

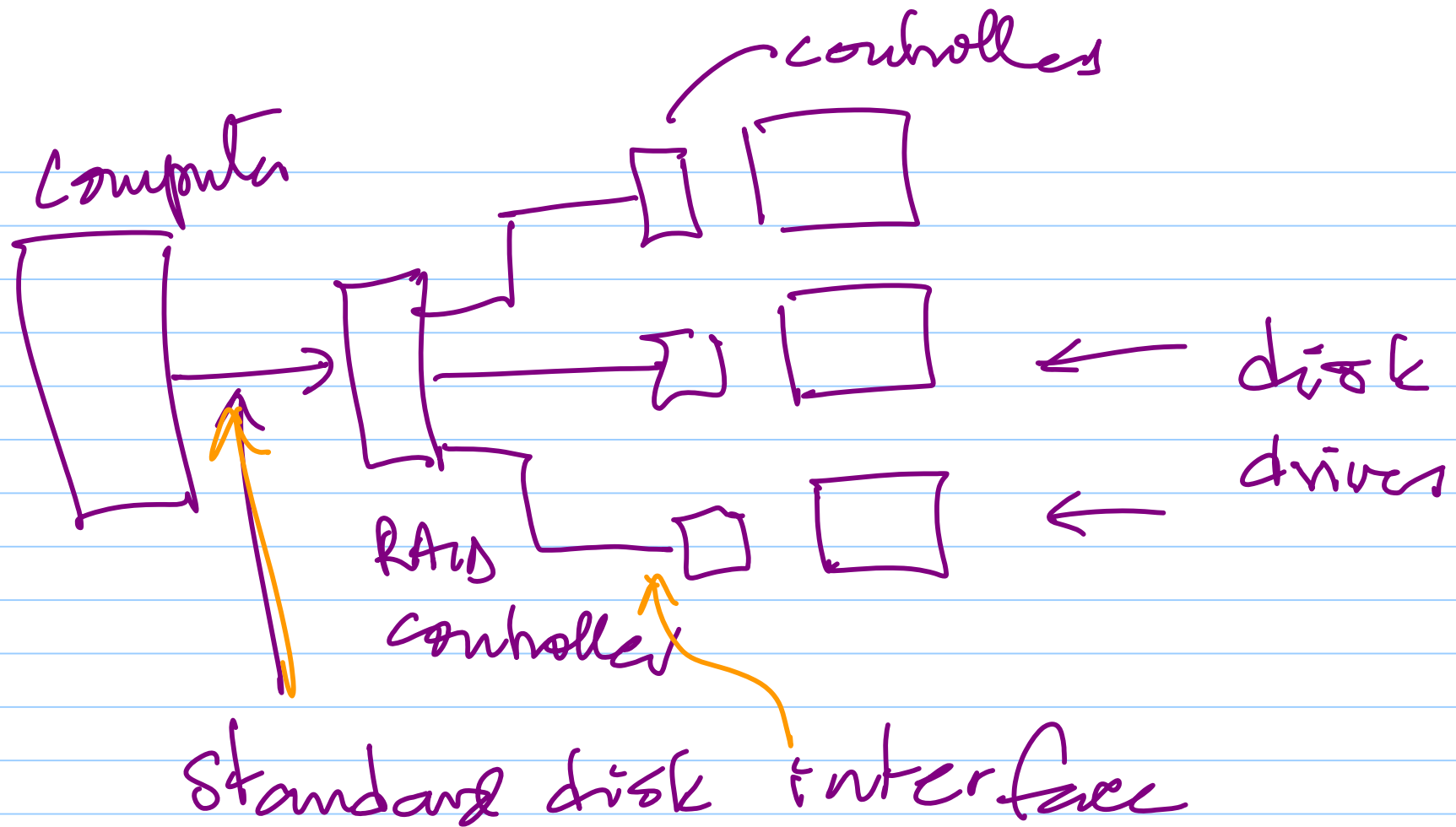
- DISK DRIVES & I/O.
    - hardware/software interfaces
    - drivers
    - scheduling
- ⇒ READ/WRITE disk blocks.

