Lab One

Kevin Mirsky

kevin.mirsky1@marist.edu

September 11, 2018

1 Problem One

What are the advantages and disadvantages of using the same system call interface for manipulating both files and devices?

This can simplify the way your interface is designed, as you do not need to create a whole separate structure for the other. In addition, commands can be reused and adding features would be hopefully easier. You will not need to maintain two versions of a properties command for example. This hopefully makes your interface easier to use by keeping things more consistent between files and devices.

The downside is treating them as the same thing may be difficult as they are not 100% alike. What should happen when you try to âĂIJopenâĂİ a printer for example? On top of this, it might be challenging trying to fit device functionality into commands meant for operating on files.

2 Problem Two

Would it be possible for the user to develop a new command interpreter using the system call interface provided by the operating system? How?

Yes, it would. An API can make the required calls available to the user, who could write a different shell that calls on that API. The API would perform the system related actions when called upon. Also, depending on the way the current shell is written, they may not even need to do that. If current shell executes programs to perform each action (instead of running the tasks internally), the new shell could call upon the same programs and have all the existing functionality in a new shell.