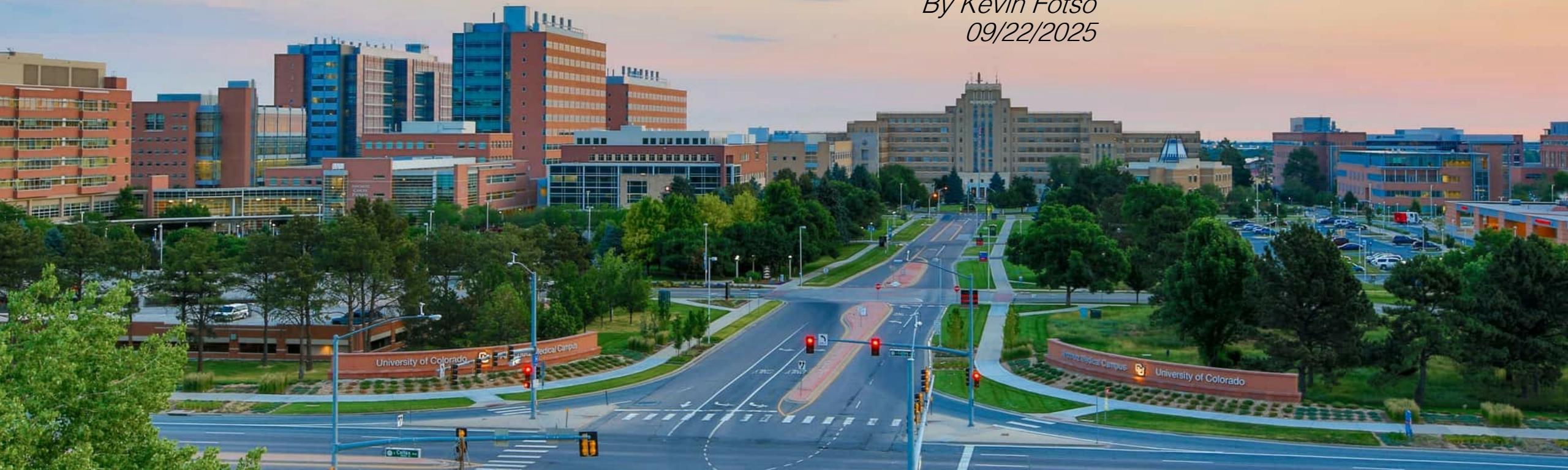




University of Colorado **Anschutz Medical Campus**

# How to use R on Alpine?

*By Kevin Fotso  
09/22/2025*



# Introduction:

This workshop will cover the following topics:

- Overview of R on Alpine.
- How to use R with Ondemand?
- How to use R with the LMOD module stack?
- How to use R with miniforge?



University of Colorado  
Anschutz Medical Campus

# Audience:

Any R user who is new to Alpine.

# Ways to use R on Alpine

- Rstudio on Ondemand with versions **4.4.1** and **4.2.2**.
- The version **4.4.1** is **strongly recommended!**



RStudio Server

## RStudio Server

This app will launch RStudio Server, an IDE for R on Alpine.

Before utilizing this application, please see the [RStudio Server](#) and [Configuring Open OnDemand interactive applications](#) sections in our documentation. This documentation includes important information regarding quitting an RStudio session. For more information on installing dependencies required by R packages, please see the [Installing dependencies for RStudio Server](#) section in our documentation.

RStudio Version

Rstudio 2024.04.2, R 4.4.1

Configuration type

Preset configuration

Preset configuration

4 cores, 4 hours

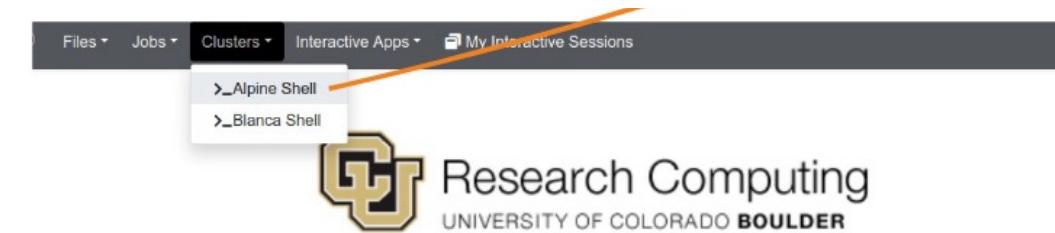
**Launch**

\* The RStudio Server session data for this session can be accessed under the [data root directory](#).

# Ways to use R on Alpine

- Under the Alpine shell, there is an R version 4.4.0 that is accessible by calling **LMOD**, the modern environment module system for HPC.

- We covered LMOD last week here:  
[https://github.com/kf-cuanschutz/CU-Anschutz-HPC-documentation/blob/main/Workshops/LMOD\\_and\\_anaconda\\_v2.pdf](https://github.com/kf-cuanschutz/CU-Anschutz-HPC-documentation/blob/main/Workshops/LMOD_and_anaconda_v2.pdf)



The screenshot shows the University of Colorado Research Computing OnDemand interface. At the top, there is a navigation bar with links for 'Files', 'Jobs', 'Clusters', 'Interactive Apps', and 'My Interactive Sessions'. A dropdown menu under 'Interactive Apps' is open, showing two options: 'Alpine Shell' (which is highlighted with an orange arrow) and 'Blanca Shell'.

**Research Computing**  
UNIVERSITY OF COLORADO BOULDER

OnDemand provides an integrated, single access point for all of your HPC resources.

## Message of the Day

Welcome to the University of Colorado Research Computing.

### Quick Links

[CU Boulder RC Status](#)

[Research Computing User Guide](#)

[Research Computing at CU Boulder](#)

[RMACC @ Ask.Cyberinfrastructure](#)

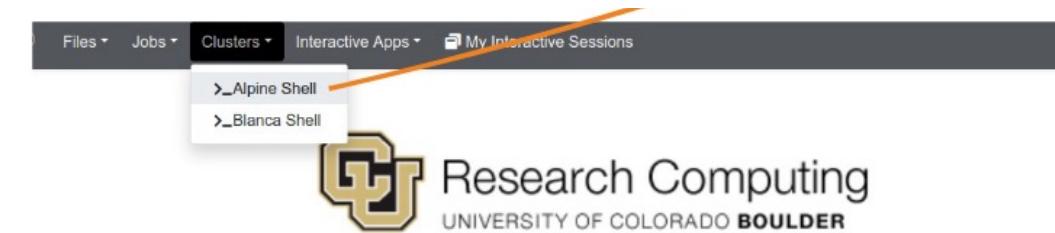
Need help? Email ([rc-help@colorado.edu](mailto:rc-help@colorado.edu))



Currently Loaded Modules:  
1) jdk/18.0.1.1 2) R/4.4.0  
  
module load R/4.4.0  
module list

# Ways to use R on Alpine

One may decide to install R using **miniforge** or anaconda. The latest R version they can install is R **4.4.3**.



The screenshot shows the Research Computing OnDemand interface. At the top, there are navigation links: Files, Jobs, Clusters, Interactive Apps (which is currently selected), and My Interactive Sessions. A dropdown menu for 'Interactive Apps' is open, showing two options: 'Alpine Shell' (highlighted with an orange arrow) and 'Blanca Shell'. To the right of the menu is the University of Colorado Boulder Research Computing logo, which consists of a gold 'CU' monogram and the text 'Research Computing UNIVERSITY OF COLORADO BOULDER'.

OnDemand provides an integrated, single access point for all of your HPC resources.

## Message of the Day

Welcome to the University of Colorado Research Computing.

### Quick Links

[CU Boulder RC Status](#)

[Research Computing User Guide](#)

[Research Computing at CU Boulder](#)

[RMACC @ Ask.Cyberinfrastructure](#)

Need help? Email ([rc-help@colorado.edu](mailto:rc-help@colorado.edu))



module load miniforge



University of Colorado **Anschutz Medical Campus**

# Part I- Ondemand R

# Ondemand Rstudio

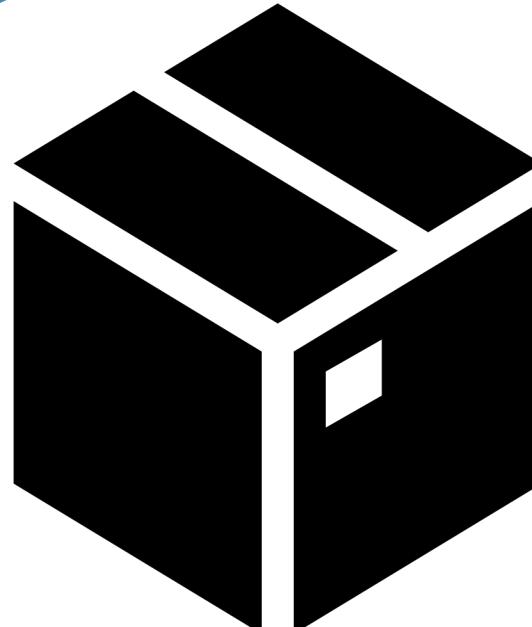
## What is Ondemand Rstudio?

- It was built on top of an apptainer container image, thus, it runs on a different OS than Alpine.
- A container is a tool that enables you to run different applications that were built on different OS than Alpine.

### RStudio Server

This app will launch RStudio Server, an IDE for R on Alpine.

Before utilizing this application, please see the [RStudio Server and Configuring Open OnDemand interactive applications](#) sections in our documentation. This documentation includes important information regarding quitting an RStudio session. For more information on installing dependencies required by R packages, please see the [Installing dependencies for RStudio Server](#) section in our documentation.



Rstudio

Container



University of Colorado  
Anschutz Medical Campus

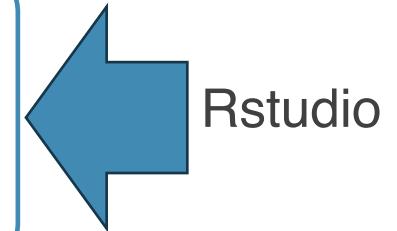
# Ondemand Rstudio

- We will cover containers as an alternative way to install packages during the workshop:  
[“How to use Containers on Alpine”](#) on  
9/29/25.
- We also have an existing guide here:  
[https://github.com/kf-cuanschutz/CU-Anschutz-HPC-documentation/blob/main/Workshops/Container\\_lab\\_workshop\\_040125.pdf](https://github.com/kf-cuanschutz/CU-Anschutz-HPC-documentation/blob/main/Workshops/Container_lab_workshop_040125.pdf)

## RStudio Server

This app will launch RStudio Server, an IDE for R on Alpine.

Before utilizing this application, please see the [RStudio Server and Configuring Open OnDemand interactive applications](#) sections in our documentation. This documentation includes important information regarding quitting an RStudio session. For more information on installing dependencies required by R packages, please see the [Installing dependencies for RStudio Server](#) section in our documentation.



# Ondemand Rstudio

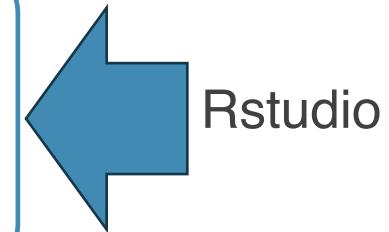
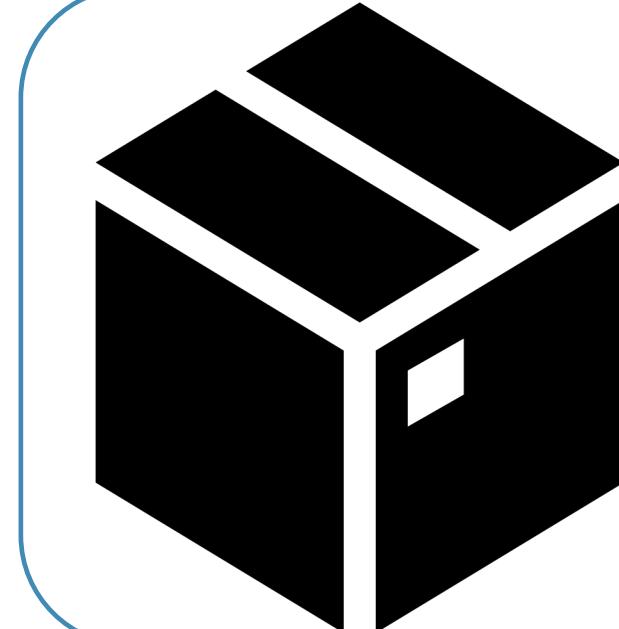
- The Rstudio .sif image is located in

*/curc/sw/containers/open\_ondemand/r  
studio-server-4.4.1.sif*

## RStudio Server

This app will launch RStudio Server, an IDE for R on Alpine.

Before utilizing this application, please see the [RStudio Server and Configuring Open OnDemand interactive applications](#) sections in our documentation. This documentation includes important information regarding quitting an RStudio session. For more information on installing dependencies required by R packages, please see the [Installing dependencies for RStudio Server](#) section in our documentation.



# How to access Rstudio?

First let's login to Alpine!



1

Go to <https://ondemand-rmacc.rc.colorado.edu>

2

It should redirect you to CILogon which is how you authenticate your Alpine session. Make sure you select "ACCESS CI (XSEDE)" as your identity provider and then press "Log On"



The image shows a two-step login process. Step 1: A consent screen titled "Consent to Attribute Release" from "Open OnDemand" asking for permission to access user identifier, name, email, and affiliation. Step 2: A "Select an Identity Provider" screen where "ACCESS CI (XSEDE)" is selected in a dropdown menu, and a "Log On" button is visible at the bottom.

6



University of Colorado  
Anschutz Medical Campus

Source: [https://github.com/kf-cuanschutz/CU-Anschutz-HPC-documentation/blob/main/Workshops/Alpine\\_Noob\\_Introduction\\_to\\_HPC\\_and\\_Alpine-120723.pdf](https://github.com/kf-cuanschutz/CU-Anschutz-HPC-documentation/blob/main/Workshops/Alpine_Noob_Introduction_to_HPC_and_Alpine-120723.pdf)

# How to access Rstudio?

First let's login to Alpine!



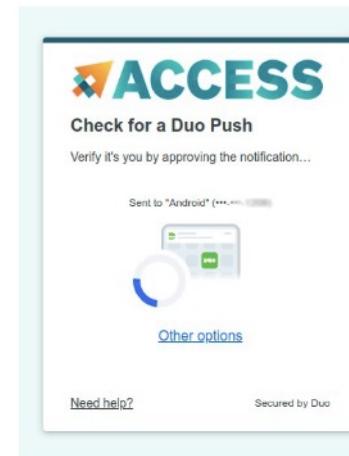
3

Next it will take you to this page where you will put in your ACCESS ID and ACCESS password and press Login. This is NOT your CU Anschutz ID!!

A screenshot of the ACCESS CILogon login page. The page has a light blue header with the ACCESS logo. Below the header, there is a form titled "Login to CILogon" with two input fields: "ACCESS Username" and "ACCESS Password", both with placeholder text. A large teal "Login" button is at the bottom. To the left of the form, there is a note: "ACCESS ID and ACCESS password not CU Anschutz credentials!!". On the right side of the page, there is a "CILogon" logo with the text "CILogon facilitates secure access to CyberInfrastructure (CI)." and links for "Register for an ACCESS Account", "Forgot your password?", and "Need Help?".

