

CS350 Term Project
Instructor: İsmail Arı

File Splitter

Arman Garip
Enes Şenel

Outline

- ❖ Why?
- ❖ Ways to solve it
 - ❖ Split files via reading and writing binary
 - ❖ Split files via manipulating inodes

Why?

- ❖ Some files are so huge and it is hard to move or share them as one piece
- ❖ Splitting makes sending, receiving and storing easier
- ❖ Easier backups

Binary Way

- ❖ Read a file in binary and split it into multiple files
- ❖ Using system calls / wrappers
- ❖ C code to split and combine any file

- ❖ System calls that we used for file management:
 - ❖ `fd = open(file, how, ...)`
 - ❖ `s = close(fd)`
 - ❖ `n = read(fd, buffer, bytes)`
 - ❖ `n = write(fd, buffer, bytes)`
 - ❖ `position = lseek(fd, offset, whence)`

❖ Splitting:

```
Enes-MacBook-Air:binarySplit enessenel$ ./split
Enter the file name to split (full path): ../testCases/testImg.bmp
How many parts: 5
```

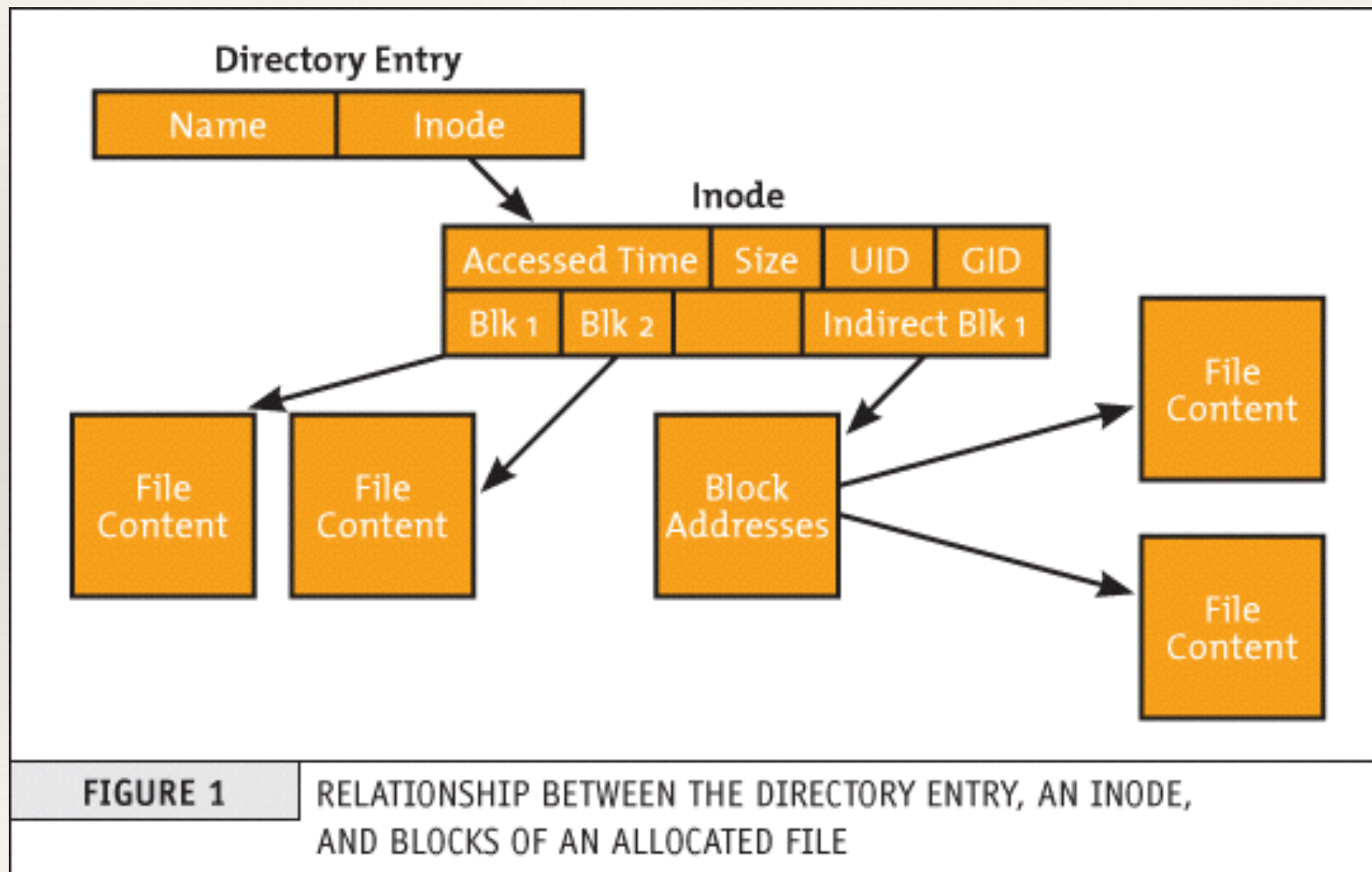
testimg.bmp	15 Jan 2015 17:15	185 KB	Windo...Image
testimg.bmp.1	Today 18:36	37 KB	Document
testimg.bmp.2	Today 18:36	37 KB	Document
testimg.bmp.3	Today 18:36	37 KB	Document
testimg.bmp.4	Today 18:36	37 KB	Document
testimg.bmp.5	Today 18:36	37 KB	Document

❖ Combining:

```
Enes-MacBook-Air:binarySplit enessenel$ ./combine
Enter the file name to combine: ../testCases/testImg.bmp
```


Better Way -> Manipulating inodes

❖ An inode?



- ❖ inodes hold the metadata for the files.
- ❖ We can change inode attributes so that we can change the file.
- ❖ Maybe we can make an inode point to some of the data and another inode point to another part of that data. So we can split a file easily?

- ❖ We can use **debugfs** tool in linux to manipulate the filesystem.
- ❖ And the **mi <inode>** function of debugfs to **modify inode**.

enes@enes-virtual-machine ~ \$ sudo debugfs -w /dev/sda1

[sudo] password for enes:

debugfs 1.42.9 (4-Feb-2014)

debugfs: mi /home/enes/Desktop/test.txt

Mode	[0100644]
User ID	[1000]
Group ID	[1000]
Size	[34]
Creation time	[1432651582]
Modification time	[1432651582]
Access time	[1432651582]
Deletion time	[0]
Link count	[1]
Block count high	[0]
Block count	[8]
File flags	[0x80000]
Generation	[0x9b2a81f6]
File acl	[0]
High 32bits of size	[0]
Fragment address	[0]
Direct Block #0	[127754]
Direct Block #1	[4]
Direct Block #2	[0]
Direct Block #3	[0]
Direct Block #4	[1]
Direct Block #5	[1155169]
Direct Block #6	[0]
Direct Block #7	[0]
Direct Block #8	[0]
Direct Block #9	[0]
Direct Block #10	[0]
Direct Block #11	[0]

- ❖ We were able to modify inodes using **debugfs** to copy or delete a file.
- ❖ However, it is not very clear how the data blocks store data so we weren't able to implement it for a general case. We can copy every file by making two inodes point the same data blocks but splitting was unsuccessful.