Scientific Computing

Feb 5, 2025

Announcements

- HW 2 due Mon. Feb. 17
- Mon, Feb 17, no in-person lecture and no office hours
- Wed, March 5, midterm exam, in person portion (in class)
- Fri, March 7, no class, extra office hours for take-home portion (time TBD)

Today

- Unix command line
- The coding process

Office Hours:

Mon + Fri

9:30am - 10:30am

Cudahy 307

Topic 3- The Unix Command Line

Unix was an OS framework developed in the 70s that is a precursor to the OSes of today (everything except Windows).

Mac and Linux have terminals where you can still issue Unix commands, and "Git for Windows," which we installed is a Unix terminal emulator.

The VS Code terminal is also a Unix emulator.

Today we'll cover just the very basic commands to navigate and manipulate the file/folder system.

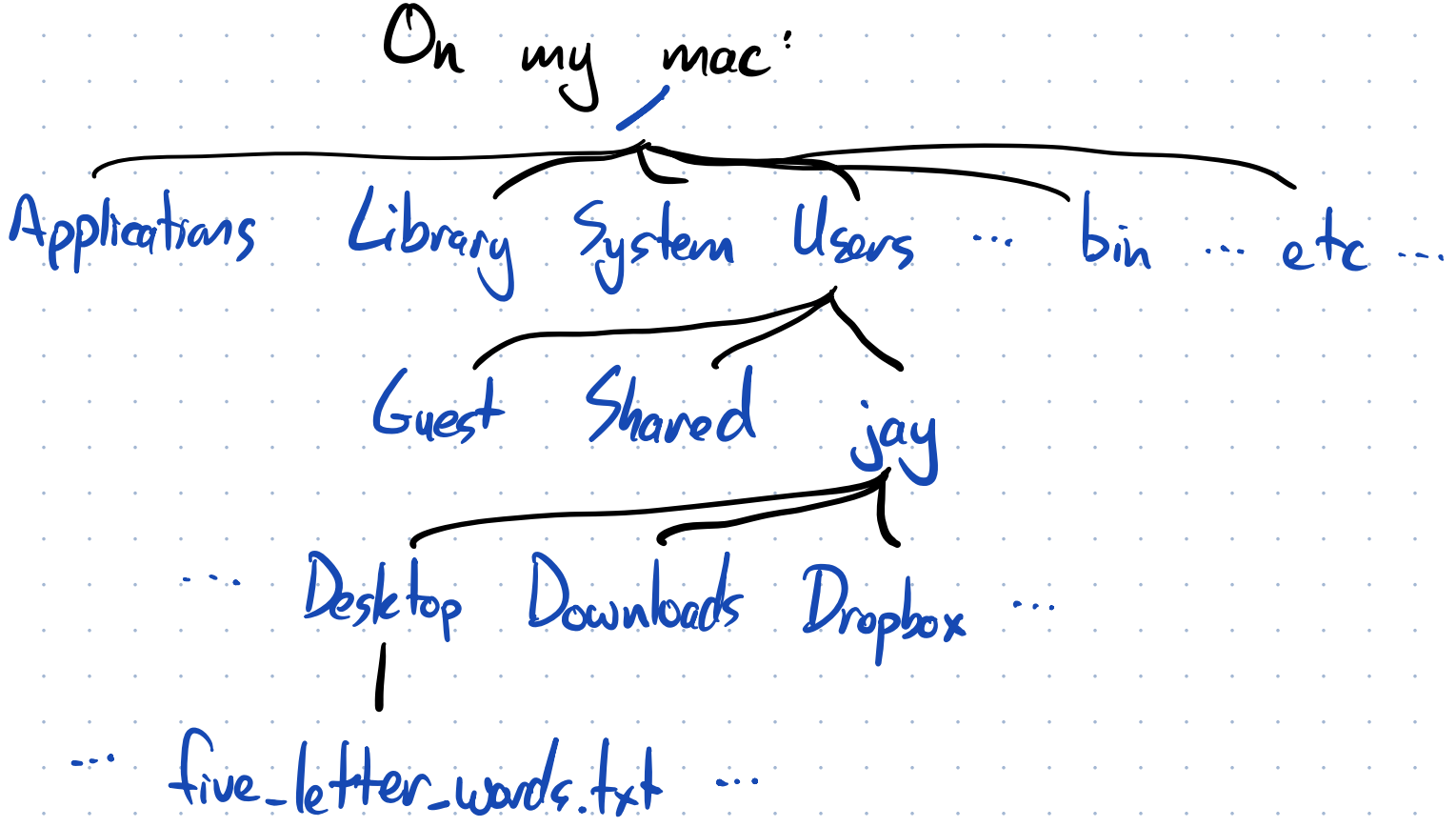
Why is this useful:

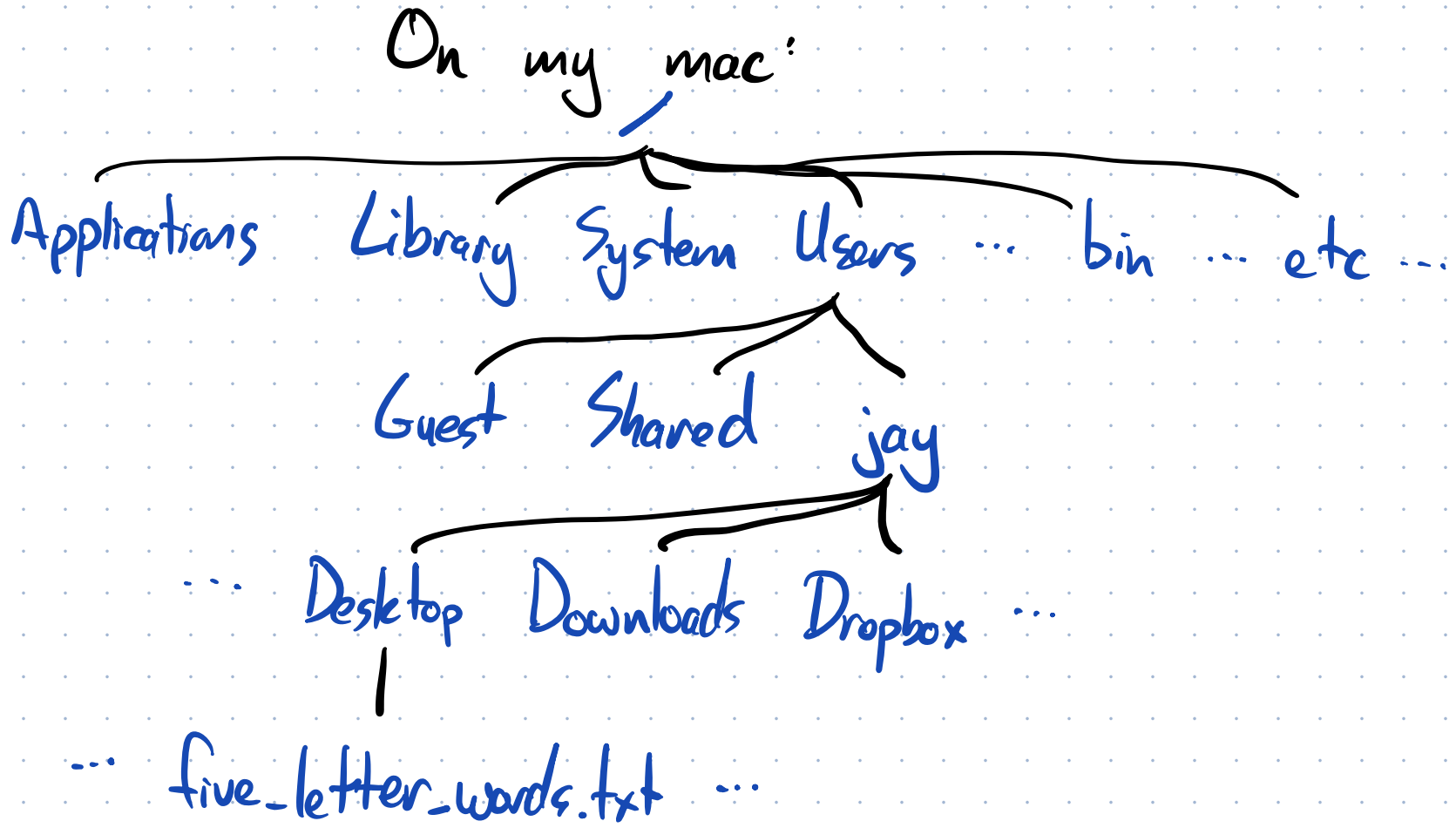
- On your own machine, a lot of it can be done with the GUI, but sometimes this is more efficient (e.g., view the first 10 lines of a 250mb text file).
- Especially though: any work you do on a server you connect to via SSH (e.g., my research server Ada).

(Software Carpentry website has great resources.)

File system:

Files in a computer are stored in a hierarchy. The very top is called "/" in unix-like systems, and usually "C:\" in Windows.





So, every file has a full address:

/Users/jay/Desktop/five_letter_words.txt

Demo: Open your terminal or Git for Windows or VS Code

(1) pwd - "present working directory", or where in the file system you are.

(2) ls - "list", display the files in the current folder

Most commands have extra arguments you can pass to change the behavior (sometimes tons of them). "flags"

Ex: `ls -l` - list the files in the current folder with extra information

To see the full "manual page" for a command, do `man command`, e.g., `man ls`. Press "q" to exit.

You can also tell ls, and many other commands, to only act on some files, using "*" as a symbol that means "anything":

```
ls -l *.txt
ls -l p*
```

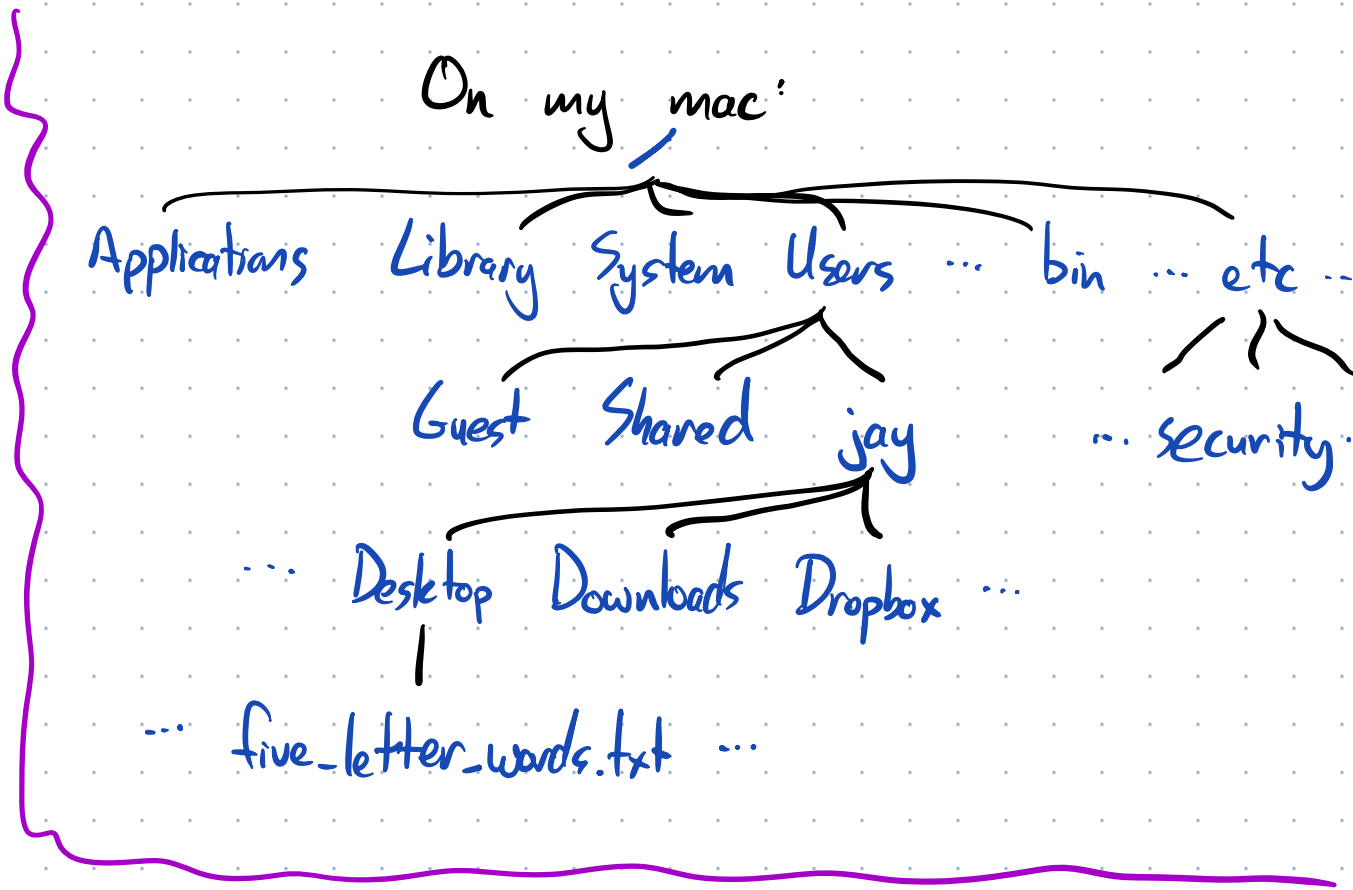
(3) cd - "change directory", move to a different place in the file structure.

cd [directory]

If you directory starts with "/", you are specifying an absolute path, exactly where a folder is.

If it doesn't start with "/", you are specifying where it is relative to your current location.

```
> cd /etc/security
> pwd
    /etc/security
> cd /Users/jay
> pwd
    /Users/jay
> cd Dropbox
> pwd
    /Users/jay/Dropbox
```

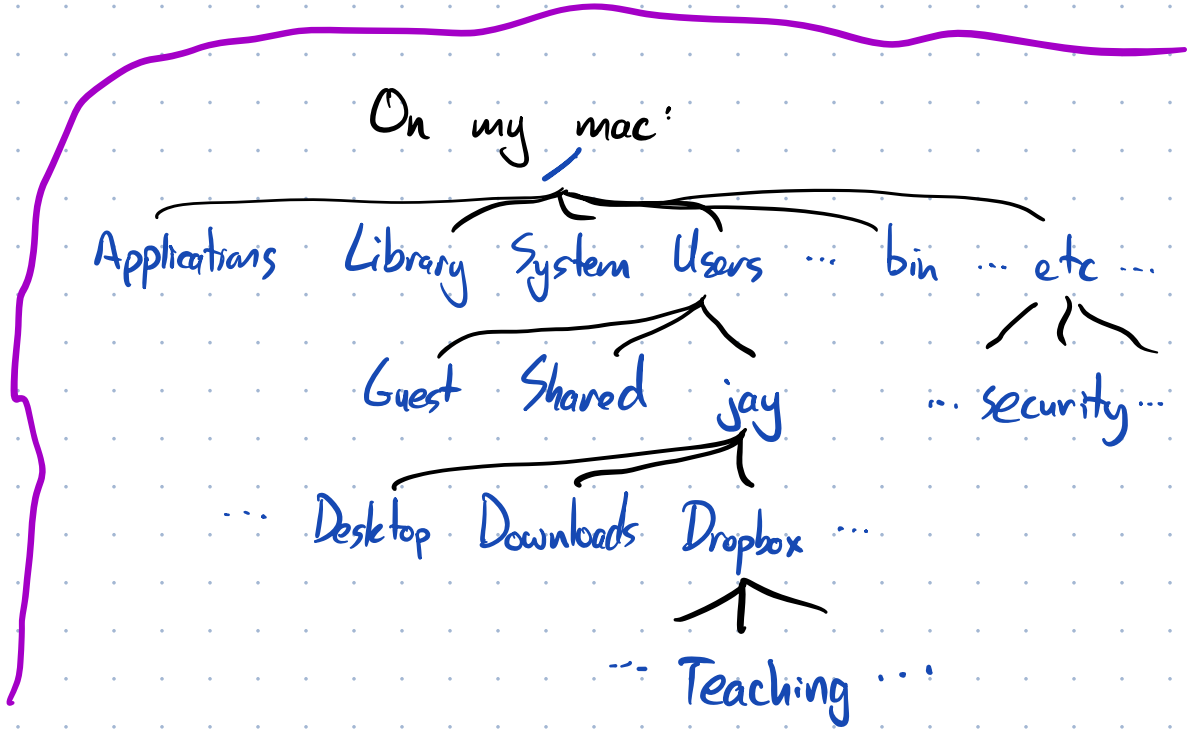


Shortcuts: "." - current folder (not helpful here)

".." - up one level

"~" - user's home folder

```
> cd ~/Dropbox/Teaching
> pwd
    /Users/jay/Dropbox/Teaching
> cd ../../Desktop
> pwd
  /Users/jay/Desktop
```



You can use the "tab" key to complete a command or filename if it's unique, or press it twice to list possibilities.

(4) mkdir [name] - "make directory" (folder)

(5) mv [current location] [source location]
- move/rename file or folder

(6) cp [source file] [destination file]
- copy a file or folder
↳ requires flags! "-R"

(7) rm [file or folder]

- "remove"/delete a file or folder
  - requires "-r" flag

⚠ Warning! This is dangerous!
"rm -rf /" will just delete all your system files. They don't go in a trash/recycle bin and can't be recovered. ⚠