4

This will prompt a DUO MFA push to your phone or whichever way you have DUO set up on your phone to authenticate. Accept the push sent to your device.



7



University of Colorado  
Anschutz Medical Campus

Source: [https://github.com/kf-cuanschutz/CU-Anschutz-HPC-documentation/blob/main/Workshops/Alpine\\_Noob\\_Introduction\\_to\\_HPC\\_and\\_Alpine-120723.pdf](https://github.com/kf-cuanschutz/CU-Anschutz-HPC-documentation/blob/main/Workshops/Alpine_Noob_Introduction_to_HPC_and_Alpine-120723.pdf)

# How to access Rstudio?

The screenshot shows the University of Colorado Research Computing HPC interface. At the top, there is a navigation bar with icons for Files, Jobs, Clusters, and Interactive Apps. The 'Interactive Apps' icon is highlighted with a red box. A dropdown menu for 'Interactive Apps' is open, showing categories: Desktops, GUIs, and Servers. Under 'Servers', the 'RStudio Server' option is highlighted with a red box. To the left of the interface, there is a large 'CU' logo and some text about OnDemand services and a message center.

CU

OnDemand pro...  
Message o...

Computing  
COLORADO BOULDER

single access point for all of your HPC resources.

Interactive Apps

- Desktops
- Core Desktop
- GUIs
- MATLAB
- Servers
- RStudio Server
- VS Code-Server

Welcome to the University of Colorado Research Computing!

## Quick Links

[CU Boulder RC Status](#)



University of Colorado  
Anschutz Medical Campus

# When to use R with OnDemand?

Use Rstudio on Alpine if:

- If you are used to Rstudio outside of Alpine
- If you need to display interactive plots.
- If you think that most of your pipelines will not require more than 16 cores or 60GB of RAM.

## RStudio Server

This app will launch [RStudio Server](#), an IDE for R on Alpine.

Before utilizing this application, please see the [RStudio Server](#) and [Configuring Open OnDemand interactive applications](#) sections in our documentation. This documentation includes important information regarding quitting an RStudio session. For more information on installing dependencies required by R packages, please see the [Installing dependencies for RStudio Server](#) section in our documentation.

RStudio Version

Rstudio 2024.04.2, R 4.4.1

Configuration type

Preset configuration

Preset configuration

✓ 1 core, 12 hours

4 cores, 4 hours

Launch

# When to use R with OnDemand?

2 versions of Rstudio available:

- R v4.2.2
- R v4.4.1
- We strongly recommend to use **R v4.4.1!**

## RStudio Server

This app will launch [RStudio Server](#), an IDE for R on Alpine.

Before utilizing this application, please see the [RStudio Server](#) and [Configuring Open OnDemand interactive applications](#) sections in our documentation. This documentation includes important information regarding quitting an RStudio session. For more information on installing dependencies required by R packages, please see the [Installing dependencies for RStudio Server](#) section in our documentation.

RStudio Version

Rstudio 2024.04.2, R 4.4.1

Configuration type

Preset configuration

Preset configuration

✓ 1 core, 12 hours

4 cores, 4 hours

Launch

# When to use R with OnDemand?

2 resource configurations possible:

- Preset configuration
- Custom configuration

## RStudio Server

This app will launch [RStudio Server](#), an IDE for R on Alpine.

Before utilizing this application, please see the [RStudio Server](#) and [Configuring Open OnDemand interactive applications](#) sections in our documentation. This documentation includes important information regarding quitting an RStudio session. For more information on installing dependencies required by R packages, please see the [Installing dependencies for RStudio Server](#) section in our documentation.

RStudio Version

Rstudio 2024.04.2, R 4.4.1

Configuration type

Preset configuration

Preset configuration

✓ 1 core, 12 hours

4 cores, 4 hours

Launch



University of Colorado  
Anschutz Medical Campus

# When to use R with OnDemand?

2 resource configurations possible:

- Preset configuration
- Custom configuration

## RStudio Server

This app will launch [RStudio Server](#), an IDE for R on Alpine.

Before utilizing this application, please see the [RStudio Server](#) and [Configuring Open OnDemand interactive applications](#) sections in our documentation. This documentation includes important information regarding quitting an RStudio session. For more information on installing dependencies required by R packages, please see the [Installing dependencies for RStudio Server](#) section in our documentation.

RStudio Version

Rstudio 2024.04.2, R 4.4.1

Configuration type

Preset configuration

Preset configuration

✓ 1 core, 12 hours

4 cores, 4 hours

Launch



University of Colorado  
Anschutz Medical Campus

# Preset configuration

- Between 1 and 4 cores.
- Up to 12 hours.
- Choose this configuration if you are still new to Alpine and you do not need more resources.

## RStudio Server

This app will launch [RStudio Server](#), an IDE for R on Alpine.

Before utilizing this application, please see the [RStudio Server](#) and [Configuring Open OnDemand interactive applications](#) sections in our documentation. This documentation includes important information regarding quitting an RStudio session. For more information on installing dependencies required by R packages, please see the [Installing dependencies for RStudio Server](#) section in our documentation.

RStudio Version

Rstudio 2024.04.2, R 4.4.1

Configuration type

Preset configuration

Preset configuration

✓ 1 core, 12 hours

4 cores, 4 hours

Launch



University of Colorado  
Anschutz Medical Campus

# Custom configuration

To access it, select “Custom configuration” under “ Configuration type”.



## RStudio Server

This app will launch [RStudio Server](#), an IDE for R on Alpine.

Before utilizing this application, please see the [RStudio Server](#) and [Configuring Open OnDemand interactive applications](#) sections in our documentation. This documentation includes important information regarding quitting an RStudio session. For more information on installing dependencies required by R packages, please see the [Installing dependencies for RStudio Server](#) section in our documentation.

### RStudio Version

Rstudio 2024.04.2, R 4.4.1

### Configuration type

Custom configuration

### Cluster

Alpine

### Account

amc-general

### Partition

ahub

### QoS

interactive

### Time

6

### Number of cores

16

### Reservation (default is None)

None

### gres options (default is None)

None

Launch

\* The RStudio Server session data for this session can be accessed under the [data root directory](#).

# Custom configuration

- Make sure that your account is amc-general!
- If the launch fails due to an incorrect account, please email [HPCSupport@cuanschutz.edu](mailto:HPCSupport@cuanschutz.edu) .



## RStudio Server

This app will launch [RStudio Server](#), an IDE for R on Alpine.

Before utilizing this application, please see the [RStudio Server](#) and [Configuring Open OnDemand interactive applications](#) sections in our documentation. This documentation includes important information regarding quitting an RStudio session. For more information on installing dependencies required by R packages, please see the [Installing dependencies for RStudio Server](#) section in our documentation.

### RStudio Version

Rstudio 2024.04.2, R 4.4.1

### Configuration type

Custom configuration

### Cluster

Alpine

### Account

amc-general

### Partition

ahub

### QoS

interactive

### Time

6

### Number of cores

16

### Reservation (default is None)

None

### gres options (default is None)

None

Launch

\* The RStudio Server session data for this session can be accessed under the [data root directory](#).

# Custom configuration

- Always select ahub, which is the correct Ondemand partition.



## RStudio Server

This app will launch [RStudio Server](#), an IDE for R on Alpine.

Before utilizing this application, please see the [RStudio Server](#) and [Configuring Open OnDemand interactive applications](#) sections in our documentation. This documentation includes important information regarding quitting an RStudio session. For more information on installing dependencies required by R packages, please see the [Installing dependencies for RStudio Server](#) section in our documentation.

### RStudio Version

Rstudio 2024.04.2, R 4.4.1

### Configuration type

Custom configuration

### Cluster

Alpine

### Account

amc-general

### Partition

ahub

### QoS

interactive

### Time

6

### Number of cores

16

### Reservation (default is None)

None

### gres options (default is None)

None

Launch

\* The RStudio Server session data for this session can be accessed under the [data root directory](#).

# Custom configuration

- The only QOS that works with the ahub partition is “interactive”.
- For more information on QOS please refer to this workshop:

[https://github.com/kf-cuanschutz/CU-Anschutz-HPC-documentation/blob/main/Workshops/Introduction\\_to\\_Alpine\\_workshop\\_09092025\\_v3.pdf](https://github.com/kf-cuanschutz/CU-Anschutz-HPC-documentation/blob/main/Workshops/Introduction_to_Alpine_workshop_09092025_v3.pdf)



## RStudio Server

This app will launch RStudio Server, an IDE for R on Alpine.

Before utilizing this application, please see the [RStudio Server](#) and [Configuring Open OnDemand interactive applications](#) sections in our documentation. This documentation includes important information regarding quitting an RStudio session. For more information on installing dependencies required by R packages, please see the [Installing dependencies for RStudio Server](#) section in our documentation.

### RStudio Version

Rstudio 2024.04.2, R 4.4.1

### Configuration type

Custom configuration

### Cluster

Alpine

### Account

amc-general

### Partition

ahub

### QoS

interactive

### Time

6

### Number of cores

16

### Reservation (default is None)

None

### gres options (default is None)

None

Launch

\* The RStudio Server session data for this session can be accessed under the [data root directory](#).

# Custom configuration

- Up to 12 hours walltime.
- However, wait times should be nonexistent.



## RStudio Server

This app will launch [RStudio Server](#), an IDE for R on Alpine.

Before utilizing this application, please see the [RStudio Server](#) and [Configuring Open OnDemand interactive applications](#) sections in our documentation. This documentation includes important information regarding quitting an RStudio session. For more information on installing dependencies required by R packages, please see the [Installing dependencies for RStudio Server](#) section in our documentation.

### RStudio Version

Rstudio 2024.04.2, R 4.4.1

### Configuration type

Custom configuration

### Cluster

Alpine

### Account

amc-general

### Partition

ahub

### QoS

interactive

### Time

6

### Number of cores

16

### Reservation (default is None)

None

### gres options (default is None)

None

Launch

\* The RStudio Server session data for this session can be accessed under the [data root directory](#).

# Custom configuration

- Up to 16 cores maximum.
- This will translate to roughly 60G mem
- $3.75\text{G-per-core} \times 16 \text{ cores} = 60 \text{ G}$



## RStudio Server

This app will launch [RStudio Server](#), an IDE for R on Alpine.

Before utilizing this application, please see the [RStudio Server](#) and [Configuring Open OnDemand interactive applications](#) sections in our documentation. This documentation includes important information regarding quitting an RStudio session. For more information on installing dependencies required by R packages, please see the [Installing dependencies for RStudio Server](#) section in our documentation.

### RStudio Version

Rstudio 2024.04.2, R 4.4.1

### Configuration type

Custom configuration

### Cluster

Alpine

### Account

amc-general

### Partition

ahub

### QoS

interactive

### Time

6

### Number of cores

16

### Reservation (default is None)

None

### gres options (default is None)

None

Launch

\* The RStudio Server session data for this session can be accessed under the [data root directory](#).

# Custom configuration

Reservation and gres do not apply here.



University of Colorado  
Anschutz Medical Campus

## RStudio Server

This app will launch [RStudio Server](#), an IDE for R on Alpine.

Before utilizing this application, please see the [RStudio Server](#) and [Configuring Open OnDemand interactive applications](#) sections in our documentation. This documentation includes important information regarding quitting an RStudio session. For more information on installing dependencies required by R packages, please see the [Installing dependencies for RStudio Server](#) section in our documentation.

### RStudio Version

Rstudio 2024.04.2, R 4.4.1

### Configuration type

Custom configuration

### Cluster

Alpine

### Account

amc-general

### Partition

ahub

### QoS

interactive

### Time

6

### Number of cores

16

### Reservation (default is None)

None

### gres options (default is None)

None

Launch

\* The RStudio Server session data for this session can be accessed under the [data root directory](#).

# Rstudio library dependencies

- Because Rstudio on Alpine is distributed through a container, it does not contain all the dependencies.
- Let's review how to install most library dependencies needed for an AMC user on Alpine.



University of Colorado  
Anschutz Medical Campus

# Rstudio library dependencies

- Select Rv4.4.1, 4 cores and finally launch a session.

The screenshot shows the RStudio Server configuration page. At the top, there's a navigation bar with links for Files, Jobs, Clusters, Interactive Apps, and My Interactive Sessions. On the far right is a Help link. Below the navigation is a breadcrumb trail: Home / My Interactive Sessions / RStudio Server. To the left is a sidebar titled "Interactive Apps" with categories: Desktops, GUIs, Servers, Jupyter Session, RStudio Server, and VS Code-Server. The "RStudio Server" item is highlighted with a blue background. The main content area is titled "RStudio Server" and contains a message: "This app will launch RStudio Server, an IDE for R on Alpine." Below this is a yellow-highlighted note: "Before utilizing this application, please see the [RStudio Server](#) and [Configuring Open OnDemand interactive applications](#) sections in our documentation. This documentation includes important information regarding quitting an RStudio session. For more information on installing dependencies required by R packages, please see the [Installing dependencies for RStudio Server](#) section in our documentation." A dropdown menu for "RStudio Version" is open, showing "✓ Rstudio 2024.04.2, R 4.4.1" (which is selected) and "Rstudio 2023.03.0, R 4.2.2". A "Configuration type" dropdown below it has "Preset configuration" selected. Another "Preset configuration" dropdown shows "4 cores, 4 hours". At the bottom is a large blue "Launch" button. A footer note at the bottom of the page states: "\* The RStudio Server session data for this session can be accessed under the [data root directory](#).

# Rstudio library dependencies

- The screen below indicates that your session is starting.

The screenshot shows the RStudio Server interface. At the top, there's a navigation bar with links for 'Interactive Apps', 'My Interactive Sessions', 'Help', and 'Logout'. Below the navigation bar, a green success message box says 'Session was successfully created.' On the left, there's a sidebar titled 'Interactive Apps' with categories: 'Desktops' (Core Desktop (Presets)), 'GUIs' (MATLAB (Presets)), and 'Servers' (Jupyter Session (Custom), Jupyter Session (Presets), RStudio Server (Custom), RStudio Server (Presets), VS Code-Server (Custom), VS Code-Server (Presets)). The main content area shows a session named 'RStudio Server (Presets) (4927252)' with details: 'Created at: 2024-02-12 16:07:24 MST', '1 node | 4 cores | Starting', 'Time Remaining: 4 hours', and 'Session ID: b69b7ba1-0363-4bd6-a203-9e4288e0a787'. A red 'Delete' button is visible next to the session name. A message at the bottom of the session card says 'Your session is currently starting... Please be patient as this process can take a few minutes.'

# Rstudio library dependencies

- Select “Connect to Rstudio Server”.

The screenshot shows the RStudio Server interface. At the top, there's a navigation bar with tabs for "Interactive Apps" and "My Interactive Sessions". Below the navigation bar, a green success message box says "Session was successfully created." On the left, a sidebar titled "Interactive Apps" lists various options like "Desktops", "Core Desktop (Presets)", "GUIs", "MATLAB (Presets)", "Servers", "Jupyter Session (Custom)", "Jupyter Session (Presets)", "RStudio Server (Custom)", "RStudio Server (Presets)", "VS Code-Server (Custom)", and "VS Code-Server (Presets)". In the center, a card for a session titled "RStudio Server (Presets) (4927252)" is displayed. The card includes details such as "1 node | 4 cores | Running", "Host: >\_c3cpu-c11-u1-4.rc.int.colorado.edu", "Created at: 2024-02-12 16:07:24 MST", "Time Remaining: 3 hours and 59 minutes", and "Session ID: b69b7ba1-0363-4bd6-a203-9e4288e0a787". A blue button labeled "Connect to RStudio Server" is highlighted with a red border. The entire screenshot has a light gray background.

