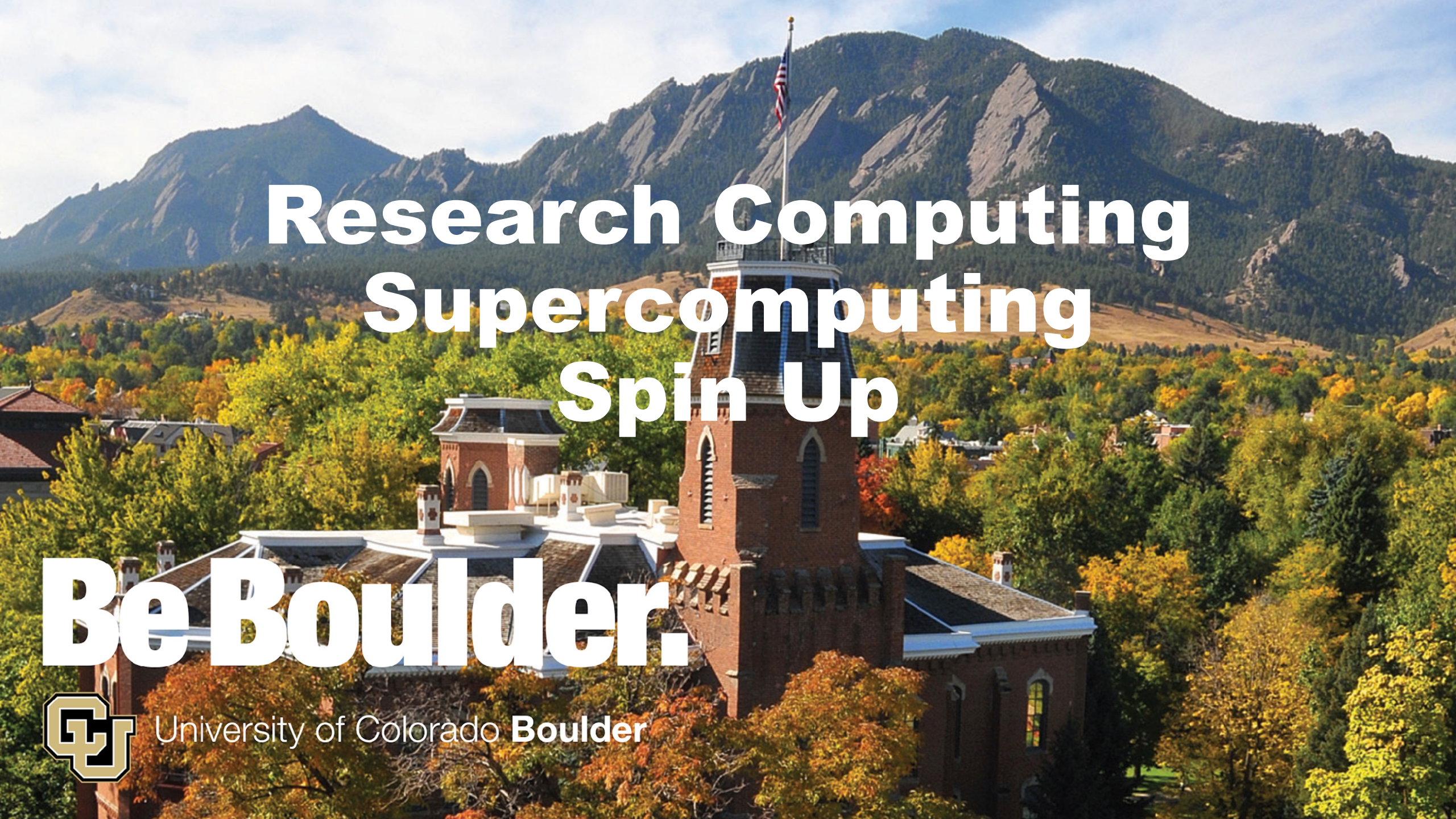

Research Computing
Supercomputing
Spin Up
Be Boulder.
University of Colorado Boulder

Supercomputing Spinup Part 1: Intro to Linux

Andy Monaghan
rc-help@colorado.edu

www.colorado.edu/rc

Slides available at:

https://github.com/ResearchComputing/Supercomputing_Spinup_Spring_2022

Funded in part by National Science Foundation under grant OAC-1925766

Outline for this presentation

- Intro to Research Computing
- Opening a Terminal
- Basic Linux commands
- File editing
- Linux filesystem
- Environment variables
- Other Linux topics (modes, wildcards)

What is Research Computing?