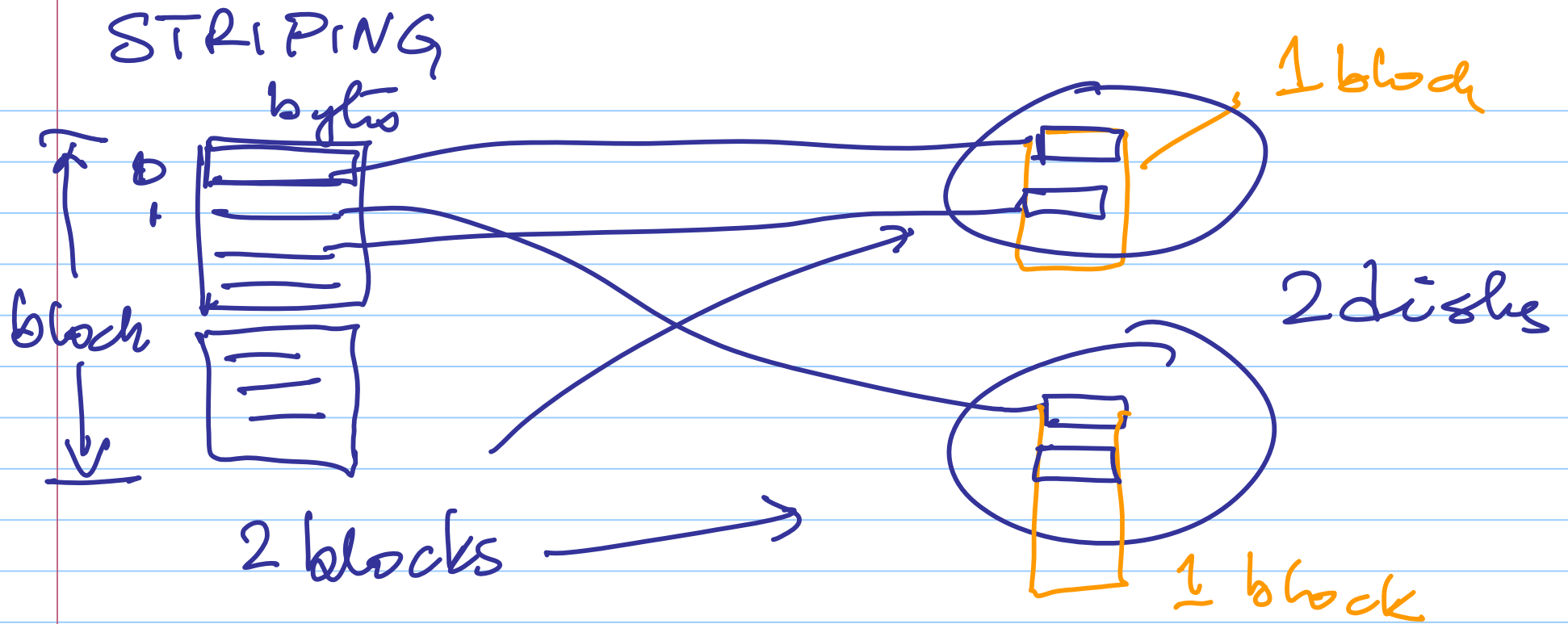
RAID

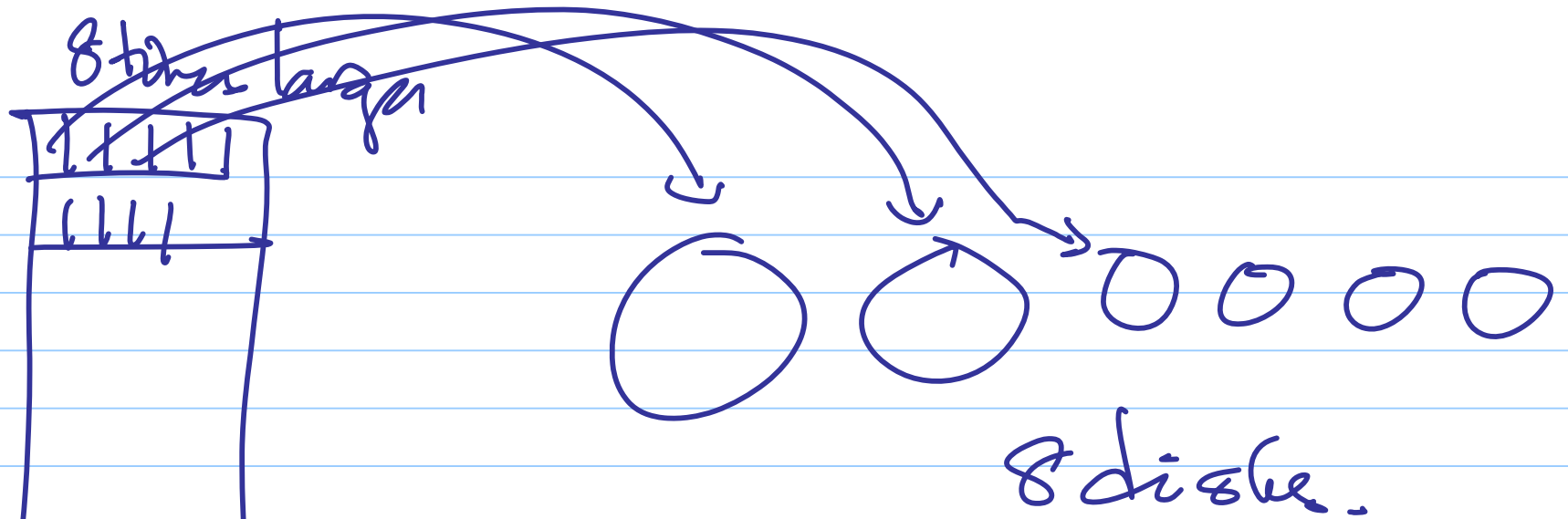
Redundant Arrays of

~~Inexpensive~~ Disks

Independent



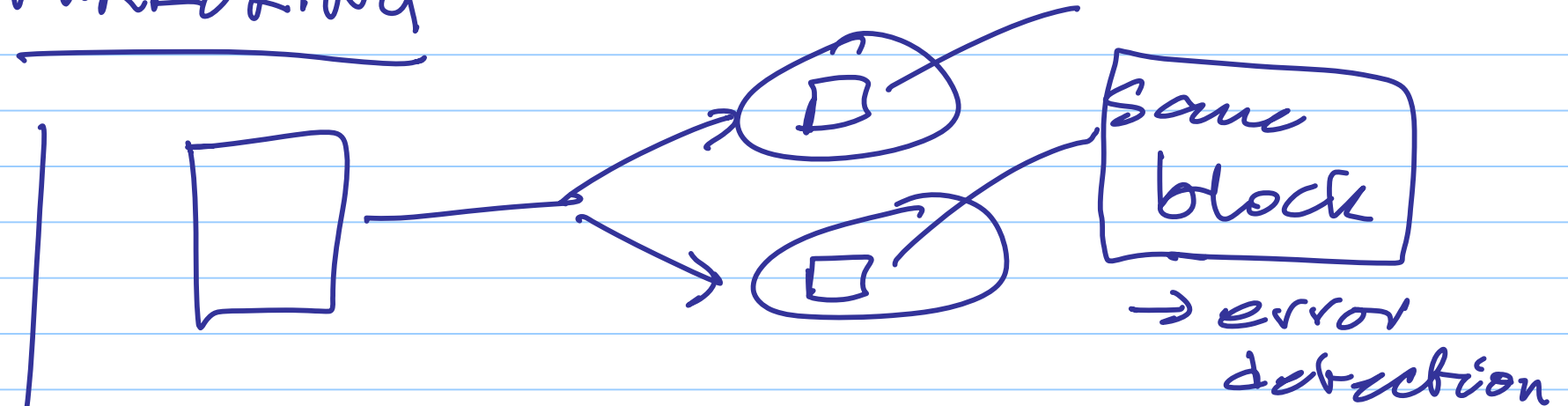