# Rstudio library dependencies

- You are now logged in!
- You have a functional Rstudio session.

The screenshot shows the RStudio desktop application interface. The top menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. The top toolbar has icons for file operations like Open, Save, and Print, along with a Go to file/function search bar and an Addins dropdown. The left pane is the Console, displaying the R startup message and workspace details:

```
R version [REDACTED] (2022-10-31) -- "Innocent and Trusting"
Copyright (C) 2022 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[Workspace loaded from ~/.RData]
```

The right pane is divided into three sections: Environment, History, Connections, and Tutorial. The Environment section shows the Global Environment with the message "Environment is empty". The History section shows a recent dataset import. The bottom-right pane is the Files browser, listing files in the current directory:

Name	Size	Modified
.RData	98 B	Sep 19, 2023, 11:39 AM
.Rhistory	688 B	Jan 5, 2024, 3:06 PM
.Rprofile	561 B	Aug 17, 2023, 8:12 PM
ondemand		
perlS		
README.mdwn	562 B	Feb 1, 2018, 8:35 AM

# Rstudio library dependencies

- However, all the library dependencies are not installed yet (e.g. zlib.h).
- In order to achieve that, we need to install them **directly inside the container!**

The screenshot shows the RStudio interface with the following details:

- Terminal:** A blue arrow points from the top-left towards the terminal window. The terminal output shows a series of gcc commands for compiling XVector code, followed by an error message:

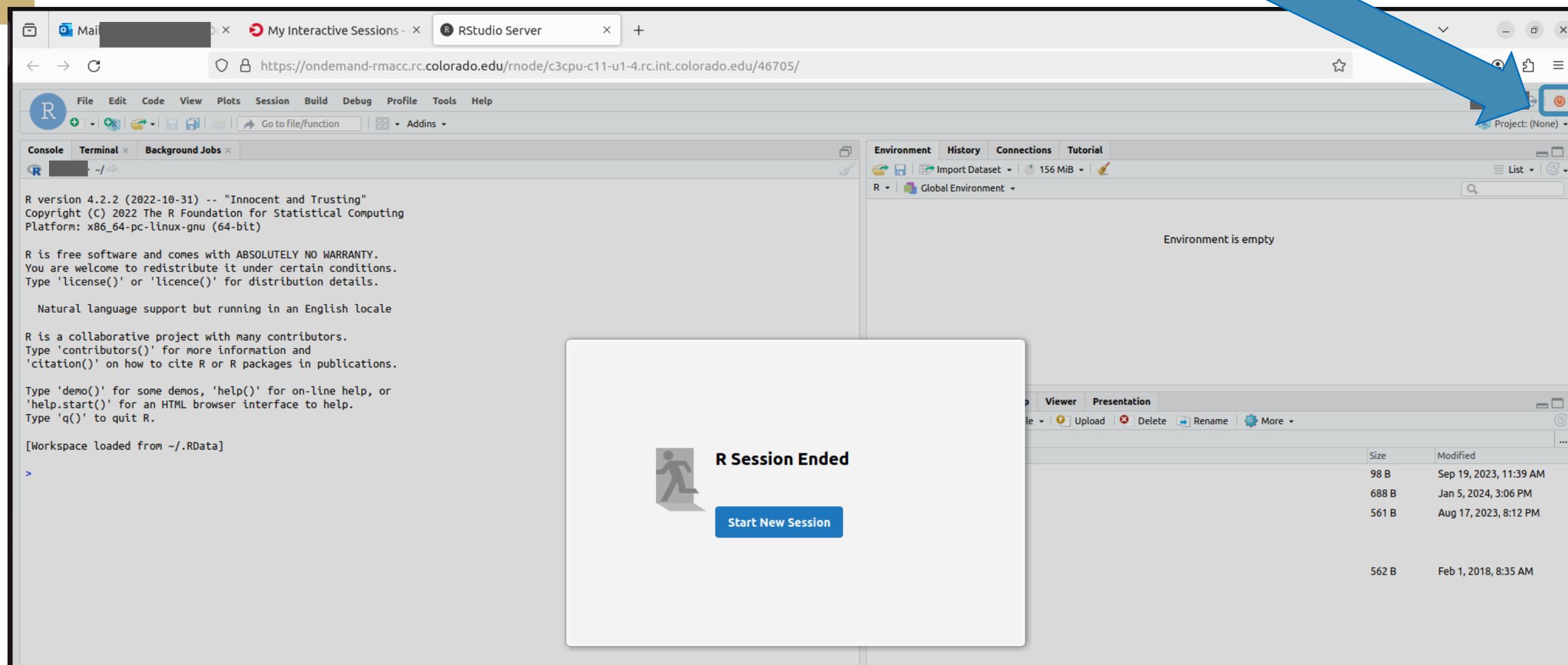
```
io_utils.c:16:10: fatal error: zlib.h: No such file or directory
  16 | #include <zlib.h>
      | ^~~~~~
compilation terminated.
make: *** [/usr/local/lib/R/etc/Makeconf:195: io_utils.o] Error 1
ERROR: compilation failed for package 'XVector'
* removing '/projects/kfotso@xsede.org/Rstudio_libs/4.4.1/Xvector'
```
- File Browser:** On the right, there is a file browser titled "Files". It shows a list of files in the current directory:

Name	Size	Modified
.R	2.6 KB	Feb 21, 2025, 11:58 AM
.RData	4.2 KB	Jul 9, 2025, 3:53 PM
.Rhistory	636 B	Aug 14, 2024, 8:57 AM
@	0 B	Sep 18, 2025, 4:23 PM
&1		
=		
\$TMPDIR		
0.2.127.lua	2.9 KB	Jan 9, 2025, 1:37 PM



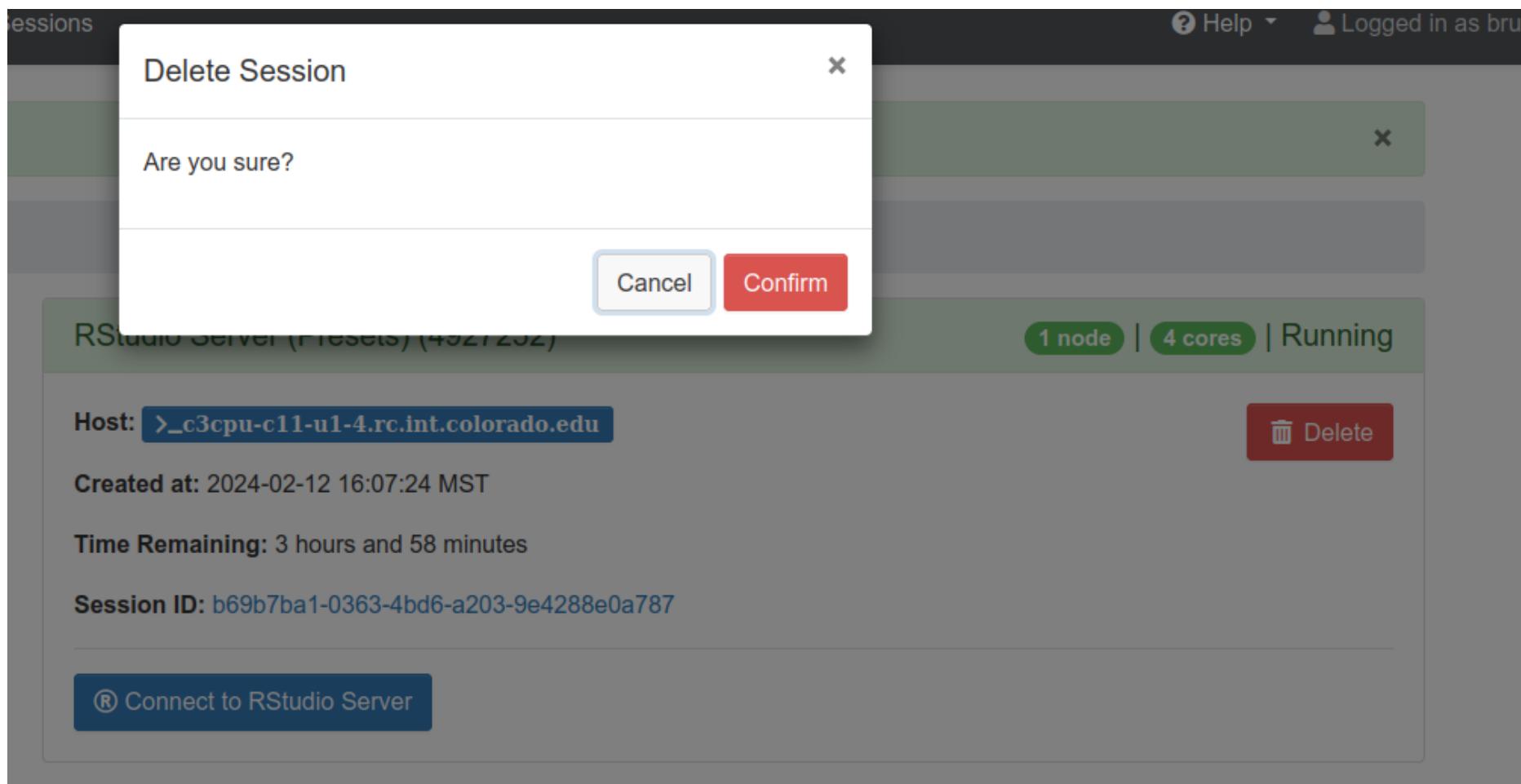
# Rstudio library dependencies

- You need to first exit the Rstudio session.
- In order to do that, please select the red button located on the top right corner.



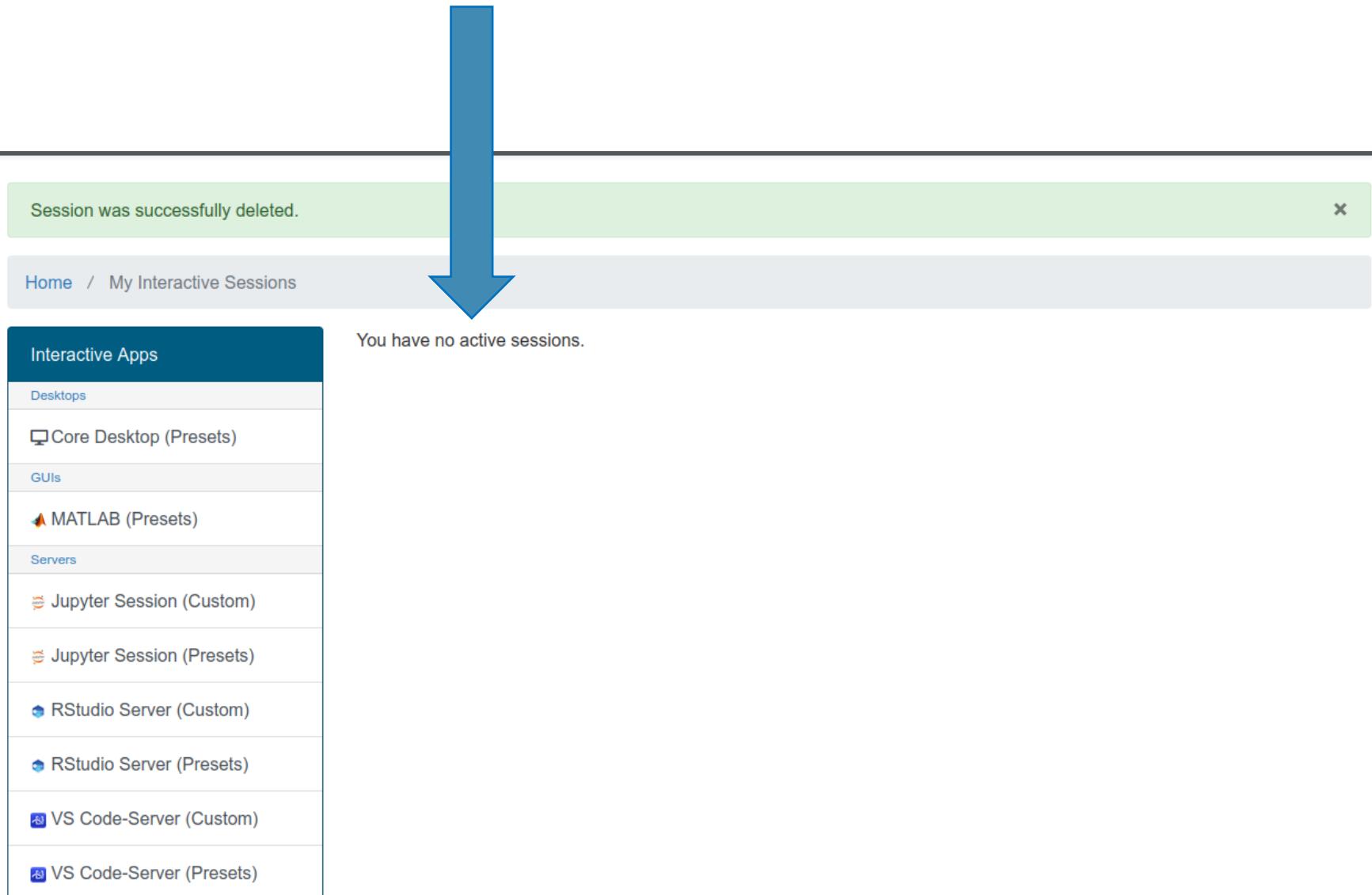
# Rstudio library dependencies

- Now go back to your interactive session tab browser.
- Make sure to press the red Delete button to end your rstudio session request.
- When asked to confirm, hit the red "Confirm" button.



# Rstudio library dependencies

- You can confirm that you are logged out of Rstudio and have killed your RStudio session by seeing the message below.



# Open an Alpine shell

First let's login to Alpine!



5

This will take to the official CURC page. Let's select the Alpine terminal

A screenshot of the Research Computing OnDemand interface. At the top, there is a navigation bar with links for "Files", "Jobs", "Clusters", "Interactive Apps", and "My Interactive Sessions". The "Interactive Apps" link is highlighted with a yellow circle containing the number "5". A dropdown menu is open under "Interactive Apps", showing two options: "Alpine Shell" and "Blanca Shell", with "Alpine Shell" being the selected option. Below the menu, the Research Computing logo (CU Boulder) and the text "Research Computing UNIVERSITY OF COLORADO BOULDER" are displayed. A message states: "OnDemand provides an integrated, single access point for all of your HPC resources." A "Message of the Day" section follows, and a "Quick Links" sidebar on the left contains links to "CU Boulder RC Status", "Research Computing User Guide", "Research Computing at CU Boulder", "RMACC @ Ask.Cyberinfrastructure", and "Need help? Email (rc-help@colorado.edu)".

