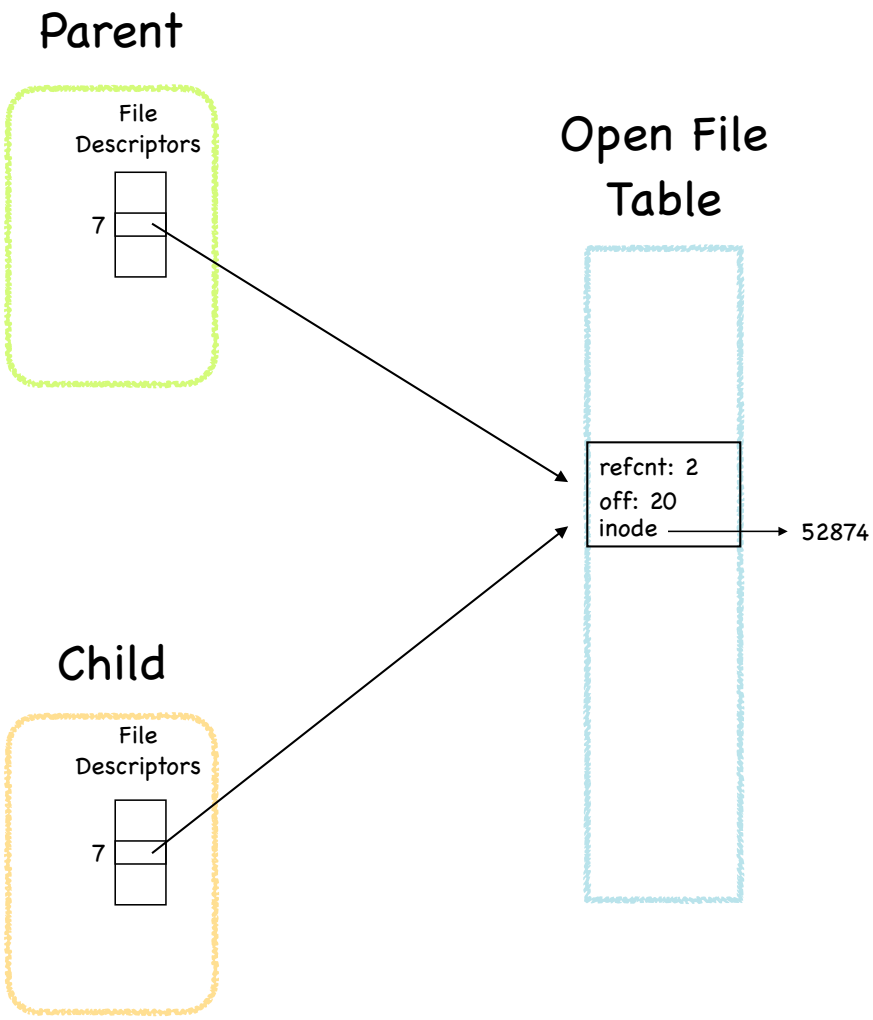
Old Friends

Remember fork()?

```
int main(int argc, char *argv[]){
    int fd = open("file.txt", O_RDONLY);
    assert (fd >= 0);
    int rc = fork();
    if (rc ==0) { /* child */
        rc = lseek(fd, 10, SEEK_SET);
        printf("child: offset %d\n", rc);
    } else if (rc > 0) { /* parent */
        (void) wait(NULL);
        printf("parent: offset %d\n",
            (int) lseek(fd, 10, SEEK_CUR));
    }
    return 0;
}
```



What does this code print?

```
child: offset 10
parent: offset 20
```

The Directory

- The directory holds mappings between human-friendly names (HFNs) and inode numbers
- It stores two types of mappings:
  - Hard links
    - map a file's HFN (its local path) to the file's inode number
  - Symbolic (soft) links
    - Logically, map a file's HFN (its local path) to the HFN of a different file
    - Implementation: maps a file's HFN to the number of an inode that contains the HFN of a different file

Hard links

- Creating file foo adds a hard link for file foo in the file's directory
- Command `ln oldpath newpath`

  adds to the directory for HFN `newpath` a hard link mapping `newpath` to the inode number of the file with HFN `oldpath`

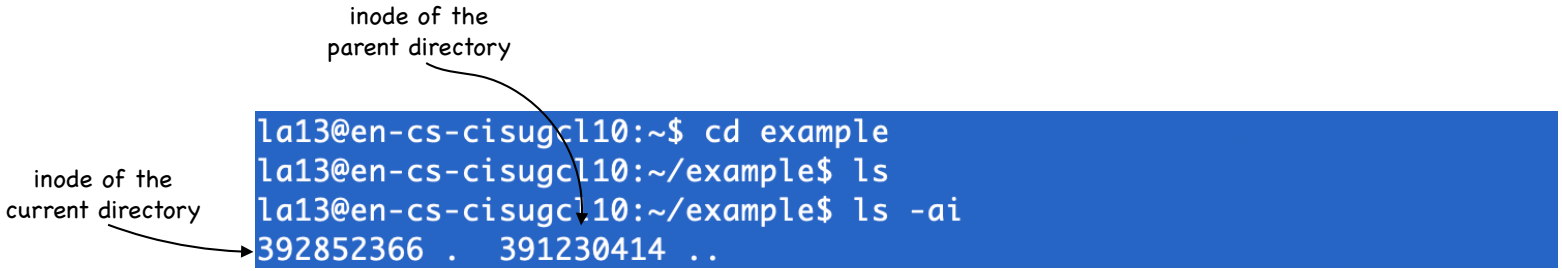
  calls `int link(const char *oldpath, const char *newpath)`
- Removing a file through the `rm [file]` command invokes a call to `int unlink(const char *pathname)`

  removes from directory the hard link between pathname and corresponding inode number
- File's inode stores the number of hard links to it

  inode reclaimed (file deleted) only when link count = 0; if file opened, wait to reclaim until file is closed