Striping → higher performance  
→ lower reliability  
→ RAID - 0

# RAID-1

## MIRRORING



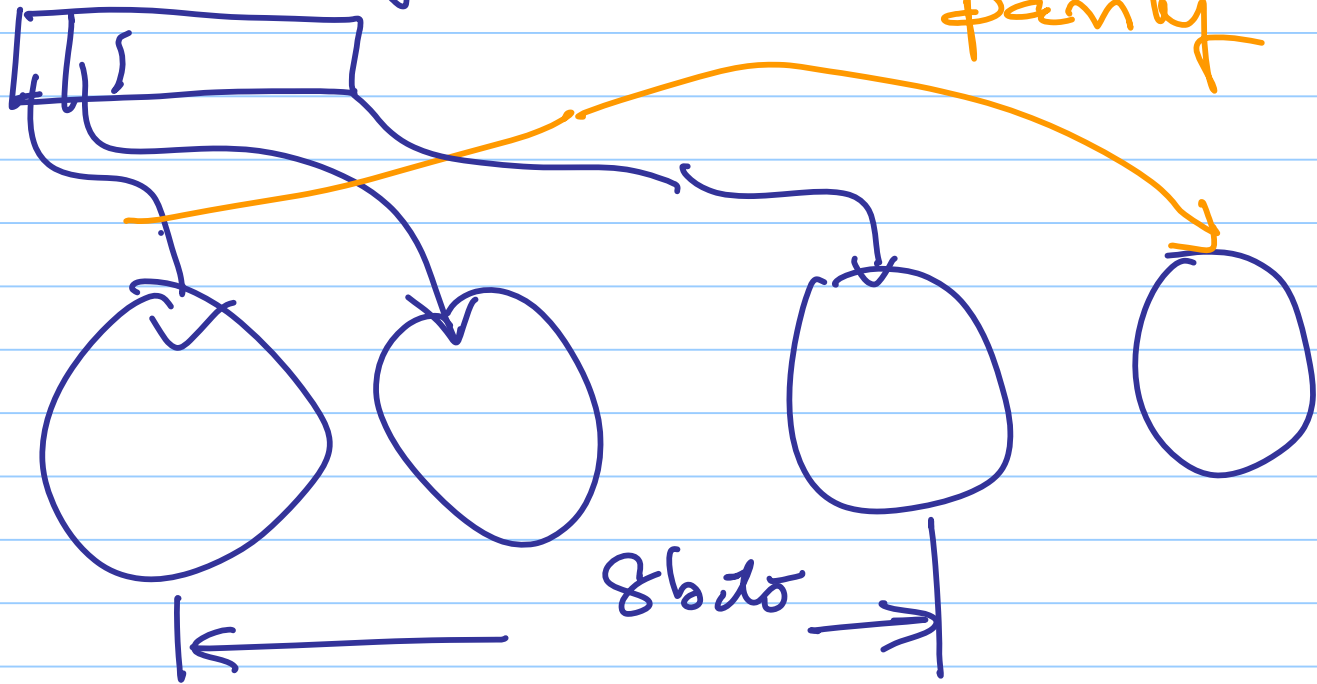
2 way mirror

→ Same performance  
