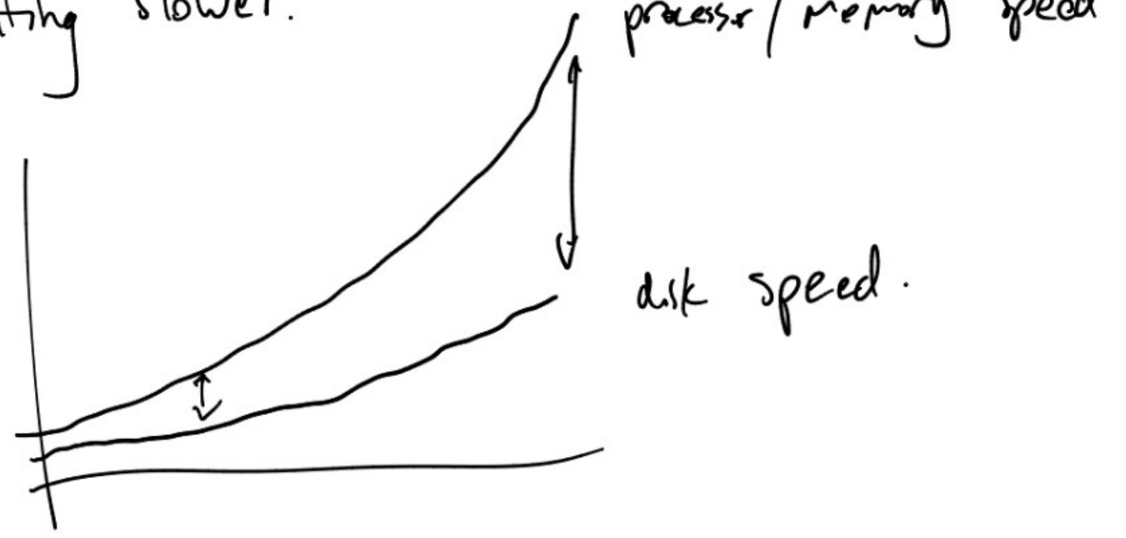
Lecture 16: RAID, Files

- RAID
- Filesystem interface
  - files directories, metadata, links, ...
  - File API
  - VFS

Disks are slow! getting slower.

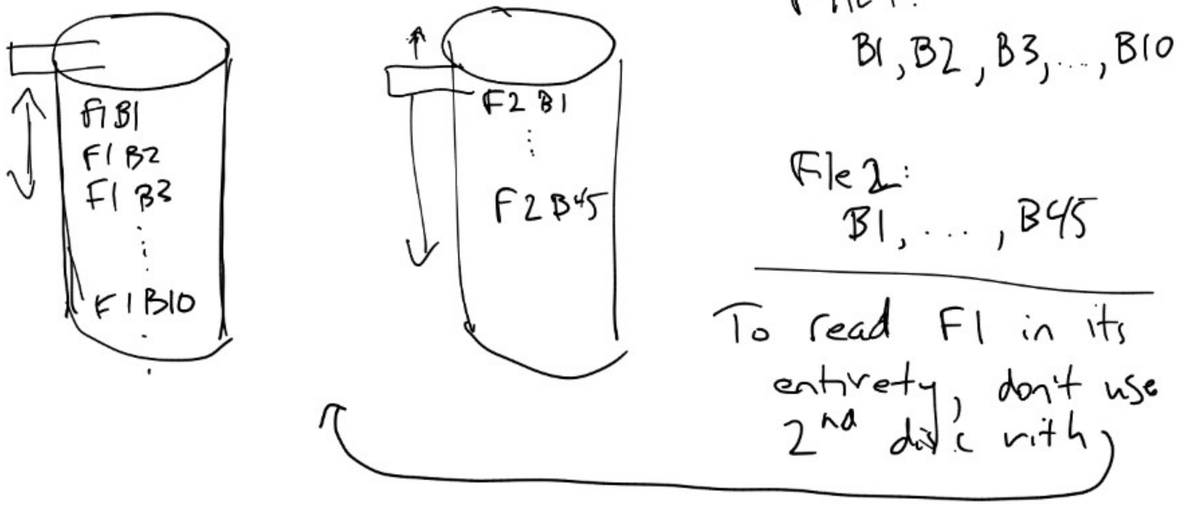


- Address by using multiple disks simultaneously.
  - Redundant array of ~~inexpensive~~ independent disks (RAID)
- Build big expensive disks. (SLED)
  single large expensive disk.

