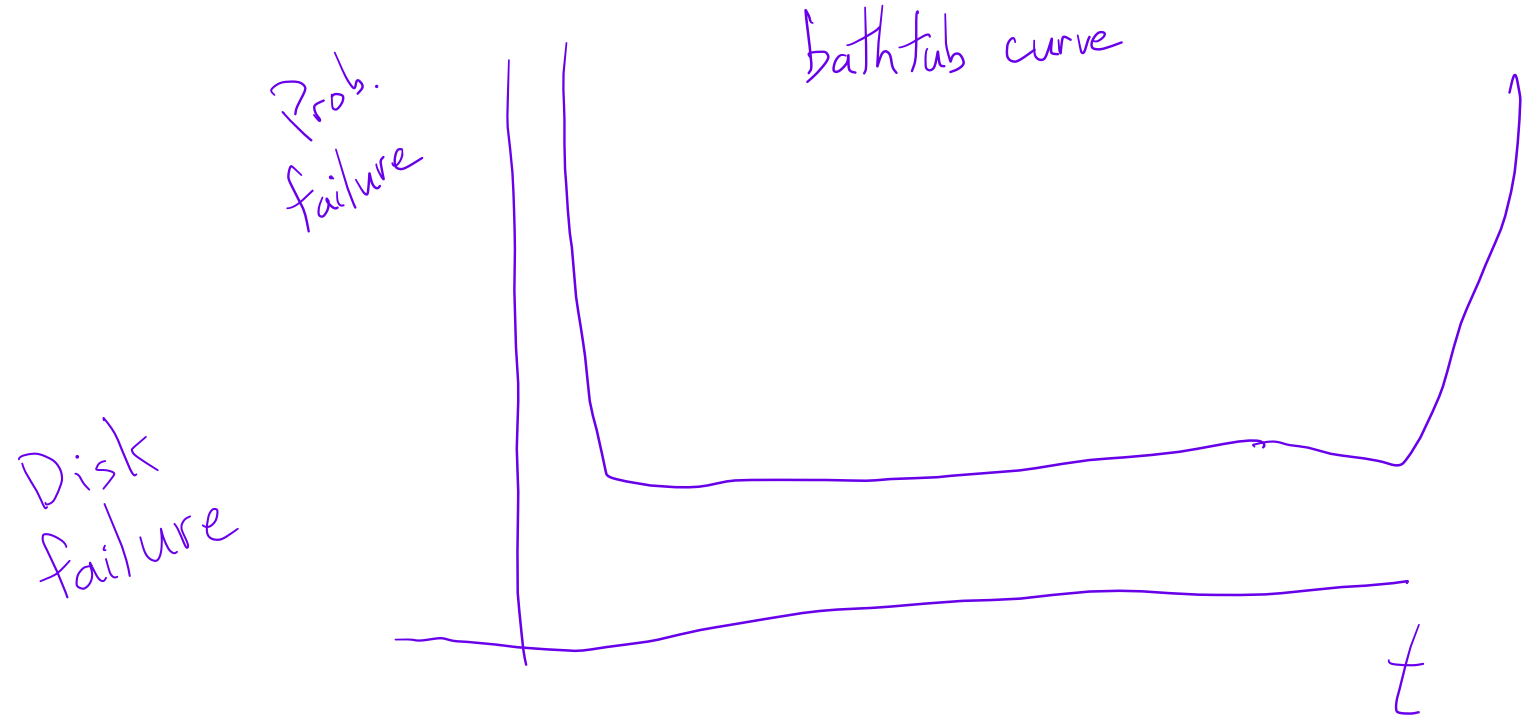
- [ ] 1. Last time
- [ ] 2. Intro to file systems
- [ ] 3. Files
- [ ] 4. Implementing files



2.

Intro to file systems

What does a FS do?

- provide persistence
- create a way to name data on the disk

FS: can be implemented in lots of places

- We focus on the disk, generalize later

Note: disk is the 1st thing we've seen that is **both** modifiable **and** persistent.

3.

Files

What is a file?

From user's view: a named, contiguous run of bytes

From FS's view: collection of disk blocks

Job of a FS:

map {file, offset in file} $\xrightarrow{\text{FS}}$ disk address

operations:

create(file), delete(file), read(), write()

Goal: operations have as few disk accesses as possible and minimal space overhead

4. Implementing files

A. Contiguous

B. Linked files

C. Indexed files

Assume for now that a given file's metadata is known to the system.