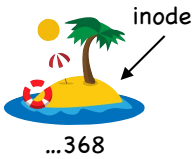
Hard link No-Nos

- Creating a hard link to a directory
  - may create a cycle in the directory tree!
- Creating a hard link to files in other volumes
  - inode numbers are unique only within a single file system


inode of the
parent directory
inode of the
current directory
la13@en-cs-cisugcl10:~$ cd example
la13@en-cs-cisugcl10:~/example$ ls
la13@en-cs-cisugcl10:~/example$ ls -ai
392852366 . 391230414 ..

Example

~/example/cornell




```
la13@en-cs-cisugcl10:~$ cd example
la13@en-cs-cisugcl10:~/example$ ls
la13@en-cs-cisugcl10:~/example$ ls -ai
392852366 .  391230414 ..
la13@en-cs-cisugcl10:~/example$ echo ezra > cornell
la13@en-cs-cisugcl10:~/example$ cat cornell
ezra
la13@en-cs-cisugcl10:~/example$ ls -ai
392852366 .  391230414 ..  392852368 cornell
```

Example

Example

~/example/bigred

~/example/cornell

…368

```
la13@en-cs-cisugcl10:~$ cd example
la13@en-cs-cisugcl10:~/example$ ls
la13@en-cs-cisugcl10:~/example$ ls -ai
392852366 .  391230414 ..
la13@en-cs-cisugcl10:~/example$ echo ezra > cornell
la13@en-cs-cisugcl10:~/example$ cat cornell
ezra
la13@en-cs-cisugcl10:~/example$ ls -ai
392852366 .  391230414 ..  392852368 cornell
la13@en-cs-cisugcl10:~/example$ ln cornell bigred
la13@en-cs-cisugcl10:~/example$ cat bigred
ezra
la13@en-cs-cisugcl10:~/example$ ls -i
392852368 bigred  392852368 cornell
```

Example

~/example/bigred

~/example/cornell

…368

~/bestivy

```
la13@en-cs-cisugcl10:~$ cd example
la13@en-cs-cisugcl10:~/example$ ls
la13@en-cs-cisugcl10:~/example$ ls -ai
392852366 .  391230414 ..
la13@en-cs-cisugcl10:~/example$ echo ezra > cornell
la13@en-cs-cisugcl10:~/example$ cat cornell
ezra
la13@en-cs-cisugcl10:~/example$ ls -ai
392852366 .  391230414 ..  392852368 cornell
la13@en-cs-cisugcl10:~/example$ ln cornell bigred
la13@en-cs-cisugcl10:~/example$ cat bigred
ezra
la13@en-cs-cisugcl10:~/example$ ls -i
392852368 bigred  392852368 cornell
la13@en-cs-cisugcl10:~/example$ ln bigred ../bestivy
la13@en-cs-cisugcl10:~/example$ ls -i
392852368 bigred  392852368 cornell
la13@en-cs-cisugcl10:~/example$ cd ..
la13@en-cs-cisugcl10:~$ cat bestivy
ezra
la13@en-cs-cisugcl10:~$ ls -i
392852368 bestivy  398842589 CS4410-2020sp-A4  392852366 example
```

Example

~/example/bigred

~/example/cornell

…368

~/bestivy

```
la13@en-cs-cisugcl10:~$ cd example
la13@en-cs-cisugcl10:~/example$ ls
la13@en-cs-cisugcl10:~/example$ ls -ai
392852366 .  391230414 ..
la13@en-cs-cisugcl10:~/example$ echo ezra > cornell
la13@en-cs-cisugcl10:~/example$ cat cornell
ezra
la13@en-cs-cisugcl10:~/example$ ls -ai
392852366 .  391230414 ..  392852368 cornell
la13@en-cs-cisugcl10:~/example$ ln cornell bigred
la13@en-cs-cisugcl10:~/example$ cat bigred
ezra
la13@en-cs-cisugcl10:~/example$ ls -i
392852368 bigred  392852368 cornell
la13@en-cs-cisugcl10:~/example$ ln bigred ../bestivy
la13@en-cs-cisugcl10:~/example$ ls -i
392852368 bigred  392852368 cornell
la13@en-cs-cisugcl10:~/example$ cd ..
la13@en-cs-cisugcl10:~$ cat bestivy
ezra
la13@en-cs-cisugcl10:~$ ls -i
392852368 bestivy  398842589 CS4410-2020sp-A4  392852366 example
la13@en-cs-cisugcl10:~$ cd example
la13@en-cs-cisugcl10:~/example$ rm cornell
la13@en-cs-cisugcl10:~/example$ rm bigred
la13@en-cs-cisugcl10:~/example$ ls -i
la13@en-cs-cisugcl10:~/example$ cd ..
```

