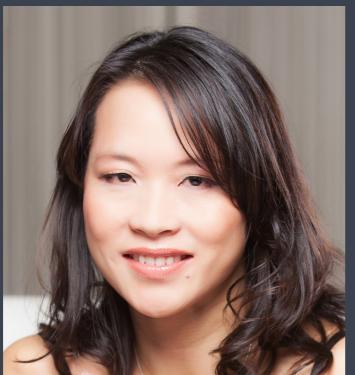


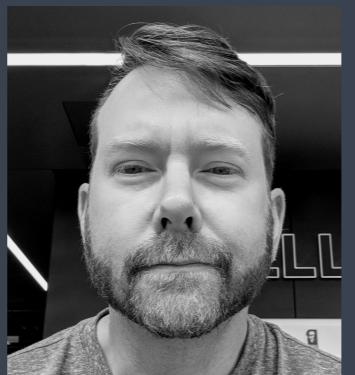
# Introduction to the command-line interface (shell)

Harvard Chan Bioinformatics Core  
in collaboration with  
HMS Research Computing

<https://tinyurl.com/hbc-shell-online>



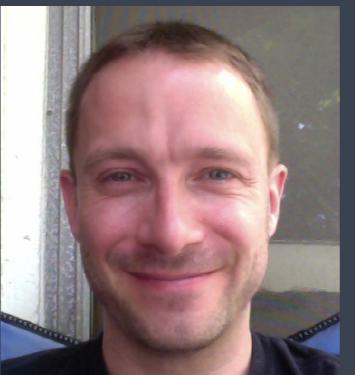
Shannan Ho Sui  
*Director*



John Hutchinson  
*Associate Director*



Victor Barrera



Rory Kirchner



Zhu Zhuo



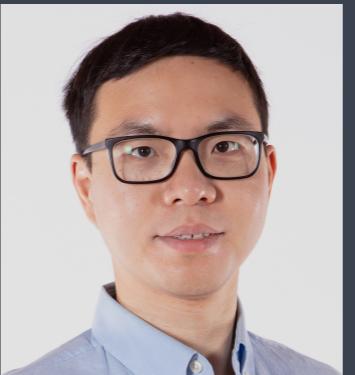
Preetida Bhetariya



Meeta Mistry



Mary Piper  
*Assoc. Training Director*



Jihe Liu



Radhika Khetani  
*Training Director*



Maria Simoneau  
*Project coordinator*



James Billingsley



Sergey Naumenko



Joon Yoon



Peter Kraft  
*Faculty Advisor*

# Consulting

- RNA-seq analysis: bulk, single cell, small RNA
- ChIP-seq and ATAC-seq analysis
- Genome-wide methylation
- WGS, resequencing, exome-seq and CNV studies
- QC & analysis of gene expression arrays
- Functional enrichment analysis
- Grant support

<http://bioinformatics.sph.harvard.edu/>



**HARVARD  
T.H. CHAN  
SCHOOL OF PUBLIC HEALTH**

NIEHS



# Training

We have divided our short workshops into 2 categories:

1. Basic Data Skills - No prior programming knowledge needed (no prerequisites)
2. Advanced Topics: Analysis of high-throughput sequencing (NGS) data - Certain “Basic” workshops required as prerequisites.

*Any participants wanting to take an advanced workshop will have to have taken the appropriate basic workshop(s) within the past 6 months.*

<http://bioinformatics.sph.harvard.edu/training/>

<https://hbctraining.github.io/main/>



# Training

We have divided our short workshops into 2 categories:

1. Basic Data Skills - No prior programming knowledge needed (no prerequisites)
2. Advanced Topics: Analysis of high-throughput sequencing (NGS) data - Certain “Basic” workshops required as prerequisites.

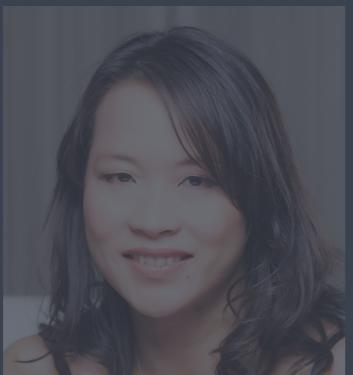
*Any participants wanting to take an advanced workshop will have to have taken the appropriate basic workshop(s) within the past 6 months.*