Access patterns to support:

- Sequential
- Random access

Ideal is good sequential + good random access performance

Candidate designs:

A. Contiguous allocation

user pre-specifies length

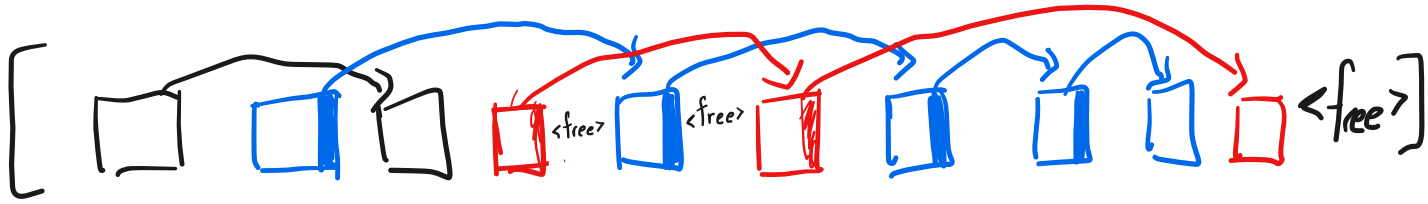
[<1 free> a1 a2 a3 <5 free> b1 b2 <1 free>]

+ fast access, both seq. and R.A.
+ simple

- fragmentation

B. Linked files

metadata is pointer (disk address) to file's first block



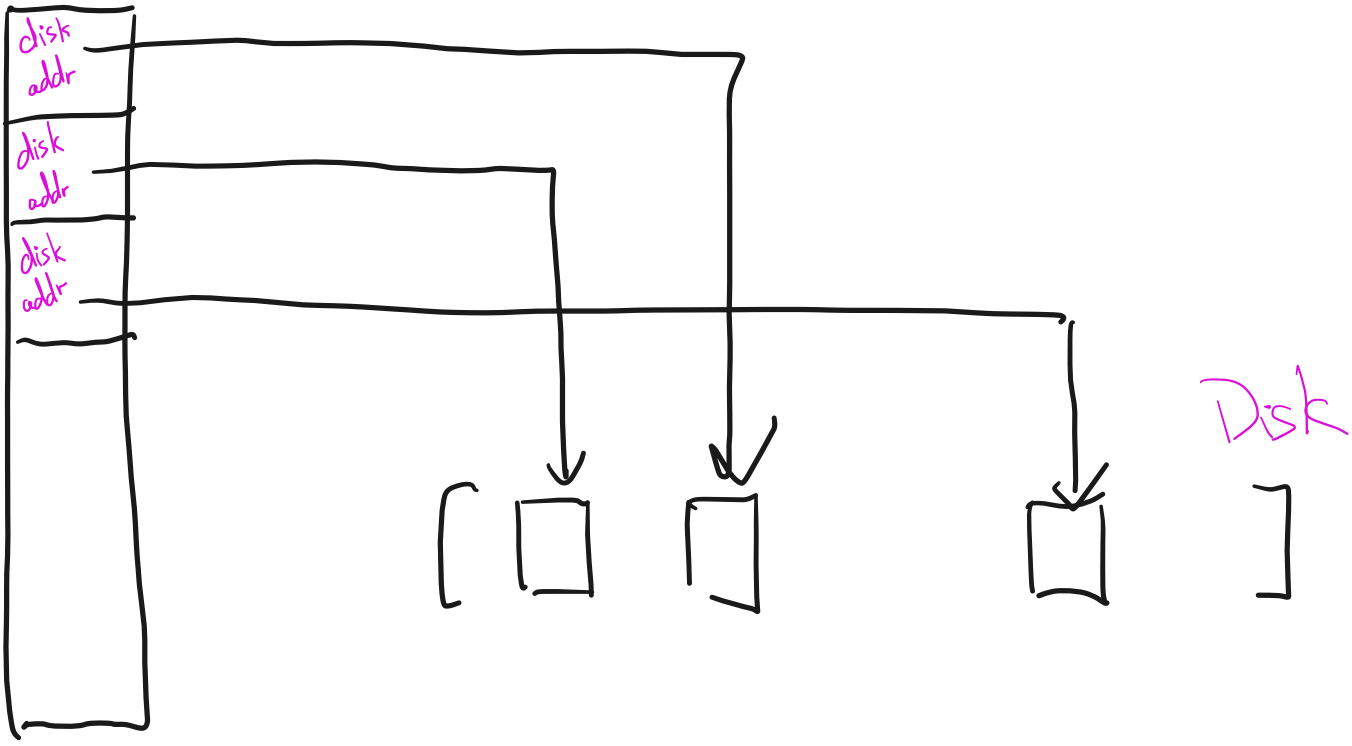
+ seq. access easy + probably fast
+ no more fragmentation

