Today

- Pipes
- File System & associated commands

"Software Carpentry"

1

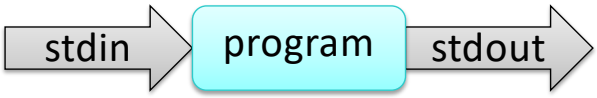
Review: Unix

- What are some components of the UNIX philosophy?
  - How do they align with software development principles?
- How do we find out more information about Unix commands?
- Name some Unix commands and what they do
- How can I execute a command that I executed earlier?
- What are some tricks we can use to make using the command-line easier/more productive?

2

Pipes

- Typical program



- Pipes: The input of one program is the output of the other

- Syntax: `programA | programB`

"The Treachery of Images"
By French surrealist painter,
René Magritte

Swarthmore CS dept t-shirt

BASIC UNIX TOOLS: FILE/DIRECTORY MANAGEMENT

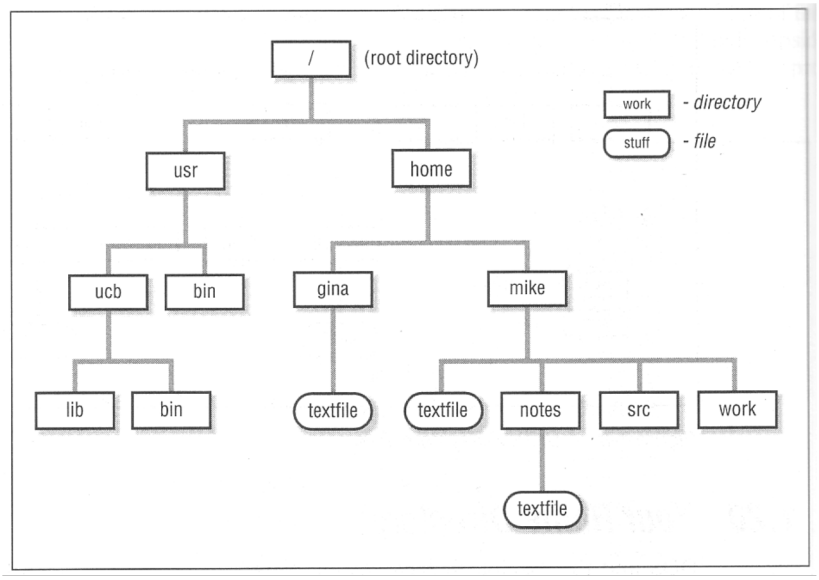
5

Directory Management Review

- How is Unix's directory structure organized?