→ higher reliability } higher price

9 disks

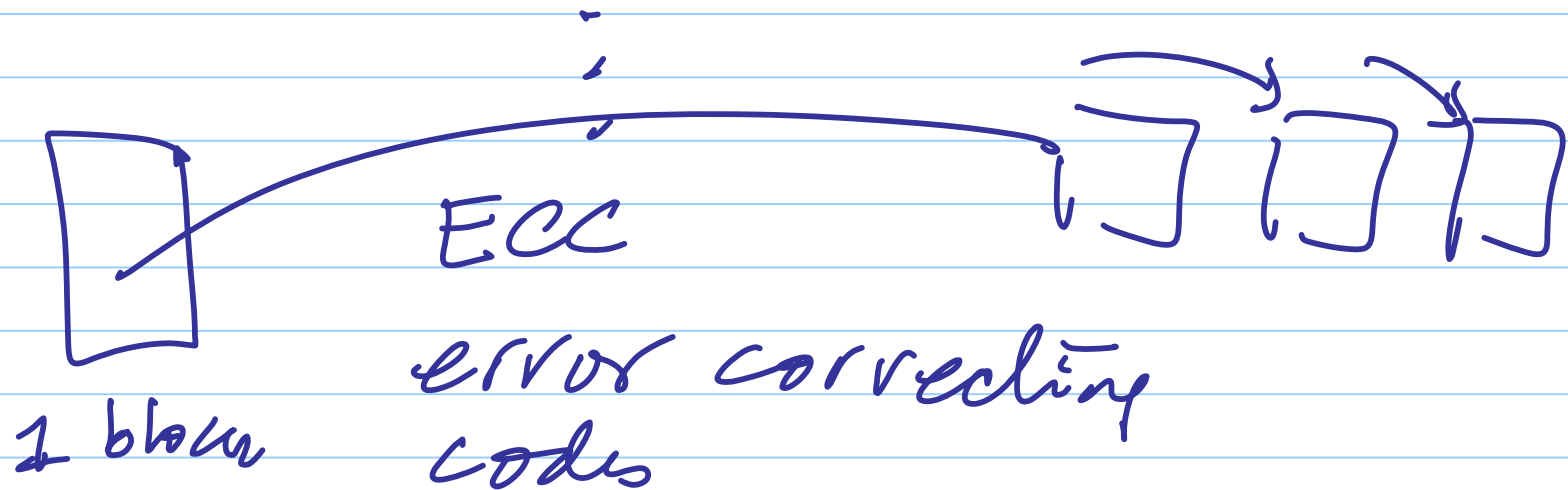
1 byte

parity



RAID - 2, 3, 4... all striping + error detection methods  
Parity - 1 bit per 8 bits X

