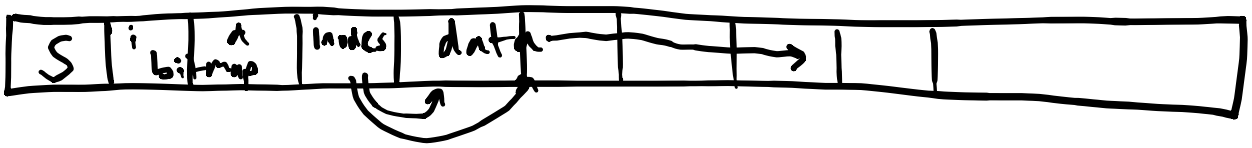
FSCK - due next fri 5/1

Final exam - comprehensive

- I will make a study guide
- 5/1 review: wrap up

Persistence: I/O device (disks)



today: what happens in a crash:

- power loss
- dropping laptop
- kernel panic
- user hard reset

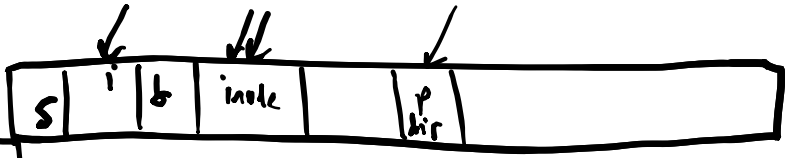
} OS suddenly interrupted

could be in the middle of updating persistent state

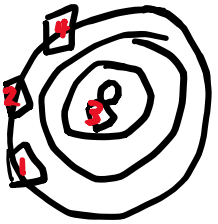
Example: File create: 3 blocks

- parent dir data block, inode
- inode bitmap
- inode itself

[Assume sector write (512B) is atomic]



EVIL "crash" scheduler



1234

It's worse: Disk scheduling "elevator"

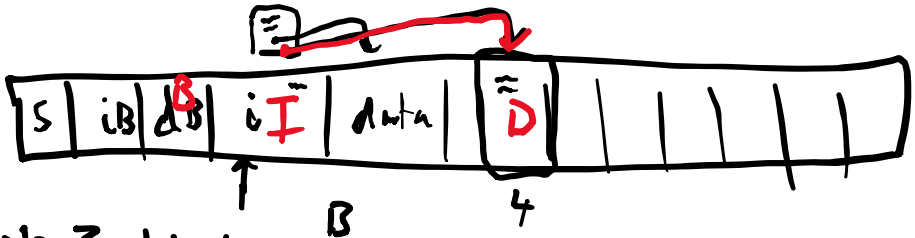
1243

i a b i

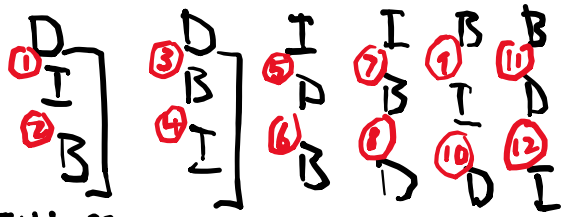
reordering of disk blocks!

Example: File append

```
fd = open("f", O_WRONLY)
lseek(fd, 0, SEEK_END);
write(fd, buf, size);
close(fd);
```



wrote 3 blocks: B I D



3,1: no problem

Fsck crashed → 12, 11, 9, 7, 6, 5, 4, 2: inode & bitmap are inconsistent → data could be overwritten; waste of space no body is pointing

10, 8: Consistent, but garbage

FS "consistency"

is consistent iff all on-disk info agrees on state of Fs

How to fix inconsistencies

in data block

Solution 1: Filesystem Checker: FSck

- scan entire on-disk structures (inodes, bitmaps, directories, indirect blocks, etc.)
- find + (fix) inconsistencies
  - e.g. if bitmap marked but no inode points to it, clear bitmap

Can be EXPENSIVE

Solution 2: Journaling (Write-ahead logging)

basic idea: on-disk structure: journal

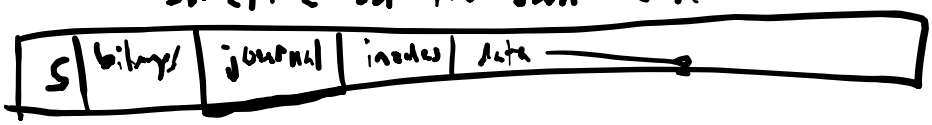
transaction:
1. write down what you're going to do
2. do it

if you crash during (2) it's OK!

figuring out based on journal (recovery)

goal: make multi-block update ATOMIC
↳ all or nothing

location of journal: usually near beginning
sometime on its own device



transaction

Protocol: 1) write [T_B, contents] to log wait
update lost ↑ 2) log T_E "commit" wait
fine ↓ 2) checkpoint: writing contents to final locations

Recovery: scan log + find all valid Tx & replay

Every data block written 2x!
↳ metadata-only journaling (e.g. ext / ext2 / ext3 / ext4) ↑ FSCK ↓ journal
write data first
" pointer never points to garbage"