...368

~/bestivy

Example

```
la13@en-cs-cisugcl10:~$ cd example
la13@en-cs-cisugcl10:~/example$ ls
la13@en-cs-cisugcl10:~/example$ ls -ai
392852366 .  391230414 ..
la13@en-cs-cisugcl10:~/example$ echo ezra > cornell
la13@en-cs-cisugcl10:~/example$ cat cornell
ezra
la13@en-cs-cisugcl10:~/example$ ls -ai
392852366 .  391230414 ..  392852368 cornell
la13@en-cs-cisugcl10:~/example$ ln cornell bigred
la13@en-cs-cisugcl10:~/example$ cat bigred
ezra
la13@en-cs-cisugcl10:~/example$ ls -i
392852368 bigred  392852368 cornell
la13@en-cs-cisugcl10:~/example$ ln bigred ../bestivy
la13@en-cs-cisugcl10:~/example$ ls -i
392852368 bigred  392852368 cornell
la13@en-cs-cisugcl10:~/example$ cd ..
la13@en-cs-cisugcl10:~$ cat bestivy
ezra
la13@en-cs-cisugcl10:~$ ls -i
392852368 bestivy  398842589 CS4410-2020sp-A4  392852366 example
la13@en-cs-cisugcl10:~$ cd example
la13@en-cs-cisugcl10:~/example$ rm cornell
la13@en-cs-cisugcl10:~/example$ rm bigred
la13@en-cs-cisugcl10:~/example$ ls -i
la13@en-cs-cisugcl10:~/example$ cd ..
la13@en-cs-cisugcl10:~$ cat bestivy
ezra
la13@en-cs-cisugcl10:~$ ls -i
392852368 bestivy  398842589 CS4410-2020sp-A4  392852366 example
la13@en-cs-cisugcl10:~$
```

Symbolic (Soft) links

- More flexible than hard links
  - can link to a directory
  - can link to files in another volume
- A map between pathnames
  - to link newpathname to existingpathname for file inode1:
    - create a hard link between newpathname and new file inode2
    - store in inode2 the existingpathname for inode1
  - so, a symbolic link is really a file (inode2 in our example) of a third type
    - neither a regular file nor a directory
- Created using ln, but with the -s flag

Example

~/example/cornell

...367

```
la13@en-cs-cisugcl05:~$ cd example
la13@en-cs-cisugcl05:~/example$ echo ezra > cornell
la13@en-cs-cisugcl05:~/example$ ls -i
392852367 cornell
```

Example

~/example/cornell

...367

~/example/bigred

```
la13@en-cs-cisugcl05:~$ cd example
la13@en-cs-cisugcl05:~/example$ echo ezra > cornell
la13@en-cs-cisugcl05:~/example$ ls -i
392852367 cornell
la13@en-cs-cisugcl05:~/example$ ln cornell bigred
la13@en-cs-cisugcl05:~/example$ ls -i
392852367 bigred  392852367 cornell
```

Example

~/highabove

~/example/cornell

...138

...367

~/bestivy

~/example/bigred

```
la13@en-cs-cisugcl05:~$ cd example
la13@en-cs-cisugcl05:~/example$ echo ezra > cornell
la13@en-cs-cisugcl05:~/example$ ls -i
392852367 cornell
la13@en-cs-cisugcl05:~/example$ ln cornell bigred
la13@en-cs-cisugcl05:~/example$ ls -i
392852367 bigred  392852367 cornell
la13@en-cs-cisugcl05:~/example$ cd ..
la13@en-cs-cisugcl05:~$ ln example/cornell bestivy
la13@en-cs-cisugcl05:~$ ln -s example/cornell highabove
la13@en-cs-cisugcl05:~$ ls -i
392852367 bestivy  398842589 CS4410-2020sp-A4  392852366 example  392971138 highabove
```

Example

~/highabove

~/example/cornell

...138

~/bestivy

~/example/bigred

...367

```
la13@en-cs-cisugcl05:~$ cd example
la13@en-cs-cisugcl05:~/example$ echo ezra > cornell
la13@en-cs-cisugcl05:~/example$ ls -i
392852367 cornell
la13@en-cs-cisugcl05:~/example$ ln cornell bigred
la13@en-cs-cisugcl05:~/example$ ls -i
392852367 bigred  392852367 cornell
la13@en-cs-cisugcl05:~/example$ cd ..
la13@en-cs-cisugcl05:~$ ln example/cornell bestivy
la13@en-cs-cisugcl05:~$ ln -s example/cornell highabove
la13@en-cs-cisugcl05:~$ ls -i
392852367 bestivy  398842589 CS4410-2020sp-A4  392852366 example  392971138 highabove
la13@en-cs-cisugcl05:~$ ls -l
total 8
-rw-r--r-- 3 la13 pug-la13    5 Apr 28 23:03 bestivy
drwxr-sr-x 4 la13 pug-la13 4096 Apr 27 11:55 CS4410-2020sp-A4
drwxr-sr-x 2 la13 pug-la13 4096 Apr 28 23:03 example
lrwxrwxrwx 1 la13 pug-la13   15 Apr 28 23:04 highabove -> example/cornell
```

