# Introduction to UNIX

Dr. Princess Rodriguez

2025-01-10

# Learning Objectives

- Log in to the VACC, a high-performance computing (HPC) cluster.
- Learn basic command-line navigation.
- Copy data into your home directory.
- List files within a directory.

# Introduction to Command Line

The command-line interface (CLI) and graphical user interface (GUI) are two different ways of interacting with a computer's operating system.

**Figure 1:** GUI-vs-CUI

## What is a Shell?

Most data processing and remote access will be command-line based. For this we need an **interpreter**. A shell is a command-line interpreter that allows users to type commands to launch programs.
The most popular UNIX shell is BASH (the Bourne Again SHell) — so named because it is derived from a shell written by Stephen Bourne.
Learning to use the shell requires time and effort.
**While a GUI presents you with choices to select, CLI does not automatically display these options to you. Instead, you will need to learn specific commands. This will resemble learning a new language!**

## Benefits of Using the Shell

Using the shell provides access to internal system controls, remote servers, and customizable workflows through scripting. With the shell, you can create, edit, and delete files, as well as perform many other tasks efficiently.

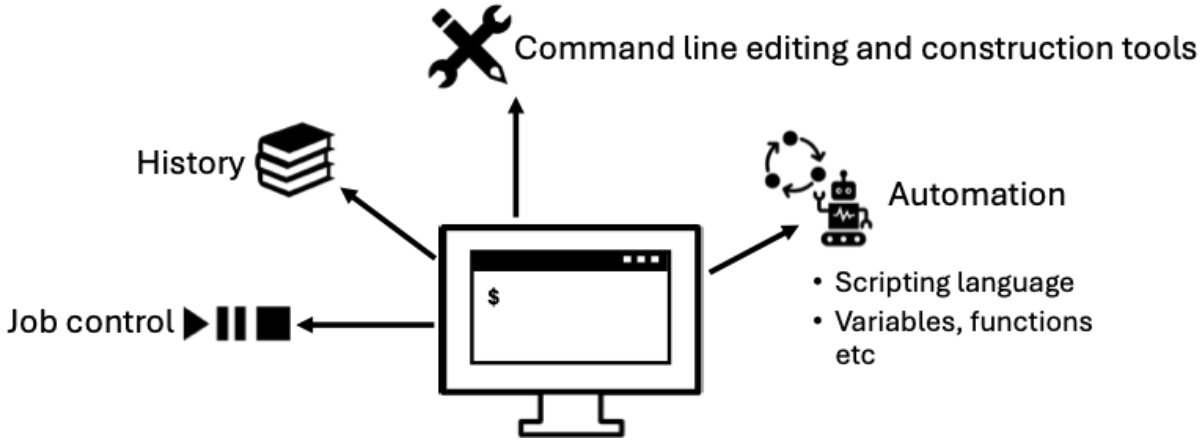



**Figure 2:** Shell-benefits

## How to access the shell

On Mac or Linux machines, you can access the shell locally through a program called **Terminal**. However, for simplicity and convenience, we will use the shell through the open-source web portal **Vermont Advanced Computing Center - Open OnDemand (VACC-OOD)**. Once you open the terminal, you will begin learning the basics of shell programming with the Bourne Again Shell (Bash).

# Working with Remote Machines: Vermont Advance Computing Center Cluster (VACC)

## Why work on the VACC?

Most data-processing tasks in bioinformatics require more computing power than we have on our workstations. For all bioinformatics projects performed in this course, you will work over a network connection with the VACC.

## Cluster Basics

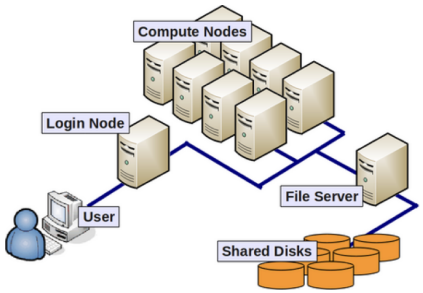



**Figure 3:** Cluster Architecture

The image above illustrates the multiple computers that make up a cluster. Each individual computer in the cluster, referred to as a "node", is significantly more powerful than a typical laptop or desktop computer. A "cluster" is a large system composed of hundreds to thousands of nodes, each serving a specific purpose.