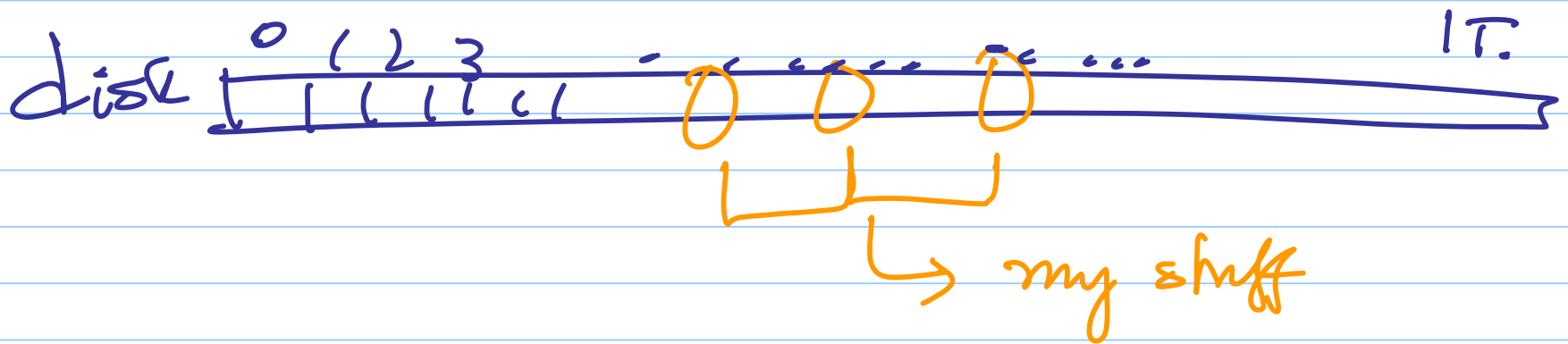
1 bit per 2 bits



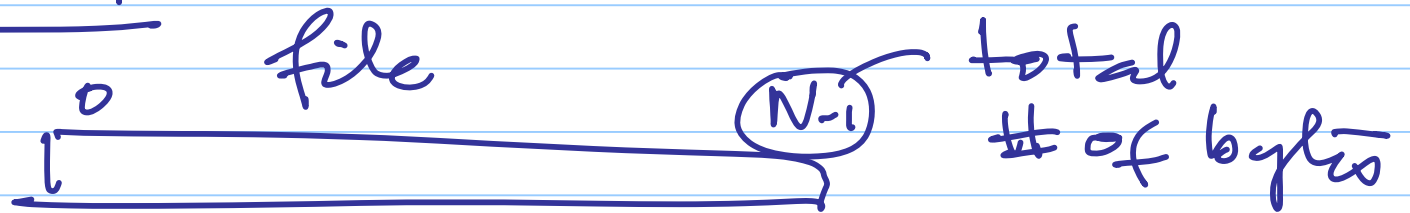


# File Systems

- user (human) & application friendly interface to disks



# File Concept



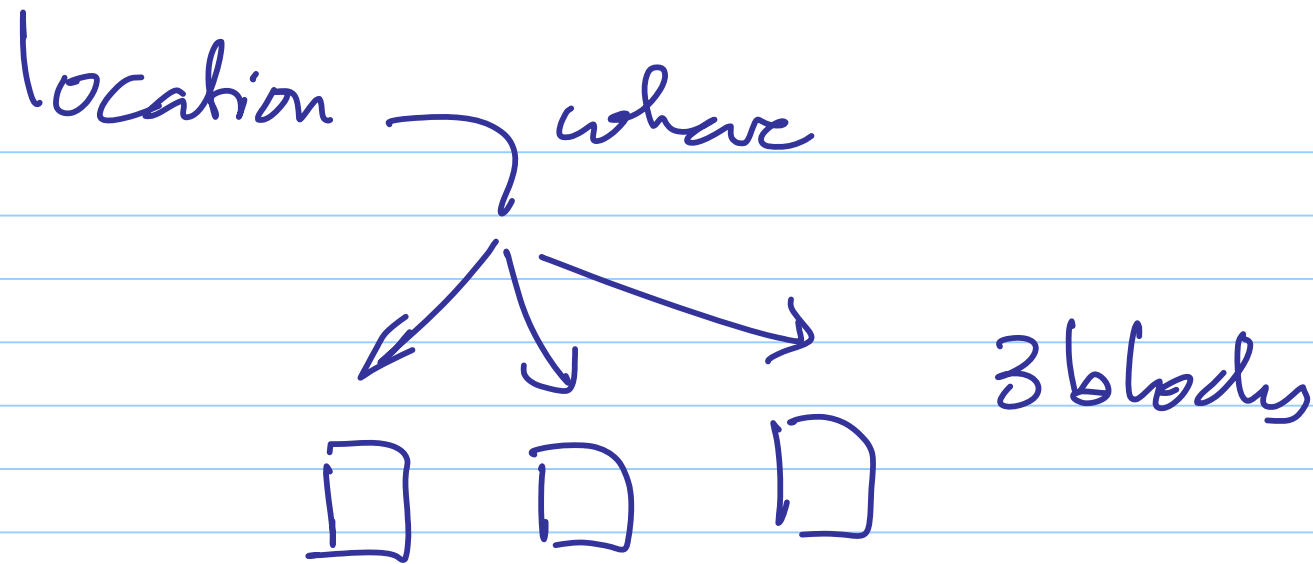
↑  
→ - sequence of bytes

file is an  
uninterpreted  
sequence of  
bytes

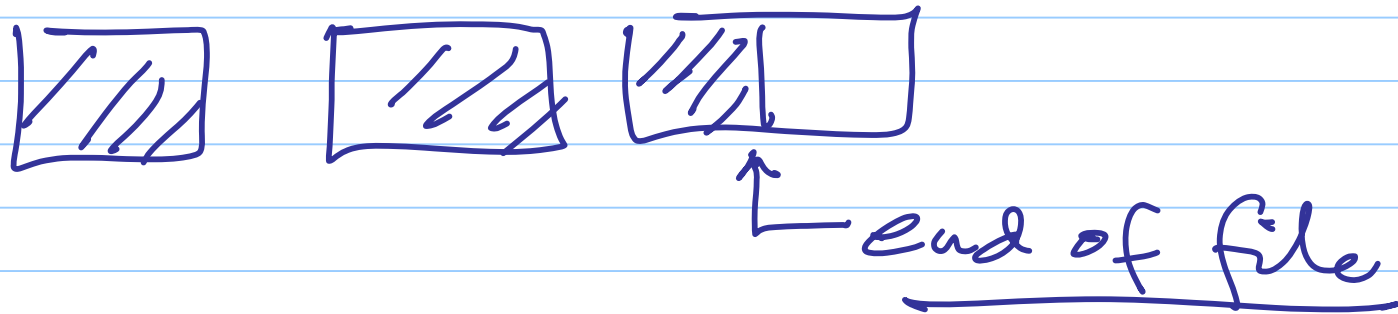
- contiguous
- mostly sequential.
- name

## File Attributes

- Name
- ID (identity, a number, system uses it to identify the file)
- location(s) → block #s
- size
- protection (access control)
- timestamps (create, modify etc)



Size  $(3 \times \text{block size})$ ? X  
