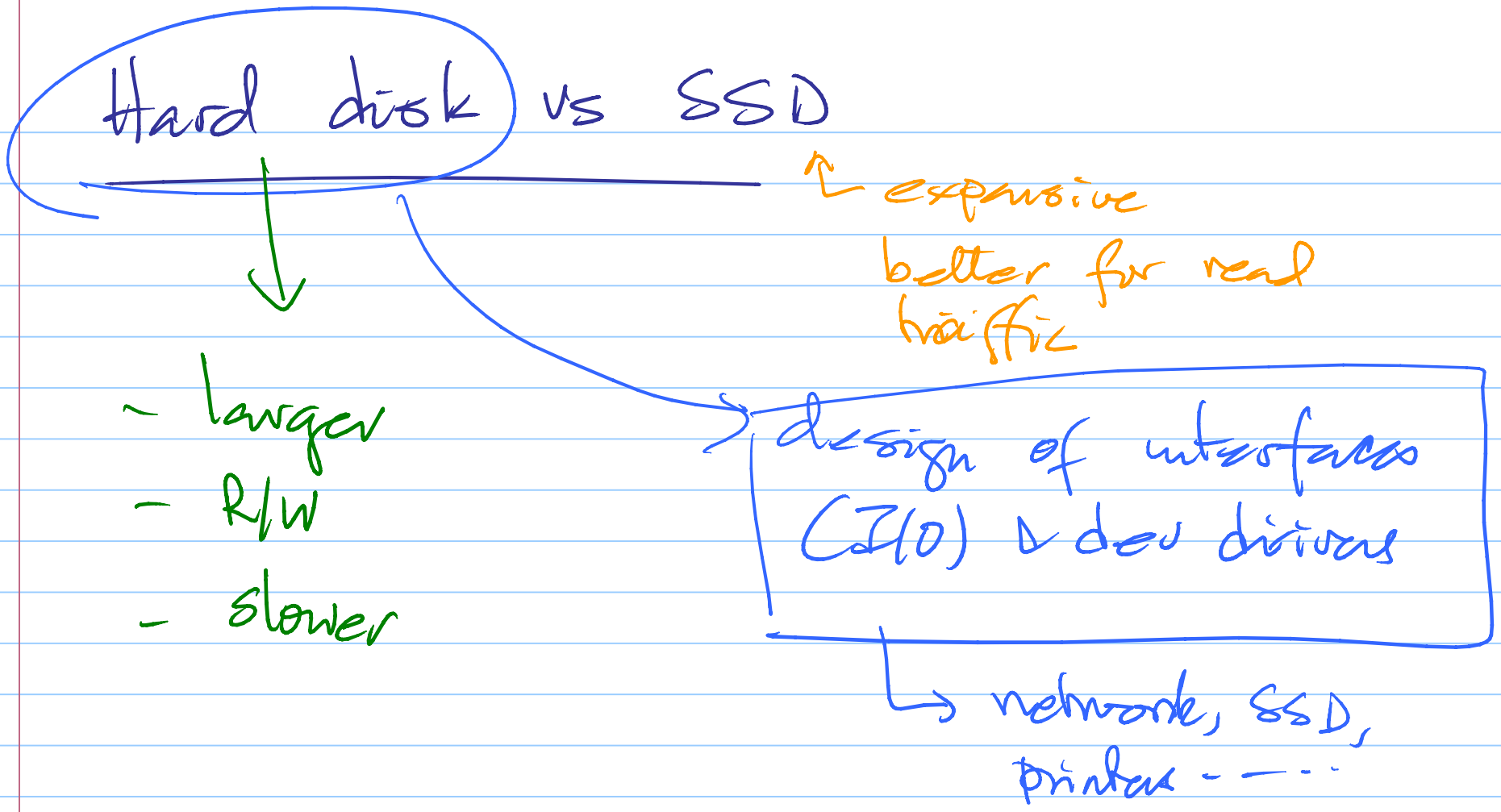
Class: 25 **Date:** 4/12

Spring 2018

Note Title



DISK HEAD SCHEDULING


Hard disk vs SSD
expensive
better for read traffic
- larger
- R/W
- slower
design of interfaces
CLI(O) & dev drivers
network, SSD, printer

