



CS 1550

Week 13 Project 4

TA: Henrique Potter

Overview

- FUSE is a **Linux kernel extension** that allows for a user space program to provide the implementations for the various file-related syscalls
- Goal: Use FUSE to create our own file system

Overview: Original File System

User

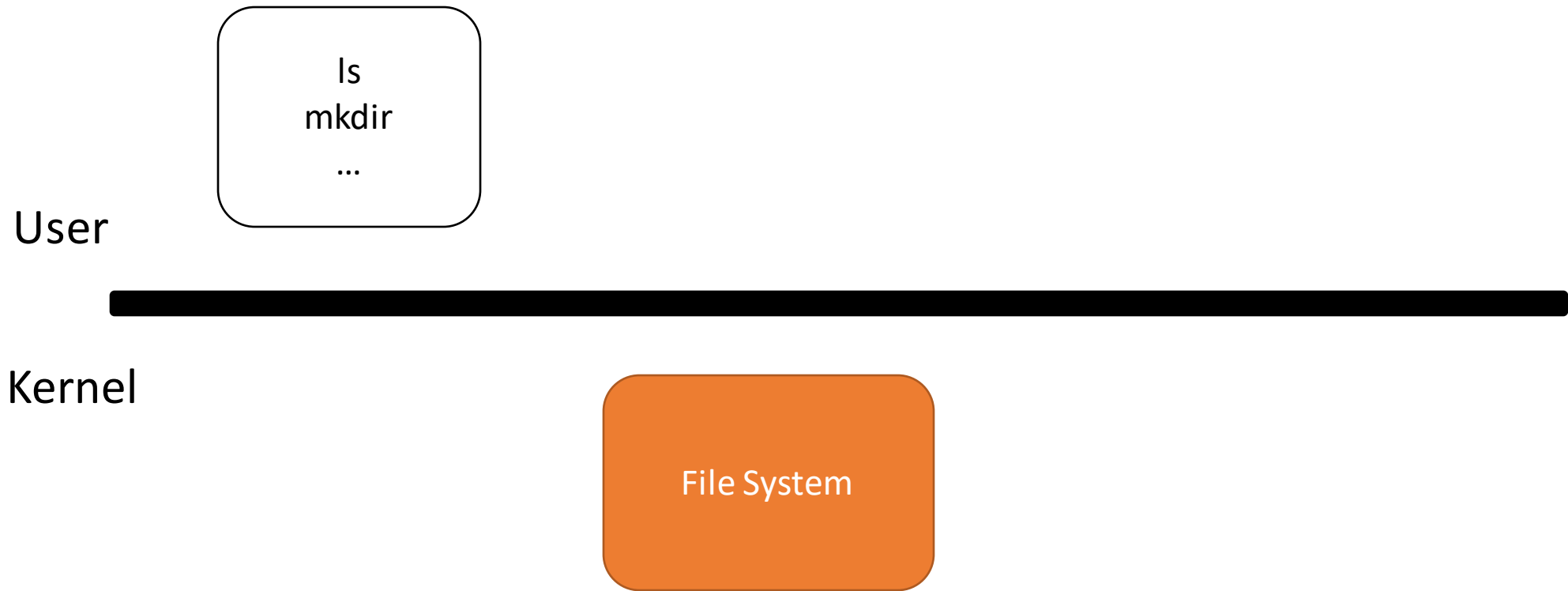