8

# Open an Alpine shell

## First let's login to Alpine!



6

This will log you into the head node of Alpine. It will always default you to your home directory.

```
Host: login-cil.rc.int.colorado.edu
Welcome to University of Colorado Boulder Research Computing!

Full documentation is available in our user guide at
https://www.rc.colorado.edu/support/user-guide. If you have a question
that's not answered there, contact us at rc-help@colorado.edu.

A number of directories have been created for you already:
* `/home/$USER`, your home directory
* `/projects/$USER`, your project directory

Run the command `module avail` to see a list of available software.

To prevent this README from being displayed at login, edit your
`.bash_profile` or `.login` files.

Welcome to CU-Boulder Research Computing.

* Website http://colorado.edu/rc
* Questions? rc-help@colorado.edu
* Subscribe to system announcements: https://curc.statuspage.io/
* Please type rc-help for the Acceptable Use Policy and a short help page.

You are using login node: login-cil

Users who had jobs in the queue prior to the planned maintenance should check
to confirm these jobs are still queued. Some jobs, particularly those scheduled
since midnight today (Wed June 7), may have been canceled during the
maintenance period.
@xsede.org@login-cil      @xsede.org]$
```

9



University of Colorado  
Anschutz Medical Campus

Source: [https://github.com/kf-cuanschutz/CU-Anschutz-HPC-documentation/blob/main/Workshops/Alpine\\_Noob\\_Introduction\\_to\\_HPC\\_and\\_Alpine-120723.pdf](https://github.com/kf-cuanschutz/CU-Anschutz-HPC-documentation/blob/main/Workshops/Alpine_Noob_Introduction_to_HPC_and_Alpine-120723.pdf)

# Access to a compute node.

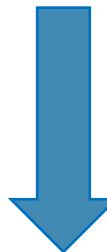
```
@xsede.org@login-ci1 ~]$ acompile --ntasks=4 --time=12:00:00  
acompile: submitting job... salloc --nodes=1 --partition=acompile --ntasks=4 --time=12:00:00 --mem-per-cpu=3480M --qos=compile --  
job-name=acompile --bell --oversubscribe srun --pty /bin/bash  
salloc: Granted job allocation 17666833  
salloc: Nodes c3cpu-a2-u32-3 are ready for job
```

- Let's access a compute node so that we can access the container.



# Rstudio library dependencies

- Change directories into /scratch/alpine/\$USER



```
| @c3cpu-a2-u32-3 ~]$ cd /scratch/alpine/$USER
| @c3cpu-a2-u32-3 [~]$ git clone https://github.com/kf-cuanschutz/Rstudio_repo_related_files.git
Cloning into 'Rstudio_repo_related_files'...
remote: Enumerating objects: 8, done.
remote: Counting objects: 100% (8/8), done.
remote: Compressing objects: 100% (7/7), done.
remote: Total 8 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (8/8), 2.33 KiB | 477.00 KiB/s, done.
| @c3cpu-a2-u32-3 [~]@xsede.org]$ cd Rstudio_repo_related_files/
| @c3cpu-a2-u32-3 Rstudio_repo_related_files]$ ls -la
total 0
drwxr-sr-x  3                               4096 Sep 21 17:22 .
drwxrws--- 484                             32768 Sep 21 17:22 ..
drwxr-sr-x  8                               4096 Sep 21 17:22 .git
-rw-r--r--  1                               91   Sep 21 17:22 README.md
-rw-r--r--  1                             508   Sep 21 17:22 step0_modify_overlay.sh
-rw-r--r--  1                             509   Sep 21 17:22 step0_r_v4_2_2_modify_overlay.sh
-rw-r--r--  1                             550   Sep 21 17:22 step1_install_os_deps.sh
[~]@xsede.org@c3cpu-a2-u32-3 Rstudio_repo_related_files]$
```

# Rstudio library dependencies

- Clone the repository below with:

```
git clone https://github.com/kf-cuanschutz/Rstudio_repo_related_files.git
```

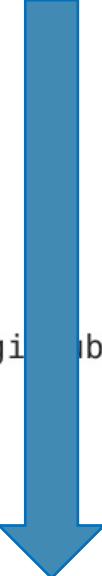


```
|      @c3cpu-a2-u32-3 ~]$ cd /scratch/alpine/$USER
|      @c3cpu-a2-u32-3 [f----o---]$ git clone https://github.com/kf-cuanschutz/Rstudio_repo_related_files.git
Cloning into 'Rstudio_repo_related_files'...
remote: Enumerating objects: 8, done.
remote: Counting objects: 100% (8/8), done.
remote: Compressing objects: 100% (7/7), done.
remote: Total 8 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (8/8), 2.33 KiB | 477.00 KiB/s, done.
|      @c3cpu-a2-u32-3          ]$ cd Rstudio_repo_related_files/
|      @c3cpu-a2-u32-3 Rstudio_repo_related_files]$ ls -la
total 08
drwxr-sr-x  3                               4096 Sep 21 17:22 .
drwxrws--- 484                             32768 Sep 21 17:22 ..
drwxr-sr-x  8                               4096 Sep 21 17:22 .git
-rw-r--r--  1                               91   Sep 21 17:22 README.md
-rw-r--r--  1                             508   Sep 21 17:22 step0_modify_overlay.sh
-rw-r--r--  1                             509   Sep 21 17:22 step0_r_v4_2_2_modify_overlay.sh
-rw-r--r--  1                             550   Sep 21 17:22 step1_install_os_deps.sh
[    @xsede.org@c3cpu-a2-u32-3 Rstudio_repo_related_files]$
```

# Rstudio library dependencies

- Then change directories into /scratch/alpine/\$USER/Rstudio\_repo\_related\_files

```
| @c3cpu-a2-u32-3 ~]$ cd /scratch/alpine/$USER
| @c3cpu-a2-u32-3 [~]$ git clone https://github.com/kf-cuanschutz/Rstudio_repo_related_files.git
Cloning into 'Rstudio_repo_related_files'...
remote: Enumerating objects: 8, done.
remote: Counting objects: 100% (8/8), done.
remote: Compressing objects: 100% (7/7), done.
remote: Total 8 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (8/8), 2.33 KiB | 477.00 KiB/s. done.
| @c3cpu-a2-u32-3 [~]$ cd Rstudio_repo_related_files/
| @c3cpu-a2-u32-3 Rstudio_repo_related_files]$ ls -la
total 68
drwxr-sr-x  3 4096 Sep 21 17:22 .
drwxrws--- 484 32768 Sep 21 17:22 ..
drwxr-sr-x  8 4096 Sep 21 17:22 .git
-rw-r--r--  1   91  Sep 21 17:22 README.md
-rw-r--r--  1  508  Sep 21 17:22 step0_modify_overlay.sh
-rw-r--r--  1  509  Sep 21 17:22 step0_r_v4_2_2_modify_overlay.sh
-rw-r--r--  1  550  Sep 21 17:22 step1_install_os_deps.sh
[...]
@xsede.org@c3cpu-a2-u32-3 Rstudio_repo_related_files]$
```



# Rstudio library dependencies

- The step0 files will enable to launch the containers.
- Step0\_modify\_overlay.sh corresponds to the Rstudio container 4.4.1
- Step0\_r\_v4\_2\_2\_modify\_overlay is a legacy version.

```
|          @c3cpu-a2-u32-3 ~]$ cd /scratch/alpine/$USER
|          @c3cpu-a2-u32-3 [~]$ git clone https://github.com/kf-cuanschu/Rstudio_repo_related_files.git
Cloning into 'Rstudio_repo_related_files'...
remote: Enumerating objects: 8, done.
remote: Counting objects: 100% (8/8), done.
remote: Compressing objects: 100% (7/7), done.
remote: Total 8 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (8/8), 2.33 KiB / 477.00 KiB/s, done.
|          @c3cpu-a2-u32-3                  ]$ cd Rstudio_repo_related_files/
|          @c3cpu-a2-u32-3 Rstudio_repo_related_files]$ ls -la
total 68
drwxr-sr-x  3                               4096 Sep 21 17:22
drwxrws--- 484                             32768 Sep 21 17:22
drwxr-sr-x  8                               4096 Sep 21 17:22
-rw-r--r--  1                               91   Sep 21 17:22 README.md
-rw-r--r--  1                               508  Sep 21 17:22 step0_modify_overlay.sh
-rw-r--r--  1                               509  Sep 21 17:22 step0_r_v4_2_2_modify_overlay.sh
-rw-r--r--  1                               550  Sep 21 17:22 step1_install_os_deps.sh
[...]
|          @xsede.org@c3cpu-a2-u32-3 Rstudio_repo_related_files]$
```



git

step0\_modify\_overlay.sh

step0\_r\_v4\_2\_2\_modify\_overlay.sh

step1\_install\_os\_deps.sh

# Rstudio library dependencies

- step1\_install\_os\_deps.sh is the main program that will install all the necessary library dependencies.

```
|          @c3cpu-a2-u32-3 ~]$ cd /scratch/alpine/$USER
|          @c3cpu-a2-u32-3 [~]$ git clone https://github.com/kf-cuanschu/Rstudio_repo_related_files.git
Cloning into 'Rstudio_repo_related_files'...
remote: Enumerating objects: 8, done.
remote: Counting objects: 100% (8/8), done.
remote: Compressing objects: 100% (7/7), done.
remote: Total 8 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (8/8), 2.33 KiB / 477.00 KiB/s, done.
|          @c3cpu-a2-u32-3           ]$ cd Rstudio_repo_related_files/
|          @c3cpu-a2-u32-3 Rstudio_repo_related_files]$ ls -la
total 68
drwxr-sr-x  3                               4096 Sep 21 17:22
drwxrws--- 484                             32768 Sep 21 17:22
drwxr-sr-x  8                               4096 Sep 21 17:22
-rw-r--r--  1                               91   Sep 21 17:22 ADME.md
-rw-r--r--  1                             508   Sep 21 17:22 step0_modify_overlay.sh
-rw-r--r--  1                             509   Sep 21 17:22 step0_r_v4_2_2_modify_overlay.sh
-rw-r--r--  1                             550   Sep 21 17:22 step1_install_os_deps.sh
[...]
@xsede.org@c3cpu-a2-u32-3 Rstudio_repo_related_files]$
```



# Rstudio library dependencies

- Let's make both step0 and step1 scripts executable.

```
[          @c3cpu-a2-u32-3 Rstudio_repo_related_files]$ chmod u+x step0_modify_overlay.sh
[          @c3cpu-a2-u32-3 Rstudio_repo_related_files]$ chmod u+x step1_install_os_deps.sh
[          @c3cpu-a2-u32-3 Rstudio_repo_related_files]$ ls
README.md  step0_modify_overlay.sh  step0_r_v4_2_2_modify_overlay.sh  step1_install_os_deps.sh
[          @xsede.org@c3cpu-a2-u32-3 Rstudio_repo_related_files]$ █
```

# Rstudio library dependencies

- Notice that now both scripts are green.

```
[          @c3cpu-a2-u32-3 Rstudio_repo_related_files]$ chmod u+x step0_modify_overlay.sh  
[          @c3cpu-a2-u32-3 Rstudio_repo_related_files]$ chmod u+x step1_install_os_deps.sh  
[          @c3cpu-a2-u32-3 Rstudio_repo_related_files]$ ls  
README.md step0_modify_overlay.sh step0_r_v4_2_2_modify_overlay.sh step1_install_os_deps.sh  
[          @xsede.org@c3cpu-a2-u32-3 Rstudio_repo_related_files]$ █
```



# Rstudio library dependencies

- Launch the container with “./step0\_modify\_overlay.sh”

```
[@xsede.org@c3cpu-a2-u32-3 Rstudio_repo_related_files]$ ./step0_modify_overlay.sh
DEBUG [U=2006460,P=1948245]setValue() Updated flag 'tmpdir' value to: /gpfs
/tmp
DEBUG [U=2006460,P=1948245]persistentPreRun() Apptainer version: 1.3.6-1.el8
DEBUG [U=2006460,P=1948245]persistentPreRun() Parsing configuration file /etc/appta
DEBUG [U=2006460,P=1948245]SetBinaryPath() Setting binary path to /usr/libexec/a
r/lpp/mmfs/bin:/usr/local/bin:/usr/bin:/usr/local/sbin:/usr/sbin:/usr/local/sbin:/usr/local/bin:
DEBUG [U=2006460,P=1948245]SetBinaryPath() Using that path for all binaries
DEBUG [U=2006460,P=1948245]handleConfDir() /home/kfotso@xsede.org/.apptainer alr
INFO [U=2006460,P=1948245]Exec() User not listed in /etc/subuid, tryin
DEBUG [U=2006460,P=1948245]UnshareRootMapped() Executing apptainer in root-mapped un
DEBUG [U=0,P=1948253] setValue() Updated flag 'tmpdir' value to: /gpfs/a
np
DEBUG [U=0,P=1948253] persistentPreRun() Apptainer version: 1.3.6-1.el8
DEBUG [U=0,P=1948253] persistentPreRun() Parsing configuration file /etc/apptain
DEBUG [U=0,P=1948253] SetBinaryPath() Setting binary path to /usr/libexec/app
lpp/mmfs/bin:/usr/local/bin:/usr/bin:/usr/local/sbin:/usr/sbin:/usr/local/sbin:/usr/local/bin:/u
DEBUG [U=0,P=1948253] SetBinaryPath() Using that path for all binaries
DEBUG [U=0,P=1948253] handleConfDir() /home/kfotso@xsede.org/.apptainer alrea
DEBUG [U=0,P=1948253] Exec() running root-mapped unprivileged
DEBUG [U=0,P=1948253] FindFake() looking for the fakeroot-sysv command
DEBUG [U=0,P=1948253] findOnPath() Found "fakeroot-sysv" at "/usr/bin/fake"
DEBUG [U=0,P=1948253] FindFake() fakeroot-sysv found at /usr/bin/fakeroo
INFO [U=0,P=1948253] Exec() Using fakeroot command combined with ro
DEBUG [U=0,P=1948253] setUmask() Saving umask 0022 for propagation into
DEBUG [U=0,P=1948253] checkEncryptionKey() Checking for encrypted system partition
DEBUG [U=0,P=1948253] Init() Image format detection
DEBUG [U=0,P=1948253] Init() Check for sandbox image format
```

# Rstudio library dependencies

- Now you can see that you are in the container, based on the prompt.

```
Apptainer> ./step1_install_os_deps.sh
Hit:1 http://archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Get:3 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
Get:4 http://archive.ubuntu.com/ubuntu jammy-backports InRelease [127 kB]
Get:5 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Packages [71.0 kB]
Get:6 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [3,642 kB]
Get:7 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [5,496 kB]
Get:8 http://archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [5,691 kB]
Get:9 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [3,327 kB]
Get:10 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1,581 kB]
Get:11 http://archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [69.1 kB]
Get:12 http://archive.ubuntu.com/ubuntu jammy-backports/main amd64 Packages [83.2 kB]
Get:13 http://archive.ubuntu.com/ubuntu jammy-backports/universe amd64 Packages [35.2 kB]
Get:14 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [1,274 kB]
Fetched 21.7 MB in 4s (5,430 kB/s)
Reading package lists... Done
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  zlib1g-dev
0 upgraded, 1 newly installed, 0 to remove and 110 not upgraded.
```