Example

~/highabove

~/example/cornell

...138

~/bestivy

~/example/bigred

...367

```
la13@en-cs-cisugcl05:~$ cd example
la13@en-cs-cisugcl05:~/example$ echo ezra > cornell
la13@en-cs-cisugcl05:~/example$ ls -i
392852367 cornell
la13@en-cs-cisugcl05:~/example$ ln cornell bigred
la13@en-cs-cisugcl05:~/example$ ls -i
392852367 bigred  392852367 cornell
la13@en-cs-cisugcl05:~/example$ cd ..
la13@en-cs-cisugcl05:~$ ln example/cornell bestivy
la13@en-cs-cisugcl05:~$ ln -s example/cornell highabove
la13@en-cs-cisugcl05:~$ ls -i
392852367 bestivy  398842589 CS4410-2020sp-A4  392852366 example  392971138 highabove
la13@en-cs-cisugcl05:~$ ls -l
total 8
-rw-r--r-- 3 la13 pug-la13    5 Apr 28 23:03 bestivy
drwxr-sr-x 4 la13 pug-la13 4096 Apr 27 11:55 CS4410-2020sp-A4
drwxr-sr-x 2 la13 pug-la13 4096 Apr 28 23:03 example
lrwxrwxrwx 1 la13 pug-la13   15 Apr 28 23:04 highabove -> example/cornell
la13@en-cs-cisugcl05:~$ cat bestivy
ezra
la13@en-cs-cisugcl05:~$ cat highabove
ezra
```

Example

~/highabove

~/example/cornell

...138

~/bestivy

~/example/bigred

...367

```
la13@en-cs-cisugcl05:~$ cd example
la13@en-cs-cisugcl05:~/example$ echo ezra > cornell
la13@en-cs-cisugcl05:~/example$ ls -i
392852367 cornell
la13@en-cs-cisugcl05:~/example$ ln cornell bigred
la13@en-cs-cisugcl05:~/example$ ls -i
392852367 bigred  392852367 cornell
la13@en-cs-cisugcl05:~/example$ cd ..
la13@en-cs-cisugcl05:~$ ln example/cornell bestivy
la13@en-cs-cisugcl05:~$ ln -s example/cornell highabove
la13@en-cs-cisugcl05:~$ ls -i
392852367 bestivy  398842589 CS4410-2020sp-A4  392852366 example  392971138 highabove
la13@en-cs-cisugcl05:~$ ls -l
total 8
-rw-r--r-- 3 la13 pug-la13    5 Apr 28 23:03 bestivy
drwxr-sr-x 4 la13 pug-la13 4096 Apr 27 11:55 CS4410-2020sp-A4
drwxr-sr-x 2 la13 pug-la13 4096 Apr 28 23:03 example
lrwxrwxrwx 1 la13 pug-la13   15 Apr 28 23:04 highabove -> example/cornell
la13@en-cs-cisugcl05:~$ cat bestivy
ezra
la13@en-cs-cisugcl05:~$ cat highabove
ezra
la13@en-cs-cisugcl05:~$ rm example/cornell
```

Example

~/highabove

...138

~/bestivy

...367

~/example/bigred

```
la13@en-cs-cisugcl05:~$ cd example
la13@en-cs-cisugcl05:~/example$ echo ezra > cornell
la13@en-cs-cisugcl05:~/example$ ls -i
392852367 cornell
la13@en-cs-cisugcl05:~/example$ ln cornell bigred
la13@en-cs-cisugcl05:~/example$ ls -i
392852367 bigred  392852367 cornell
la13@en-cs-cisugcl05:~/example$ cd ..
la13@en-cs-cisugcl05:~$ ln example/cornell bestivy
la13@en-cs-cisugcl05:~$ ln -s example/cornell highabove
la13@en-cs-cisugcl05:~$ ls -i
392852367 bestivy  398842589 CS4410-2020sp-A4  392852366 example  392971138 highabove
la13@en-cs-cisugcl05:~$ ls -l
total 8
-rw-r--r-- 3 la13 pug-la13    5 Apr 28 23:03 bestivy
drwxr-sr-x 4 la13 pug-la13 4096 Apr 27 11:55 CS4410-2020sp-A4
drwxr-sr-x 2 la13 pug-la13 4096 Apr 28 23:03 example
lrwxrwxrwx 1 la13 pug-la13   15 Apr 28 23:04 highabove -> example/cornell
la13@en-cs-cisugcl05:~$ cat bestivy
ezra
la13@en-cs-cisugcl05:~$ cat highabove
ezra
la13@en-cs-cisugcl05:~$ rm example/cornell
la13@en-cs-cisugcl05:~$ cat bestivy
ezra
```

Example

~/highabove

...138

~/bestivy

...367

~/example/bigred

```
la13@en-cs-cisugcl05:~$ cd example
la13@en-cs-cisugcl05:~/example$ echo ezra > cornell
la13@en-cs-cisugcl05:~/example$ ls -i
392852367 cornell
la13@en-cs-cisugcl05:~/example$ ln cornell bigred
la13@en-cs-cisugcl05:~/example$ ls -i
392852367 bigred  392852367 cornell
la13@en-cs-cisugcl05:~/example$ cd ..
la13@en-cs-cisugcl05:~$ ln example/cornell bestivy
la13@en-cs-cisugcl05:~$ ln -s example/cornell highabove
la13@en-cs-cisugcl05:~$ ls -i
392852367 bestivy  398842589 CS4410-2020sp-A4  392852366 example  392971138 highabove
la13@en-cs-cisugcl05:~$ ls -l
total 8
-rw-r--r-- 3 la13 pug-la13    5 Apr 28 23:03 bestivy
drwxr-sr-x 4 la13 pug-la13 4096 Apr 27 11:55 CS4410-2020sp-A4
drwxr-sr-x 2 la13 pug-la13 4096 Apr 28 23:03 example
lrwxrwxrwx 1 la13 pug-la13   15 Apr 28 23:04 highabove -> example/cornell
la13@en-cs-cisugcl05:~$ cat bestivy
ezra
la13@en-cs-cisugcl05:~$ cat highabove
ezra
la13@en-cs-cisugcl05:~$ rm example/cornell
la13@en-cs-cisugcl05:~$ cat bestivy
ezra
la13@en-cs-cisugcl05:~$ cat highabove
cat: highabove: No such file or directory
la13@en-cs-cisugcl05:~$
```