<http://bioinformatics.sph.harvard.edu/training/>

<https://hbctraining.github.io/main/>



# Introductions!



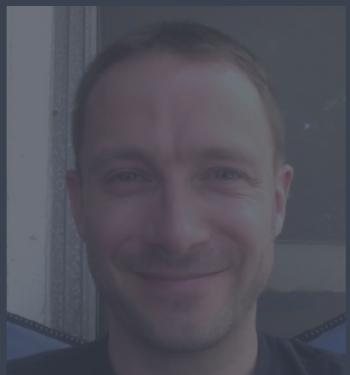
Shannan Ho Sui  
*Director*



John Hutchinson  
*Associate Director*



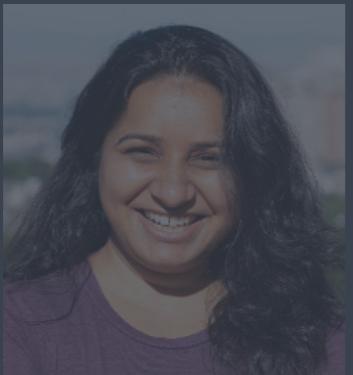
Victor Barrera



Rory Kirchner



Zhu Zhuo



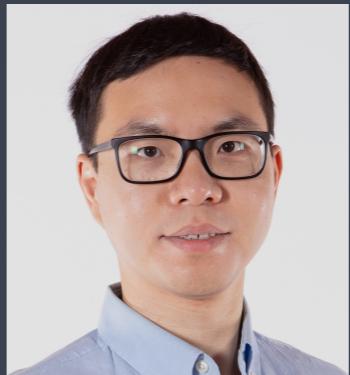
Preetida Bhetariya



Meeta Mistry



Mary Piper  
*Assoc. Training Director*



Jihe Liu



Radhika Khetani  
*Training Director*



Maria Simoneau  
*Project coordinator*



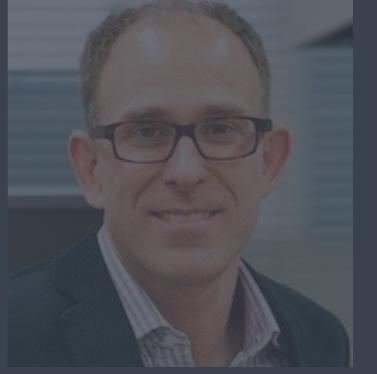
James Billingsley



Sergey Naumenko

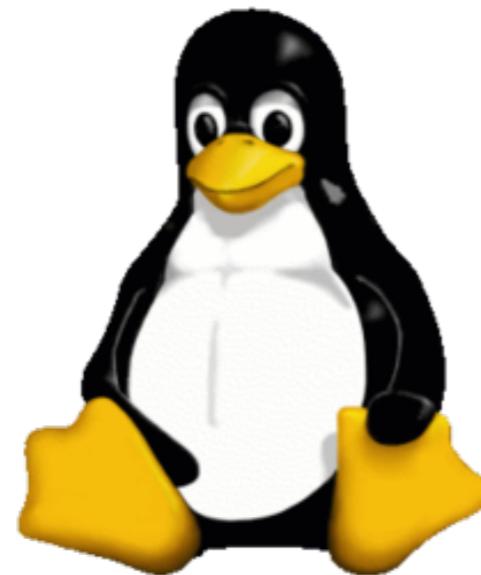


Joon Yoon



Peter Kraft  
*Faculty Advisor*

# Workshop scope



```
rkhetani — rsk27@clarinet002-072: ~ — ssh — 75x51
rsk27@clarinet002-072:~$ ll -htr unix_workshop/
total 177K
drwxrwsr-x 2 rsk27 rsk27 62 May 23 2016 reference_data
-rw-rw-r-- 1 rsk27 rsk27 377 May 23 2016 README.txt
drwxrwsr-x 2 rsk27 rsk27 78 May 23 2016 genomics_data
drwxrwsr-x 2 rsk27 rsk27 257 May 23 2016 raw_fastq
drwxrwsr-x 2 rsk27 rsk27 695 May 23 2016 other
drwxrwsr-x 6 rsk27 rsk27 972 May 24 2016 rnaseq_project
rsk27@clarinet002-072:~$
```

*“Unix is user-friendly.*

*It's just very selective about who its friends are.”*

# The Unix command-line interface

- ♦ Unix is a stable, efficient and powerful operating system
- ♦ It can easily coordinate the use and sharing of a computer's (or a system's) resources, i.e. built to allow multi-user functionality
- ♦ Can easily handle complex and repetitive tasks easily on large and small datasets

# The Unix command-line interface

- ◆ Unix is a stable, efficient and powerful operating system
- ◆ It can easily coordinate the use and sharing of a computer's (or a system's) resources, i.e. built to allow multi-user functionality
- ◆ Can easily handle complex and repetitive tasks easily on large and small datasets
- ◆ Usually, written commands are used to work with this OS, instead of the pointing and clicking used with operating systems like Windows and OSX

# The Unix command-line interface

- ◆ Unix is a stable, efficient and powerful operating system
- ◆ It can easily coordinate the use and sharing of a computer's (or a system's) resources, i.e. built to allow multi-user functionality
- ◆ Can easily handle complex and repetitive tasks easily on large and small datasets
- ◆ Usually, written commands are used to work with this OS, instead of the pointing and clicking used with operating systems like Windows and OSX