 $\rightarrow$  last block contains partial data



**EOF** symbol @ end of file.

# File types

↳ text file, <sup>word</sup> doc, PDF, JPEG  
→ executable

## UNIX

↳ files are bytes

LINUX

↳ magic numbers

OSX

↳ has file type

WINDOWS

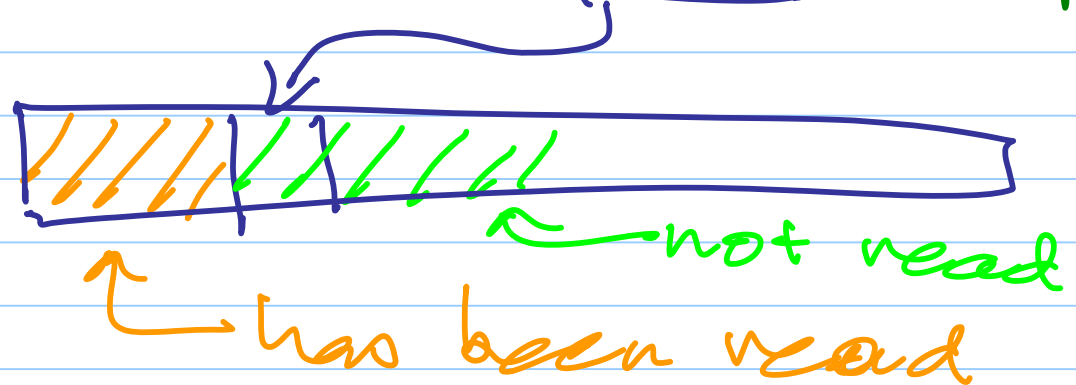
↳ Extension

# File structure

- ↳ Same as file types
- structure is defined by application

# File Access

→ Sequential. file pointer R/W pointer

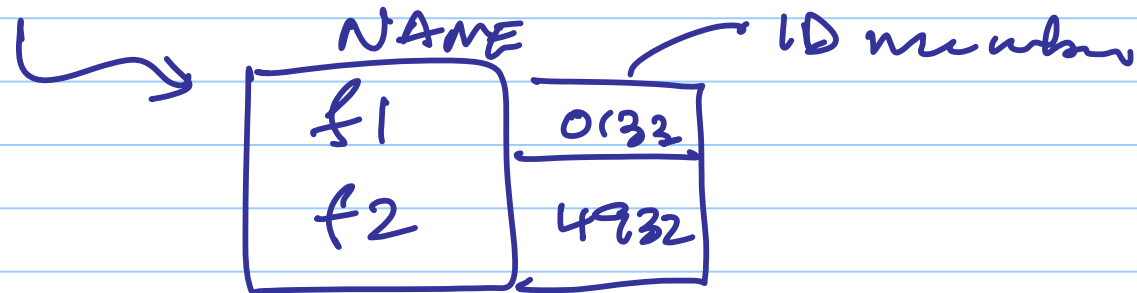


→ direct access x

→ random access → R/W pointer can be moved

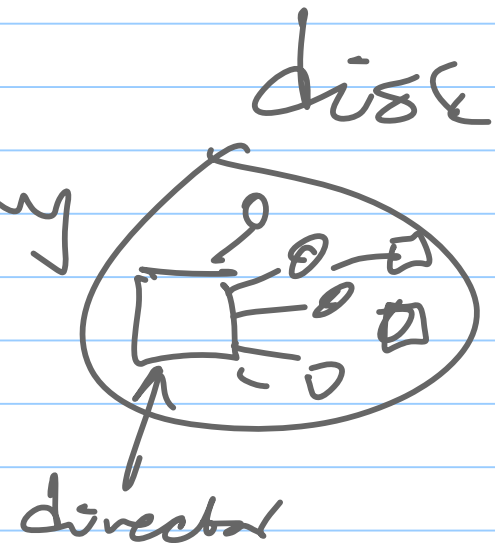


Directories → this names files.