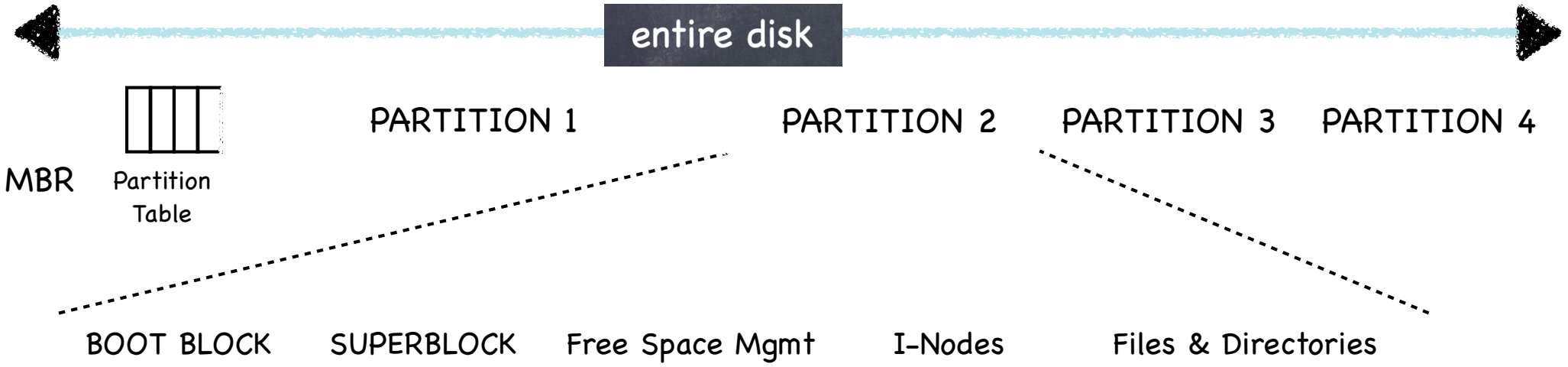
Permission Bits

```
la13@en-cs-cisugcl05:~$ ls -l
total 8
-rw-r--r-- 3 la13 pug-la13    5 Apr 28 23:03 bestivy
drwxr-sr-x 4 la13 pug-la13 4096 Apr 27 11:55 CS4410-2020sp-A4
drwxr-sr-x 2 la13 pug-la13 4096 Apr 28 23:03 example
lrwxrwxrwx 1 la13 pug-la13   15 Apr 28 23:04 highabove -> example/cornell
```

- File bestivy
  - leading - says bestivy is a regular file
    - d is for directory; l is for soft link
  - Next nine characters are permission bits
    - rwx for owner, group, everyone
      - owner can read and write; group and others can just read
      - x set in a regular file means means file can be executed
      - x set in a directory that user/group/everybody is allow to cd to that directory
    - can be set using chmod

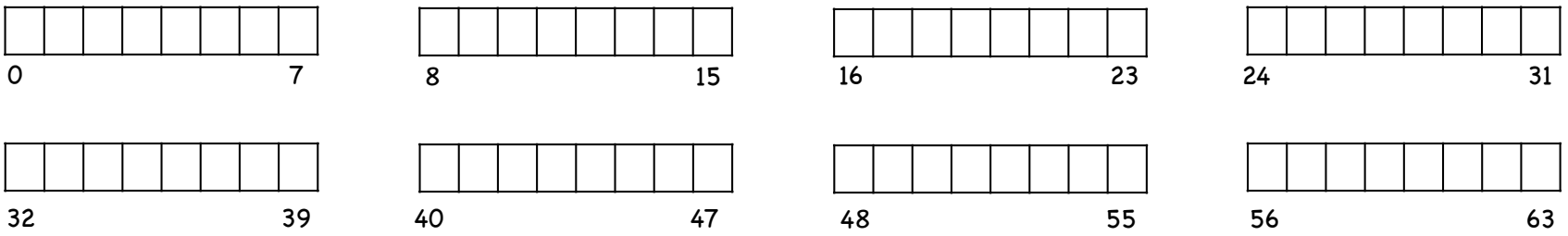
File System Layout

- File System is stored on disks
  - disk can be divided into one or more partitions
  - Sector 0 of disk: Master Boot Record (MBR). It contains:
    - bootstrap code (loaded and executed by firmware)
    - partition table (addresses of where partitions start & end)
  - First block of each partition has boot block
    - loaded by executing code in MBR and executed on boot