scheduling – based on head movement

FCFS

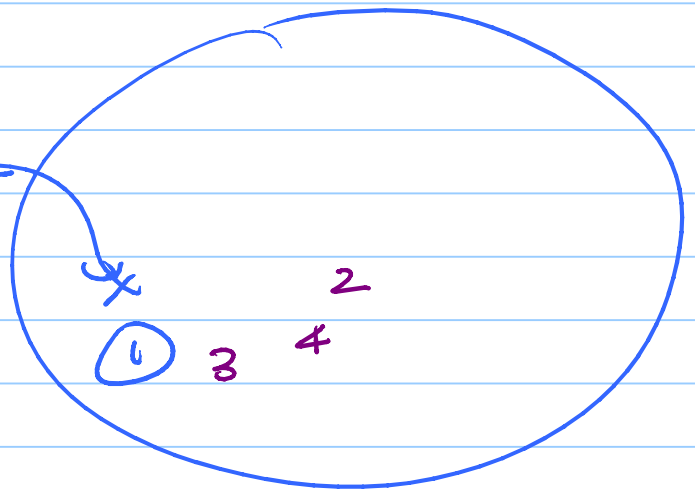
↳ too much head movement

---

waiting

2 3 4 5

current

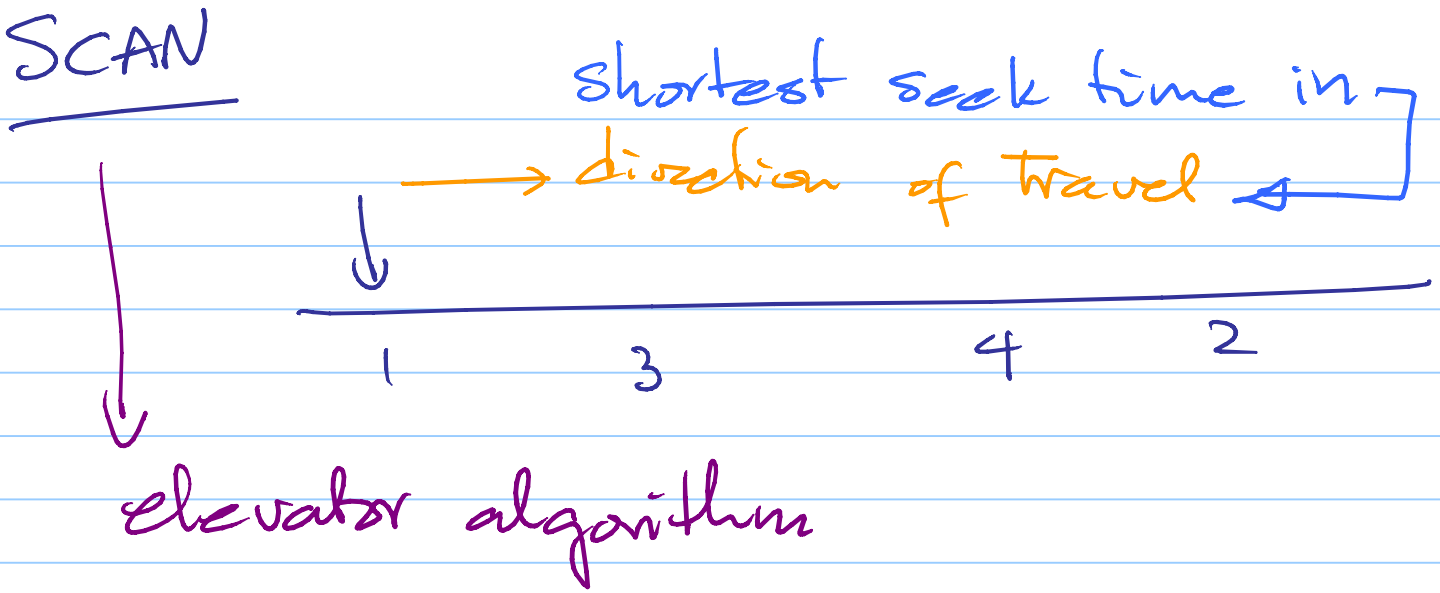


SSTF

→ shortest seek time first

new ones may get added

SCAN



elevator algorithm

circular

C-SCAN

↳ go to end & retract to beginning

only 1 direction of travel

LOOK → do not go to end (stop at last one.