Disk read/write in units of sectors (½k - 4k)

Break up data into blocks to put in sectors

(analogous to frames / pages)

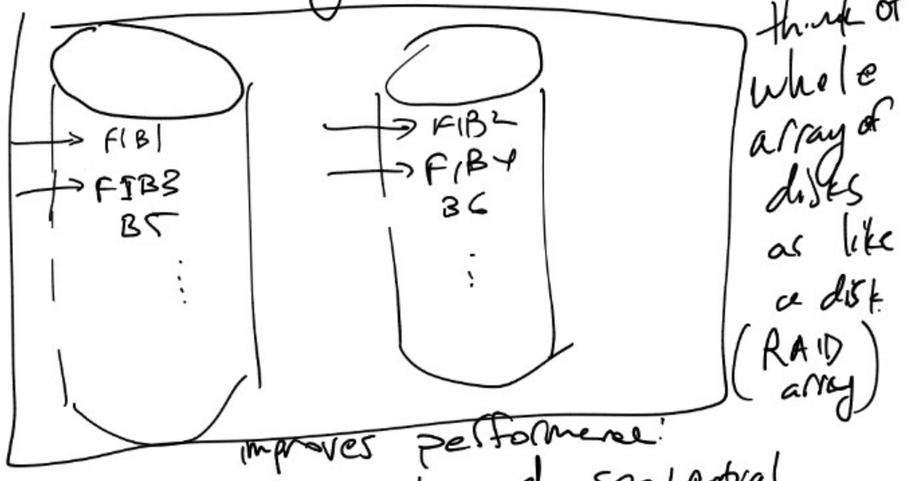


File 1:
B1, B2, B3, ..., B10

File 2:
B1, ..., B45

To read F1 in its entirety, don't use 2nd disk with

Striping (RAID 0)

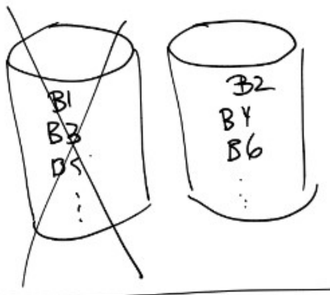


think of whole array of disks as like a disk (RAID array)

improves performance: effectively do sequential reads & writes 2x as fast

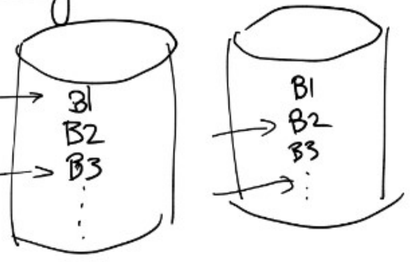
Disks fail!

- local failures (bad sector)
- disk failures. (entire disk)



with striping: failure is worse (more likely for 1 of 2 to fail, than just 1, either failing ⇒ file unusable)

Mirroring (RAID 1)

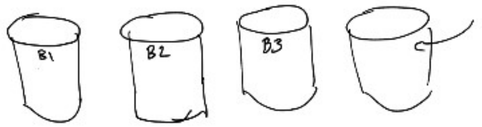


write speed:
- slower of 2 disks.

read speed:
- almost as fast as striping ⇒ can read from both @ same time.

pro: Great fault tolerance

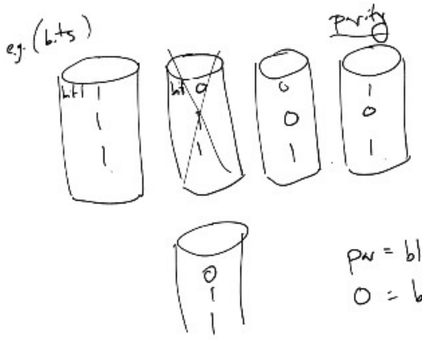
con: Compared to striping, not quite as fast (have to write 2x).



parity of B1⊕2⊕3
i.e. XOR of B1,2,3

parity is sum, mod 2

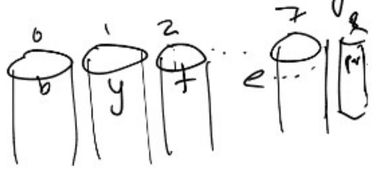
useful: any one of the disks is the parity of the other 3.