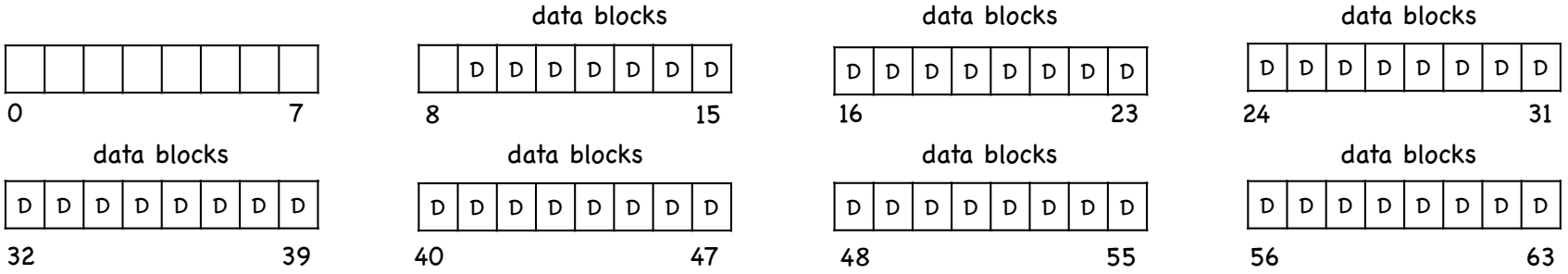
Peeking Inside

- Persistent storage modeled as a sequence of N blocks
  - from 0 to N-1
    - in this example , 64 blocks, each 4KB
  - some blocks store data



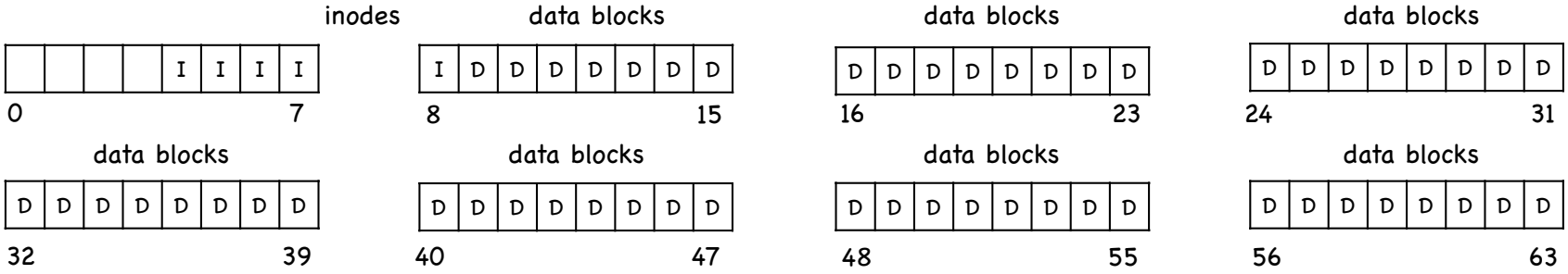
Peeking Inside

- Persistent storage modeled as a sequence of N blocks
  - from 0 to N-1
    - in this example , 64 blocks, each 4KB
  - some blocks store data
  - other blocks store metadata
    - an array of inodes
      - at 256 bytes, 16 per block: with 5 blocks for inodes, file system can have up to 80 files



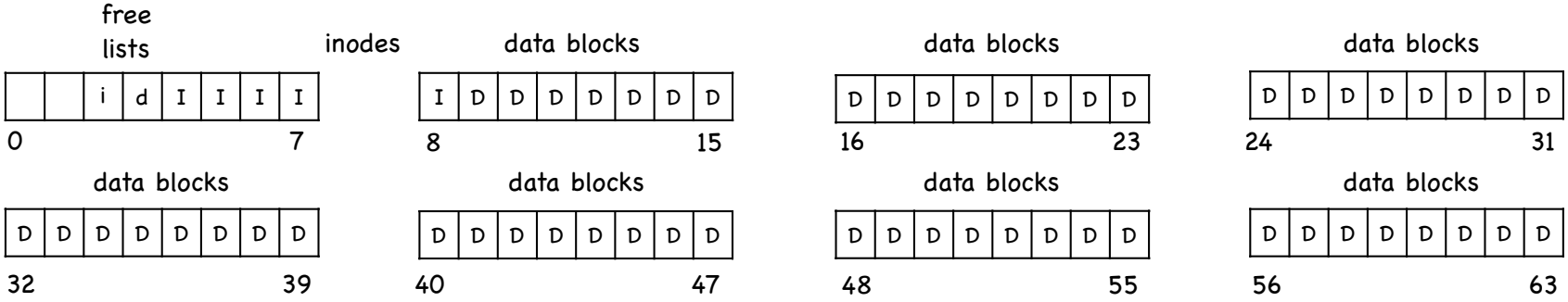
Peeking Inside

- Persistent storage modeled as a sequence of N blocks
  - from 0 to N-1
    - in this example , 64 blocks, each 4KB
  - some blocks store data
  - other blocks store metadata
    - an array of inodes
      - at 256 bytes, 16 per block: with 5 blocks for inodes, file system can have up to 80 files