# The Unix command-line interface

- ◆ Unix is a stable, efficient and powerful operating system
- ◆ It can easily coordinate the use and sharing of a computer's (or a system's) resources, i.e. built to allow multi-user functionality
- ◆ Can easily handle complex and repetitive tasks easily on large and small datasets
- ◆ Usually, written commands are used to work with this OS, instead of the pointing and clicking used with operating systems like Windows and OSX

*Bioinformatics:*

# The Unix command-line interface

- ◆ Unix is a stable, efficient and powerful operating system
- ◆ It can easily coordinate the use and sharing of a computer's (or a system's) resources, i.e. built to allow multi-user functionality
- ◆ Can easily handle complex and repetitive tasks easily on large and small datasets
- ◆ Usually, written commands are used to work with this OS, instead of the pointing and clicking used with operating systems like Windows and OSX

## *Bioinformatics:*

- ◆ A lot of NGS-analysis tools are created for the Unix OS

# The Unix command-line interface

- ◆ Unix is a stable, efficient and powerful operating system
- ◆ It can easily coordinate the use and sharing of a computer's (or a system's) resources, i.e. built to allow multi-user functionality
- ◆ Can easily handle complex and repetitive tasks easily on large and small datasets
- ◆ Usually, written commands are used to work with this OS, instead of the pointing and clicking used with operating systems like Windows and OSX

## *Bioinformatics:*

- ◆ A lot of NGS-analysis tools are created for the Unix OS
- ◆ High-performance compute clusters which are necessary to analyze large datasets require a working knowledge of Unix

# Linux

- ❖ Linux is a free, open-source operating system based on Unix
- ❖ It has the same components as the original, but the open source community is involved in active development of various distinct distributions of Linux

# Linux

- ♦ Linux is a free, open-source operating system based on Unix
- ♦ It has the same components as the original, but the open source community is involved in active development of various distinct distributions of Linux



ubuntu



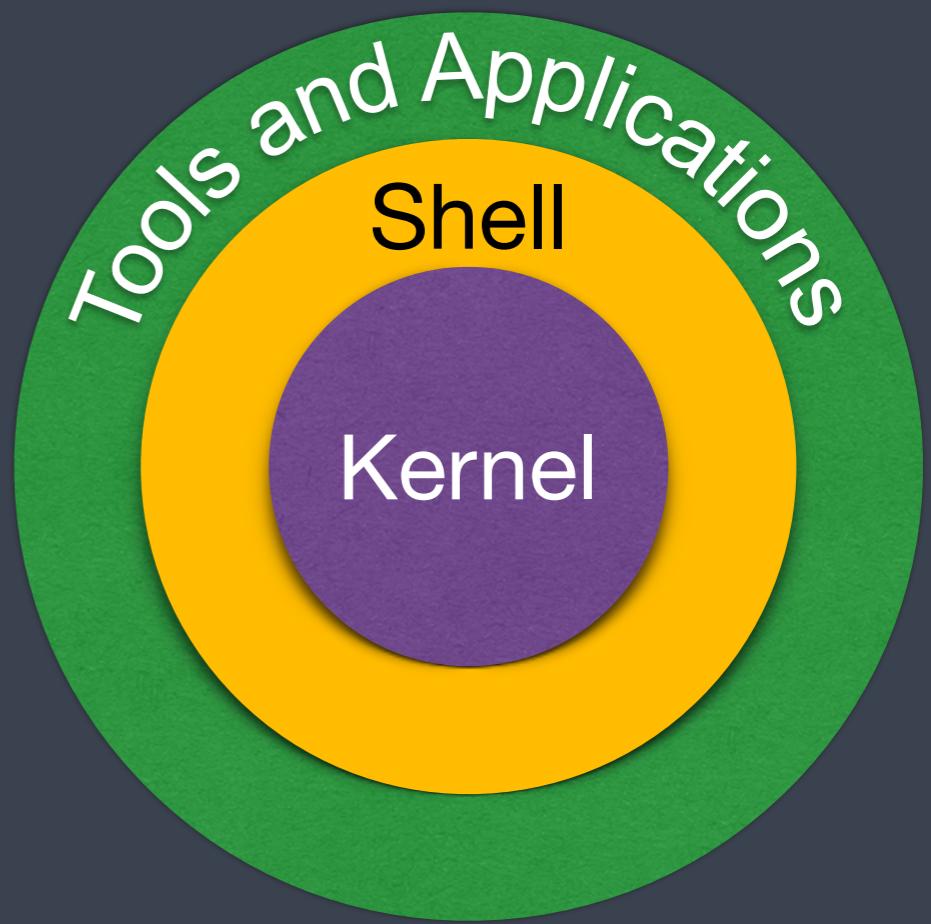
fedora



# Components

The Unix/Linux system is functionally organized at 3 levels:

- ◆ **The kernel**, which schedules tasks and manages storage: *the brain of the system*
- ◆ **The shell**, *an interpreter* that helps interprets our input for the kernel
- ◆ **Utilities, tools and applications**, which use the shell to communicate with the kernel



# The “shell”

- ◆ The shell is **an interpreter**

# The “shell”

- ◆ The shell is **an interpreter**
- ◆ It is independent of the operating system

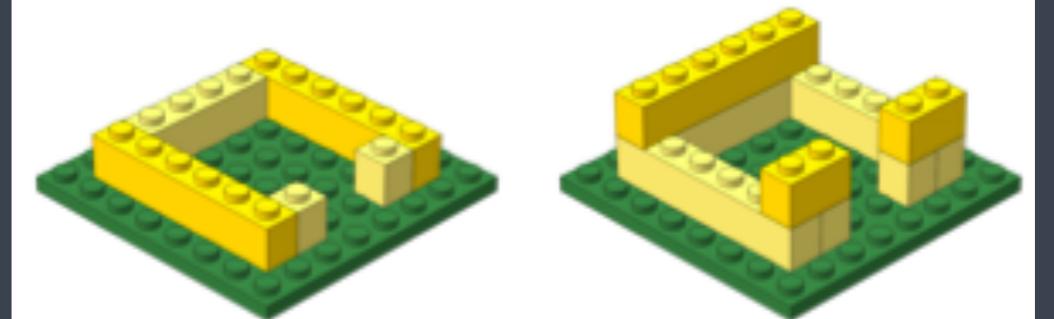
# The “shell”

- ◆ The shell is **an interpreter**
- ◆ It is independent of the operating system
- ◆ Dozens of shells have been developed throughout UNIX history, and a lot of them are still in use

# The “shell”

- ◆ The shell is **an interpreter**
- ◆ It is independent of the operating system
- ◆ Dozens of shells have been developed throughout UNIX history, and a lot of them are still in use
- ◆ The most commonly used shell is **bash**

# Learning Objectives



- ✓ Learn what a “shell” is and become comfortable with the command-line interface
  - Find your way around a filesystem using written commands
  - Work with small and large data files
  - Become more efficient when performing repetitive tasks
- ✓ Understand what a computational cluster is and why we need it

# Logistics

# Course webpage

<https://tinyurl.com/hbc-shell-online>

# Course schedule online

## Workshop Schedule

### Day 1

Time	Topic	Instructor
9:30 - 10:10	Workshop introduction	Meeta
10:10 - 11:40	Introduction to Shell	Mary
11:40 - 12:00	Overview of self-learning materials and homework submission	Jihe

Before the next class:

1. Please **study the contents** and **work through all the code** within the following lessons:

- Wildcards and shortcuts in Shell
- Examining and creating files
- Searching and redirection
- Shell scripts and variables in Shell

**NOTE:** To run through the code above, you will need to be **logged into O2** and **working on a compute node** (i.e. your command prompt should have the word `compute` in it).

1. Log in using `ssh rc_trainingXX@o2.hms.harvard.edu` and enter your password (replace the "XX" in the username with the number you were assigned in class).
2. Once you are on the login node, use `srun --pty -p interactive -t 0-2:30 --mem 1G`

# Course materials online



## Introduction to the command line interface (shell)

[View on GitHub](#)

### Learning Objectives

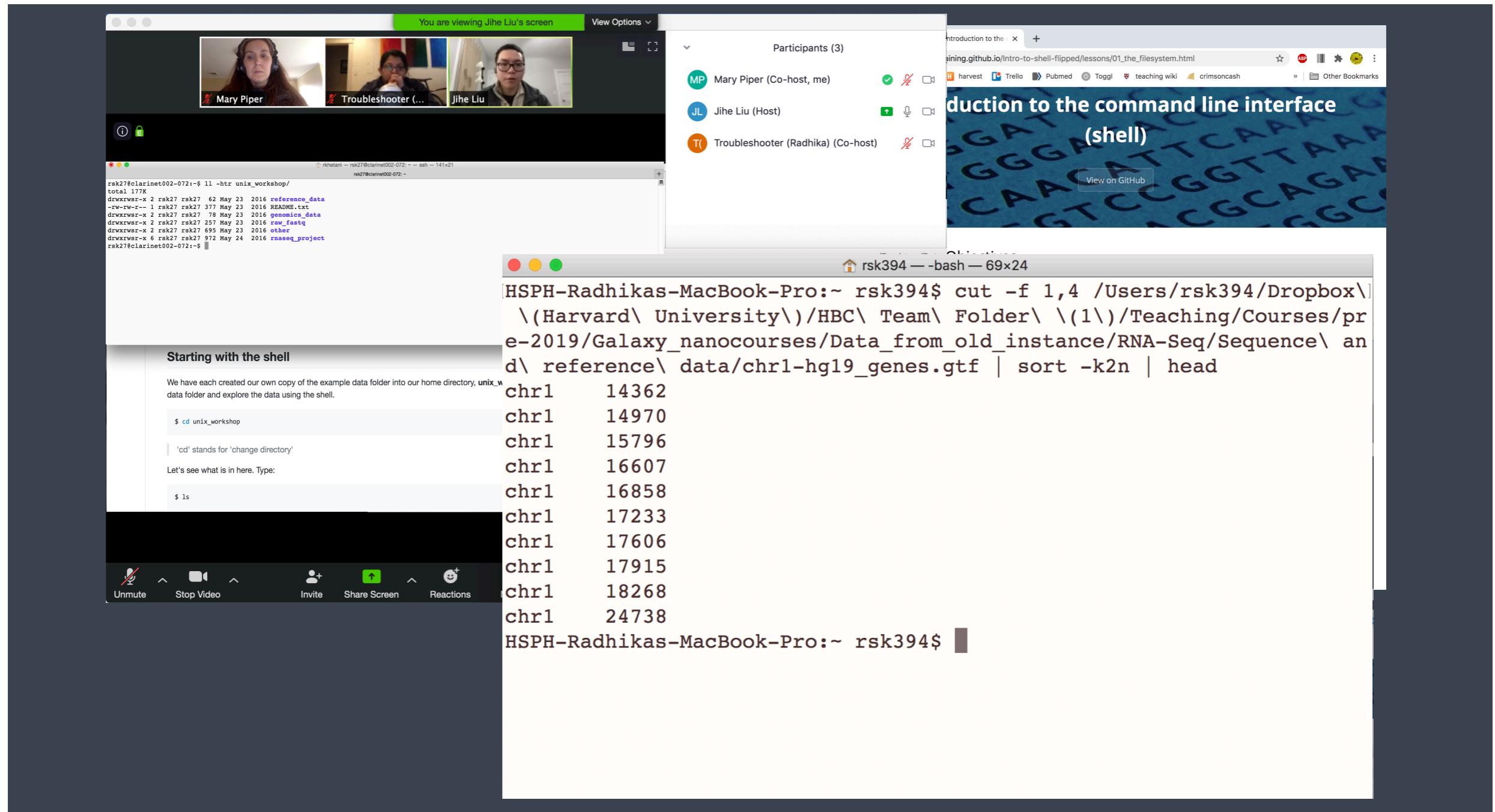
- How do you access the shell?
- How do you use it?
  - Getting around the Unix file system
  - looking at files
  - manipulating files
  - automating tasks
- What is it good for?

### Setting up

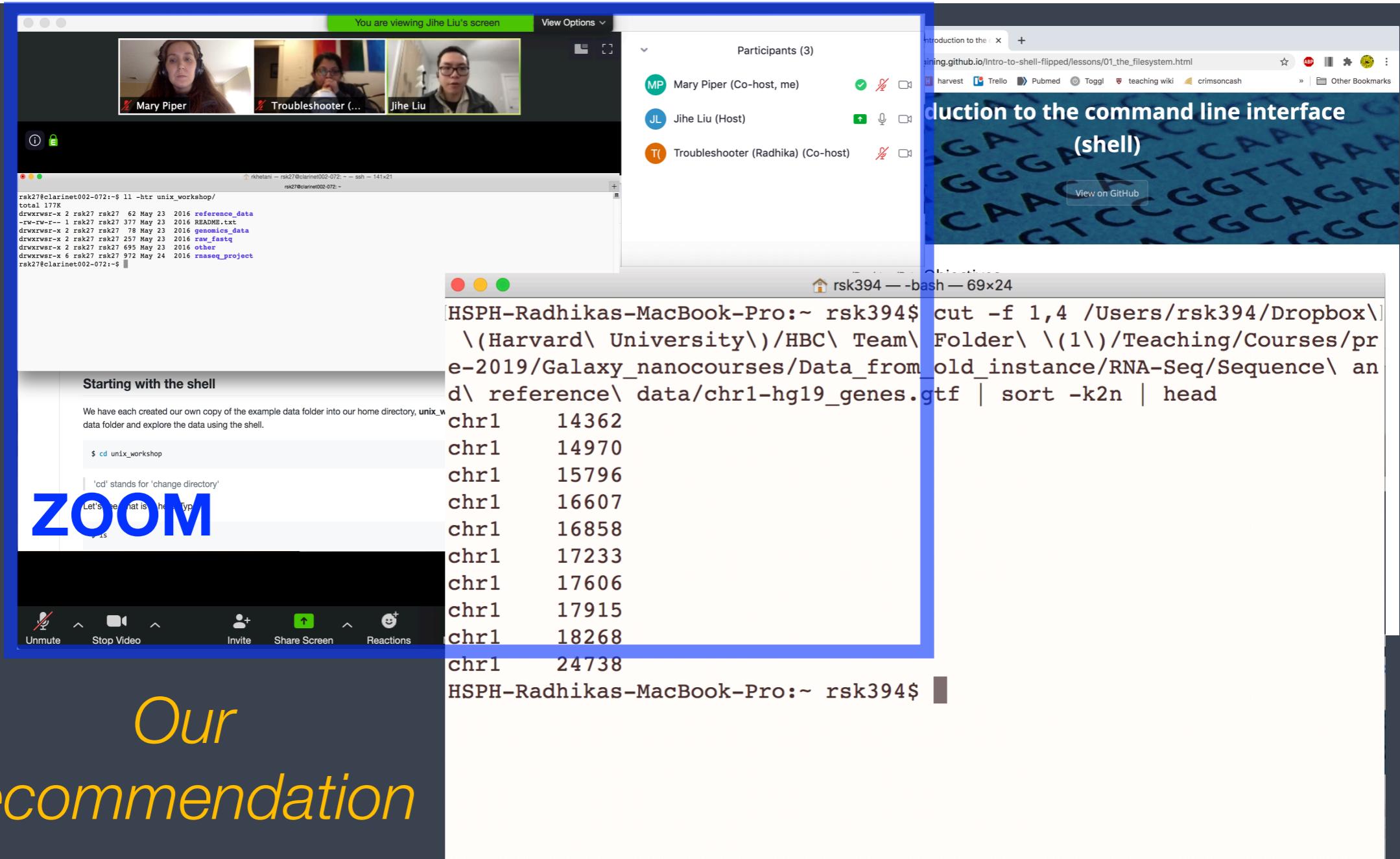
We will spend most of our time learning about the basics of the shell command-line interface (CLI) by exploring experimental data on the **O2** cluster. So, we will need to log in to this remote compute cluster first before we can start with the basics.

Let's take a quick look at the basic architecture of a cluster environment and some cluster-specific jargon prior to logging in.

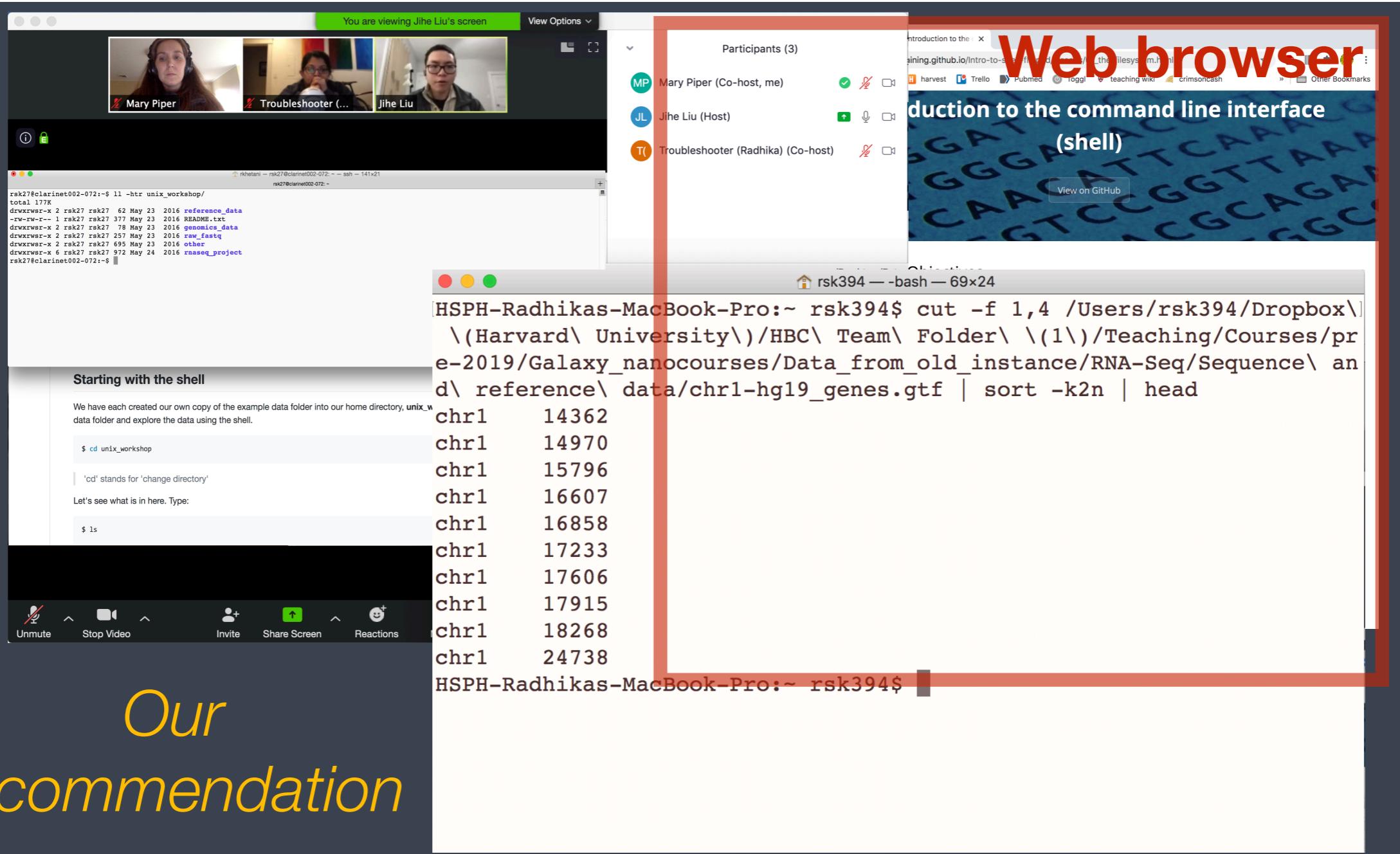
# Single screen & 3 windows?