Nodes are generally classified by their roles: login nodes and compute nodes. Login nodes are used for accessing the cluster, setting up jobs, and managing workflows, while compute nodes handle the actual computational analysis or work. Most clusters have a few login nodes and many compute nodes to efficiently handle diverse workloads.

**Common characteristics of a Cluster:**

- Large memory
- Storage shared across nodes
- High speed interconnection network; suitable for high-throughput applications
- Shared by many users

**As of March 2022, the VACC provides three Clusters:**

- BlackDiamond
- Bluemoon
- DeepGreen

**Figure 4:** Bluemoon cluster

We will primarily use the **Bluemoon** cluster for any downstream analysis.

Please note that more information can always be found at the Vermont Advanced Computing Center website.

# Connecting to the VACC

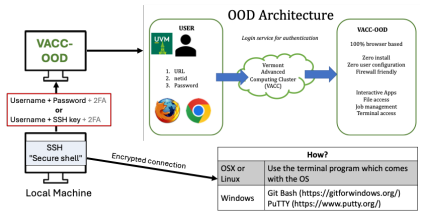



**Figure 5:** VACC-OOD Architecture

To connect to the VACC cluster you can either use SSH or VACC-OOD.

## VACC-OOD Overview

Each student has been provided with their own personal VACC account that they can use to access VACC-Open OnDemand (OOD).

## What is Open OnDemand (OOD)?

Open OnDemand (OOD) is an open source web portal for high performance computing (HPC) that provides users with an *easy-to-use* web interface to HPC clusters.

**Benefits of using OOD:**

1. Web-based, no additional software needs to be installed on your local machine
2. The easiest way to run graphical user interface (GUI) applications remotely on a cluster
3. Typical computing with command-line requires a *high learning curve* whereas OOD is easy to use and simple to learn