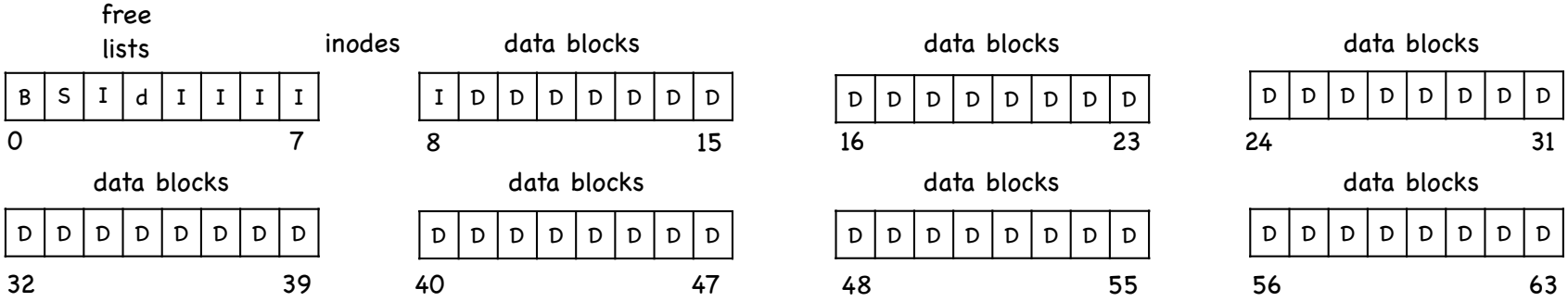
Peeking Inside

- Persistent storage modeled as a sequence of N blocks
  - from 0 to N-1
    - in this example , 64 blocks, each 4KB
  - some blocks store data
  - other blocks store metadata (remember stat()?)
    - an array of inodes
      - at 256 bytes, 16 per block: with 5 blocks for inodes, file system can have up to 80 files
    - bitmaps tracking free inodes and data blocks;



Peeking Inside

- Persistent storage modeled as a sequence of N blocks
  - from 0 to N-1
    - in this example , 64 blocks, each 4KB
  - some blocks store data
  - other blocks store metadata (remember stat()?)
    - an array of inodes
      - at 256 bytes, 16 per block: with 5 blocks for inodes, file system can have up to 80 files
    - bitmaps tracking free inodes and data blocks; Superblock; Boot block



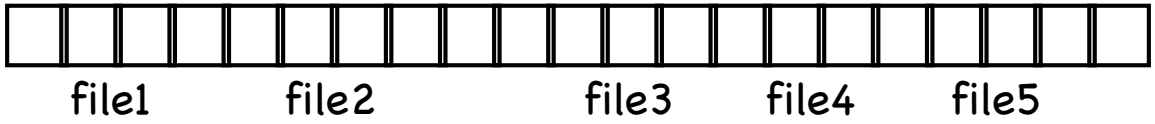
The  uperblock

- One logical superblock per file system
  - at a well-known location
  - contains metadata about the file system, including
    - how many inodes
    - how many data blocks
    - where the inode table begins
    - may contain info to manage free inodes/data blocks
  - read first when mounting a file system

Storing Files

- Files can be allocated in different ways
  - Contiguous allocation
    - all bytes together, in order
  - Linked Structure
    - Each points to the next block
  - Indexed Structure
    - Index block, pointing to many other blocks
- Which is best?
  - For sequential access? Random access?
  - Large files? Small files? Mixed?

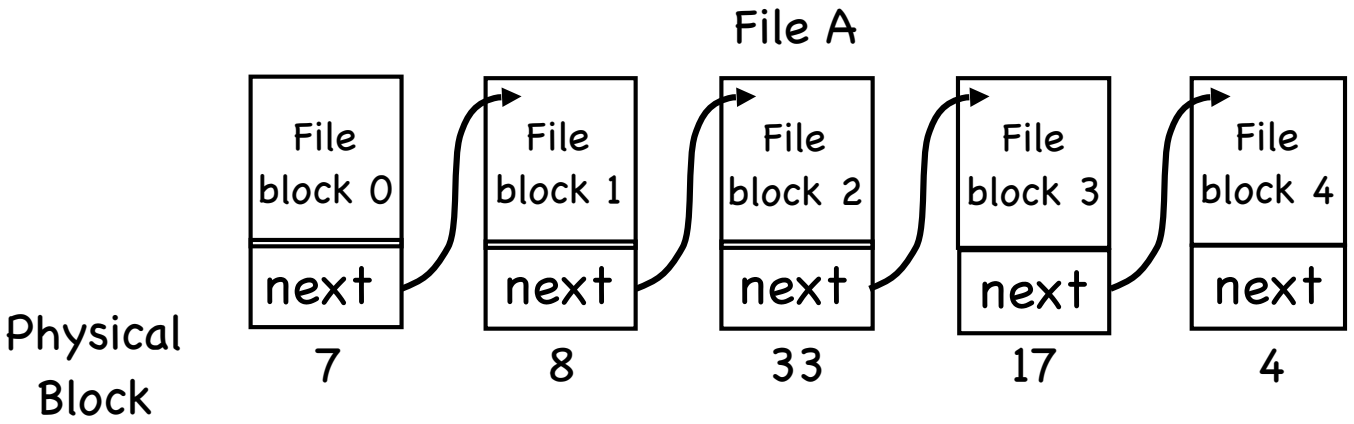
Contiguous Allocation



- All bytes together, in order
  - Simple: only need start block and size
  - Efficient: one seek to read entire file
  - Fragmentation: external, and can be serious
  - Usability: User need to know file's size at time of creation