- R.A. is a disaster
- alignment of data can get messed up

C. Indexed files

attempt 1

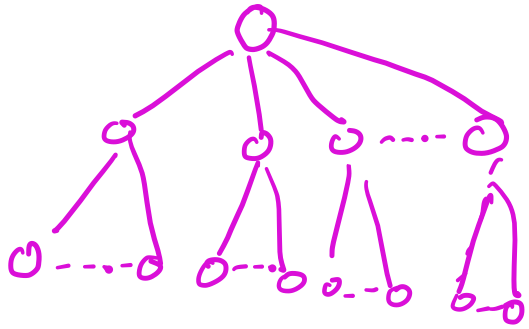
metadata:



+ seq, R.A. easy

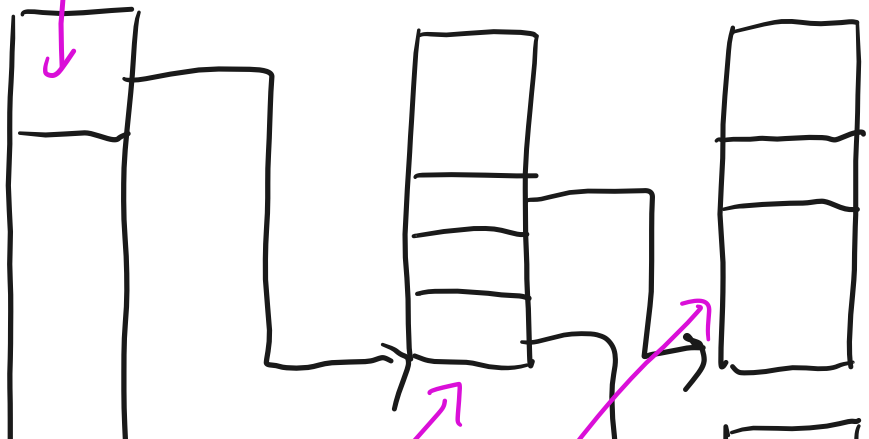
+

- storing this array is impractical



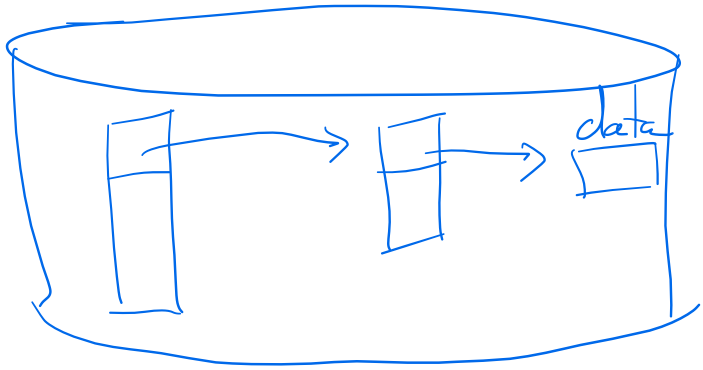
