

Lab One

Aidan Carr

September 2, 2024

1 WHAT ARE THE ADVANTAGES AND DISADVANTAGES OF USING THE SAME SYSTEM CALL INTERFACE FOR MANIPULATING BOTH FILES AND DEVICES?

1.1 USER EXPERIENCE

In terms of user experience, using the same system call interface for manipulating both files and devices can be confusing. When searching through and deleting files, it is not necessary to view devices on the system. Accidents can happen (like deleting a keyboard), and the extra devices or files visible may slow down a user who only needs to access one of these categories.

However, with an experienced user in an operating system like TSOS, using the same system call interface for this manipulation can be easier in terms of time and complexity. Having all similar operations in one place can be beneficial when performing similar manipulation functions.

1.2 SOFTWARE COMPLEXITY

Manipulating files and devices in the same system call interface may lead to complex portions of code. For example, renaming a filename may have different effects than renaming a device. Other more in-depth edits may require two different functions for this seemingly similar manipulation.

Creating another interface for devices is going to be more work for the programmer. However, the separation of these two different categories may help separate code that should not be together. Duplicating a file and duplicating a device may entail different processes and should not be grouped together because of a similar function name.

2 WOULD IT BE POSSIBLE FOR THE USER TO DEVELOP A NEW COMMAND INTERPRETER USING THE SYSTEM CALL INTERFACE PROVIDED BY THE OPERATING SYSTEM? HOW?

The user is able to develop a new command interpreter using the system call interface. The operating system will be able to run programs written in low-level Assembly operations. These small operations are the basis to every computer—storing data in memory, adding, looping, retrieving data from memory, and outputting information onto the system—and therefore can be used to build out something complex like a new command interpreter. A user can therefore create a program containing pseudo-commands stored in memory that can be recalled with user input.