- Provide services for researchers that include:
  - Large scale computing
  - Data storage
  - High speed data transfer
  - Data management support
  - Consulting
  - Training

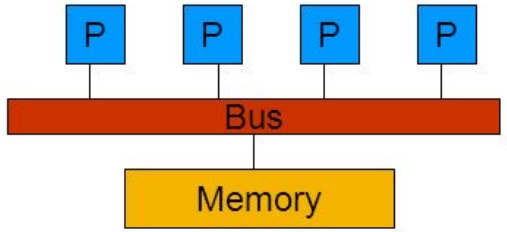
- We are likely best known for:
  - Summit and Alpine Supercomputers (~12,000 cores each)
  - Blanca ”condo” cluster (~4,000 cores)
  - PetaLibrary storage

Why would I Use CURC resources?

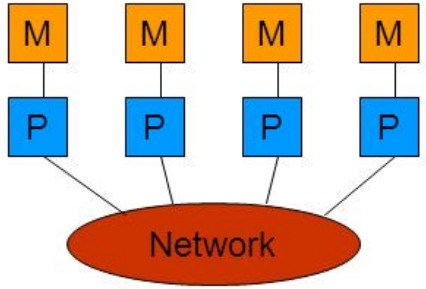
Solving large problems that require more:

- Memory than you have on your personal computer
- Cores/nodes/power than you have on your personal computer

CURC resources are set up for both shared memory (single node) and distributed memory (multi-node) parallelization.



- **Shared memory**



- **Distributed memory**

Source: https://images.slideplayer.com/25/7599921/slides/slide_4.jpg

Opening a Terminal

- Mac:
  - Go to: Applications->Utilities->Terminal

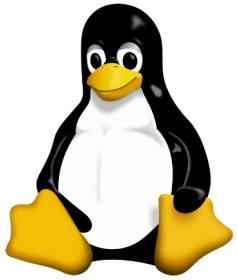
- Windows:
  - Download a terminal emulator. Lots of options, e.g.:
    - PuTTY: https://www.putty.org
    - Git BASH: https://gitforwindows.org

- For practice, you can use an online emulator:
  - https://cocalc.com/app?anonymous=terminal

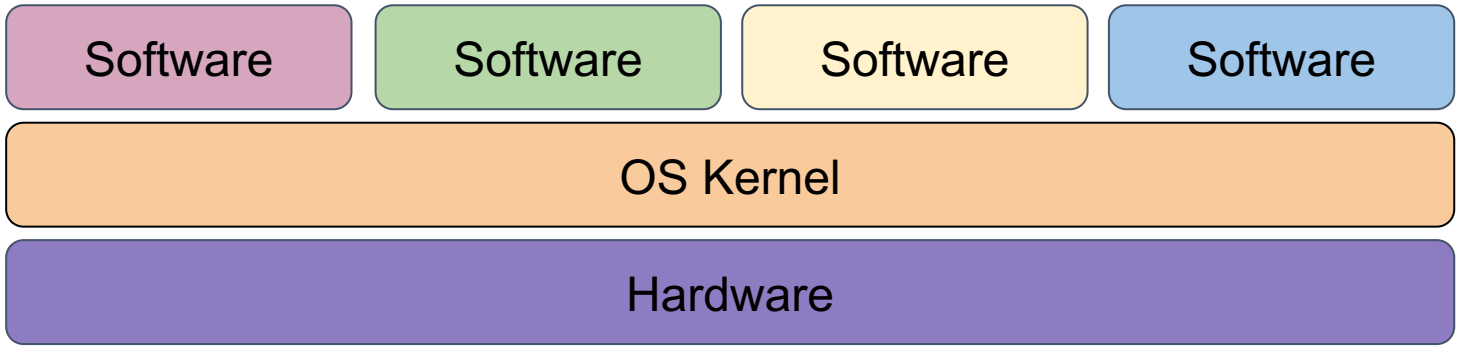
Optional: Logging In to CURC

- `ssh <identikey>@login.rc.colorado.edu`
- Enter your `identikey_password`
- Authenticate by accepting the Duo push to your smartphone
  - Can also authenticate by text message, phone call, or token
- More info here: https://curc.readthedocs.io/en/latest/access/logging-in.html

What is Linux?