6

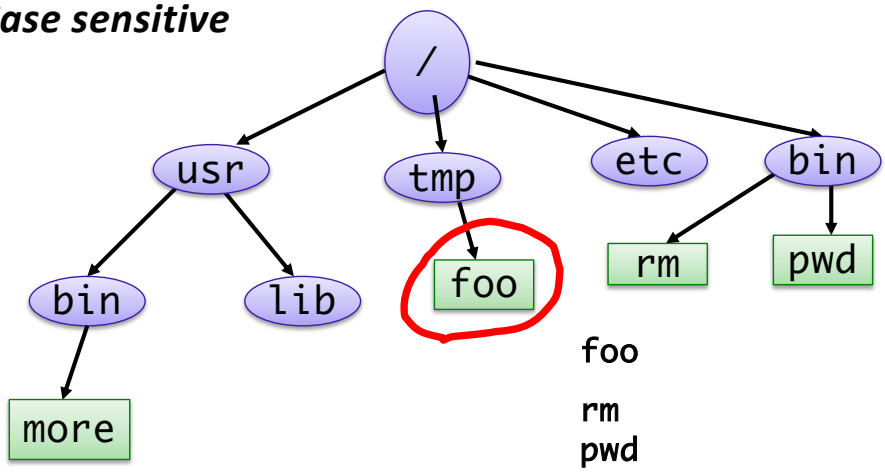
The UNIX File Hierarchy



Definition: Filename

A sequence of characters other than slash

Case sensitive



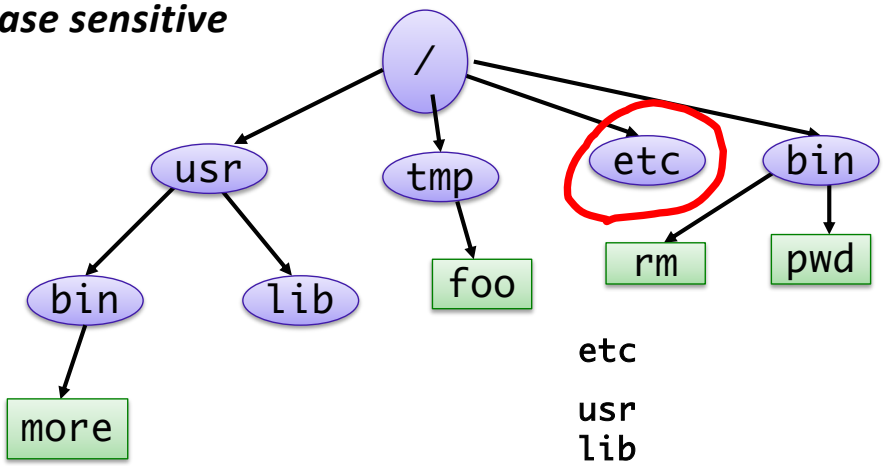

```
foo

rm
pwd
more
```

Definition: Directory

Holds a set of files or other directories

Case sensitive



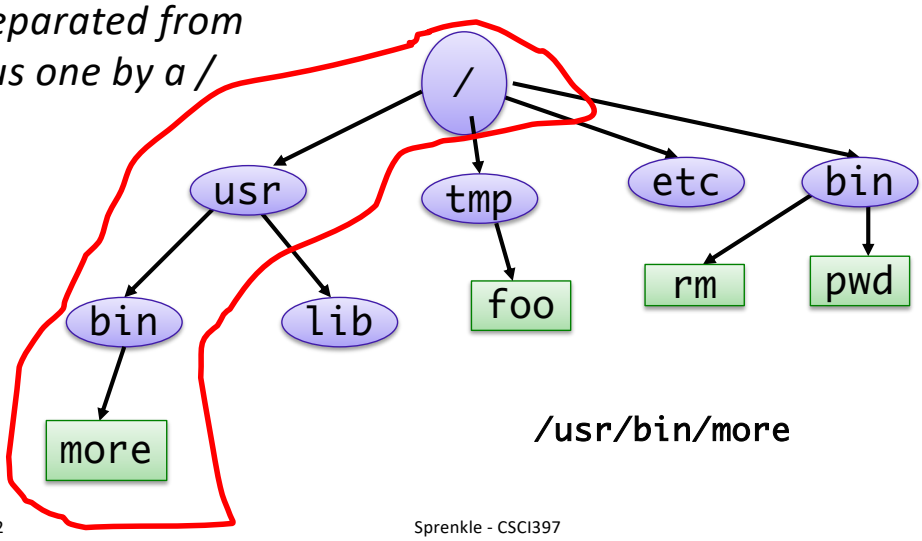

```
etc

usr
lib
bin
```


9

Definition: Pathname

A sequence of directory names followed by a simple filename, each separated from previous one by a /



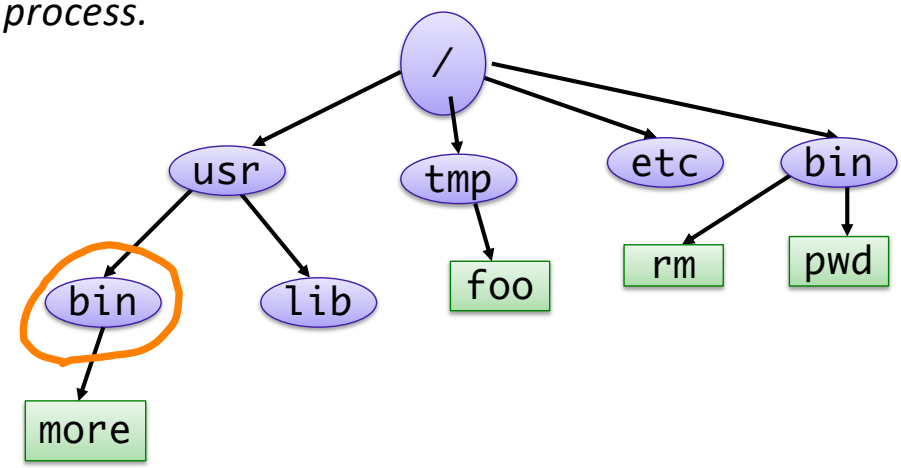

```
/usr/bin/more
```


10

Definition: Working Directory

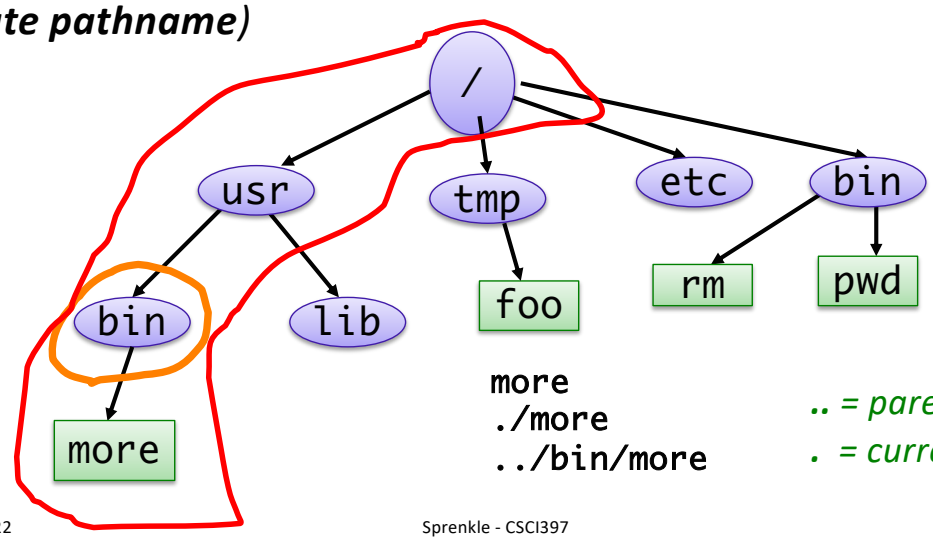
Directory the process is currently in.

