Help Topic: Renaming and Copying Files

Spring 2020

Michael Benjamin, mikerb@mit.edu
Department of Mechanical Engineering
MIT, Cambridge MA 02139

Moving/Renaming and Copying Files on the Command Line

Moving (renaming) and copying files and directories are done on the command line using the `mv` (move) command and the `cp` (copy) command.

Moving and Renaming a file with the `mv` command

The jobs of moving and renaming files are both handled by the `mv` command. In the simplest case for example, `file1` can be renamed to file `file2` like this:

```
$ ls
file1
$ mv file1 file2
$ ls
file2
```

The `mv` command may also be used to move one or many files into a directory:

```
$ ls -F
file1   file2   file3   file4   file5   project/
$ mv file* project
$ ls -F
project/
$ ls project
file1   file2   file3   file4   file5
```

If you try to move a file into a directory that doesn't exist you will get an error.

```
$ ls -F
file1   project/
$ mv file1 project23/
mv: rename file1 to project23/: No such file or directory
```

By default the `mv` command will let you rename one file to another, even if the target file already exists. Most people find this a bit unsettling to allow this without prompting for confirmation of intention. If the `mv` command is used with the `-i` option, the user will indeed be prompted for confirmation if the target file already exists. For this reason most people employ an alias for the `mv`

command that makes the interactive mode the default:

```
$ alias mv 'mv -i'       (if using tcsh)
$ alias mv='mv -i'       (if using bash)
```

See the help topics on aliases and shell configuration to do this. Once this is done, you can always use the `-f` (force) command line option to force the renaming of file to a target, even if the target already exists, without prompting the user for confirmation.

Copying files and directories with the `cp` command

Files and whole directories can be copied with the `cp` command. The simplest case of copying one file to another:

```
$ ls
file1
$ cp file1 file2
$ ls
file1 file2
```

Copying a directory with `cp` requires a couple additional arguments on the command line:

```
$ ls -F
project/
$ cp -rp project project_backup
$ ls
project/ project_backup/
```

The first argument `-r` stands for *recursive* and indicates that all subcomponents of the folder being copied should also be copied, following the tree structure recursively in each subfolder's subfolder and so on. The second argument `-p` is not required. It ensures that certain file attributes are preserved when copying. This includes the read-write permissions, and date of last modification. When copying a directory it is a good habit to always use this option.

A note of caution. If you are copying a file to a file that already exists, the `cp` does not, by default, prompt you for a confirmation of your intentions. I highly recommend you change this default behavior. See the following additional help topic:

http://oceanai.mit.edu/ivpman/help/cmdline_interactive_rm_mv_cp