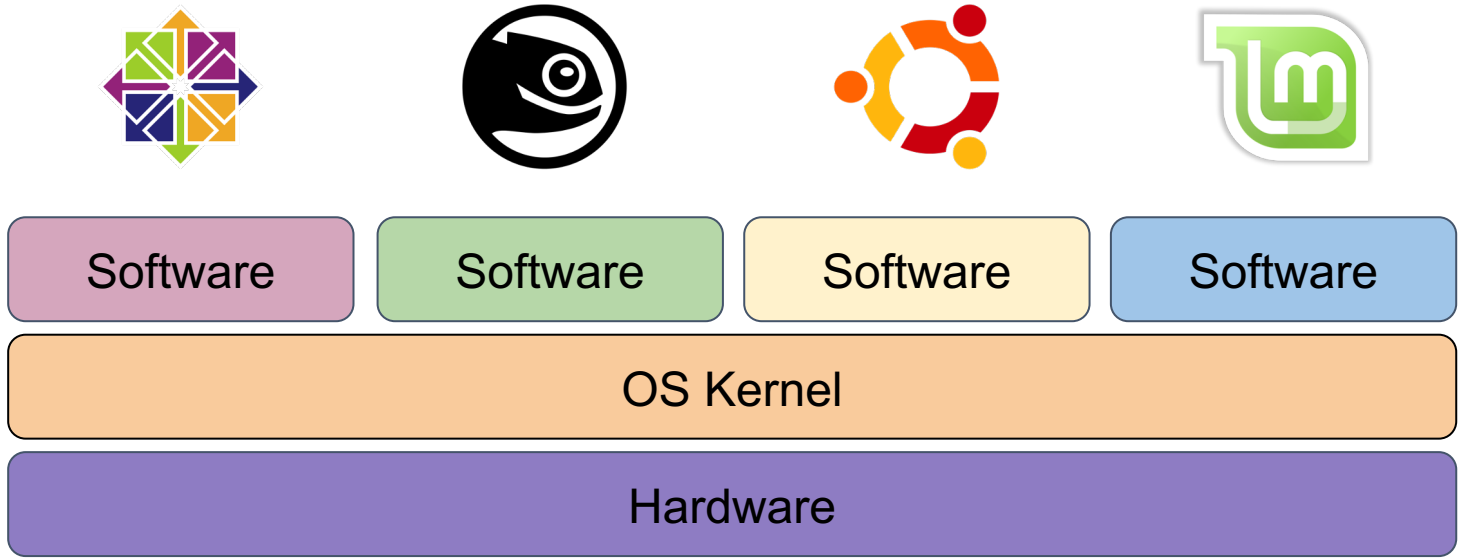
- Part of the Unix-like family of operating systems.
- Started in early ‘90s by Linus Torvalds.
- Typically refers only to the kernel with software from the GNU project and elsewhere layered on top to form a complete OS. Most is open source.



images courtesy of wikicommons

What is Linux?

- Several distributions are available; from enterprise-grade, like RHEL or SUSE, to more consumer-focused, like Ubuntu.
- Runs on everything from embedded systems to supercomputers.



images courtesy of wikicommons

Why Use Linux?