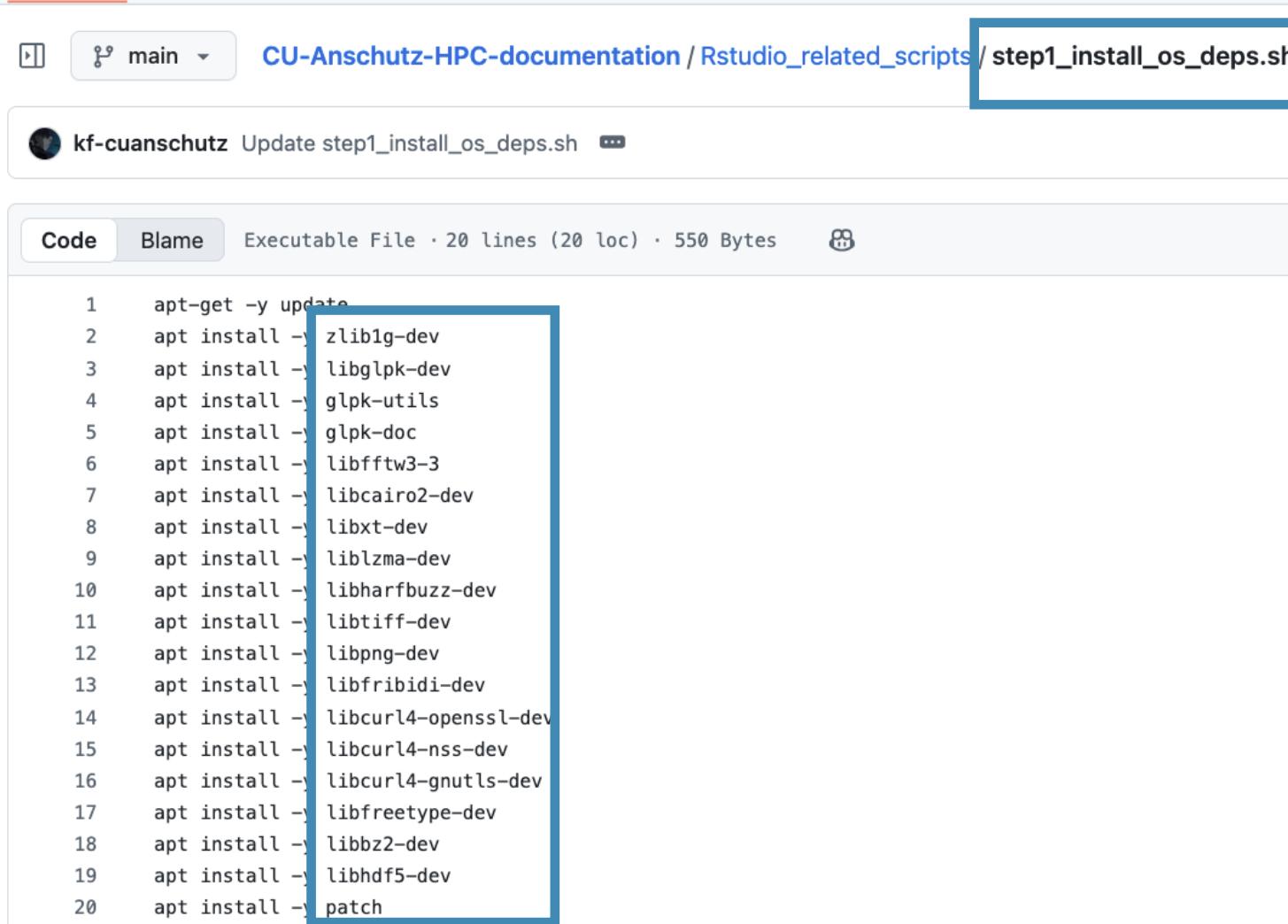
# Rstudio library dependencies

- Install the library dependencies with “.step1\_install\_os\_deps.sh”

```
Apptainer> ./step1_install_os_deps.sh
Hit:1 http://archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Get:3 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
Get:4 http://archive.ubuntu.com/ubuntu jammy-backports InRelease [127 kB]
Get:5 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Packages [71.0 kB]
Get:6 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [3,642 kB]
Get:7 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [5,496 kB]
Get:8 http://archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [5,691 kB]
Get:9 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [3,327 kB]
Get:10 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1,581 kB]
Get:11 http://archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [69.1 kB]
Get:12 http://archive.ubuntu.com/ubuntu jammy-backports/main amd64 Packages [83.2 kB]
Get:13 http://archive.ubuntu.com/ubuntu jammy-backports/universe amd64 Packages [35.2 kB]
Get:14 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [1,274 kB]
Fetched 21.7 MB in 4s (5,430 kB/s)
Reading package lists... Done
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  zlib1g-dev
0 upgraded, 1 newly installed, 0 to remove and 110 not upgraded.
```

# step1\_install\_os.deps.sh script

You can see the list of dependencies it installs below.



The screenshot shows a GitHub code editor interface. At the top, there's a navigation bar with a repository icon, a dropdown menu set to 'main', and the full repository path: 'CU-Anschutz-HPC-documentation / Rstudio\_related\_scripts / step1\_install\_os\_deps.sh'. A blue rectangular box highlights the repository path. Below the navigation bar, a commit message from 'kf-cuanschutz' is shown: 'Update step1\_install\_os\_deps.sh'. Underneath the commit message, there are two tabs: 'Code' (which is selected) and 'Blame'. To the right of the tabs, it says 'Executable File · 20 lines (20 loc) · 550 Bytes'. On the far right of the header, there's a small octocat icon. The main area contains the content of the script, with each line numbered from 1 to 20. Lines 1 through 20 are all 'apt install -y' commands followed by a package name. A second blue rectangular box highlights the first 19 lines of the script, starting from 'apt-get -y update' and ending with 'patch'. The packages listed are: zlib1g-dev, libglpk-dev, glpk-utils, glpk-doc, libfftw3-3, libcairo2-dev, libxt-dev, liblzma-dev, libharfbuzz-dev, libtiff-dev, libpng-dev, libfribidi-dev, libcurl4-openssl-dev, libcurl4-nss-dev, libcurl4-gnutls-dev, libfreetype-dev, libbz2-dev, libhdf5-dev, and patch.

```
1 apt-get -y update
2 apt install -y zlib1g-dev
3 apt install -y libglpk-dev
4 apt install -y glpk-utils
5 apt install -y glpk-doc
6 apt install -y libfftw3-3
7 apt install -y libcairo2-dev
8 apt install -y libxt-dev
9 apt install -y liblzma-dev
10 apt install -y libharfbuzz-dev
11 apt install -y libtiff-dev
12 apt install -y libpng-dev
13 apt install -y libfribidi-dev
14 apt install -y libcurl4-openssl-dev
15 apt install -y libcurl4-nss-dev
16 apt install -y libcurl4-gnutls-dev
17 apt install -y libfreetype-dev
18 apt install -y libbz2-dev
19 apt install -y libhdf5-dev
20 apt install -y patch
```

# step1\_install\_os.deps.sh script

- Once the installation is done, make sure to exit the container and the interactive session!

```
debconf: delaying package configuration, since apt-utils is not ins  
Selecting previously unselected package patch.  
(Reading database ... 51738 files and directories currently install  
Preparing to unpack .../patch_2.7.6-7build2_amd64.deb ...  
Unpacking patch (2.7.6-7build2) ...  
Setting up patch (2.7.6-7build2) ...  
Apptainer> exit  
exit  
DEBUG [U=0,P=1948253] CleanupContainer()  
DEBUG [U=0,P=1948253] umount()  
DEBUG [U=0,P=1948253] stop()  
DEBUG [U=0,P=1948253] umount()  
DEBUG [U=0,P=1948253] stop()  
DEBUG [U=0,P=1948253] umount()  
DEBUG [U=0,P=1948253] umount()  
DEBUG [U=0,P=1948253] stop()  
DEBUG [U=0,P=1948253] Master()  
cleanup co  
Unmount /va  
Waiting fo  
Unmount /va  
Waiting fo  
Sync of /v  
Unmount /va  
Waiting fo  
Child_exit  
@xsede.org@c3cpu-a2-u32-3 Rstudio_repo_related_files]$ exit  
logout  
Connection to c3cpu-a2-u32-3 closed.  
@xsede.org@login-ci1 ~]$
```

# Install a package

- We go back to Ondemand and start a new Rstudio session

The screenshot shows the Ondemand interface. At the top, there's a navigation bar with 'Interactive Apps' and 'My Interactive Sessions'. Below it, a success message 'Session was successfully created.' is displayed. The main content area shows a list of interactive apps. A specific session is highlighted: 'RStudio Server (Presets) (4927252)'. This session is running on 1 node with 4 cores. It was created at 2024-02-12 16:07:24 MST and has 3 hours and 59 minutes remaining. The session ID is b69b7ba1-0363-4bd6-a203-9e4288e0a787. A red box highlights the 'Connect to RStudio Server' button.

Session was successfully created.

Home / My Interactive Sessions

Interactive Apps

- Desktops
- Core Desktop (Presets)
- GUIs
- MATLAB (Presets)
- Servers
- Jupyter Session (Custom)
- Jupyter Session (Presets)
- RStudio Server (Custom)
- RStudio Server (Presets)
- VS Code-Server (Custom)
- VS Code-Server (Presets)

RStudio Server (Presets) (4927252)

1 node | 4 cores | Running

Host: >\_c3cpu-c11-u1-4.rc.int.colorado.edu

Created at: 2024-02-12 16:07:24 MST

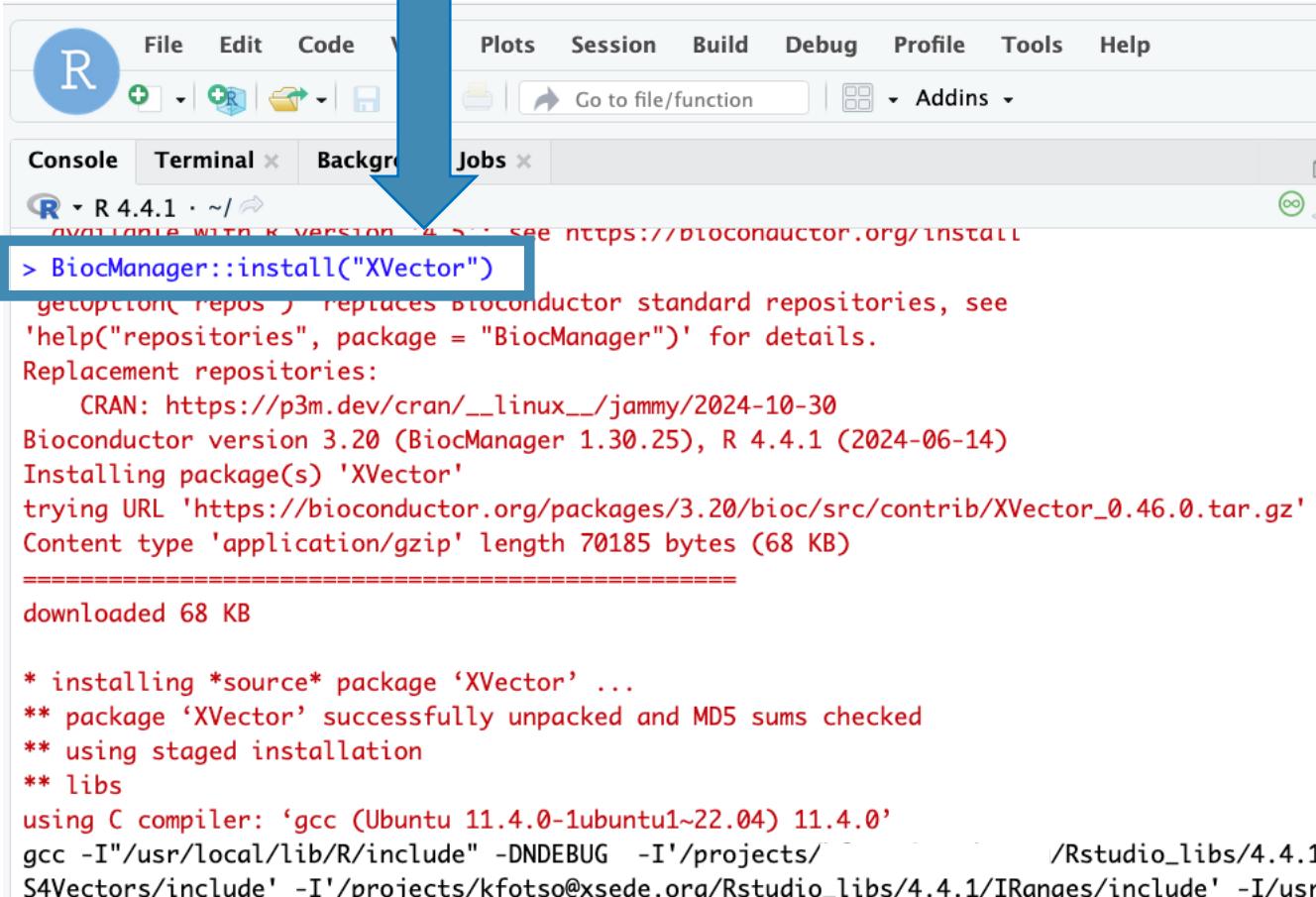
Time Remaining: 3 hours and 59 minutes

Session ID: b69b7ba1-0363-4bd6-a203-9e4288e0a787

Connect to RStudio Server

# Install a package

- Now we can install "Xvector"!



```
> BiocManager::install("XVector")
getOption('repos') replaces Bioconductor standard repositories, see
'help("repositories", package = "BiocManager")' for details.
Replacement repositories:
CRAN: https://p3m.dev/cran/_linux_/jammy/2024-10-30
Bioconductor version 3.20 (BiocManager 1.30.25), R 4.4.1 (2024-06-14)
Installing package(s) 'XVector'
trying URL 'https://bioconductor.org/packages/3.20/bioc/src/contrib/XVector_0.46.0.tar.gz'
Content type 'application/gzip' length 70185 bytes (68 KB)
=====
downloaded 68 KB

* installing *source* package 'XVector' ...
** package 'XVector' successfully unpacked and MD5 sums checked
** using staged installation
** libs
using C compiler: 'gcc (Ubuntu 11.4.0-1ubuntu1~22.04) 11.4.0'
gcc -I"/usr/local/lib/R/include" -DNDEBUG -I'/projects/.../Rstudio_libs/4.4.1/
S4Vectors/include' -I'/projects/kfotso@xsede.org/Rstudio_libs/4.4.1/IRanges/include' -I/usr/
local/include -fpic -g -O2 -fstack-protector-strong -Wformat -Werror=format-security -Wd
ate-time -D_FORTIFY_SOURCE=2 -g -c IRanges_stubs.c -o IRanges_stubs.o
gcc -I"/usr/local/lib/R/include" -DNDEBUG -I'/projects/.../Rstudio_libs/4.4.1/
S4Vectors/include' -I'/projects/kfotso@xsede.org/Rstudio_libs/4.4.1/IRanges/include' -I/usr/
local/include -fpic -g -O2 -fstack-protector-strong -Wformat -Werror=format-security -Wd
ate-time -D_FORTIFY_SOURCE=2 -g -c IRanges_stubs.c -o IRanges_stubs.o
```

# Install a package

- Now we can install "Xvector"!

```
vector_copy.o vapply_summarization_methods.o -lz -L/usr/local/lib  
installing to /projects/...@xsede.org/Rstudio_libs/4.4.1/lib  
s  
** R  
** inst  
** byte-compile and prepare package for lazy loading  
** help  
*** installing help indices  
** building package indices  
** testing if installed package can be loaded from temporary  
** checking absolute paths in shared objects and dynamic libr  
** testing if installed package can be loaded from final loca  
** testing if installed package keeps a record of temporary i  
* DONE (XVector)
```

The downloaded source packages are in

```
'/projects/...@xsede.org/.rstudioserver/rstudio-4.  
ded_packages'  
Old packages: 'boot', 'foreign', 'MASS', 'Matrix', 'nlme', 's  
Update all/some/none? [a/s/n]:
```

# Install a package

Please refer to this guide for more details:

[https://github.com/kf-cuanschutz/CU-Anschutz-HPC-documentation/tree/main/Rstudio\\_related\\_scripts](https://github.com/kf-cuanschutz/CU-Anschutz-HPC-documentation/tree/main/Rstudio_related_scripts)

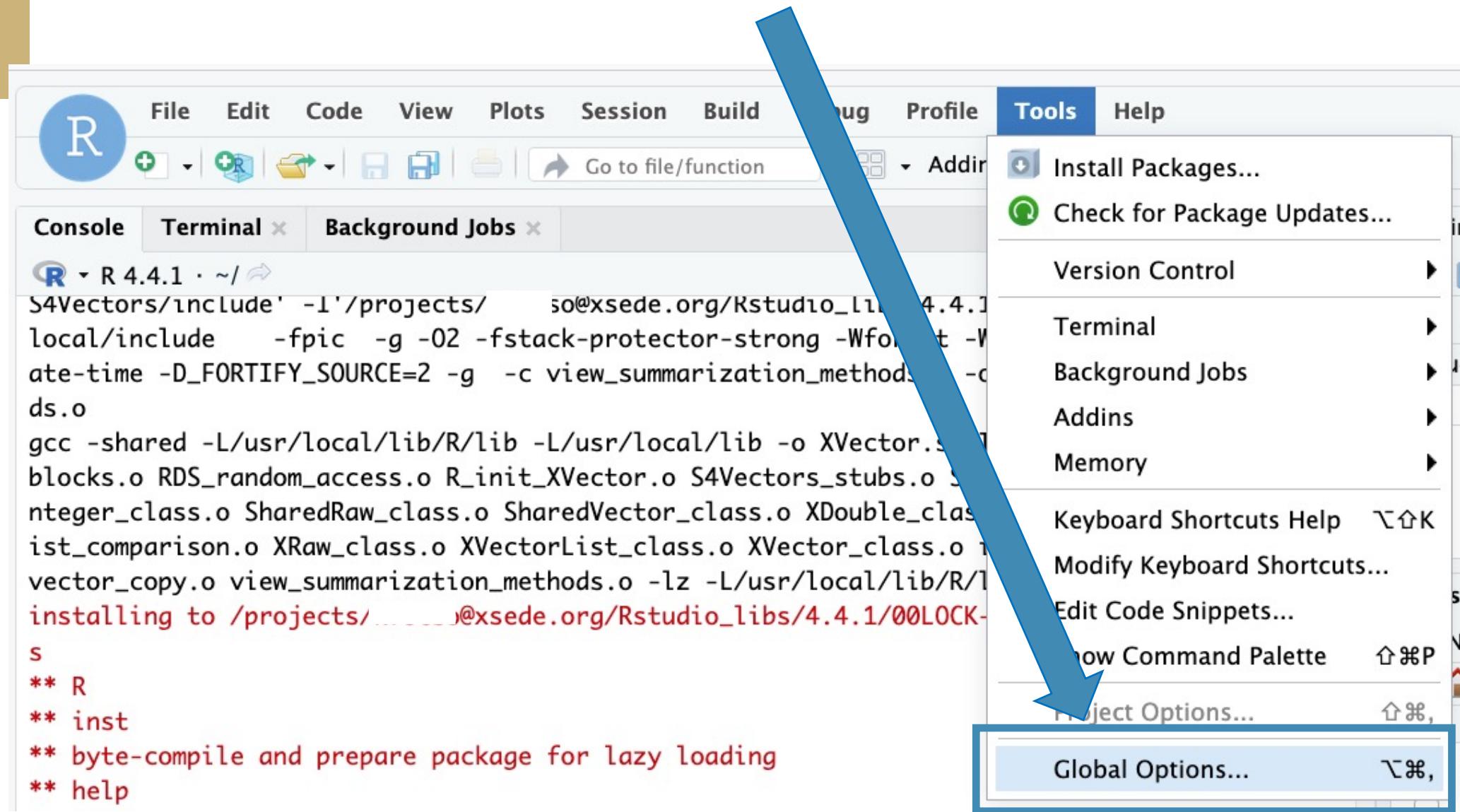
```
vector_copy.o view_summarization_methods.o -lz -L/usr/local/lib  
installing to /projects/`whoami`@xsede.org/Rstudio_libs/4.4.1/lib  
S  
** R  
** inst  
** byte-compile and prepare package for lazy loading  
** help  
*** installing help indices  
** building package indices  
** testing if installed package can be loaded from temporary  
** checking absolute paths in shared objects and dynamic libraries  
** testing if installed package can be loaded from final location  
** testing if installed package keeps a record of temporary indices  
* DONE (XVector)
```

The downloaded source packages are in

```
'/projects/`whoami`@xsede.org/.rstudioserver/rstudio-4.  
ded_packages'  
Old packages: 'boot', 'foreign', 'MASS', 'Matrix', 'nlme', 's  
Update all/some/none? [a/s/n]:
```

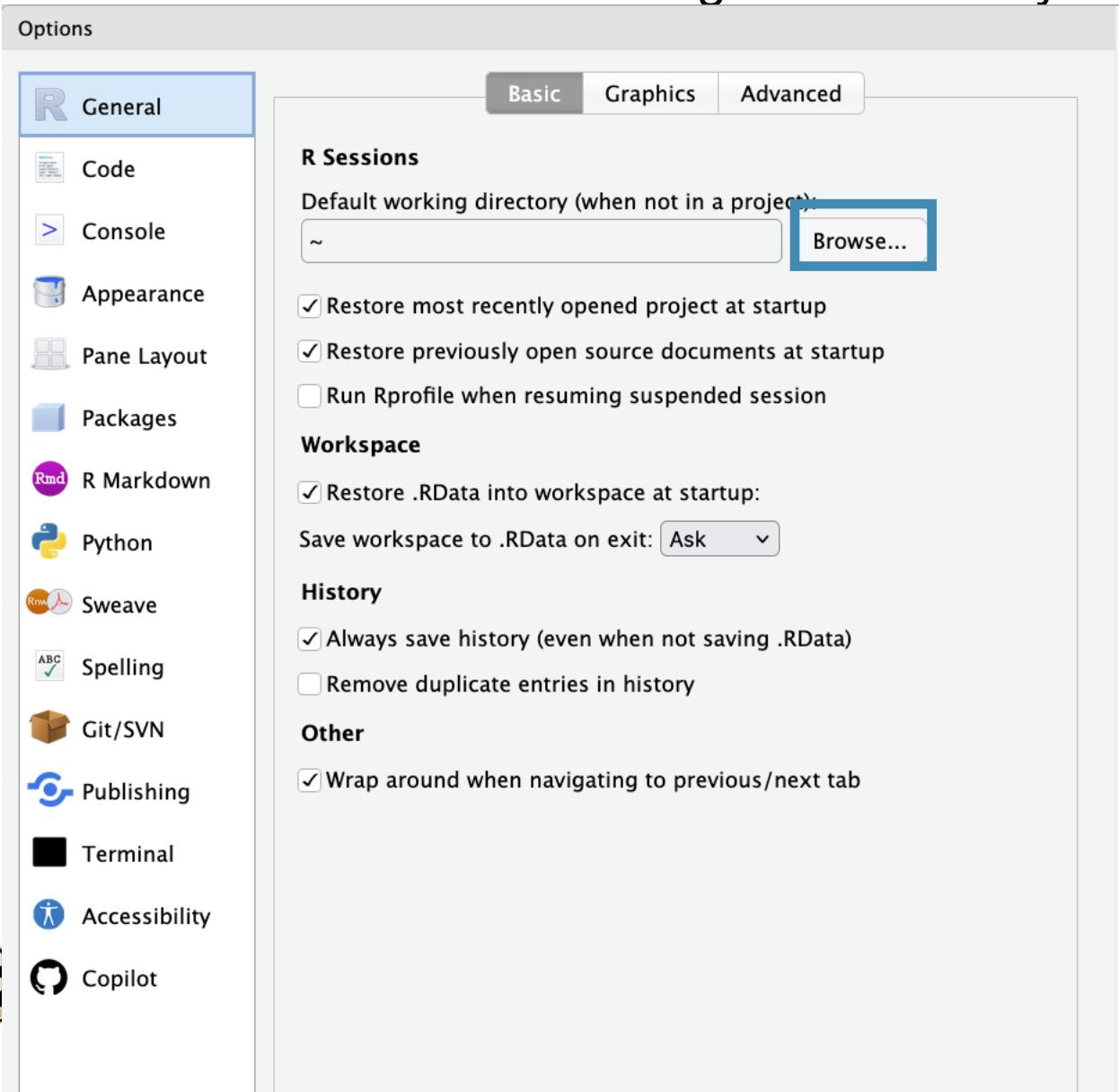
# Change my working directory

- Select Tools -> Global Options



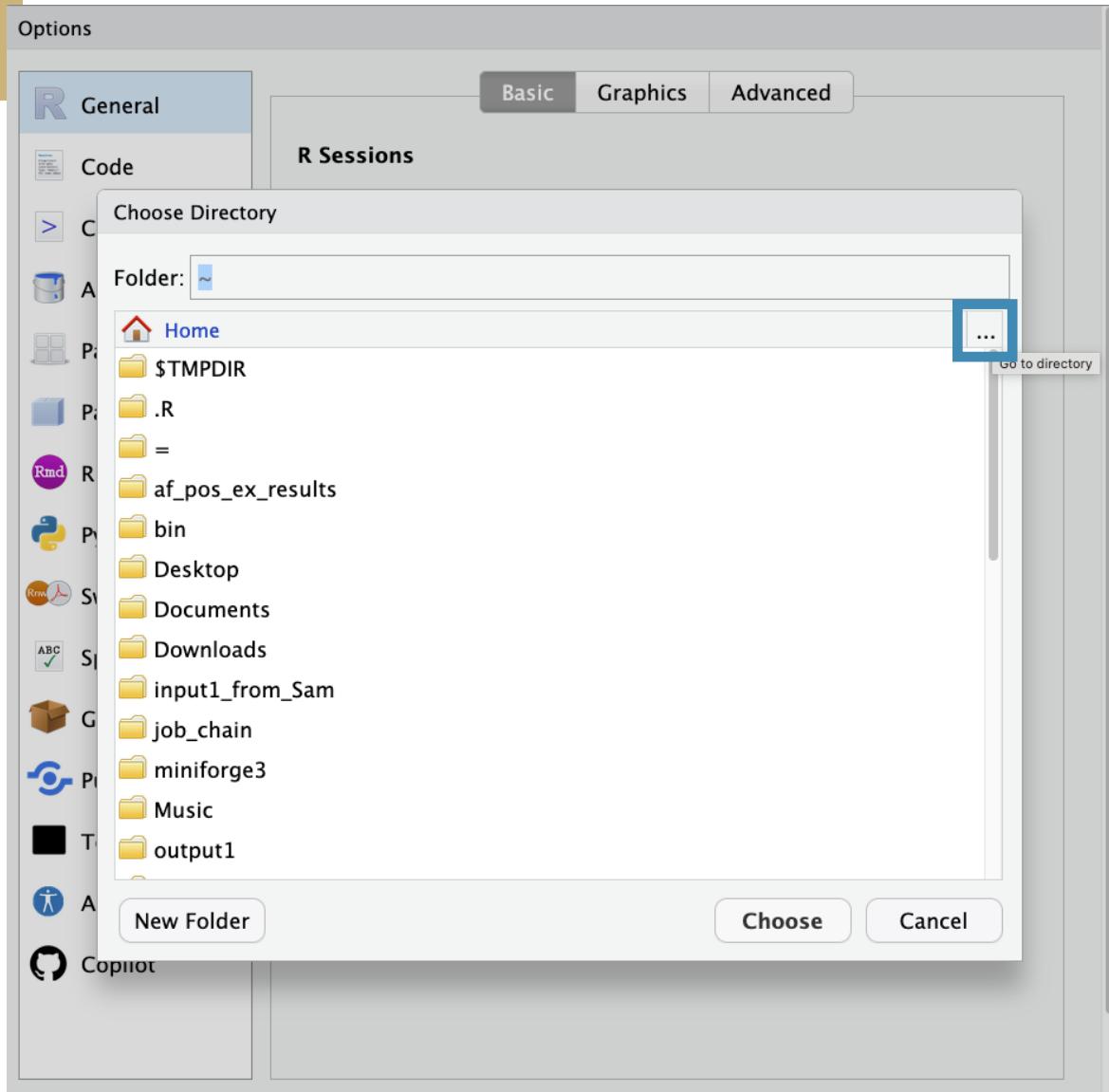
# Change my working directory

- Select “Browse” to change the directory.



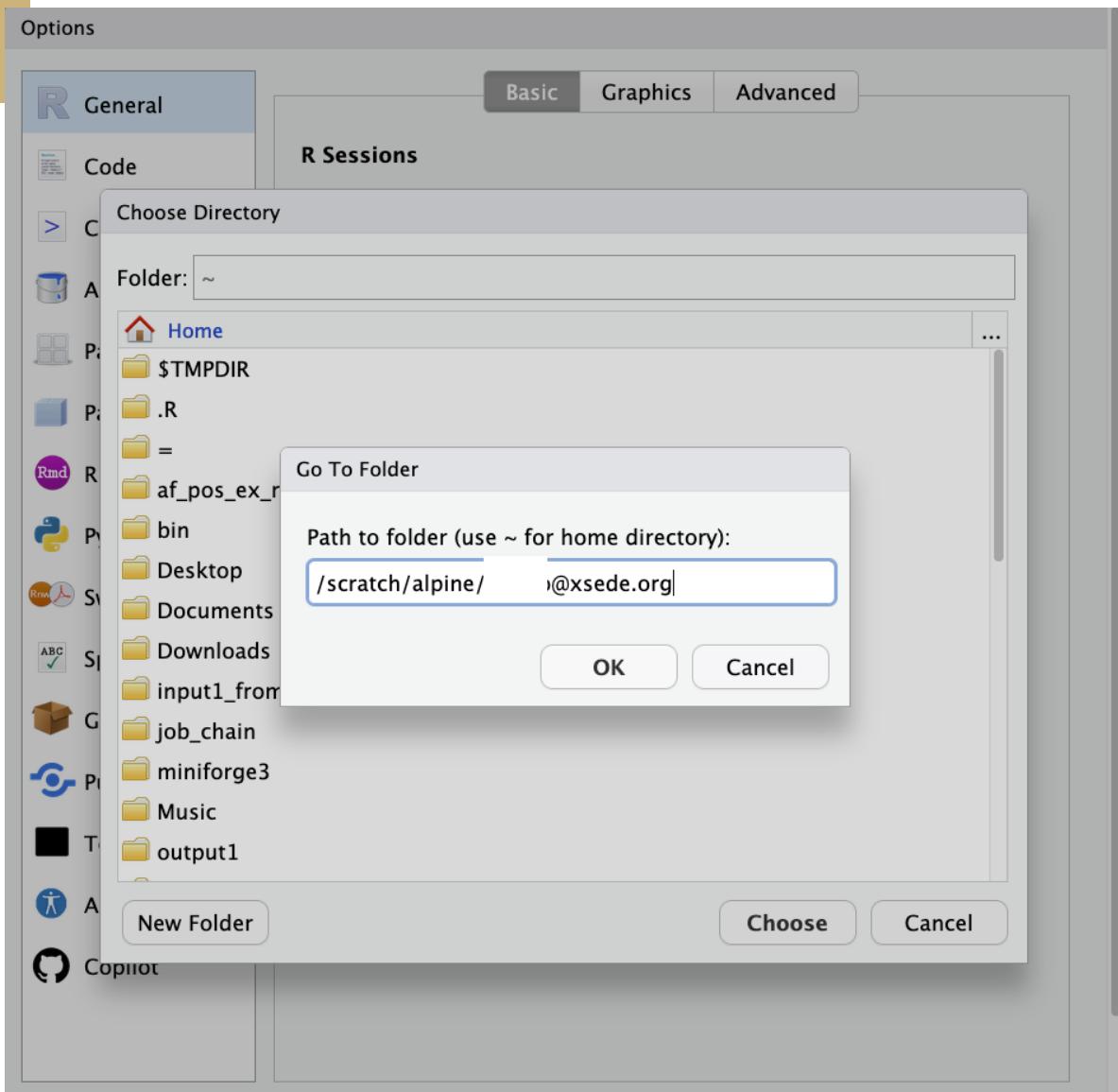
# Change my working directory

- Select the three dots on the right to enter your directory manually.



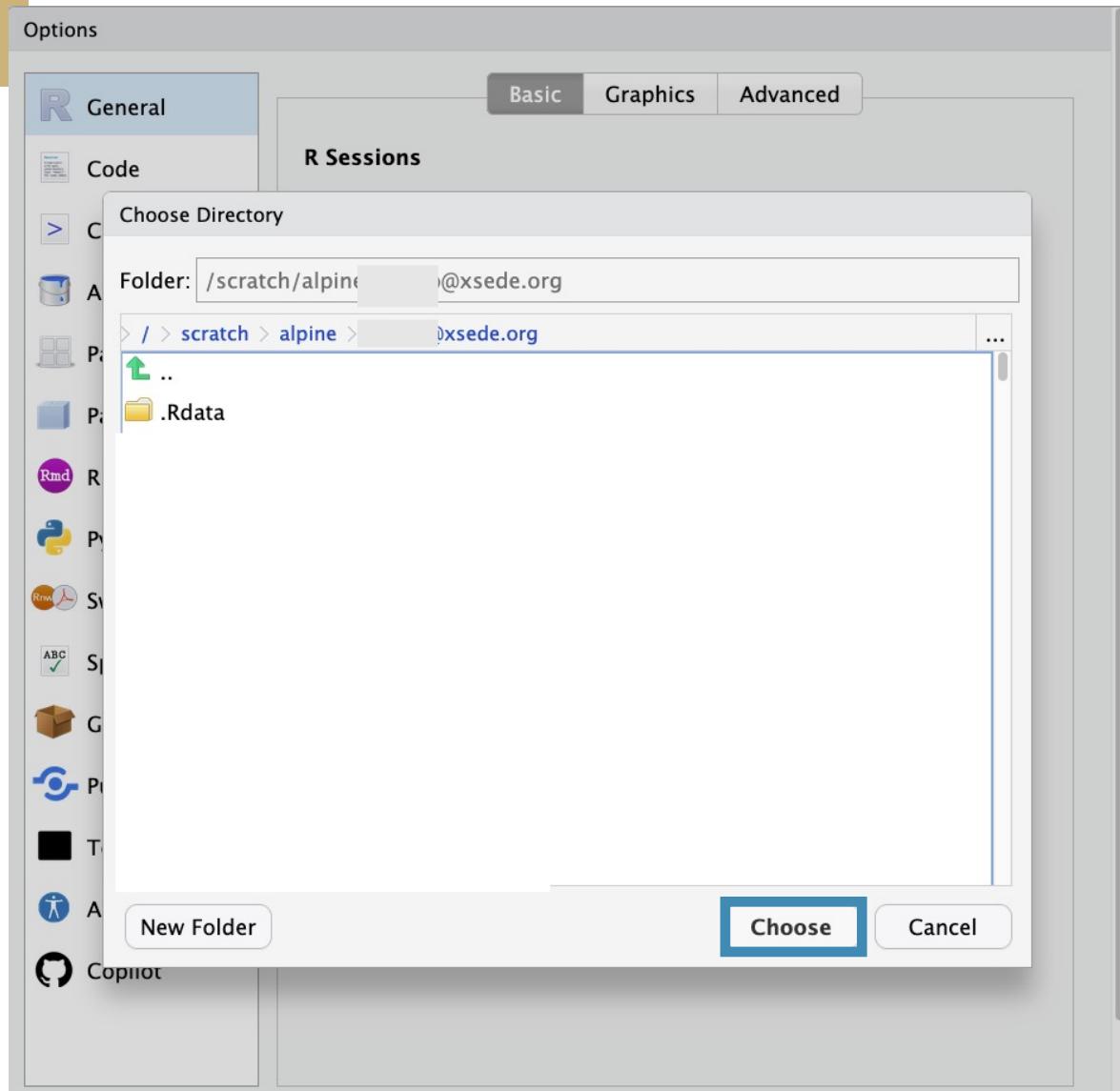
# Change my working directory

- Here, I enter my scratch directory as my working directory.



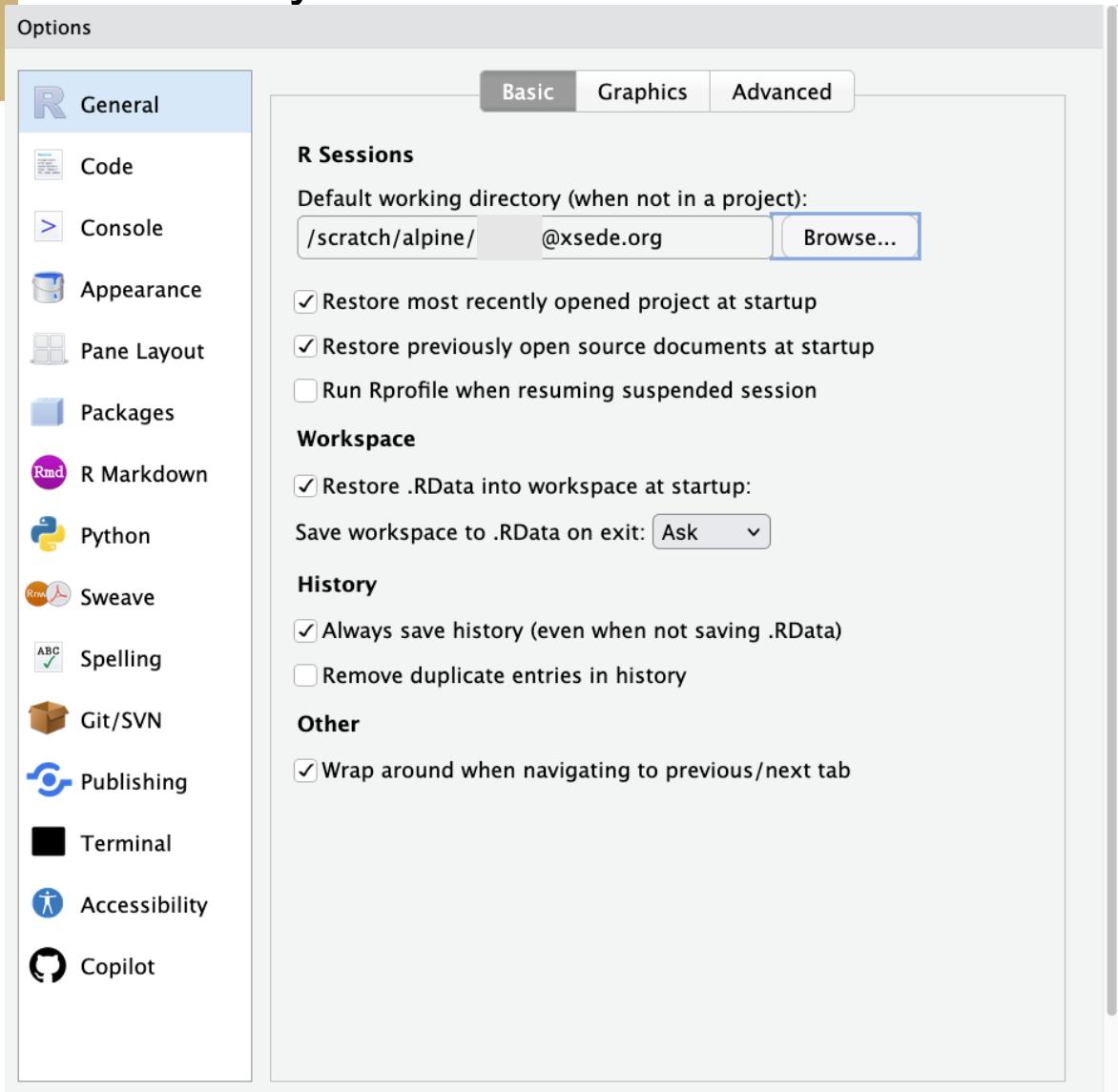
# Change my working directory

- Select “Choose”.



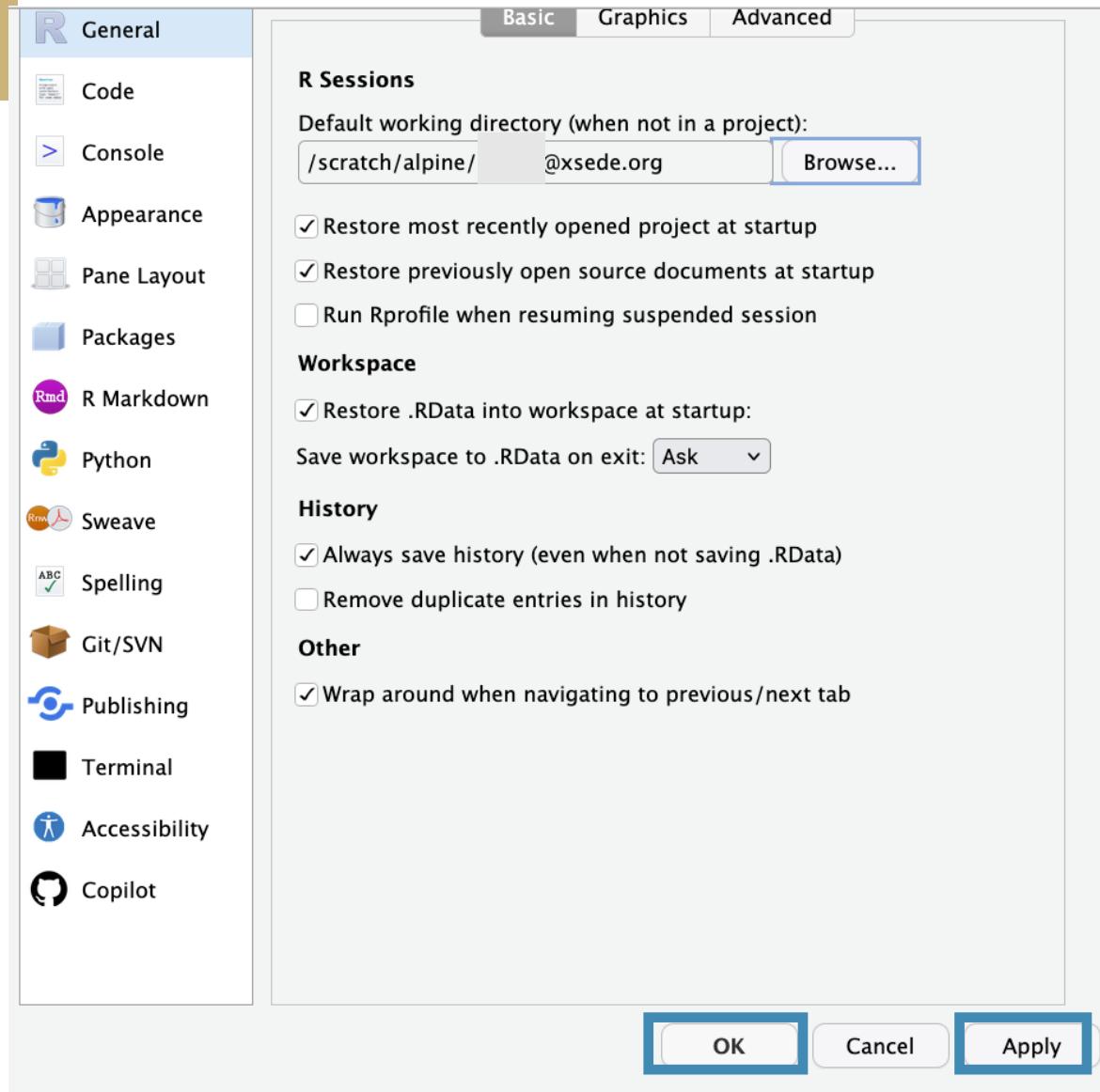
# Change my working directory

- The screen below confirms that we've selected the scratch directory as the working directory.



# Change my working directory

- Select “Apply” and then “OK” to save the change.



# Slurm script with Rstudio environment

- If you need more RAM (>60G).
- If you need more cores (>16).
- If you need to reuse your R environment for longer walltime (>12 hours).