Used in CD-ROm, DVDs

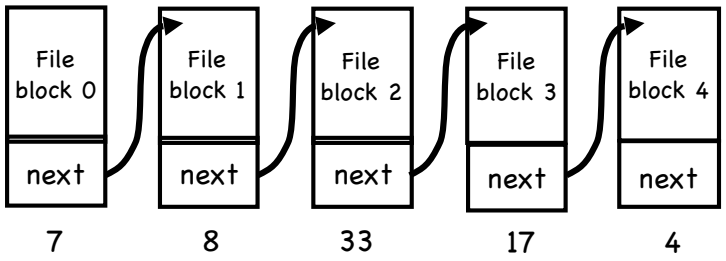
Linked List Allocation

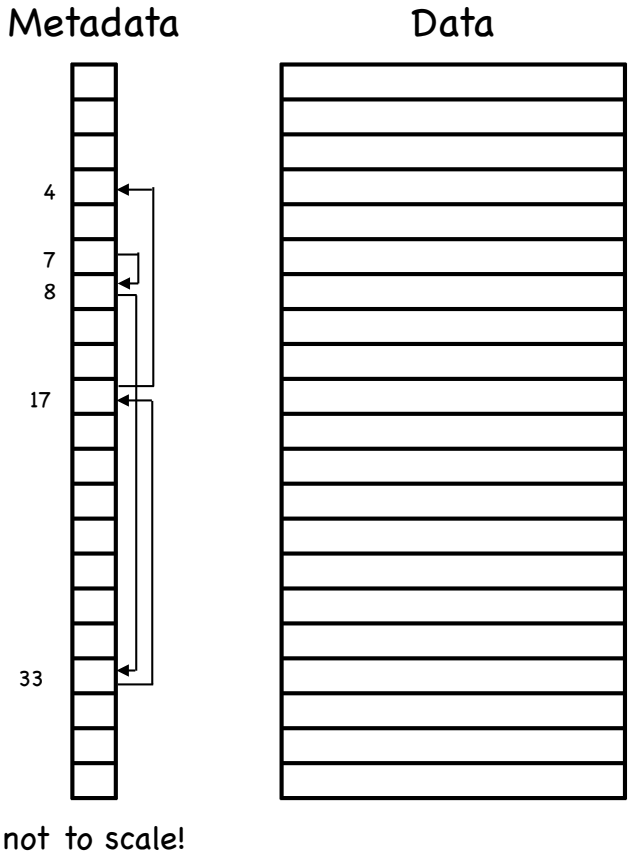


- Each file is stored as a linked list of blocks
  - first word of each block points to next block
  - the rest of the block is data
- Space utilization: no external fragmentation
- Simplicity: only need to find first block of each file
- Performance: random access is slow
- Implementation: blocks mix data and metadata

File Allocation Table (FAT) FS

- Decouple data and metadata





Microsoft, late 70s

still widely used today

thumb drives, camera cards, CD ROMs

FAT File system

Index Structures

File Allocation Table (FAT)

- array of 32-bit entries
- one entry per block
- file represented as a linked list of FAT entries
- file # = index of first FAT entry

Free space map

- If data block i is free, then FAT[i] = 0
- find free blocks by scanning FAT

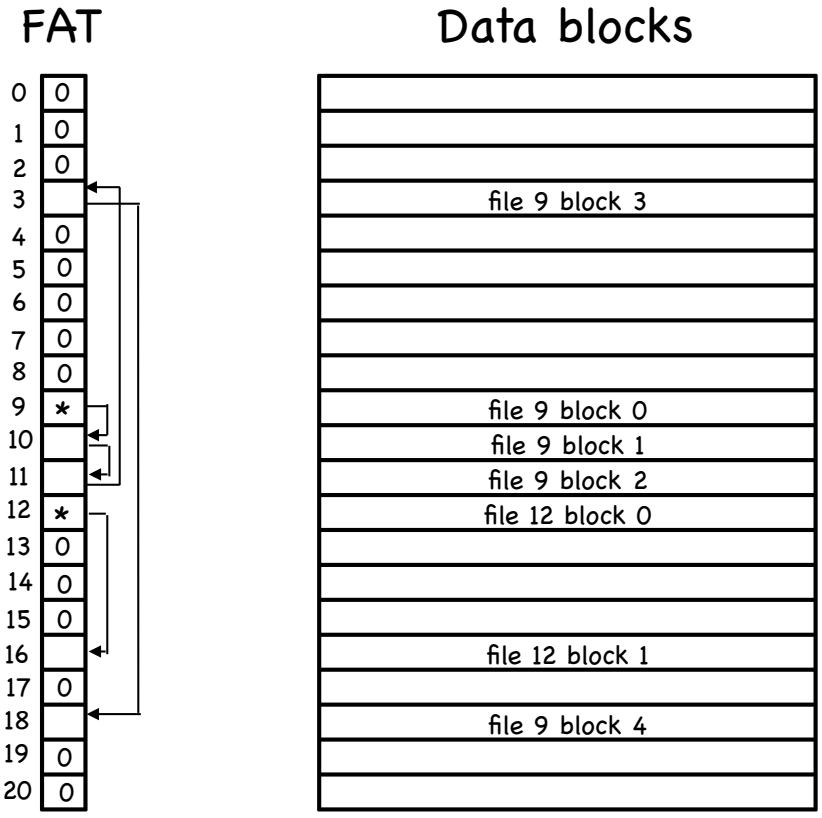
Locality heuristics

- As simple as next fit: scan sequentially from last allocated entry and return next free entry
- Can be improved through defragmentation

Directory

- Maps file name to FAT index

Directory	
jack.txt	12
jill.txt	9



FAT File system

Advantages
- simple!
  - per file, needs only start block
- widely supported
- no external fragmentation
- no conflating data and metadata in the same block

Disadvantages
- Poor locality
  - many file seeks unless entire FAT in memory
  - 1 TB (2^{40} bytes) disk, 4kb (2^{12} bytes block, 2^{28} FAT entries; at 4B/entry, 1 GB (!)
- Poor random access
  - needs sequential traversal
- Limited access control
  - no file owner or group ID
  - any user can read/write any file
- No support for hard links
- Volume and file size are limited
  - FAT entry is 32 bits, but top 4 are reserved
  - no more than 2^{28} blocks
  - with 4kB blocks, at most 1TB FS
  - file no bigger than 4GB
- No support for advanced reliability techniques