# Single screen & 3 windows?



# Single screen & 3 windows?



# Single screen & 3 windows?

The image shows a video conference interface with three main windows:

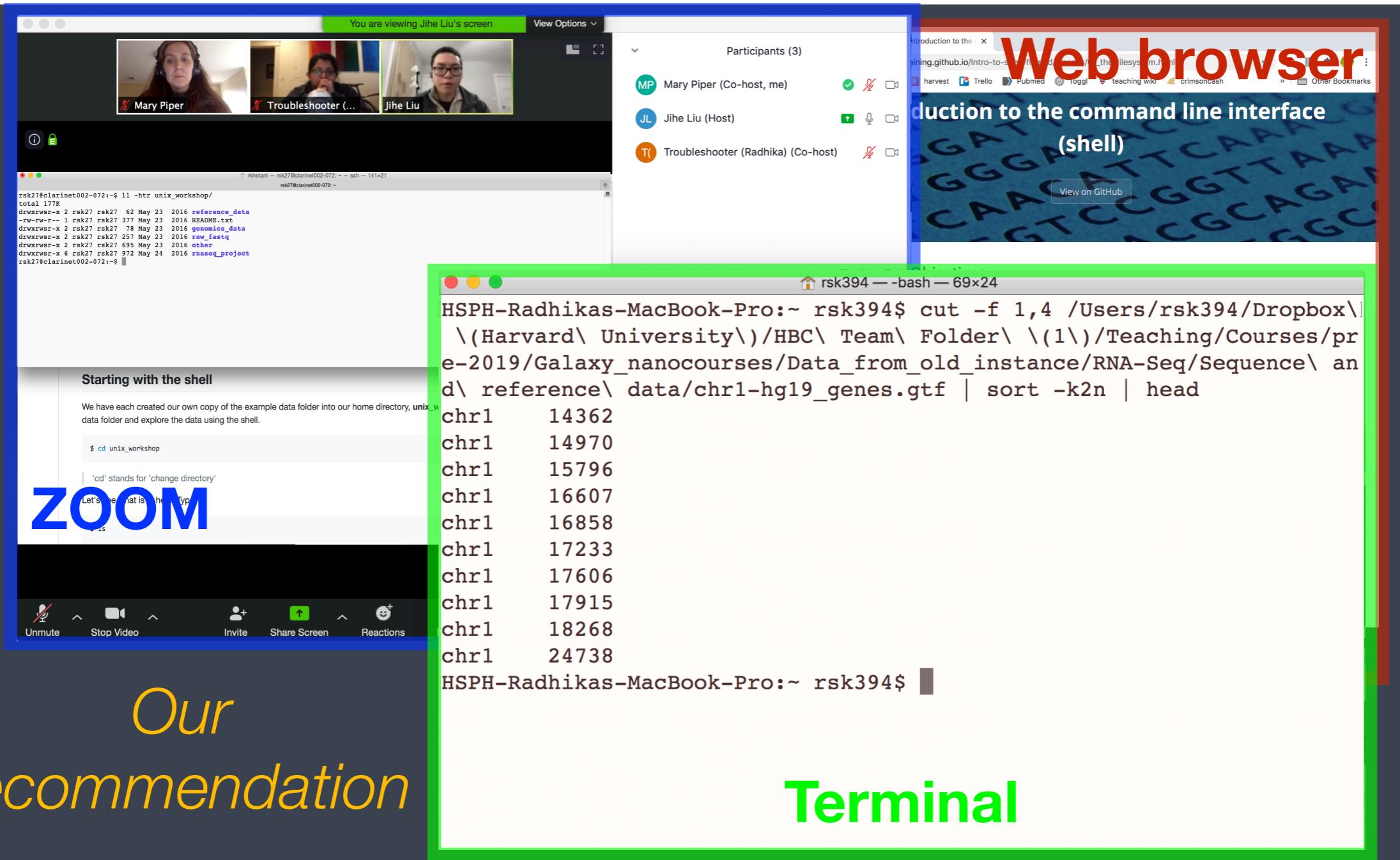
- Video Feed:** Shows three participants: Mary Piper, Troubleshooter (Radhika), and Jihe Liu.
- Participants List:** Shows three participants: Mary Piper (Co-host, me), Jihe Liu (Host), and Troubleshooter (Radhika) (Co-host).
- Terminal Window:** A green-bordered window showing a command-line session. The session starts with a GitHub link to an introduction to the command line interface (shell). It then runs a command to extract specific lines from a GTF file and sort them by chromosome number (k2n). The output shows the first few entries for chromosome 1 (chr1):

```
HSPH-Radhikas-MacBook-Pro:~ rsk394$ cut -f 1,4 /Users/rsk394/Dropbox\\(Harvard\\ University\\)/HBC\\ Team\\ Folder\\ \\(1\\)/Teaching/Courses/pre-2019/Galaxy_nanocourses/Data_from_old_instance/RNA-Seq/Sequence\\ and\\ reference\\ data/chr1-hg19_genes.gtf | sort -k2n | head
chr1    14362
chr1    14970
chr1    15796
chr1    16607
chr1    16858
chr1    17233
chr1    17606
chr1    17915
chr1    18268
chr1    24738
```