One per process.



Definition: Relative Pathname

*A pathname **relative** to the working directory (as opposed to **absolute pathname**)*




```
more
./more
../bin/more
```

.. = parent directory

. = current directory

Files and Directories

- Files are just a sequence of bytes
  - No file types (data vs. executable)
  - Example of UNIX philosophy
- Directories are a list of files and status of the files:
  - Creation date
  - Attributes
  - etc.

Directory Management Review

- How do you see a directory's contents?
  - How can you find out more information about the contents?
  - How can you list the content in time order?
- How do you go into a directory?
  - Home directory?
  - Parent directory?
- How can you help avoid a lot of typing and errors when you're trying to go into a directory?

Tilde Expansion

- Each user has a *home* directory
- Most shells support ~ operator:
  - ~ expands to my home directory
    - `~/myfile → /home/sprenkle@ad.wlu.edu/myfile`
  - **~user** expands to user's home directory
    - `~unixtool/file2 → /home/unixtool@ad.wlu.edu/file2`
- Useful because home directory locations vary by machine
- HOME environment variable

What is your home directory?

Directory Management Review

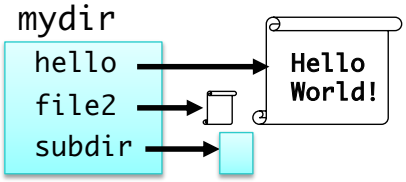
- How do you know what directory you're in?
- How do you make a new directory?
  - How do you make a series of directories, for example `~/cs397/practice/tmp`, in one command?
  - What if `~/cs397/practice/` doesn't exist?
- How do you delete an *empty* directory?

File Management Review

- How do you copy a file?
  - A directory and its contents?
- How do you move/rename a file?
- How do you delete a file?
- How do you delete a directory and all of its content?

Links

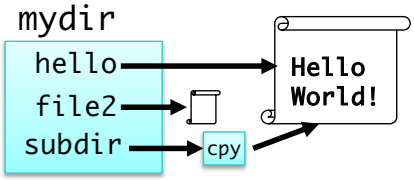
- Directories are lists of files and directories
- Each directory entry *links* to a file on the disk



- Links (hard links): Two different directory entries can link to the same file
  - Essentially gives same file another name
  - In same directory or across different directories
  - Cannot make a hard link to a directory

Links

- `ln` command: `ln <target> <dest>`

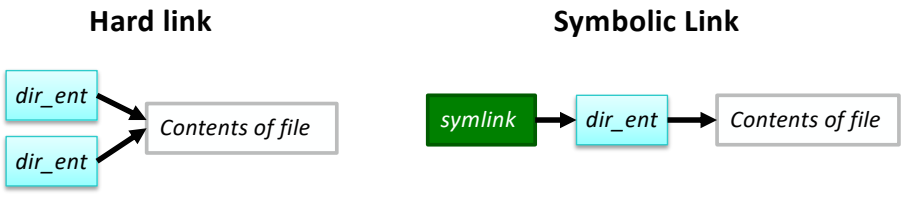



```
ln hello subdir/cpy
```

- Two different directory entries can link to the same file
  - In same directory or across different directories
- FYI: `mv` ing a file does not actually move any data around
  - Creates link in new location
  - Deletes link in old location

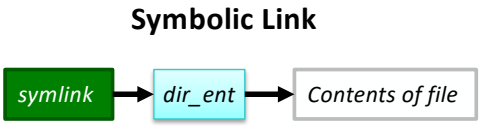
Symbolic links

- ***Symbolic*** links are different than regular links (often called **hard links**)
  - Created with `ln -s`
- Can be thought of as a directory entry that points to the ***name*** of another file



Symbolic links

- ***Symbolic*** links are different than regular links (often called **hard links**)
  - Created with `ln –s target dest`
- Can be thought of as a directory entry that points to the ***name*** of another file
- Does not change link count for file
  - When original deleted, symbolic link remains
- They exist because
  - Hard links don't work across file systems
  - Hard links only work for regular files, not directories



Create a symbolic link to the course's directory
Check if that worked by cding to the link

Jan 14, 2022

21