disk addr of metadata

attempt 2



disk addr of 1st datablock

disk addr of 2nd datablock

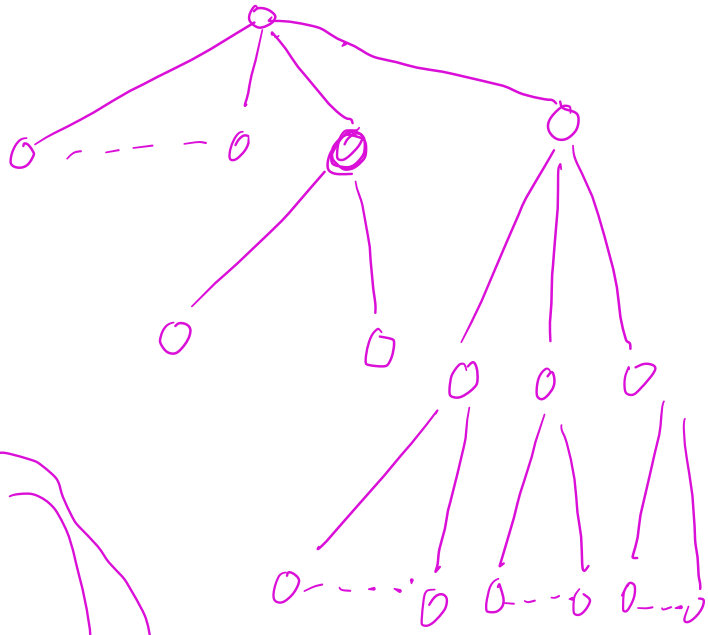


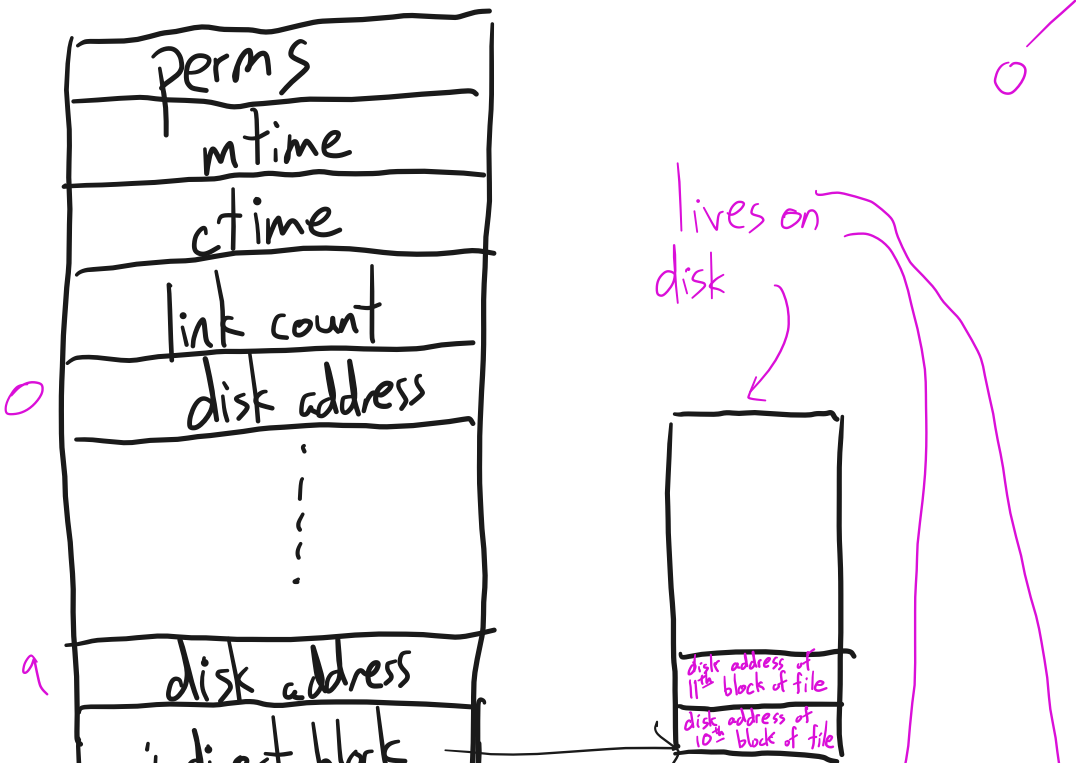
+ metadata is compact

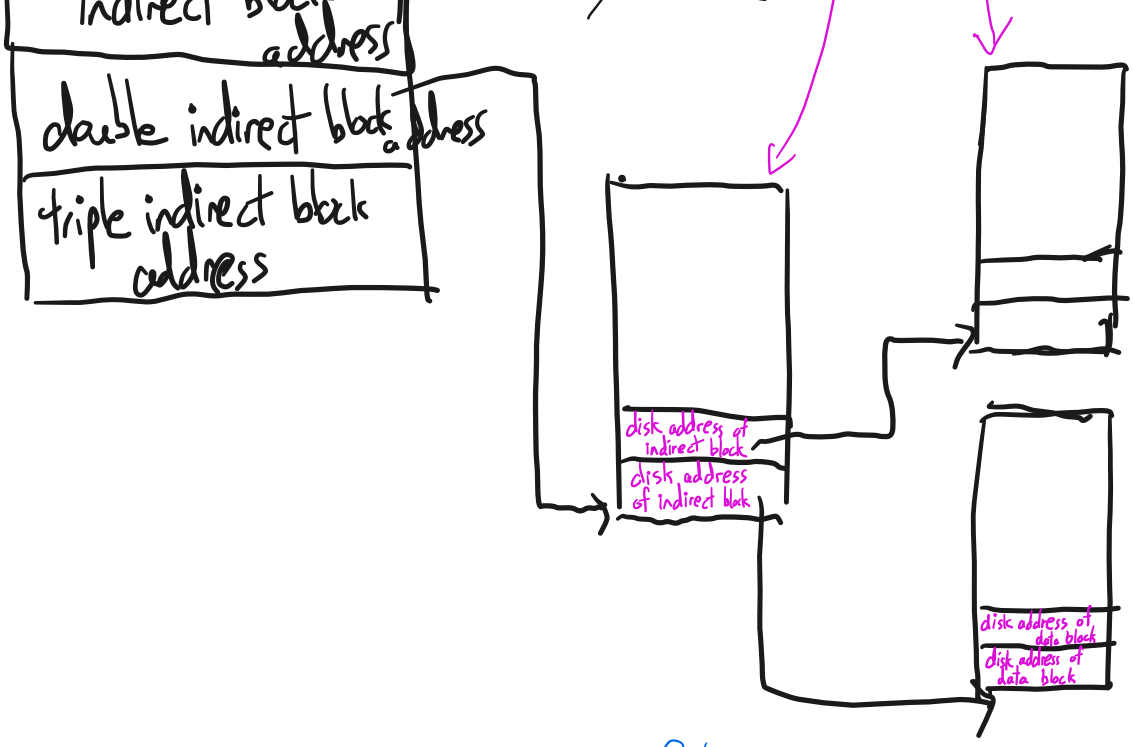
- looking up any block requires many disk accesses