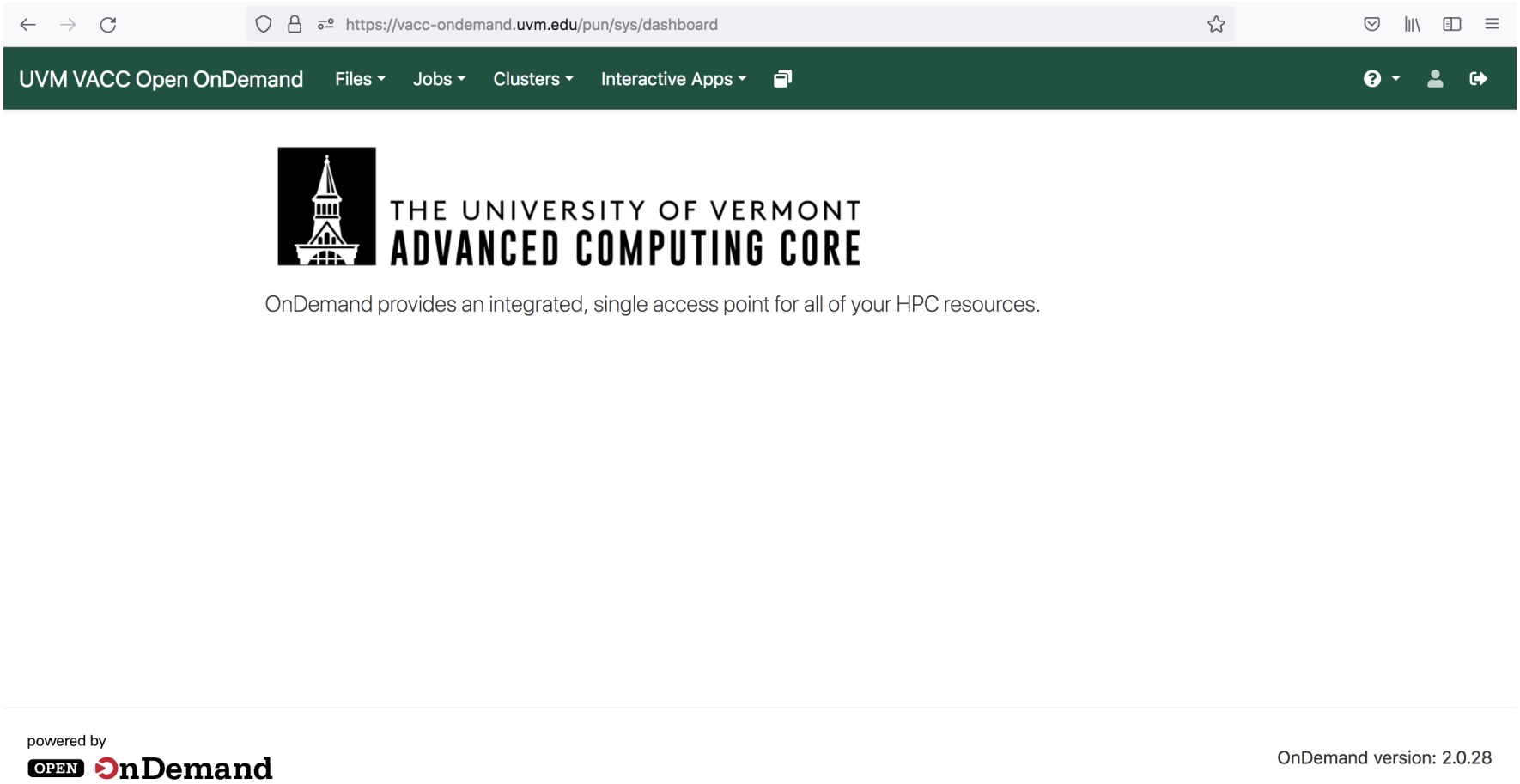
**GUI applications offered by VACC-OOD:**

- Equipped with **Terminal**: this is used to perform tasks on the command line (shell), both locally and on remote machines.
- RStudio: an integrated development environment for R

**Figure 6:** RStudio-Logo

# How to log-in to VACC-OOD:

1. Use the VACC-OOD link to access the site
2. Add your UVM netid and password
3. You should be viewing the following dashboard



## Connecting to VACC with SSH

If you already had a VACC account and/or are currently working towards generating and analyzing your own data, you may want to learn to log-in without VACC-OOD.

- To do so, first open your terminal locally on your computer.
- Once you open your terminal, your screen should look similar to below:

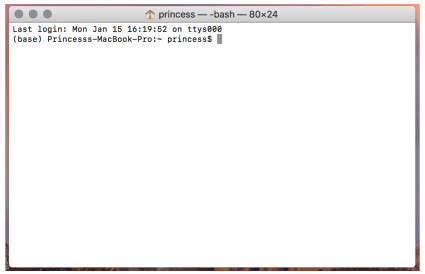



**Figure 9:** Terminal View

## What is SSH?

There are many ways to connect to another machine over a network, but by far the most common is through the secure shell (SSH). We use SSH because its encrypted. This makes it secure to send passwords and edit private data files.

## Step-by-step instructions:

1. Type in the `ssh` command at the command prompt followed by a space, and then type your username (e.g. uvm net id) plus the address of the cluster `@vacc-user1.uvm.edu`.

```
ssh username@vacc-user1.uvm.edu
```

2. Press the return/enter key and you should receive a prompt for your password. Type in your password and note that **the cursor will not move as you type** it in! This is normal and know that the computer is receiving and transmitting your typed password to the remote system, i.e. the VACC cluster.

3. If this is the first time you are connecting to the cluster, **a warning will pop up** and will ask you if you are sure you want to do this; **type `Yes` or `Y`**.

Once logged in, you should see a new command prompt.

## Using VACC-OOD OFF-campus

To use OFF-campus you will need to VPN first. See install-cisco-vpn for more information!