**Our recommendation**

**Terminal**

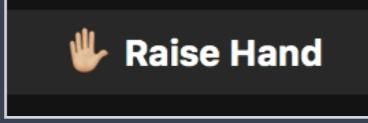
# Single screen & 3 windows?

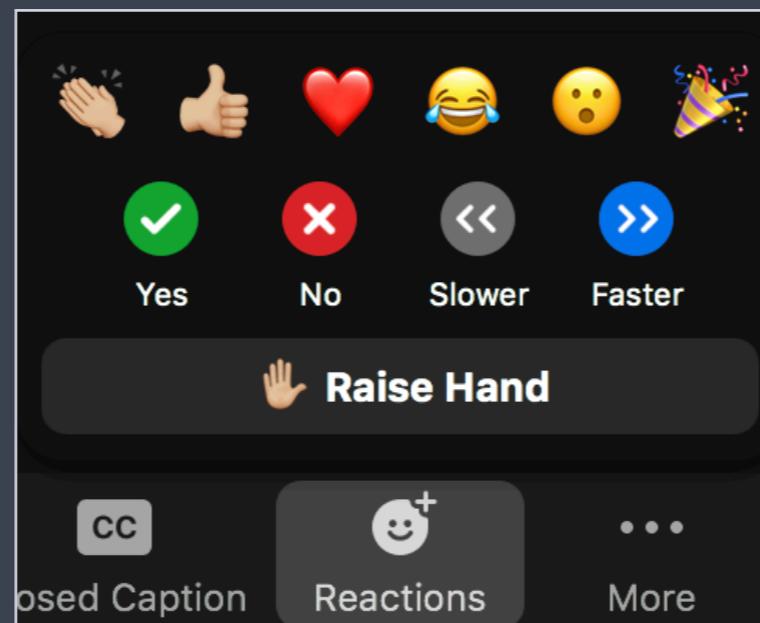


# Odds and Ends

- ❖ Quit/minimize all applications that are not required for class

# Odds and Ends

- ❖ Quit/minimize all applications that are not required for class
- ❖ Are you all set?
  -  = "agree", "I'm all set" (equivalent to a **green post-it**)
  - Type “X” in chat = "disagree", “I’m not ready” (equivalent to a **red post-it**)



# Odds and Ends

- ❖ Quit/minimize all applications that are not required for class
- ❖ Are you all set?
  -  = "agree", "I'm all set" (equivalent to a **green post-it**)
  - Type "X" in chat = "disagree", "I'm not ready" (equivalent to a **red post-it**)
- ❖ Questions for the presenter?
  - Post the question in the Chat window OR
  - Let the moderator know you have a question for the speaker

# Odds and Ends

- ❖ Quit/minimize all applications that are not required for class
- ❖ Are you all set?
  -  = "agree", "I'm all set" (equivalent to a **green post-it**)
  - Type "X" in chat = "disagree", "I'm not ready" (equivalent to a **red post-it**)
- ❖ Questions for the presenter?
  - Post the question in the Chat window OR
  - Let the moderator know you have a question for the speaker
- ❖ Technical difficulties with logging in or running commands?
  - Start a private chat with the *Troubleshooter* with a description of the problem.

# Thanks!

- Kathleen Keating and Andy Bergman from HMS-RC
- Data Carpentry

*These materials have been developed by members of the teaching team at the [Harvard Chan Bioinformatics Core \(HBC\)](#). These are open access materials distributed under the terms of the [Creative Commons Attribution license \(CC BY 4.0\)](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.*



# Contact us!

*HBC training team:* [hbctraining@hsph.harvard.edu](mailto:hbctraining@hsph.harvard.edu)

*O2 (HMS-RC):* [rchelp@hms.harvard.edu](mailto:rchelp@hms.harvard.edu)

*HBC consulting:* [bioinformatics@hsph.harvard.edu](mailto:bioinformatics@hsph.harvard.edu)

## Twitter

*HBC:* @bioinfocore

*HMS-RC:* @hms\_rc