- Default operating system on virtually all HPC systems, and is the foundation for many business services globally
- Extremely flexible
- Fast and powerful
- Many potent tools for software development
- You can get started with a few basic commands and build from there

Anatomy of a Linux command

- command [flags] [target(s)]
  - ls -l myworkdir/
- Case is important!
- Help on commands is available through the ”man” command (short for manual). E.g.,
  - man ls

File and directory related commands

- **pwd** – prints full path to current directory
- **cd** – changes directory; can use full or relative path as target
- **mkdir** – creates a subdirectory in the current directory
- **rmdir** – removes an empty directory
- **rm** – removes a file (`rm -r` removes a directory and all of its contents)
- **cp** – copies a file
- **mv** – moves (or renames) a file or directory
- **ls** – lists the contents of a directory (`ls -l` gives detailed listing)

File-viewing commands

- **more** – displays a file one screen at a time
- **cat** – prints entire file to the screen
- **head** – prints the first few lines of a file
- **tail** – prints the last few lines of a file (with -f shows in real time the end of a file that may be changing)
- **diff** – shows differences between two files
- **grep** – prints lines containing a string or other regular expression (ps –ef | grep XX)
- **sort** – sorts lines in a file
- **find** – searches for files that meet specified criteria
- **wc** – count words, lines, or characters in a file

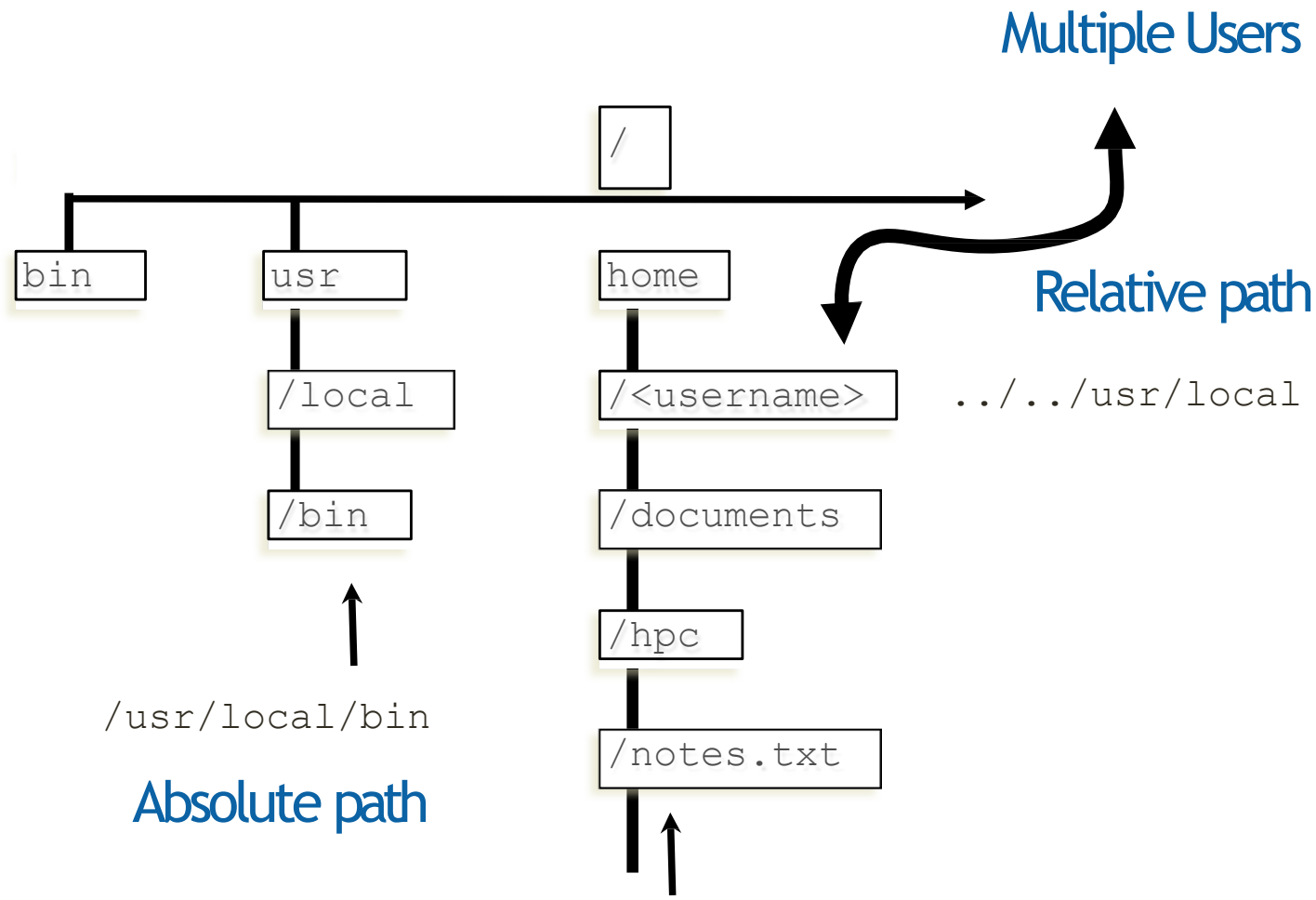
File editing with nano

- To edit a file:
  - nano myfile.txt
- From within Nano:
  - Ctrl+o save (need to confirm filename)
  - Ctrl+x exit
  - Ctrl+k cut
  - Ctrl+u paste
- Other population Linux editors: vi, emacs

The Linux Filesystem

- System of arranging files on disk
- Consists of directories (folders) that can contain files or other directories
- Levels in full paths separated by *forward* slashes, e.g.
- /home/user/scripts/analyze_data.sh
- Case-sensitive; spaces in names discouraged
- Some shorthand:
  - . (the current directory)
  - .. (the directory one level above)
  - ~ (home directory)

Filesystem


Multiple Users
/
bin
usr
home
Relative path
/local
/<username>
../../usr/local
/bin
/documents
/hpc
/usr/local/bin
/notes.txt
Absolute path
/home/<username>/documents/hpc/notes.txt

Your personal directories on CURC

- /home/<username>
  - Very small: 2GB.
  - Backed up daily.
  - Good for ‘can’t lose’ files
- /projects/<username>
  - 250 GB
  - Backed up regularly
  - Good for storing scripts, self-installed software, some data
- /scratch/<resource>/<username>
  - 10 TB
  - Good for jobs with lots of I/O –highly performant!)
  - Not backed up
  - Temporary: data deleted 90 days from creation.

Environment variables

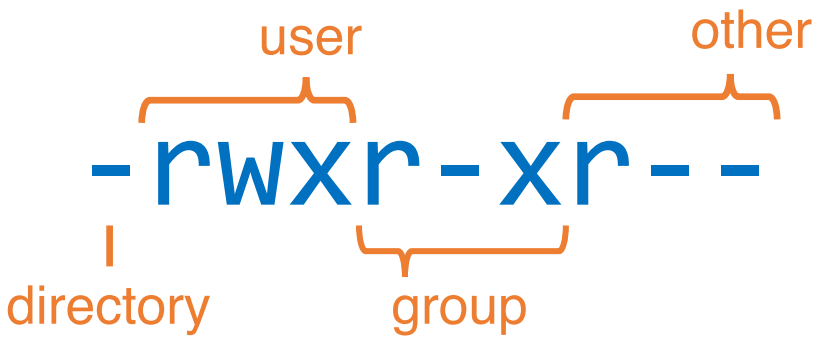
- Environment variables store important information needed by Linux users, programs, etc.

- Type ‘env’ to see your currently set environment variables

- Useful Environment variables:
  - `PATH`: directories to search for commands
  - `HOME`: home directory
  - `PWD`: current working directory
  - `USER`: username
  - `LD_LIBRARY_PATH`: directories to search for shared objects (dynamically-loaded libs)

Modes (aka permissions)

- View file/directory permissions: "ls -l"
- 3 classes of users:
  - User (u), *aka "owner"*
  - Group (g)
  - Other (o)
- 3 types of permissions:
  - Read (r)
  - Write (w)
  - Execute (x)



Modes (continued)

- `chmod` changes modes:

- To add write and execute permission for your group:

```
chmod g+wx filename
```

- To remove execute permission for others:

```
chmod o-x filename
```

Shell Wildcards and Special Characters

- `*` - matches zero or more characters
- `?` - matches a single character
- `#` - comment; rest of the line is ignored
- `\` - escape; don't interpret the next character

Questions?

Presenter: Andrew Monaghan
Email rc-help@colorado.edu

Link to course evaluation:
http://tinyurl.com/curc-survey18

CU Research Computing Documentation:
https://curc.readthedocs.io

Slides:
https://github.com/ResearchComputing/Supercomputing_Spinup_Spring_2022