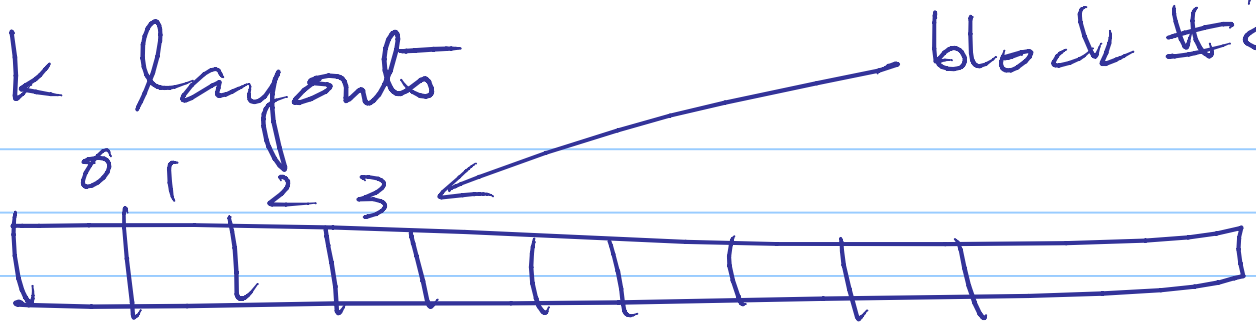
C-LOOK ↑ same but circular

where in the driver is the disk head scheduler?

```
read (or write)
    { P(disk_mutex)
```

FIFO → change the ordering of the queue here

Disk layouts



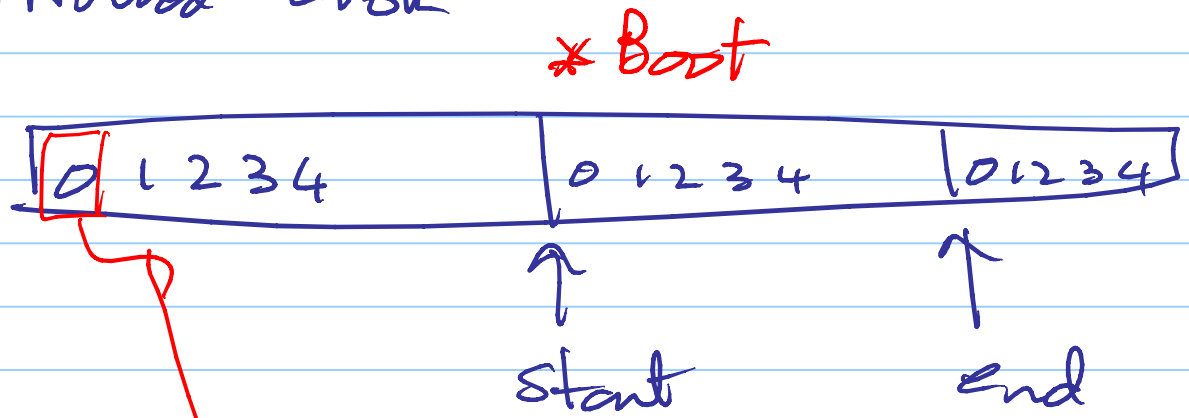
- Partitions → Partition table

	0	start	END
Boot *	1		
	2		

↑ Block #s

- Boot Partitions
- MBR (master Boot Record)