attempt 3

Metadata: inode



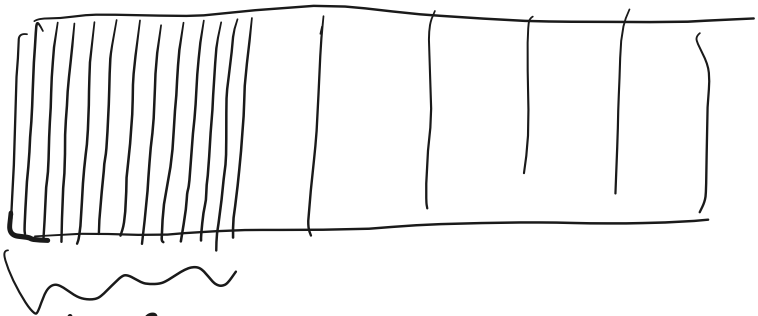




+ fast access to small files
+ Max length can be enormous
+ Simple, easy to build
- worst case # of accesses: pretty bad
- " " space ovhd.
- locality issues

inodes: stored in a fixed-size array, known location

vocab: "inumber"



stat (&sb);

slots for inodes

---

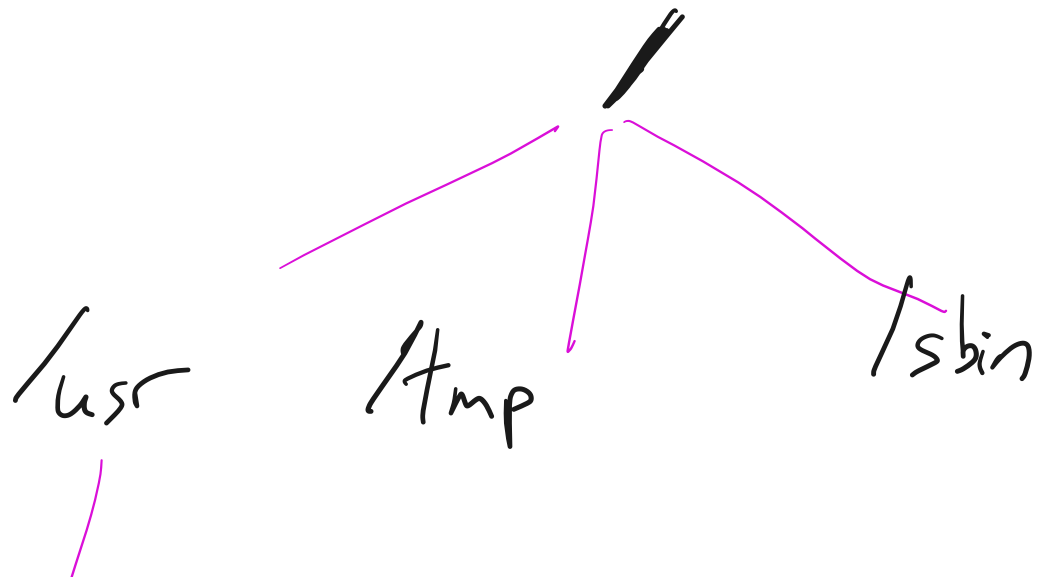
5. Directories (next time)

/bin/

/sbin

/usr

/tmp




/usr/mw
/lab4
kernel.c ...