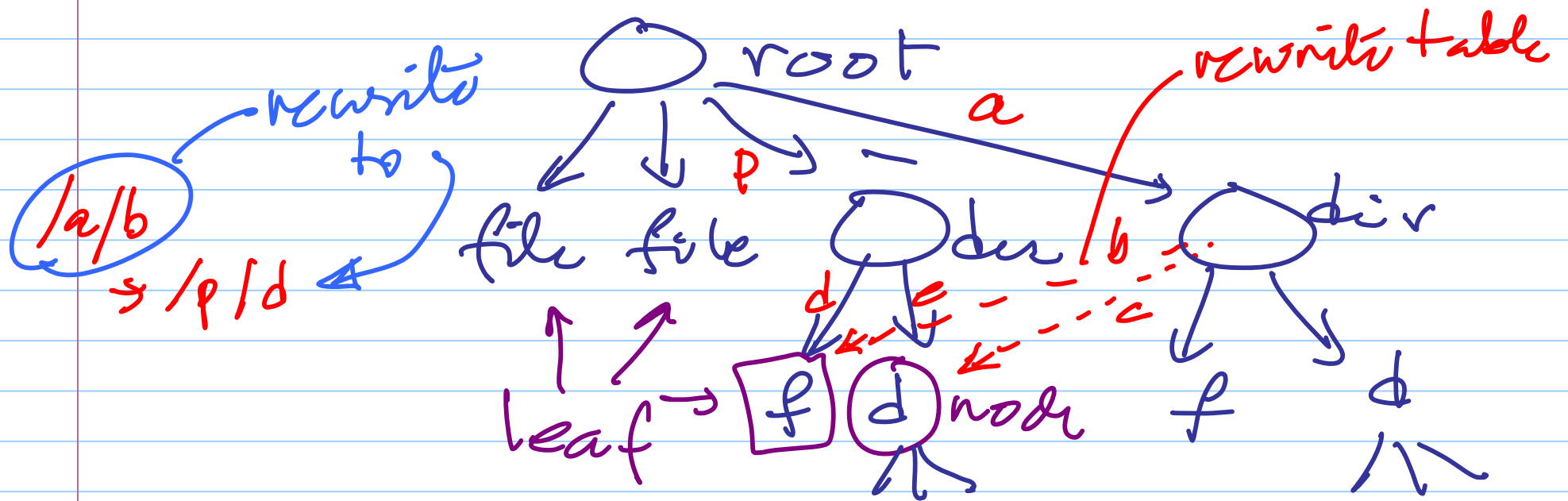


→ Single level directory

→ Two level     "     "