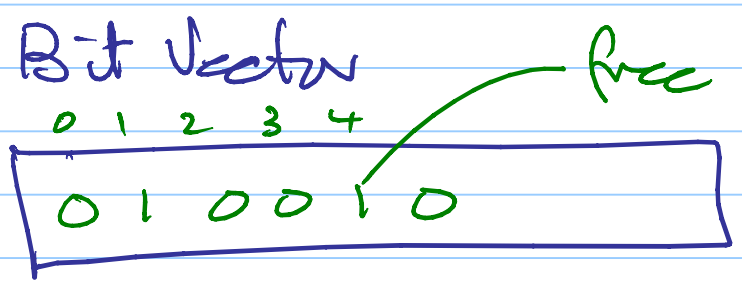
Partitioned disk



- Bios will pick one from the boot partitions

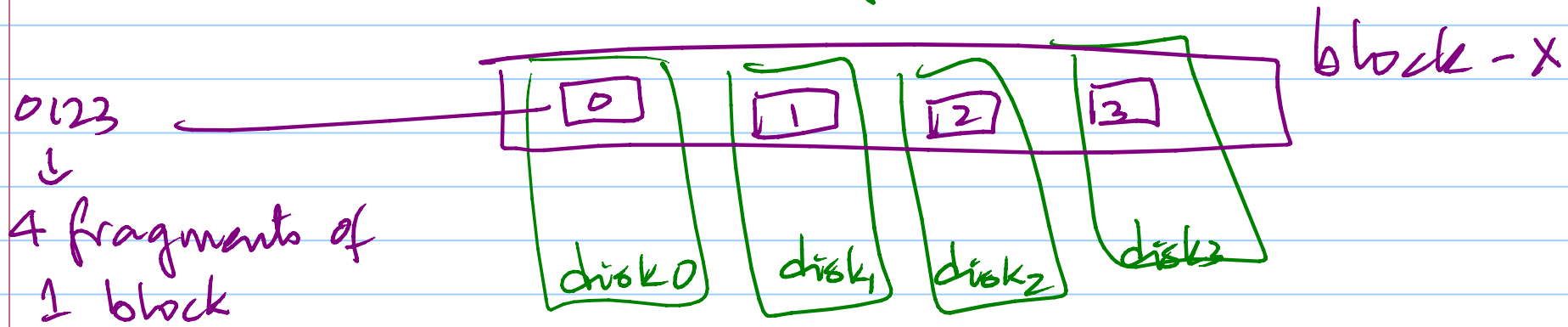
FREE Space (per partition)

