

CSE 30341: Home Work Project 5

Assigned: Apr 18, 2008

Due: Apr 30, 2008 10:40AM

Late submissions will not be accepted

Group effort

In this project, you will read the source code for an actual file system and report on the functionality implemented by the various source functions. You do not need to understand every single line of code – make your best guess on what that code might be doing (you can use the comments as a guide). For example, one can make a reasonable guess for a line such as:

```
MSDOS_I(inode)->i_attrs = attr & ATTR_UNUSED;  
mark_inode_dirty(inode);
```

up:

```
mutex_unlock(&inode->i_mutex);
```

A high level description should suffice. I strongly suggest that you download the Linux kernel for your analysis (the source code of Mac OSX and Windows are also available on the expsys machines). It is usually easier to understand Linux kernels because of their extensive documentation (and Google). The suggested course text-book on Linux Kernel Development is an useful tool for this project. (The book should be available in the Engineering reserves. Please let me know if you encounter any problems). Within the Linux kernel, the directory 'fs' contains the source code for various file systems. For example, the source code for UNIX file system is available in fs/ufs/. Similarly, the source code for DOS file system is available in fs/fat/. For example, if you are analyzing the DOS file system, here is what I expect:

The structure fat_file_operations identifies the C function that implement typical file system functionality (must be familiar to you after HWP #4).

The function fat_getattr() in file.c sets generic file attributes as well as file attributes specific to MSDOS such a file system block size.

[and so on for each function]

Your grades will depend on the thoroughness of your observation. Please note that this project was designed to be familiar with some aspect of an actual file system; this was not designed to be tricky or time consuming. Do not delve too deeply if you do not understand a particular function. If you are not familiar with C, you can form a super group with others who are more familiar with C.