# Tree directory - windows / linux / osx

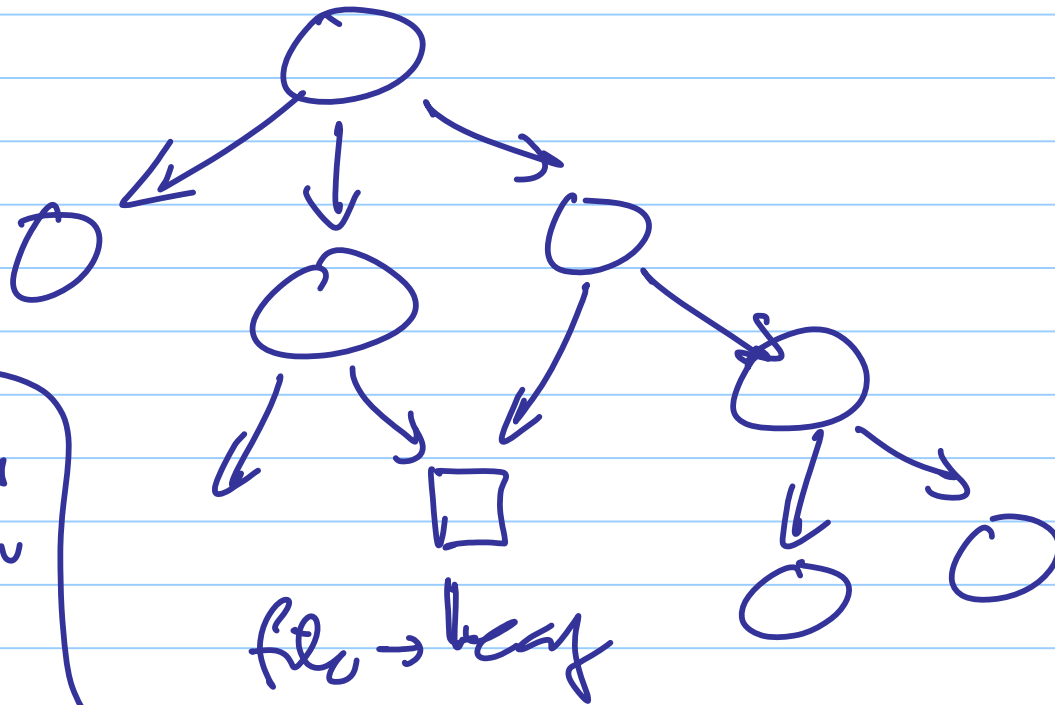


# Acyclic Graph für - LINUX

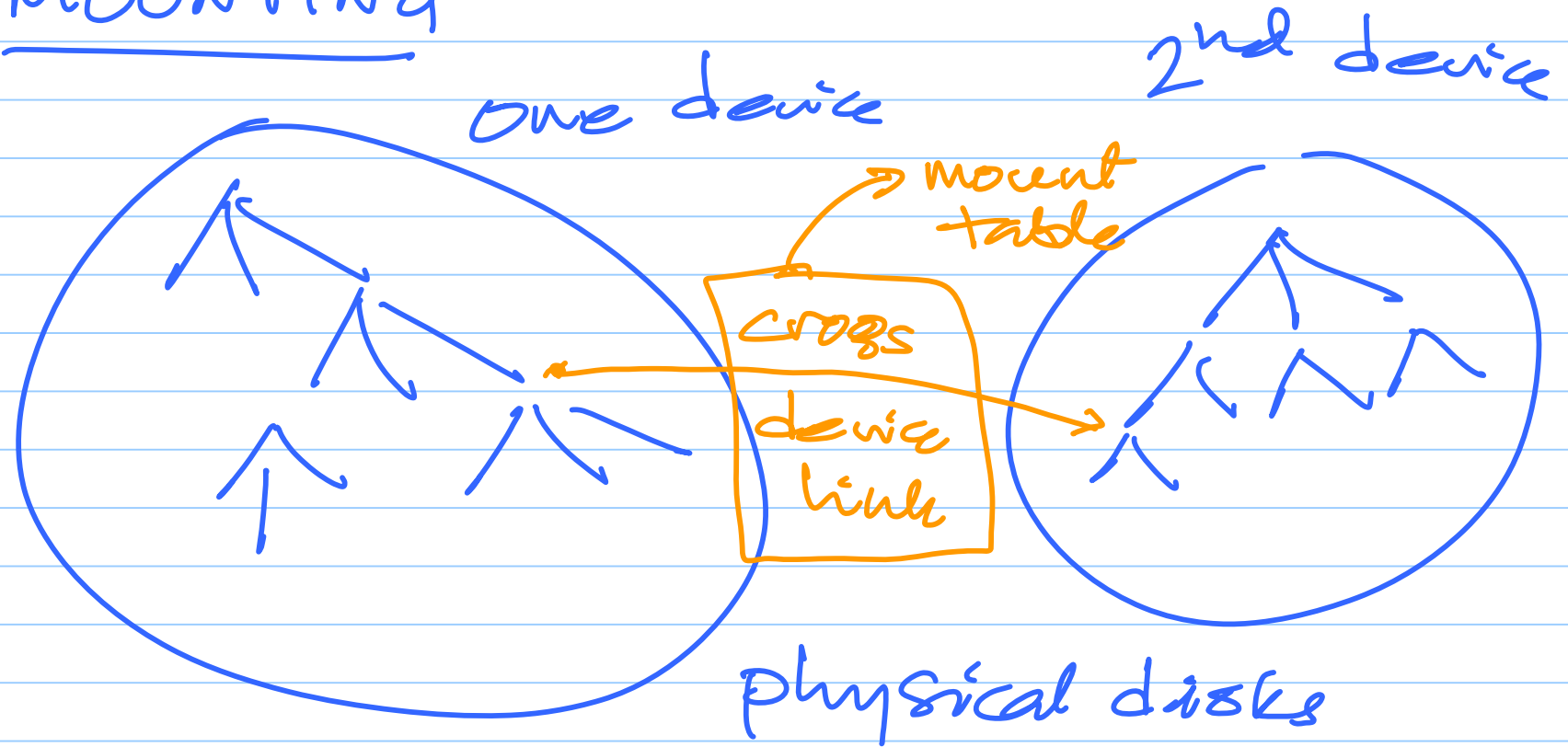
LINUX



2 directories  
can contain  
same file



# MOUNTING



MOUNTING → Remote mounting

↳ sharing  
between  
networked  
computers