Kernel

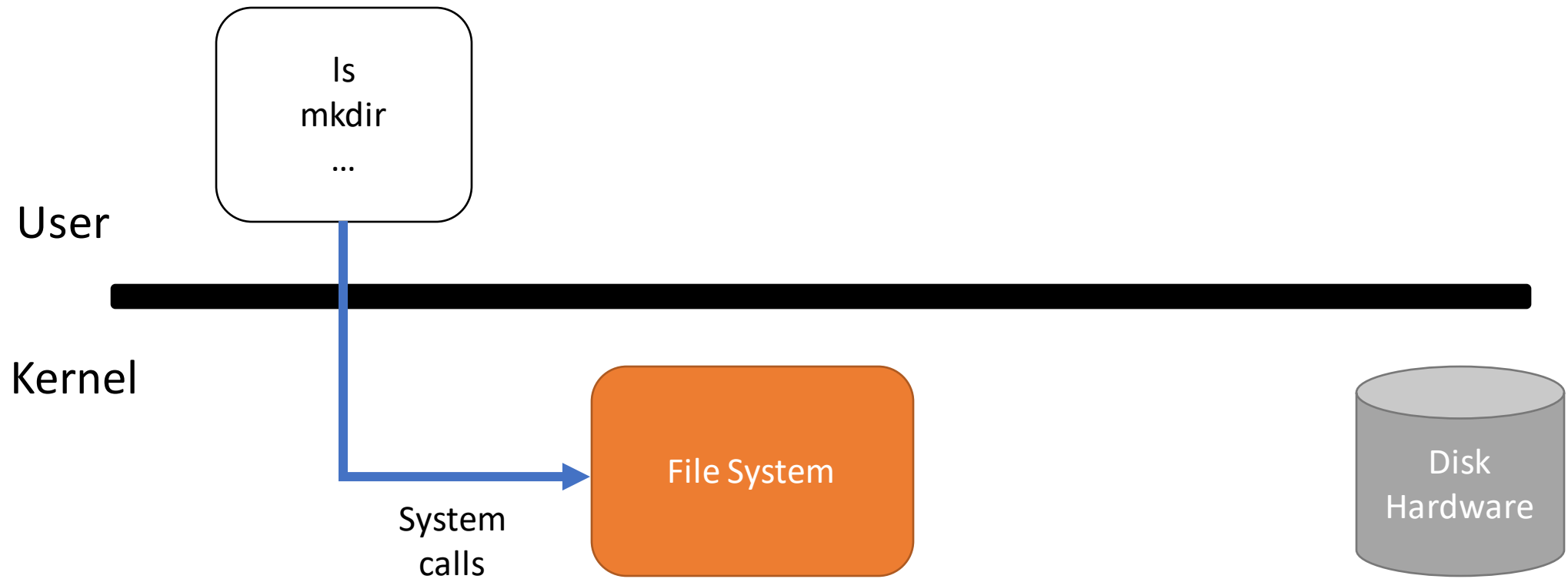
Overview: Original File System



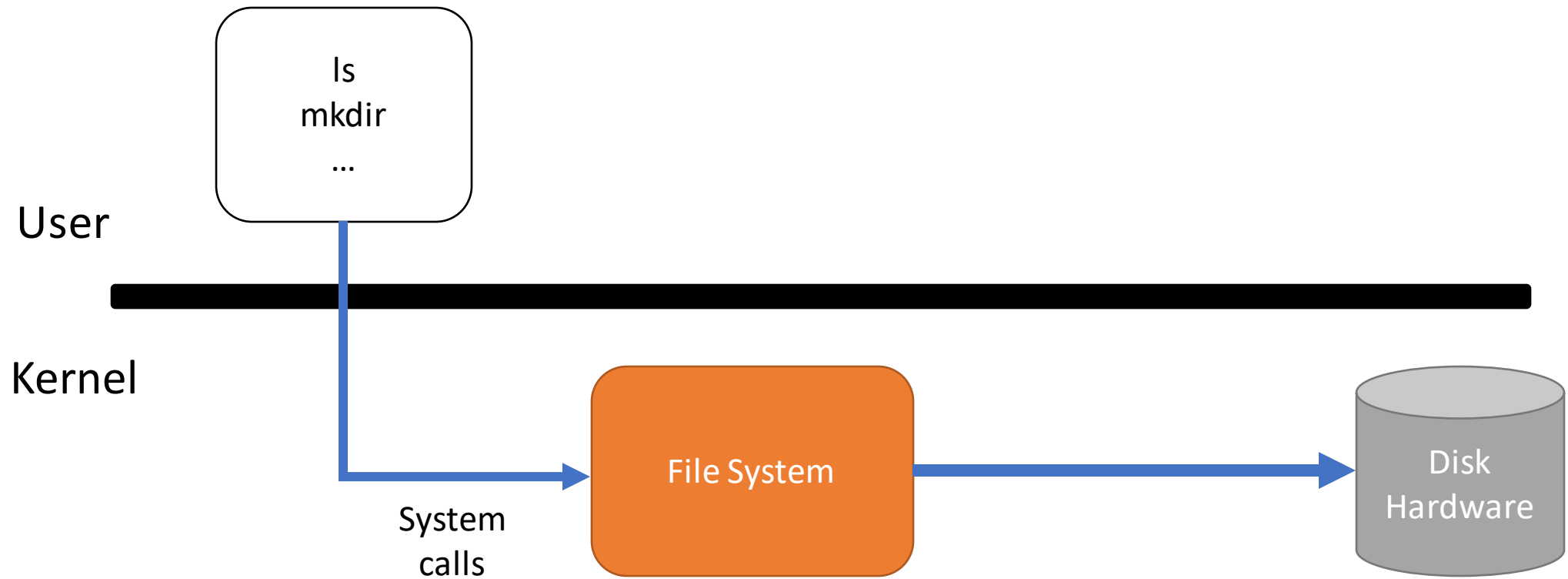
Overview: Original File System



Overview: Original File System



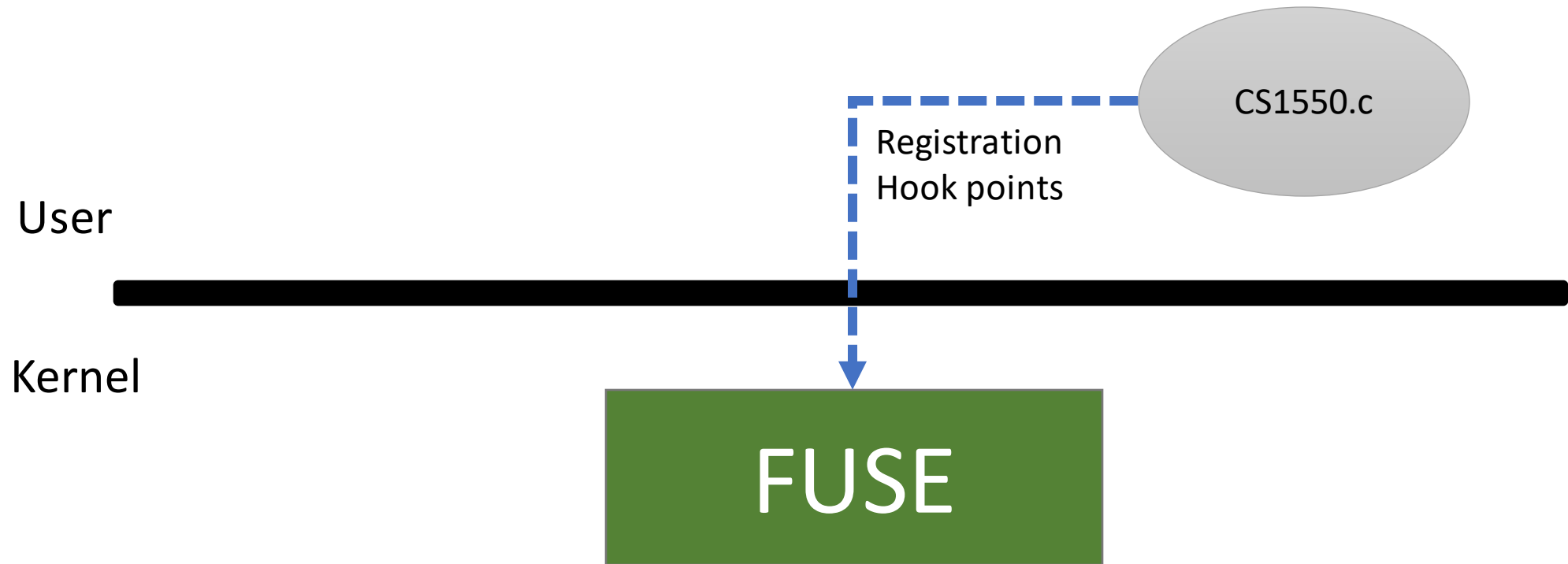
Overview: Original File System



Overview: User Space File System



Overview: User Space File System