↳ Bit Vector



RAID Redundant Arrays of ~~Inexpensive~~ Disks
Independant

RAID-0 → striping

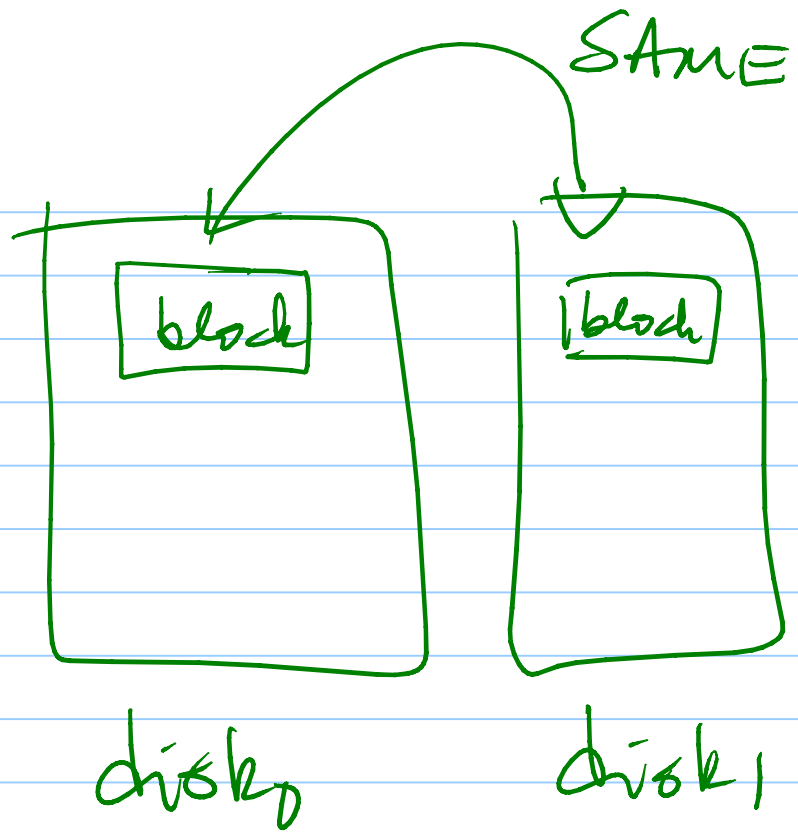


RAID - 1

mirroring

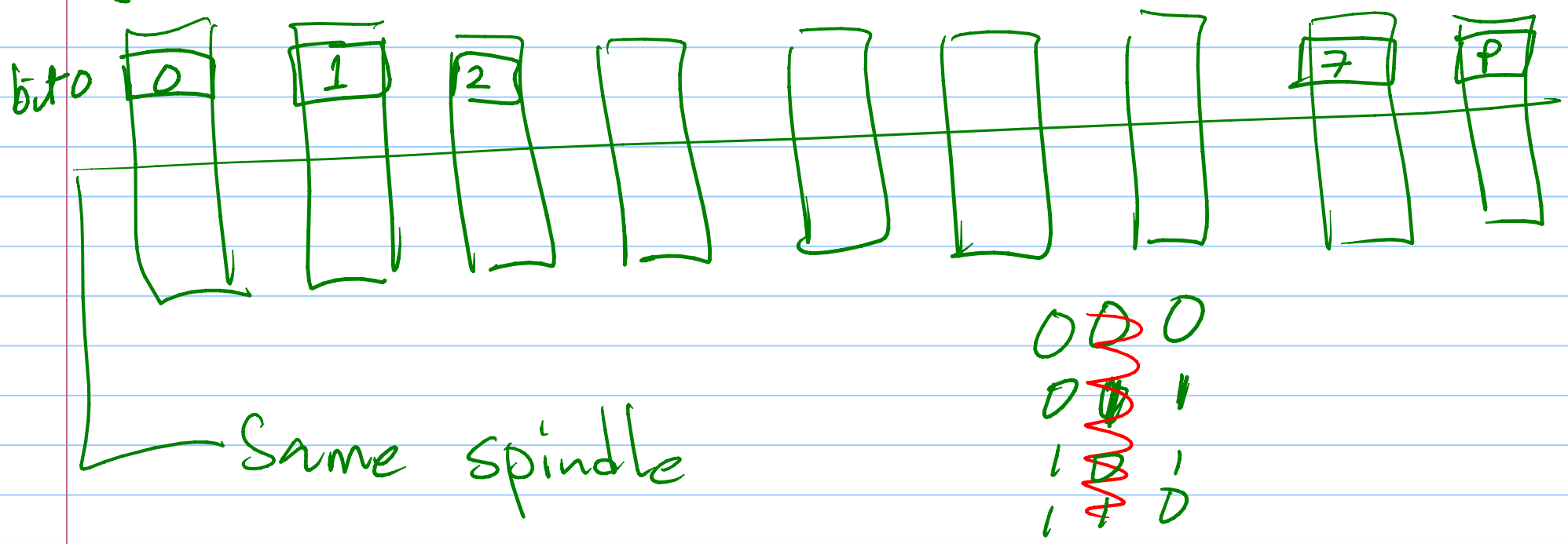
↓

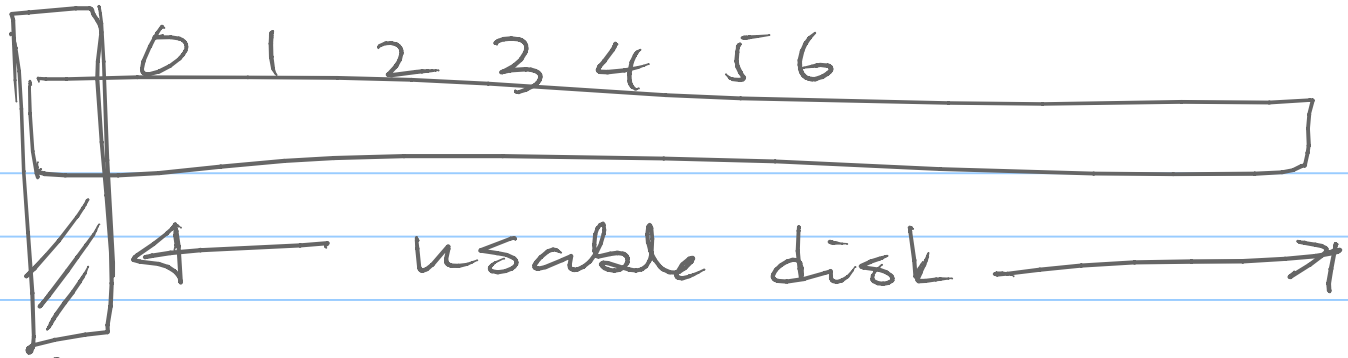
2 copies of everything.