# Running Commands On Terminal

Now that we are logged-in to the VACC, lets explore terminal. Your screen should look similar to the following:

**Figure 10:** Login Explained

To run a program, we will follow these basic steps:

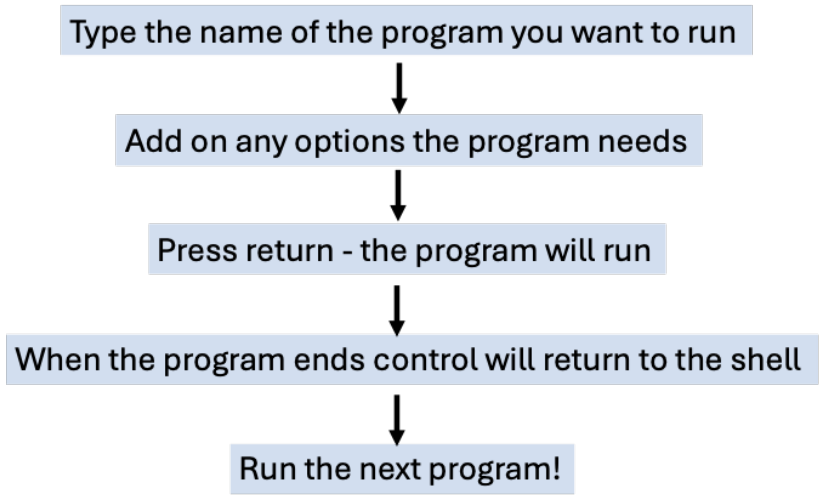



**Figure 11:** Steps to Running a Program/Command

The “$” is called the **“command prompt”**.

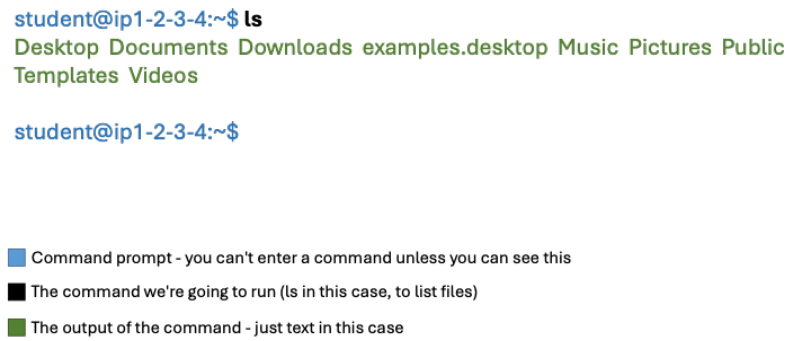



**Figure 12:** Running A Command

## How to get more information on Arguments

Most commands will take additional arguments that control their behavior. How do we know what arguments are available for a particular command? The most commonly used shell commands have a manual available that can be accessed using the `man` command. Let's try this command with `ls`:

```
man ls
```

| Core Programs | Non-Core Programs |
|---|---|
| Included with the install | Additional installs e.g analysis tools |
| Manual page (always) | Help Page (usually) |
| `man [program]` | `[program] --help (or -h)` |

**Figure 13:** figure-out

# Summary of Commands

```
cd
  + Change Directory
      +   used to move throughout the filesystem of a computer

ls
  + List
      +   list the contents of a directory

rm
  + Remove
      + used to remove a file
```

## Citation

*This lesson has 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.*

- *The materials used in this lesson were derived from work that is Copyright © Data Carpentry (http://datacarpentry.org/). All Data Carpentry instructional material is made available under the Creative Commons Attribution license (CC BY 4.0).* Adapted from the lesson by Tracy Teal. Original contributors: Paul Wilson, Milad Fatenejad, Sasha Wood and Radhika Khetani for Software Carpentry (http://software-carpentry.org/)* *authors: Sheldon McKay, Mary Piper, Radhika Khetani, Meeta Mistry, Jihe Liu date posted: September 28, 2020*

*Other parts of this lesson were derived from: