week 10b
CS6640
03/13 2026
https://naizhengtan.github.io/26spring/

☐ 1. egos rwfs design
☐ 2. intro to concurrency
☐ 3. consistency model

Lab7?

Production fs:

• file size?	N	N	Y	Y
• #files?	N	N	Y	Y
• #files w/ a ~~dir~~ folder?	N	N	N	Y
• #files under a dir?	N	N	N	Y

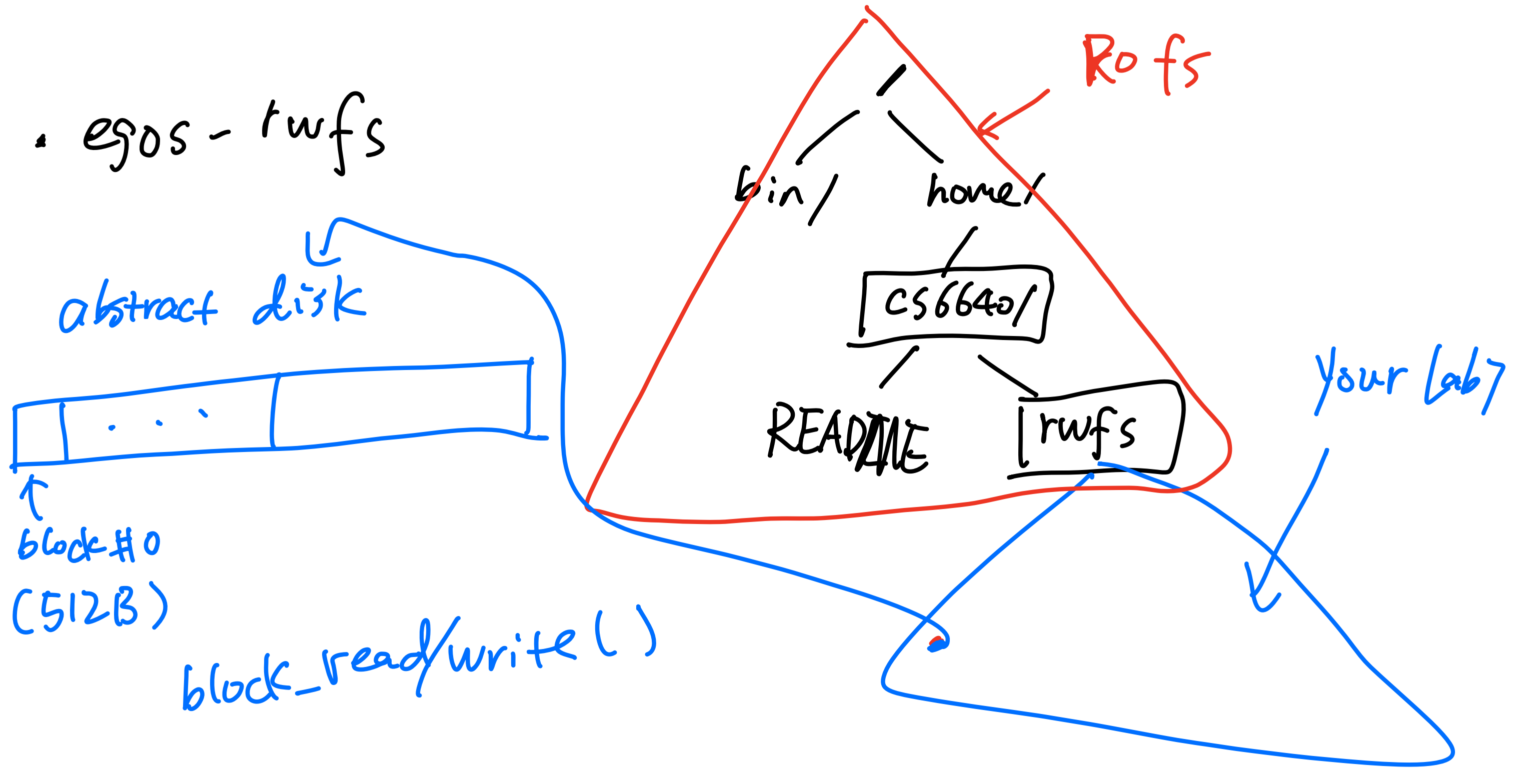
• fs size? (#blocks) 10^{50} B

• depth of a path?

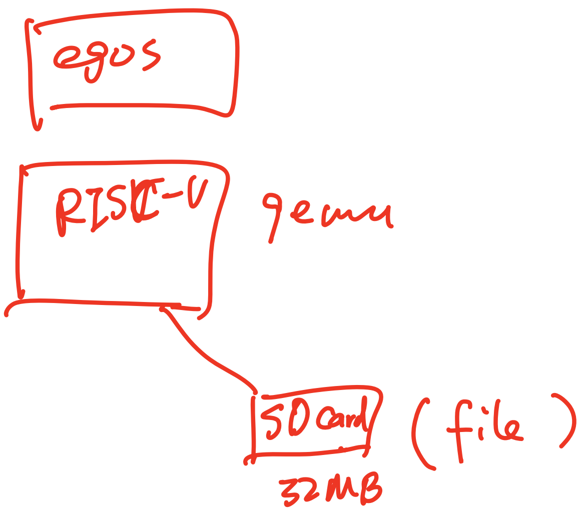
• name length of a file?

$\underbrace{/a/b/c\ldots}$

• egos-rwfs



$$\begin{tikzcd}
fs1 \arrow[dr] & fs2 \arrow[d] & \cdots & fsN \arrow[dll] \\
& \text{block\_read/write} \arrow[dl] \arrow[d] \arrow[dr] \arrow[drr] & & \\
SDCard & SSD & HDD & \cdots
\end{tikzcd}$$

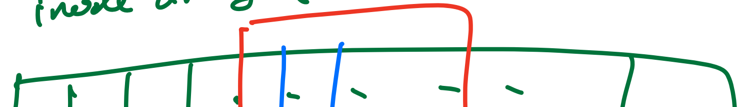


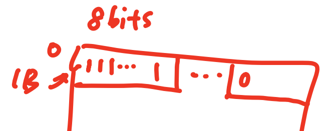
fs_read ($\underbrace{\text{"/home/cs1640/r}}_{\text{Rofs}}$ | $\underbrace{\text{ufs/file1.txt"}}_{\text{rufs}}$, offset=100, len=512, $\overset{\text{char *}}{\text{buf}}$)

$\hookrightarrow$ ino = 10

$\frac{512B}{64B} \times 10 = 80$ (files)

inode array (10 blocks)





inode#0
ino#10
block
511

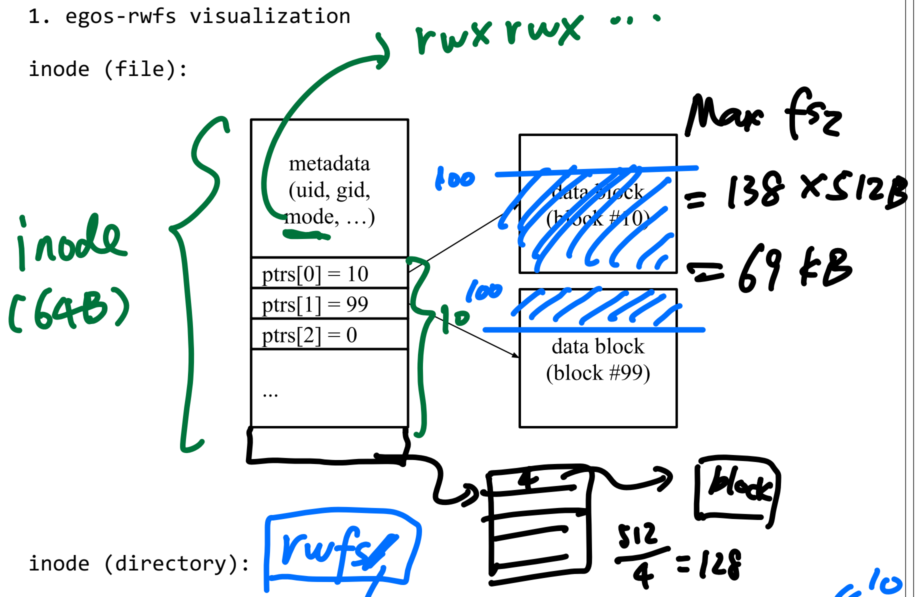
#blocks = 512×8 = 4096

fs size = 4096 × 512B = 2 MB

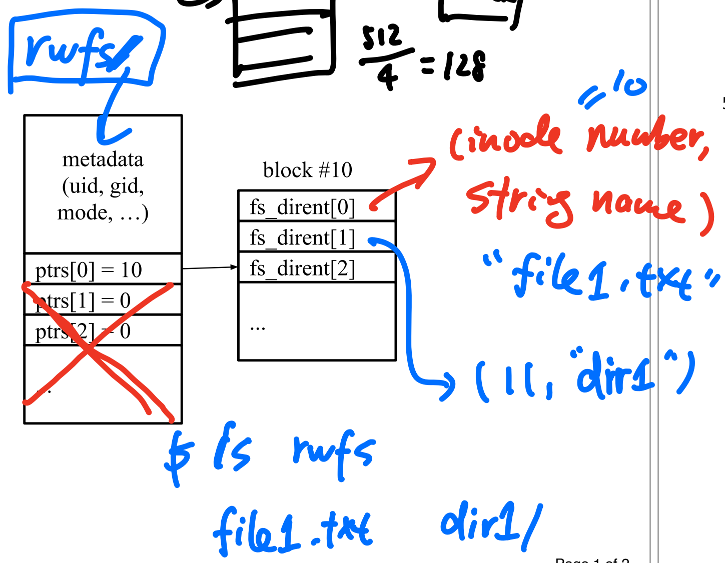
OSI handout week10.b

1. egos-rwfs visualization

inode (file):



inode (directory):



2. egos-rwfs block layout

```
                          |<-inode arr->|<-data blocks->|
   +-------+-------+---+---+---+---+---+---+----------------+
   | super | block |   |   |...|   |   |...|            |
   | block | bitmap|   |   |   |   |   |   |            |
   +-------+-------+---+---+---+---+---+---+----------------+
   block  0       1     2  3   ...11 12...
```

metadata   real data

3. inode mode

```
        file or dir
        |<-    ->|                       |<- S-app ->|<- U-app ->|
        +---+---+---+---+---+---+---+---+---+---+---+---+---+
        | F | D |   |   |   | … |   | R | W | X | R | W | X |
        +---+---+---+---+---+---+---+---+---+---+---+---+---+
```

777 → rwx rwx rwx

664 → rw rw r

4. page cache

```
    typedef struct fs_struct {
        ...

        // page caches
        uint buffer_blk_id;
        block_t buffer_cache;
    } fs_t;

    inode_t *__load_inode(int ino)
    void flush_inode(int ino)
```

sys_file

5. fs interfaces

```
  fs_init - constructor (i.e. put your init code here)
  fs_getsize - get size of a file/directory
  fs_read - read data from a file
  fs_write - write data to a file
```

mkdir
creat
rm

- Concurrency
  - single-core
  - multi-core
  - distributed setup