RAID - 2 (parity)

byte is striped

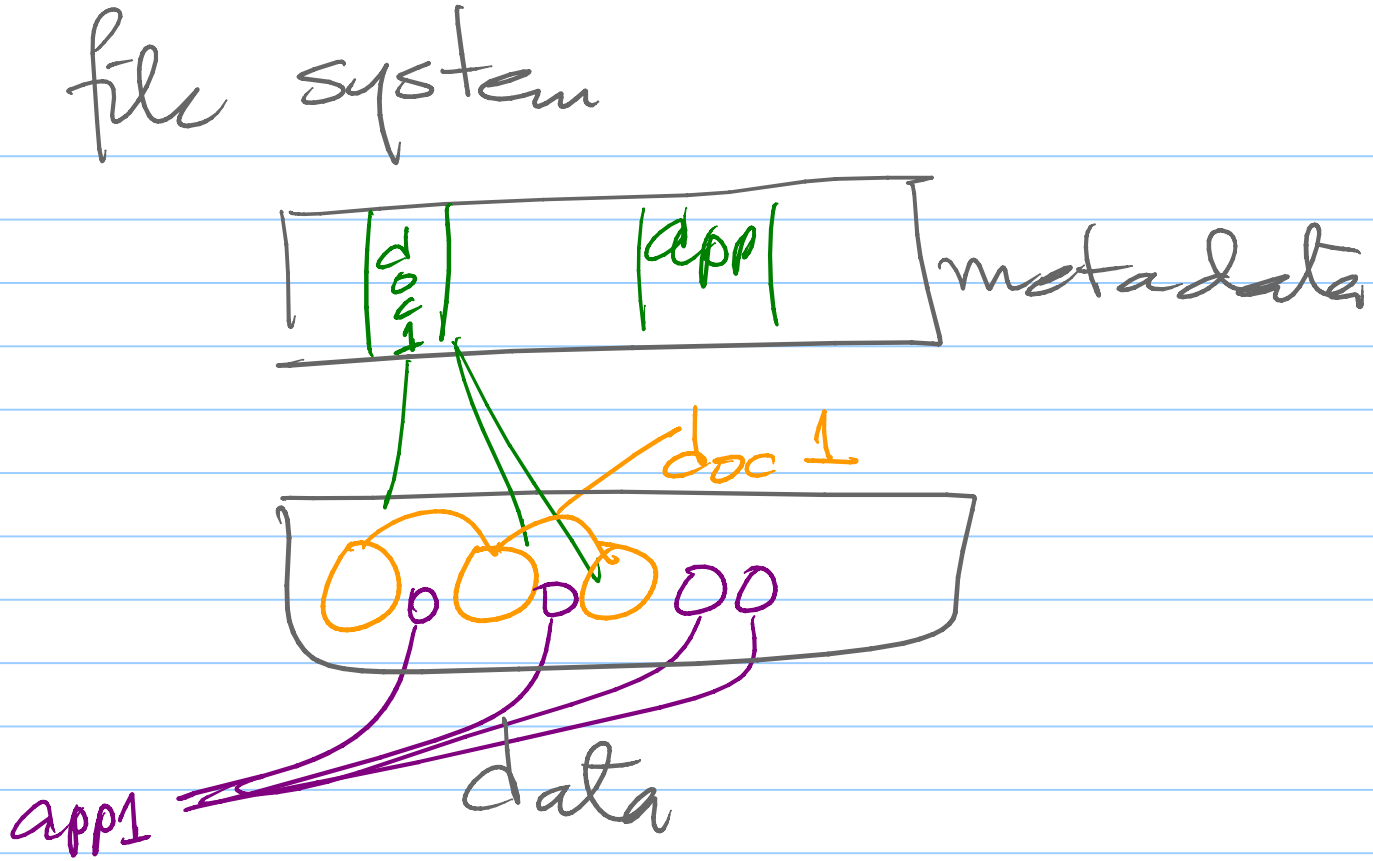




reserved for ...

↳ file system

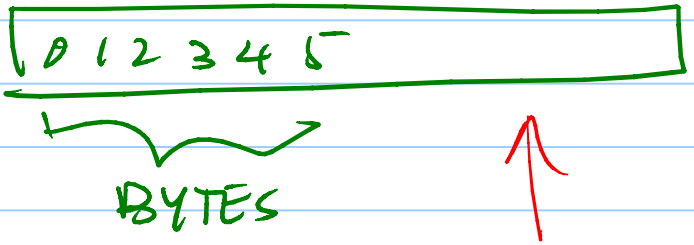
(disk) data organization done by the OS


file system
doc1
app
metadata
doc 1
app1
data

file → a sequential set of uninterpreted bytes

has attributes

- NAME
- OWNER
- creation date
- permissions
- size



contents

file types — none!

↳ extension

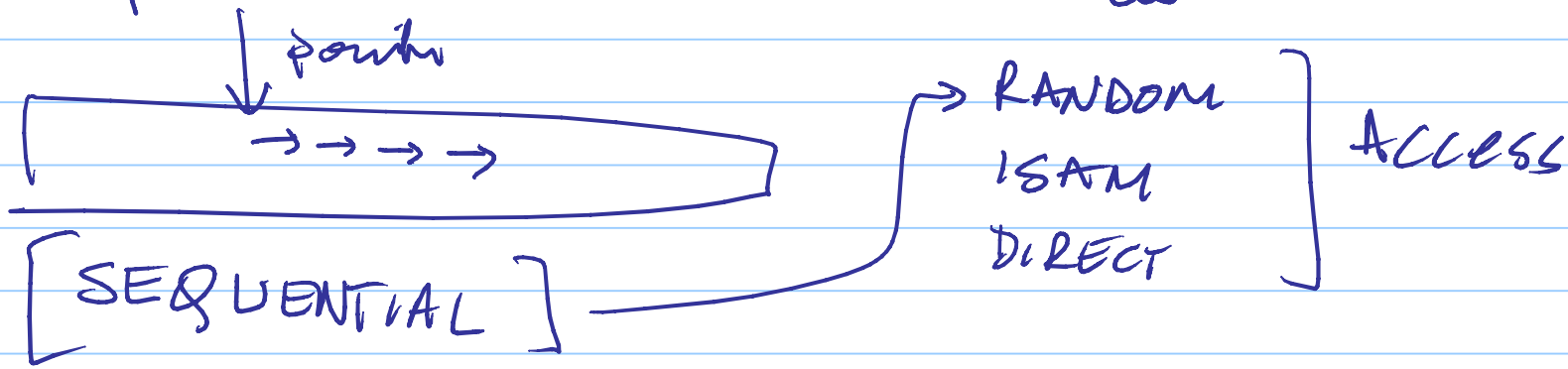
↳ magic numbers

File functions [set permission]