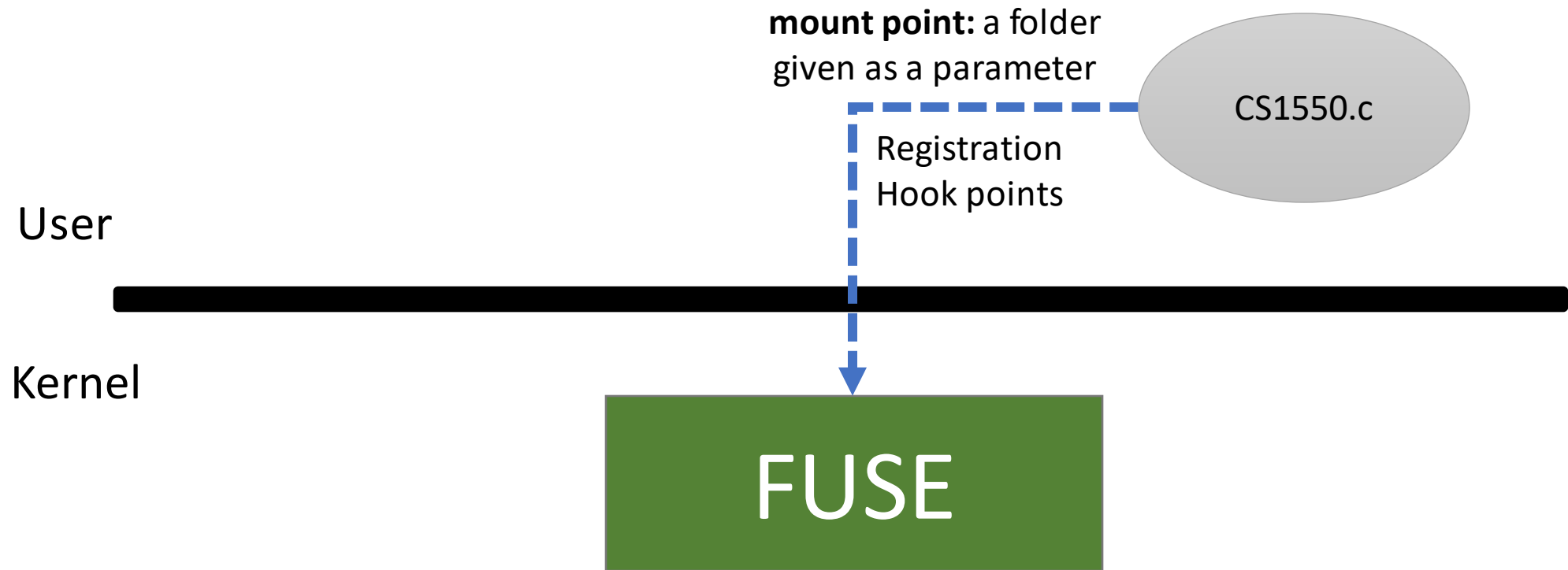


Overview: User Space File System

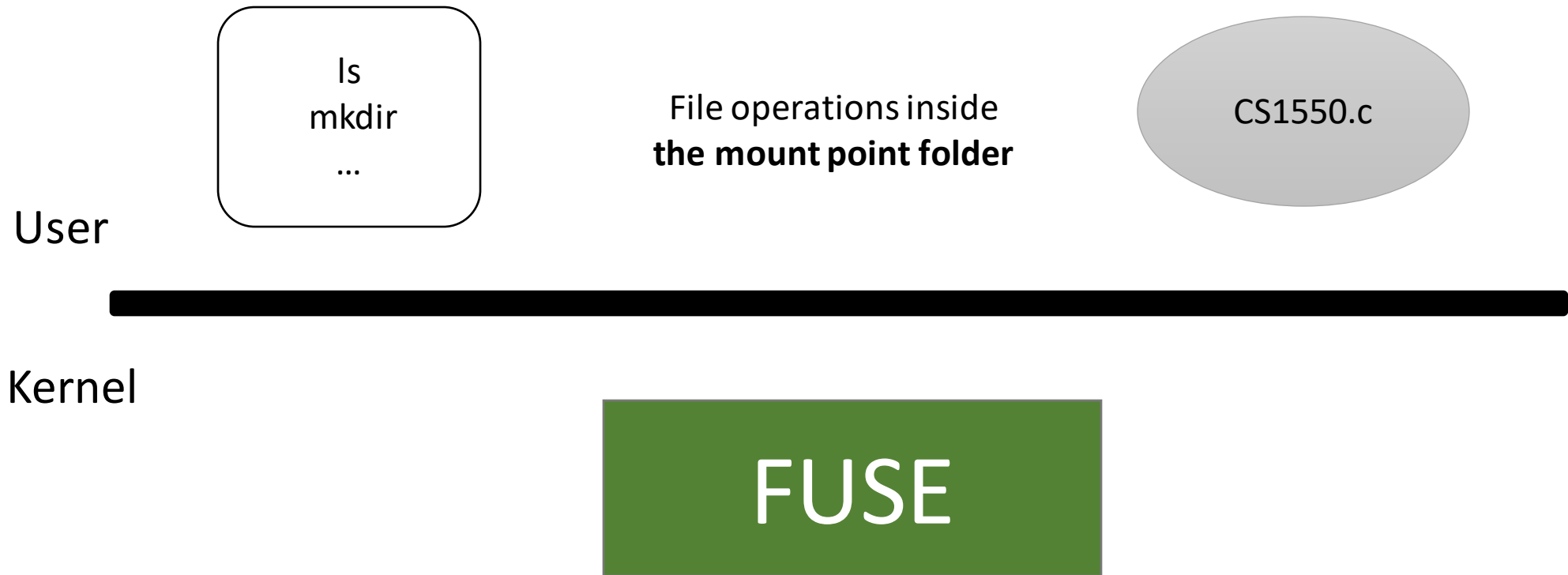
- Example of hooks in hello.c

```
static struct fuse_operations hello_oper = {  
    .getattr  = hello_getattr,  
    .readdir  = hello_readdir,  
    .open     = hello_open,  
    .read     = hello_read,  
};
```