$pw = b1 \oplus b2 \oplus b3$

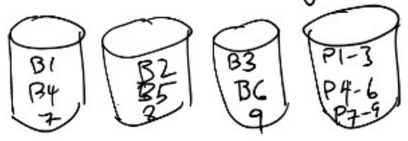
$0 = b1 \oplus b2 \oplus b3 \oplus par$

RAID 2: bit-level parity



RAID 3: byte-level parity

RAID 4: block-level parity



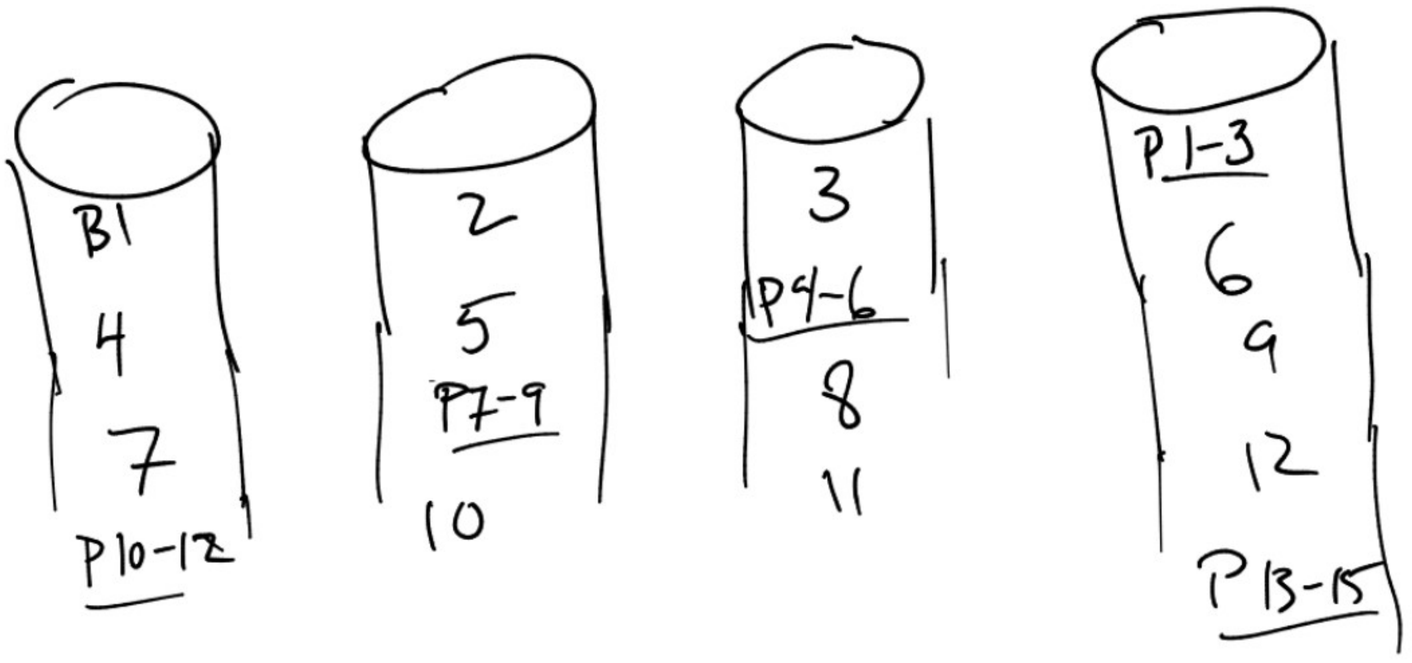
pro: - fault tolerance: can lose 1 of n disks, still recover
- reads as fast as striping across n-1 disks

con: - parity disk doesn't improve performance

every write writes to parity disk (also reads parity)
⇒ more wear
⇒ maybe more seeking.

- to write: read old data, and old parity, recompute new parity, write new data + new parity.

RAID 5: put parity on different disk for each "row" of blocks

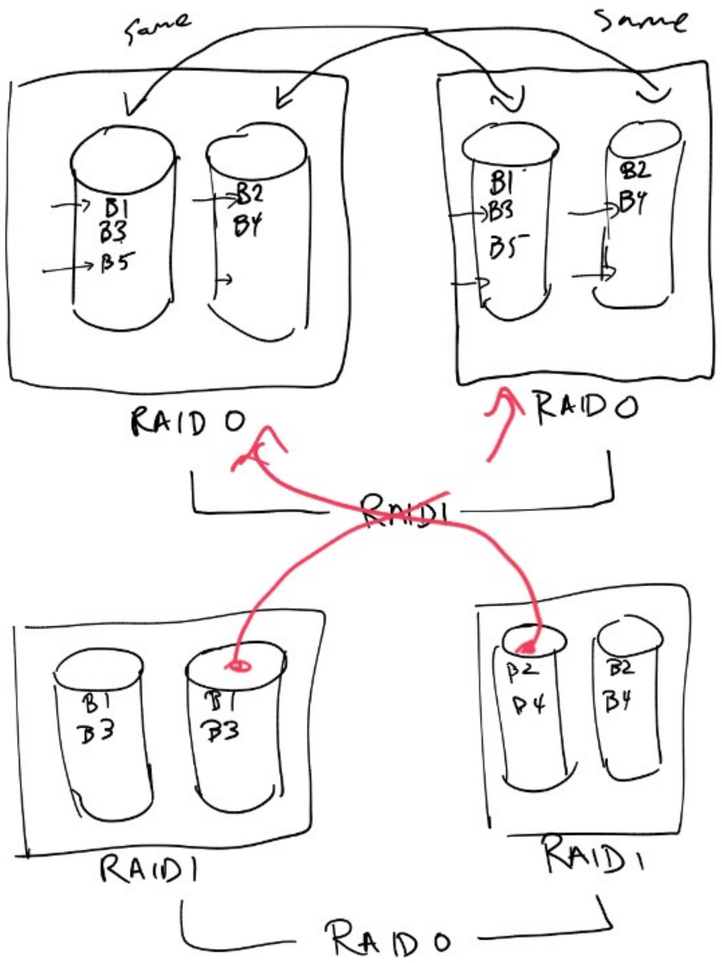


pros/cons:

Same as RAID 4, except for non-uniform wear.

Con: requires complicated controller.

Raid 1+0 or 0+1

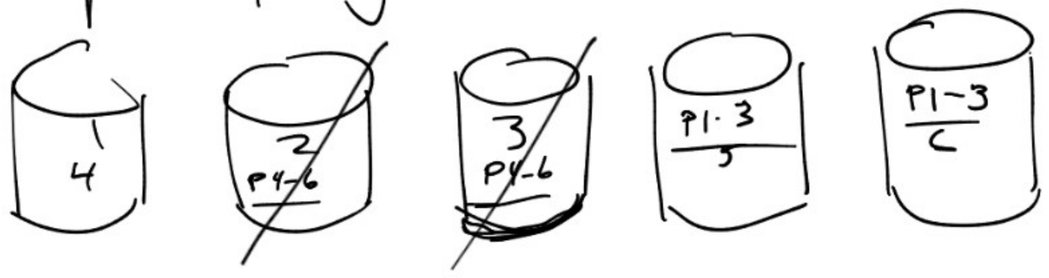


pro: - good fault tolerance
- simple
- good read performance
- decent write perform.

effectively the same.

RAID 6: Tolerating 1 failure might not be enough.

- recovery takes a long time for large disks
- lots of disks. long time = high(er) prob. of 2 simultaneous failures.
- 2 striped parity blocks



- Reed-Solomon codes: generalize parity to handle 2 or more failures (By holding more "parity"-like bits)