- create
- read, write
- delete

] system calls

position & access methods

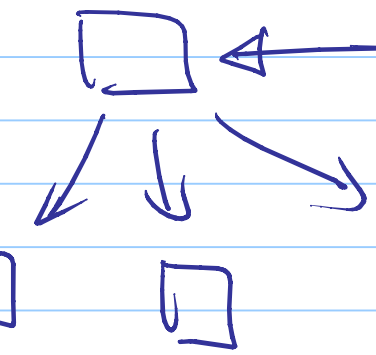


directories (folders) → containers for files

flat file system
↳ one directory, contains files

→ 2 level directories

multilevel directories



a directory contains
- files & directories

Linux only

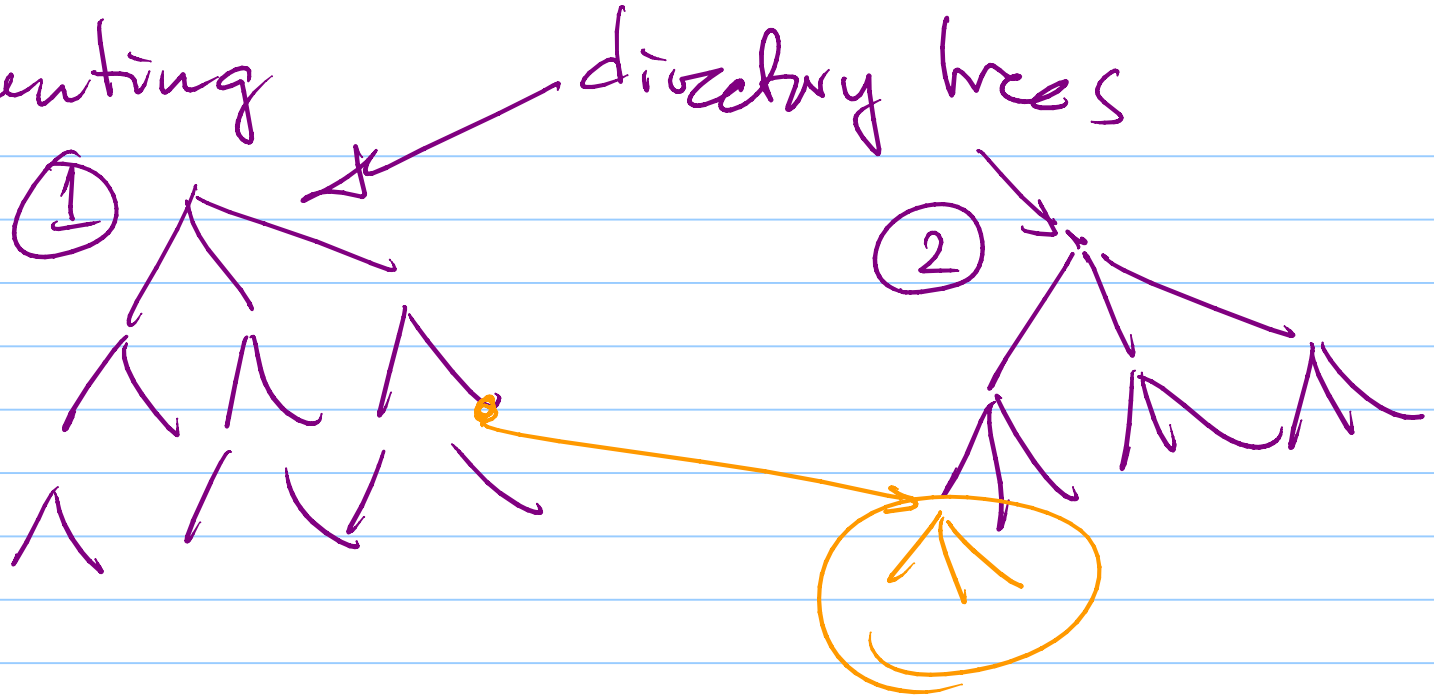
① TREE

② ACYCLIC GRAPHS

③ ~~GENERAL GRAPHS~~

④ symbolic links

Mounting directory trees



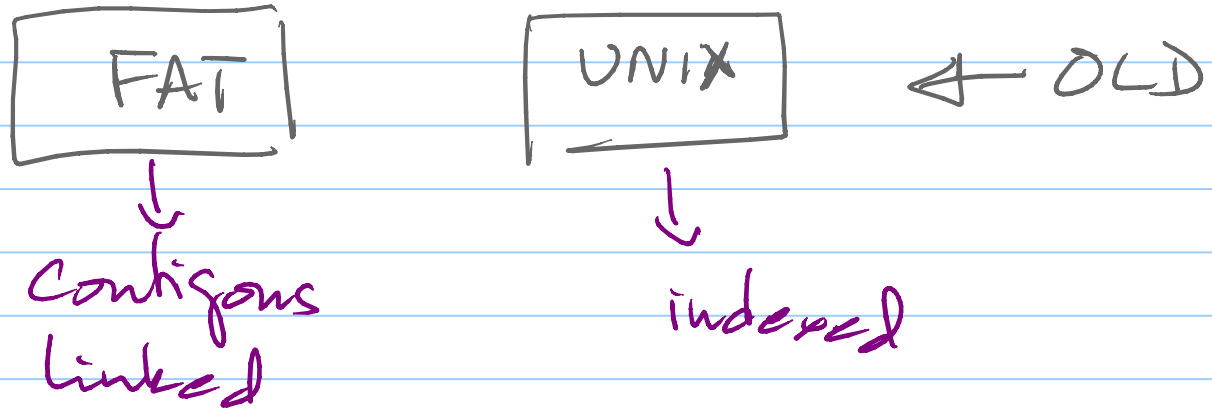
↳ Remote mounting (Linux) → NFS

↳ sharing (WIN) → SMB

appears in 2 trees

How are file systems implemented

→ complicated



→ journalling, log structured

NTFS/ext

inode
block
name entry
data blocks