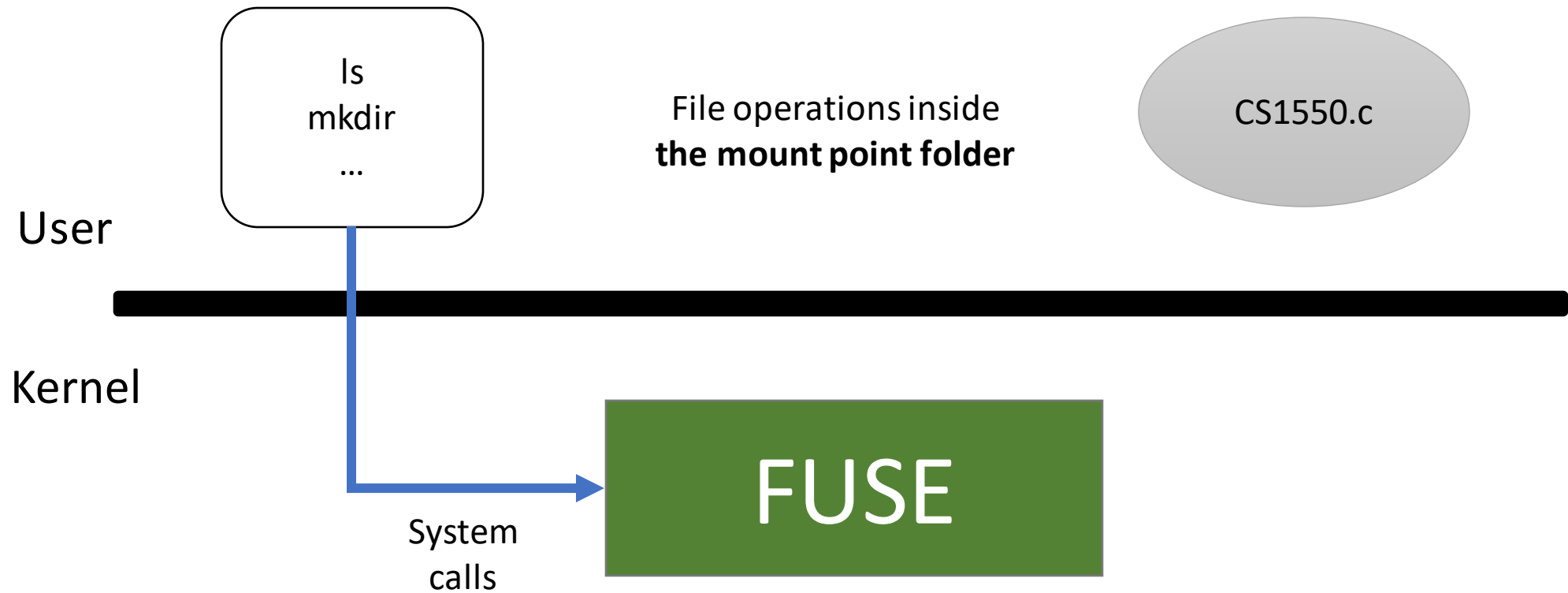
Overview: User Space File System



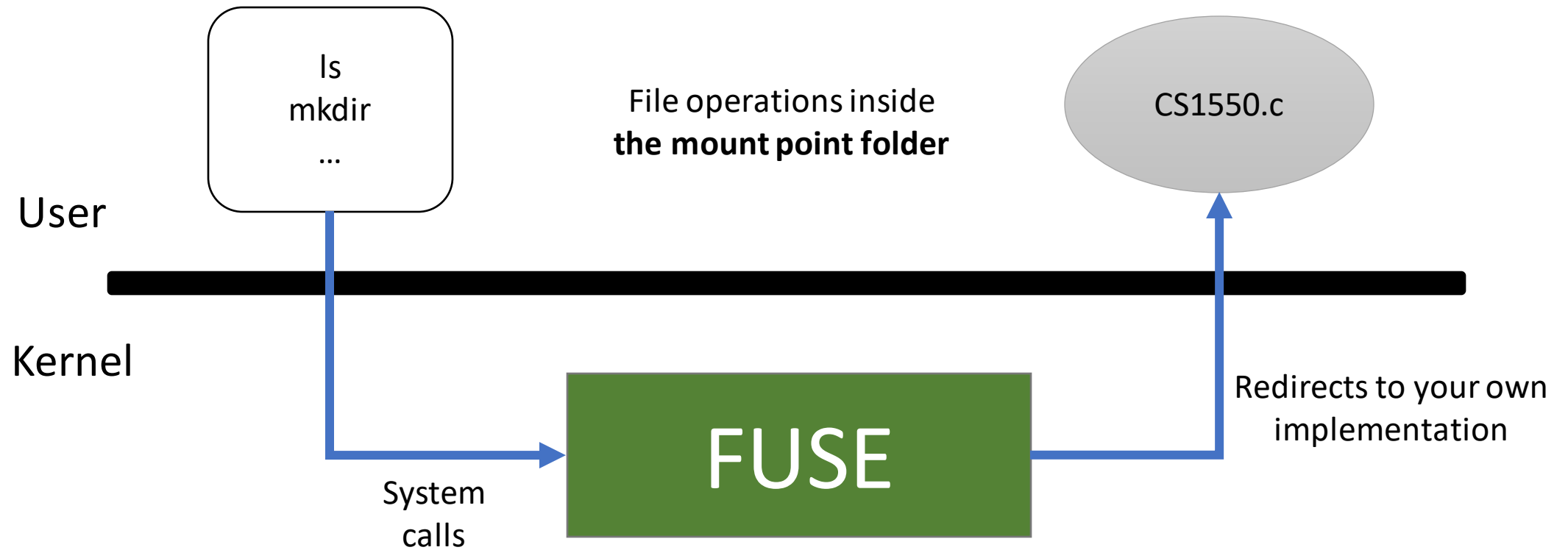
Overview: User Space File System



Overview: User Space File System



Overview: User Space File System



Installation of FUSE

- Kernel is installed
- Install libraries and example programs

```
cd /u/OSLab/USERNAME
```

```
cp /u/OSLab/original/fuse-2.7.0.tar.gz .
```

```
tar xvfz fuse-2.7.0.tar.gz
```

```
cd fuse-2.7.0
```

```
./configure
```

```
make
```

Installation of FUSE

- Kernel is installed
- Install libraries and example programs

```
cd /u/OSLab/USERNAME
```

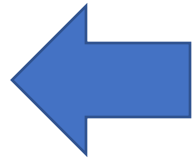
```
cp /u/OSLab/original/fuse-2.7.0.tar.gz .
```

```
tar xvfz fuse-2.7.0.tar.gz
```

```
cd fuse-2.7.0
```

```
./configure
```

```
make
```



This compiles the examples.

FUSE Example

```
cd /u/OSLab/USERNAME/
```

FUSE Example

```
cd /u/OSLab/USERNAME/  
cd fuse-2.7.0/example
```

FUSE Example

```
cd /u/OSLab/USERNAME/
```

```
cd fuse-2.7.0/example
```

```
mkdir testmount (create mount point)
```

A mount point is a location in the UNIX hierarchical file system where a new device or file system is located

FUSE Example

```
cd /u/OSLab/USERNAME/
```

```
cd fuse-2.7.0/example
```

```
mkdir testmount (create mount point)
```

A mount point is a location in the UNIX hierarchical file system where a new device or file system is located

```
ls -al testmount
```

← Still using Original File System

FUSE Example

```
cd /u/OSLab/USERNAME/
```

```
cd fuse-2.7.0/example
```

```
mkdir testmount (create mount point)
```

A mount point is a location in the UNIX hierarchical file system where a new device or file system is located

```
ls -al testmount
```

← Still using Original File System

```
./hello testmount
```

FUSE Example

```
cd /u/OSLab/USERNAME/
```

```
cd fuse-2.7.0/example
```

```
mkdir testmount (create mount point)
```

A mount point is a location in the UNIX hierarchical file system where a new device or file system is located

```
ls -al testmount      ← Still using Original File System
```

```
./hello testmount
```

```
ls -al testmount
```

FUSE Example

```
cd /u/OSLab/USERNAME/
```

```
cd fuse-2.7.0/example
```

```
mkdir testmount (create mount point)
```

A mount point is a location in the UNIX hierarchical file system where a new device or file system is located

```
ls -al testmount      ← Still using Original File System
```

```
./hello testmount
```

```
ls -al testmount      ← Redirect to your implementation in Hello.c
```

Should see . , .., hello

FUSE Example

- “ls -al” requests
 - .getattr
 - .readdir

```
drwxr-xr-x 2 root  root          0 Dec 31  1969 .  
drwxr-xr-x 5 yud42 UNKNOWN1 4096 Apr  6 15:43 ..  
-r--r--r-- 1 _root  root        13 Dec 31  1969 hello
```


FUSE Example

- “ls -al” requests
 - .getattr
 - .readdir

```
drwxr-xr-x 2 root  root          0 Dec 31  1969 .
drwxr-xr-x 5 yud42 UNKNOWN1 4096 Apr  6 15:43 ..
-r--r--r-- 1 _root  root        13 Dec 31  1969 hello
```

- ‘.’ → current directory
- ‘..’ → parent directory

- The first column indicates file type and permissions

“drwxr-xr-x” → “**directory**, owner can Read/Write/eXecute, group can R/X, all other users can R/X”

FUSE Example

- “.readdir” redirected to “hello_readdir”

```
static int hello_readdir(const char *path, void *buf, fuse_fill_dir_t filler,
                        off_t offset, struct fuse_file_info *fi)
{
    (void) offset;
    (void) fi;

    if (strcmp(path, "/") != 0)
        return -ENOENT;

    filler(buf, ".", NULL, 0);
    filler(buf, "..", NULL, 0);
    filler(buf, hello_path + 1, NULL, 0);

    return 0;
}
```

FUSE Example

- “.readdir” redirected to “hello_readdir”

```
static int hello_readdir(const char *path, void *buf, fuse_fill_dir_t filler,
                        off_t offset, struct fuse_file_info *fi)
{
    (void) offset;
    (void) fi;

    if (strcmp(path, "/") != 0)
        return -ENOENT;

    filler(buf, ".", NULL, 0);
    filler(buf, "..", NULL, 0);
    filler(buf, hello_path + 1, NULL, 0);

    return 0;
}
```

Parse requested path. This example only checks if path is root

FUSE Example

- “.readdir” redirected to “hello_readdir”

```
static int hello_readdir(const char *path, void *buf, fuse_fill_dir_t filler,
                        off_t offset, struct fuse_file_info *fi)
{
    (void) offset;
    (void) fi;

    if (strcmp(path, "/") != 0)
        return -ENOENT;

    filler(buf, ".", NULL, 0);
    filler(buf, "..", NULL, 0);
    filler(buf, hello_path + 1, NULL, 0);

    return 0;
}
```

Parse requested path. This example only checks if path is root

Please be careful about buffer overrun in string operations. E.g., if you use `strcmp()` / `strcpy()`, you need to make sure the string is null terminated (`\0`). Alternatively, you can simply use `strncmp()` / `strncpy()`.

FUSE Example

- “.readdir” redirected to “hello_readdir”

```
static int hello_readdir(const char *path, void *buf, fuse_fill_dir_t filler,
                        off_t offset, struct fuse_file_info *fi)
{
    (void) offset;
    (void) fi;

    if (strcmp(path, "/") != 0)
        return -ENOENT;

    filler(buf, ".", NULL, 0);
    filler(buf, "..", NULL, 0);
    filler(buf, hello_path + 1, NULL, 0);

    return 0;
}
```

drwxr-xr-x	2	root	root	0	Dec	31	1969	.
drwxr-xr-x	5	yud42	UNKNOWN1	4096	Apr	6	15:43	..
-r--r--r--	1	root	root	13	Dec	31	1969	hello

FUSE Example

- “.getattr” redirected to “hello_getattr”

```
static int hello_getattr(const char *path, struct stat *stbuf)
{
    int res = 0;

    memset(stbuf, 0, sizeof(struct stat));
    if (strcmp(path, "/") == 0) {
        stbuf->st_mode = S_IFDIR | 0755;
        stbuf->st_nlink = 2;
    } else if (strcmp(path, hello_path) == 0) {
        stbuf->st_mode = S_IFREG | 0444;
        stbuf->st_nlink = 1;
        stbuf->st_size = strlen(hello_str);
    } else
        res = -ENOENT;

    return res;
}
```

FUSE Example

- “.getattr” redirected to “hello_getattr”

```
static int hello_getattr(const char *path, struct stat *stbuf)
{
    int res = 0;

    memset(stbuf, 0, sizeof(struct stat));

    if (strcmp(path, "/") == 0) {
        stbuf->st_mode = S_IFDIR | 0755;
        stbuf->st_nlink = 2;
    } else if (strcmp(path, hello_path) == 0) {
        stbuf->st_mode = S_IFREG | 0444;
        stbuf->st_nlink = 1;
        stbuf->st_size = strlen(hello_str);
    } else
        res = -ENOENT;

    return res;
}
```

The Case if it is root directory
Mode: Is directory | Permission code
Links: s and “.”

FUSE Example

- “.getattr” redirected to “hello_getattr”

```
static int hello_getattr(const char *path, struct stat *stbuf)
{
    int res = 0;

    memset(stbuf, 0, sizeof(struct stat));
    if (strcmp(path, "/") == 0) {
        stbuf->st_mode = S_IFDIR | 0755;
        stbuf->st_nlink = 2;
    } else if (strcmp(path, hello_path) == 0) {
        stbuf->st_mode = S_IFREG | 0444;
        stbuf->st_nlink = 1;
        stbuf->st_size = strlen(hello_str);
    } else
        res = -ENOENT;

    return res;
}
```

The Case if it is regular file
Mode: Is File | Permission code
Links: the file name
Size of the file

FUSE Example

- Permission code (4 octal digits)

0 (0) 7 5 5

- 0 -> 000 (set-user-id = 0 | set-group-id = 0 | sticky bit = 0)
 - 7 -> 111 Owner permission:(readable | writable | executable)
 - 5 -> 101 Group permission:(readable | unwritable | executable)
 - 5 -> 101 Others permission:(readable | unwritable | executable)
-
- 0755 → rwx r-x r-x

FUSE Example cont.

- Run your file system in front

./hello testmount

- You can check which system call is been invoked
- Launch another session to check your file system
- Can simply unmount by exiting

- Remember to **unmount** the file system when you are done or need to make changes

fusermount -u testmount

Solve “fusemount permission denied” error

1. Add /u/fuse/bin into PATH env by typing the following 2 commands:

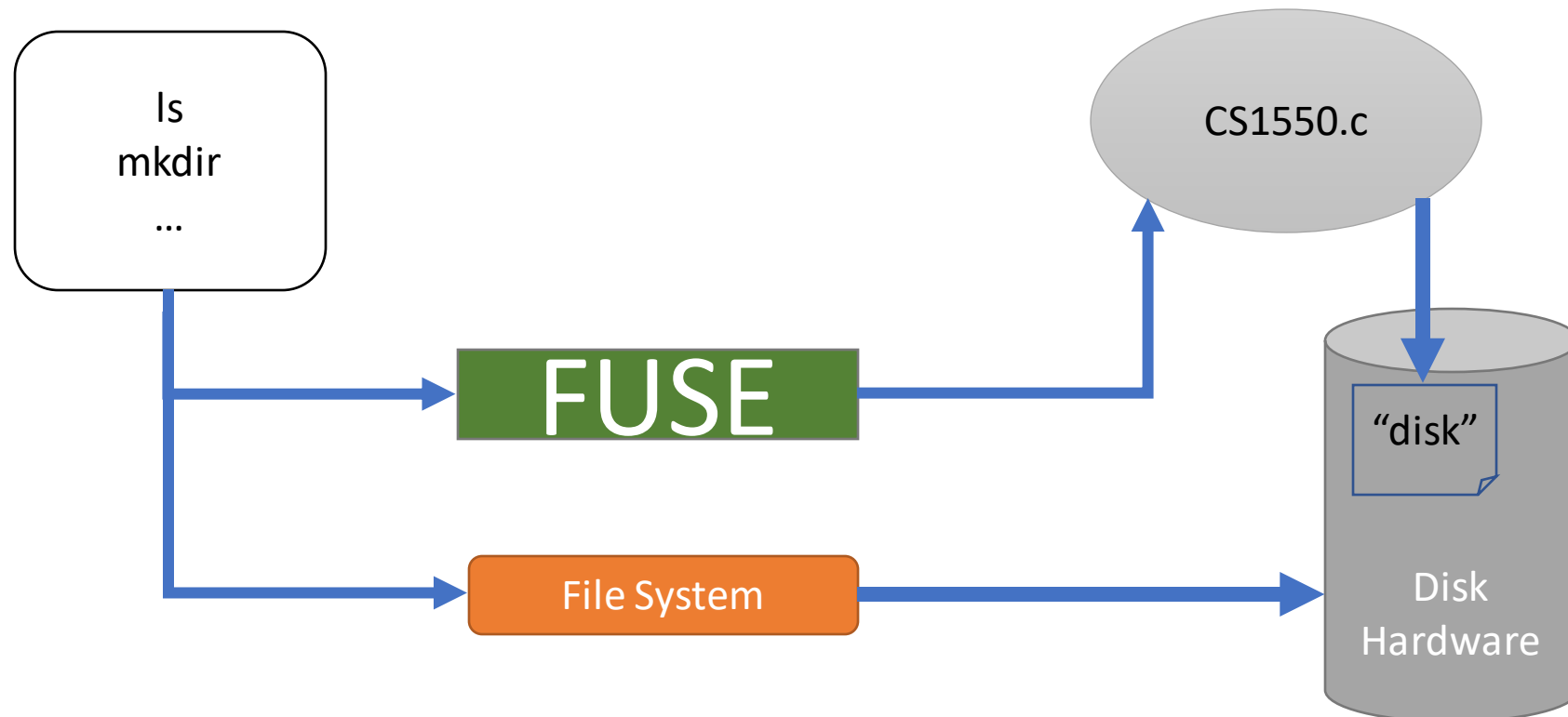
```
echo "export PATH=\"$PATH:/u/fuse/bin\"" >> ~/.bash_profile  
source ~/.bash_profile
```

2. Alternatively, add an alias:

```
echo "alias fusemount='/u/fuse/bin/fusemount'" >> ~/.bash_profile  
source ~/.bash_profile
```

Create a file as “disk”

- We need to create a file as the “disk” for our file system. All metadata and file data in our file system will be stored in this “disk”.
 - Create a 5MB file: `dd bs=1K count=5K if=/dev/zero of=.disk`

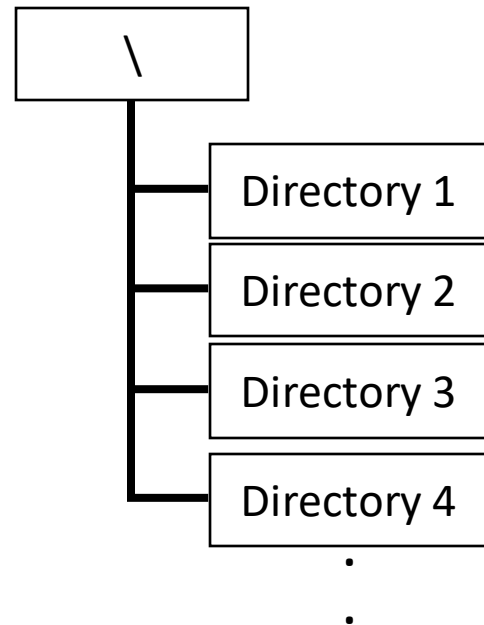


File System

- Two-level directory system
 - The root directory “\” will only contain other subdirectories, and no regular files.

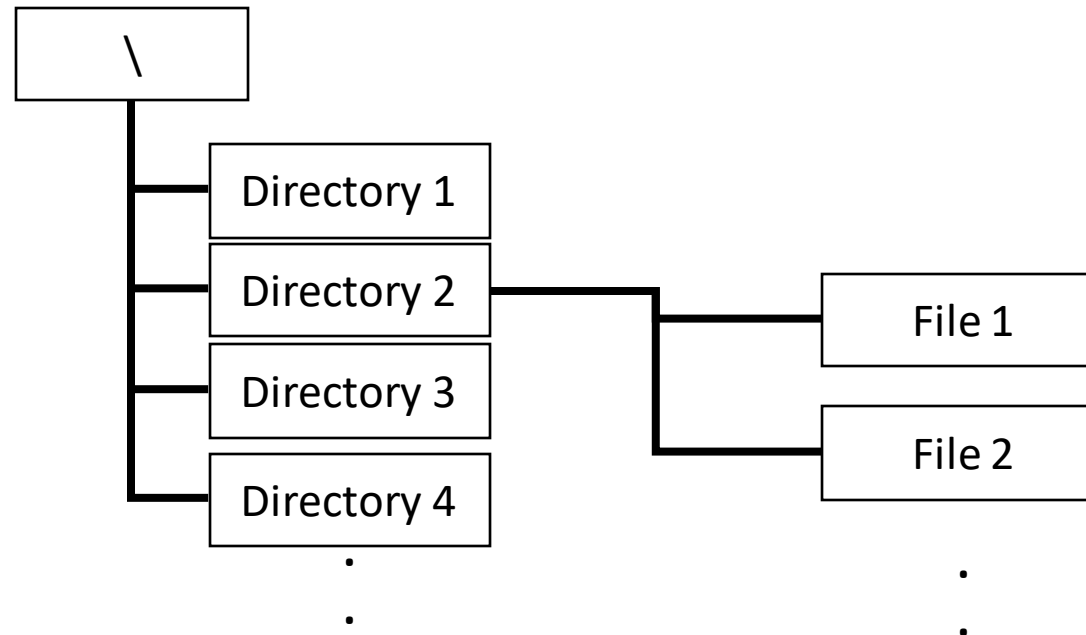
File System

- Two-level directory system
 - The root directory “\” **will only contain other subdirectories**, and no regular files.



File System

- Two-level directory system
 - The root directory “\” will only contain other subdirectories, and no regular files.
 - The subdirectories **will only contain regular files**, and no subdirectories of their own.



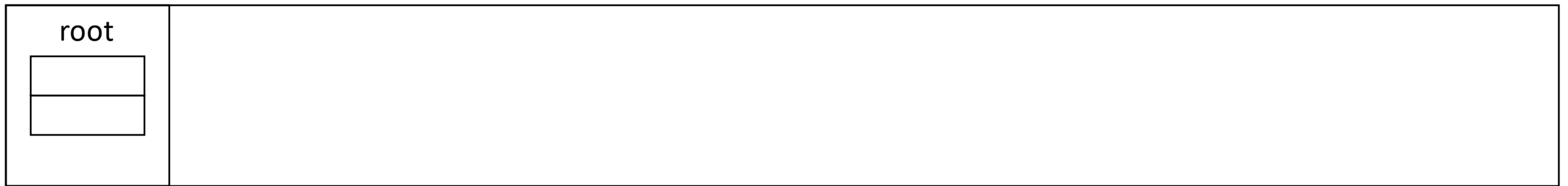
File System

- Two-level directory system
 - The root directory “\” will only contain other subdirectories, and no regular files.
 - The subdirectories will only contain regular files, and no subdirectories of their own.
 - **This project does not require you to support reading/writing files.**

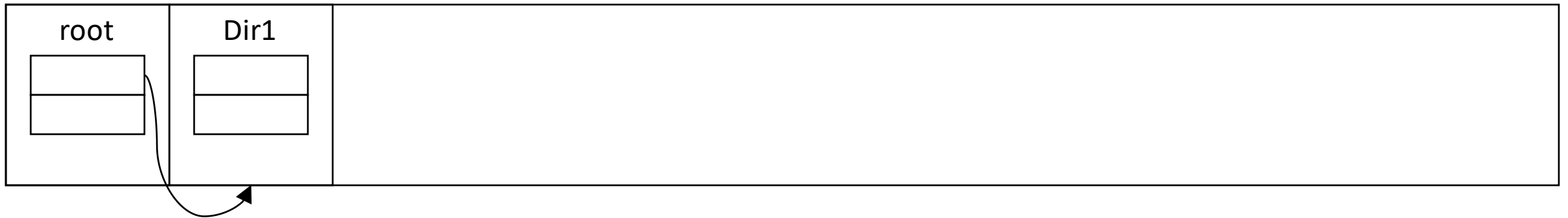
“disk” layout

Block-0	Block-1	...	Block-N
512B	512B		512B

“disk” layout



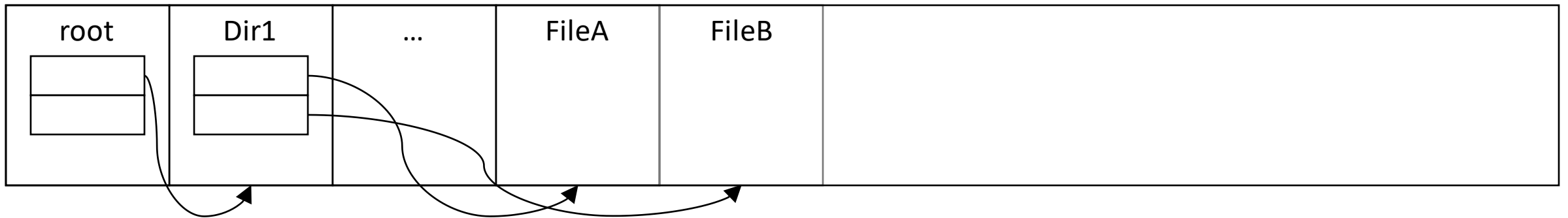
“disk” layout



“disk” layout



“disk” layout



“disk” layout