# Slurm script with Rstudio environment

```
#!/bin/bash
#SBATCH --nodes=1
#SBATCH --time=00:00:05
#SBATCH --qos=normal
#SBATCH --partition=amilan
#SBATCH --ntasks=1
#SBATCH --account=amc-general
#SBATCH --job-name=R_job
#SBATCH --output=Rjob.%j.out
#SBATCH --error=Rjob.%j.err
#SBATCH --mail-user=foo@cuanschutz.edu
#SBATCH --mail-type=BEGIN,END,FAIL
```

Number of nodes. **Always use 1** unless you are using MPI, gnu parallel or sparkcluster!

```
# Temporar█ directories
export ALPINE_SCRATCH=/gpfs/alpine1/scratch/$USER
export APPTAINER_TMPDIR=$ALPINE_SCRATCH/apptainer/tmp
export APPTAINER_CACHEDIR=$ALPINE_SCRATCH/apptainer/cache
mkdir -pv $APPTAINER_CACHEDIR $APPTAINER_TMPDIR
```

# Slurm script with Rstudio environment

```
#!/bin/bash
#SBATCH --nodes=1
#SBATCH --time=00:00:05 ← Duration of the slurm job.
#SBATCH --qos=normal
#SBATCH --partition=amilan
#SBATCH --ntasks=1
#SBATCH --account=amc-general
#SBATCH --job-name=R_job
#SBATCH --output=Rjob.%j.out
#SBATCH --error=Rjob.%j.err
#SBATCH --mail-user=foo@cuanschutz.edu
#SBATCH --mail-type=BEGIN,END,FAIL
```

```
# Temporar█ directories
export ALPINE_SCRATCH=/gpfs/alpine1/scratch/$USER
export APPTAINER_TMPDIR=$ALPINE_SCRATCH/apptainer/tmp
export APPTAINER_CACHEDIR=$ALPINE_SCRATCH/apptainer/cache
mkdir -pv $APPTAINER_CACHEDIR $APPTAINER_TMPDIR
```

# Slurm script with Rstudio environment

```
#!/bin/bash
#SBATCH --nodes=1
#SBATCH --time=00:00:05
#SBATCH --qos=normal
#SBATCH --partition=amilan
#SBATCH --ntasks=1
#SBATCH --account=amc-general
#SBATCH --job-name=R_job
#SBATCH --output=Rjob.%j.out
#SBATCH --error=Rjob.%j.err
#SBATCH --mail-user=foo@cuanschutz.edu
#SBATCH --mail-type=BEGIN,END,FAIL
```

Quality of service

```
# Temporar█ directories
export ALPINE_SCRATCH=/gpfs/alpine1/scratch/$USER
export APPTAINER_TMPDIR=$ALPINE_SCRATCH/apptainer/tmp
export APPTAINER_CACHEDIR=$ALPINE_SCRATCH/apptainer/cache
mkdir -pv $APPTAINER_CACHEDIR $APPTAINER_TMPDIR
```

# Slurm Quality of service (qos)

- Used to modify or constrain characteristics that a job can have.
- **--qos=normal** corresponds to a walltime of 24 hours and is the default.
- **--qos=long** corresponds to a walltime of up to 7 days
- **--qos=mem** corresponds to high memory jobs only (up to 2TB)



# Slurm Quality of service (qos)

- Used to modify or constrain characteristics that a job can have.
- **--qos=normal** corresponds to a walltime of 24 hours and is the default.
- **--qos=long** corresponds to a walltime of up to 7 days
- **--qos=mem** corresponds to high memory jobs only (up to 2TB)
- It is now mandatory to use **#SBATCH --qos** when interacting with Slurm!



# Slurm script with Rstudio environment

```
#!/bin/bash
#SBATCH --nodes=1
#SBATCH --time=00:00:05
#SBATCH --qos=normal
#SBATCH --partition=amilan
#SBATCH --ntasks=1
#SBATCH --account=amc-general
#SBATCH --job-name=R_job
#SBATCH --output=Rjob.%j.out
#SBATCH --error=Rjob.%j.err
#SBATCH --mail-user=foo@cuanschutz.edu
#SBATCH --mail-type=BEGIN,END,FAIL
```



Main CPU partition name

```
# Temporar█ directories
export ALPINE_SCRATCH=/gpfs/alpine1/scratch/$USER
export APPTAINER_TMPDIR=$ALPINE_SCRATCH/apptainer/tmp
export APPTAINER_CACHEDIR=$ALPINE_SCRATCH/apptainer/cache
mkdir -pv $APPTAINER_CACHEDIR $APPTAINER_TMPDIR
```

# Partitions on Alpine

- amilan -> CPU
- amem -> High memory
- aa100 -> NVIDIA GPU partition
- al40 -> NVIDIA GPU partition
- ami100 -> AMD GPU partition

# Slurm script with Rstudio environment

```
#!/bin/bash
#SBATCH --nodes=1
#SBATCH --time=00:00:05
#SBATCH --qos=normal
#SBATCH --partition=amilan
#SBATCH --ntasks=1 ← Number of CPU cores.
#SBATCH --account=amc-general
#SBATCH --job-name=R_job
#SBATCH --output=Rjob.%j.out
#SBATCH --error=Rjob.%j.err
#SBATCH --mail-user=foo@cuanschutz.edu
#SBATCH --mail-type=BEGIN,END,FAIL
```

```
# Temporar█ directories
export ALPINE_SCRATCH=/gpfs/alpine1/scratch/$USER
export APPTAINER_TMPDIR=$ALPINE_SCRATCH/apptainer/tmp
export APPTAINER_CACHEDIR=$ALPINE_SCRATCH/apptainer/cache
mkdir -pv $APPTAINER_CACHEDIR $APPTAINER_TMPDIR
```

# Slurm script with Rstudio environment

```
#!/bin/bash
#SBATCH --nodes=1
#SBATCH --time=00:00:05
#SBATCH --qos=normal
#SBATCH --partition=amilan
#SBATCH --ntasks=1
#SBATCH --account=amc-general
#SBATCH --job-name=R_job
#SBATCH --output=Rjob.%j.out
#SBATCH --error=Rjob.%j.err
#SBATCH --mail-user=foo@cuanschutz.edu
#SBATCH --mail-type=BEGIN,END,FAIL
```

Account name. All AMC users should use the **amc-general account!**

```
# Temporar█ directories
export ALPINE_SCRATCH=/gpfs/alpine1/scratch/$USER
export APPTAINER_TMPDIR=$ALPINE_SCRATCH/apptainer/tmp
export APPTAINER_CACHEDIR=$ALPINE_SCRATCH/apptainer/cache
mkdir -pv $APPTAINER_CACHEDIR $APPTAINER_TMPDIR
```

# Slurm script with Rstudio environment

```
#!/bin/bash
#SBATCH --nodes=1
#SBATCH --time=00:00:05
#SBATCH --qos=normal
#SBATCH --partition=amilan
#SBATCH --ntasks=1
#SBATCH --account=amc-general
#SBATCH --job-name=R_job
#SBATCH --output=Rjob.%j.out
#SBATCH --error=Rjob.%j.err
#SBATCH --mail-user=foo@cuanschutz.edu
#SBATCH --mail-type=BEGIN,END,FAIL
```

Job name



```
# Temporar█ directories
export ALPINE_SCRATCH=/gpfs/alpine1/scratch/$USER
export APPTAINER_TMPDIR=$ALPINE_SCRATCH/apptainer/tmp
export APPTAINER_CACHEDIR=$ALPINE_SCRATCH/apptainer/cache
mkdir -pv $APPTAINER_CACHEDIR $APPTAINER_TMPDIR
```

# Slurm script with Rstudio environment

```
#!/bin/bash
#SBATCH --nodes=1
#SBATCH --time=00:00:05
#SBATCH --qos=normal
#SBATCH --partition=amilan
#SBATCH --ntasks=1
#SBATCH --account=amc-general
#SBATCH --job-name=R_job
#SBATCH --output=Rjob.%j.out
#SBATCH --error=Rjob.%j.err
#SBATCH --mail-user=foo@cuanschutz.edu
#SBATCH --mail-type=BEGIN,END,FAIL
```

- 
- Job output file.
  - Similar to stdout in linux.

```
# Temporar█ directories
export ALPINE_SCRATCH=/gpfs/alpine1/scratch/$USER
export APPTAINER_TMPDIR=$ALPINE_SCRATCH/apptainer/tmp
export APPTAINER_CACHEDIR=$ALPINE_SCRATCH/apptainer/cache
mkdir -pv $APPTAINER_CACHEDIR $APPTAINER_TMPDIR
```

# Slurm script with Rstudio environment

```
#!/bin/bash
#SBATCH --nodes=1
#SBATCH --time=00:00:05
#SBATCH --qos=normal
#SBATCH --partition=amilan
#SBATCH --ntasks=1
#SBATCH --account=amc-general
#SBATCH --job-name=R_job
#SBATCH --output=Rjob.%j.out
#SBATCH --error=Rjob.%j.err
#SBATCH --mail-user=foo@cuanschutz.edu
#SBATCH --mail-type=BEGIN,END,FAIL
```



%j for the jobID  
number

```
# Temporar█ directories
export ALPINE_SCRATCH=/gpfs/alpine1/scratch/$USER
export APPTAINER_TMPDIR=$ALPINE_SCRATCH/apptainer/tmp
export APPTAINER_CACHEDIR=$ALPINE_SCRATCH/apptainer/cache
mkdir -pv $APPTAINER_CACHEDIR $APPTAINER_TMPDIR
```

# Slurm script with Rstudio environment

```
#!/bin/bash
#SBATCH --nodes=1
#SBATCH --time=00:00:05
#SBATCH --qos=normal
#SBATCH --partition=amilan
#SBATCH --ntasks=1
#SBATCH --account=amc-general
#SBATCH --job-name=R_job
#SBATCH --output=Rjob.%j.out
#SBATCH --error=Rjob.%j.err
#SBATCH --mail-user=foo@cuanschutz.edu
#SBATCH --mail-type=BEGIN,END,FAIL
```

Job error file.

```
# Temporar█ directories
export ALPINE_SCRATCH=/gpfs/alpine1/scratch/$USER
export APPTAINER_TMPDIR=$ALPINE_SCRATCH/apptainer/tmp
export APPTAINER_CACHEDIR=$ALPINE_SCRATCH/apptainer/cache
mkdir -pv $APPTAINER_CACHEDIR $APPTAINER_TMPDIR
```

# Slurm script with Rstudio environment

```
#!/bin/bash
#SBATCH --nodes=1
#SBATCH --time=00:00:05
#SBATCH --qos=normal
#SBATCH --partition=amilan
#SBATCH --ntasks=1
#SBATCH --account=amc-general
#SBATCH --job-name=R_job
#SBATCH --output=Rjob.%j.out
#SBATCH --error=Rjob.%j.err
#SBATCH --mail-user=foo@cuanschutz.edu
#SBATCH --mail-type=BEGIN,END,FAIL
```



Email address and request to receive a notification.

```
# Temporar█ directories
export ALPINE_SCRATCH=/gpfs/alpine1/scratch/$USER
export APPTAINER_TMPDIR=$ALPINE_SCRATCH/apptainer/tmp
export APPTAINER_CACHEDIR=$ALPINE_SCRATCH/apptainer/cache
mkdir -pv $APPTAINER_CACHEDIR $APPTAINER_TMPDIR
```

# Slurm script with Rstudio environment

```
#!/bin/bash
#SBATCH --nodes=1
#SBATCH --time=00:00:05
#SBATCH --qos=normal
#SBATCH --partition=amilan
#SBATCH --ntasks=1
#SBATCH --account=amc-general
#SBATCH --job-name=R_job
#SBATCH --output=Rjob.%j.out
#SBATCH --error=Rjob.%j.err
#SBATCH --mail-user=foo@cuanschutz.edu
#SBATCH --mail-type=BEGIN,END,FAIL
```

```
# Temporar█ directories
export ALPINE_SCRATCH=/gpfs/alpine1/scratch/$USER
export APPTAINER_TMPDIR=$ALPINE_SCRATCH/apptainer/tmp
export APPTAINER_CACHEDIR=$ALPINE_SCRATCH/apptainer/cache
mkdir -pv $APPTAINER_CACHEDIR $APPTAINER_TMPDIR
```

We need to redirect the temporary files to the scratch filesystem.

# Slurm script with Rstudio environment

“\${r\_app\_version}.sif” is the name of the container image.

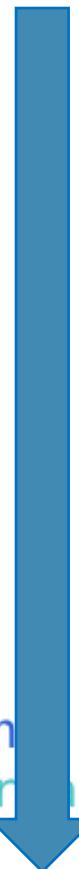


```
# Current R version is 4.4.1
export r_app_version="4.4.1"

apptainer exec --bind /projects,/scratch/alpine,$CURC_CONTAINER_DIR_00D \
    --fakeroot \
    --overlay \
    /projects/$USER/.rstudioserver/rstudio-${r_app_version}/rstudio-server-${r_app_version}_overlay.img:ro \
    /curc/sw/containers/open_ondemand/rstudio-server-${r_app_version}.sif \
    Rscript R_test.R
```

# Slurm script with Rstudio environment

- “exec” is going to run the container.



```
# Current R version is 4.4.1
export r_app_version="4.4.1"

apptainer exec --bind /projects,/scratch/alpine,$CURC_CONTAINER_DIR_00D \
    --fakeroot \
    --overlay \
    /projects/${USER}/rstudioserver/rstudio-${r_app_version}/rstudio-server-${r_app_version}_overlay.img:ro \
    /curc/sw/containers/open_ondemand/rstudio-server-${r_app_version}.sif \
    Rscript R_test.R
```

# Slurm script with Rstudio environment

- “--bind” will bring Alpine directories into the container.

```
# Current R version is 4.4.1
export r_app_version="4.4.1"

apptainer exec --bind /projects,/scratch/alpine,$CURC_CONTAINER_DIR_00D \
    --fakeroot \
    --overlay \
    /projects/$USER/.rstudioserver/rstudio-${r_app_version}/rstudio-server-${r_app_version}_overlay.img:ro \
    /curc/sw/containers/open_ondemand/rstudio-server-${r_app_version}.sif \
    Rscript R_test.R
```

# Slurm script with Rstudio environment

“\${r\_app\_version}.sif” is the name of the container image (v.4.4.1).

```
# Current R version is 4.4.1
export r_app_version="4.4.1"

apptainer exec --bind /projects,/scratch/alpine,$CURC_CONTAINER_DIR_00D \
    --fakeroot \
    --overlay \
    /projects/$USER/.rstudioserver/rstudio-${r_app_version}/rstudio-server-${r_app_version}_overlay.img:ro \
    /curc/sw/containers/open_ondemand/rstudio-server-${r_app_version}.sif \
    Rscript R_test.R
```

# Slurm script with Rstudio environment

- We call R with Rscript.
- R\_test.R is the name of the R script. This is the only variable that you will have to worry about.



```
# Current R version is 4.4.1
export r_app_version="4.4.1"

apptainer exec --bind /projects,/scratch/alpine,$CURC_CONTAINER_DIR_00D \
    --fakeroot \
    --overlay \
    /projects/$USER/.rstudioserver/rstudio-${r_app_version}/rstudio-server-${r_app_version}_overlay.img:ro \
    /curc/sw/containers/open_ondemand/rstudio-server-${r_app_version}.sif \
    Rscript R_test.R
```

# Slurm script with Rstudio environment

Please refer to this guide for more details:

[https://github.com/kf-cuanschutz/CU-Anschutz-HPC-documentation/blob/main/Rstudio\\_on\\_Slurm\\_.md](https://github.com/kf-cuanschutz/CU-Anschutz-HPC-documentation/blob/main/Rstudio_on_Slurm_.md)

```
# Current R version is 4.4.1
export r_app_version="4.4.1"

apptainer exec --bind /projects,/scratch/alpine,$CURC_CONTAINER_DIR_00D \
    --fakeroot \
    --overlay \
    /projects/$USER/.rstudioserver/rstudio-${r_app_version}/rstudio-server-${r_app_version}_overlay.img:ro \
    /curc/sw/containers/open_ondemand/rstudio-server-${r_app_version}.sif \
    Rscript R_test.R
```



University of Colorado **Anschutz Medical Campus**

# Part II- LMOD R

# LMOD R

What is LMOD R?

- It is accessible through the use of **LMOD**, the modern environment module system for HPC.
- You need to be on a compute node interactively or you need to submit a slurm batch script.
- We covered LMOD during the previous workshop last week:

[https://github.com/kf-cuanschutz/CU-Anschutz-HPC-documentation/blob/main/Workshops/LMOD\\_and\\_anaconda\\_v2.pdf](https://github.com/kf-cuanschutz/CU-Anschutz-HPC-documentation/blob/main/Workshops/LMOD_and_anaconda_v2.pdf)



University of Colorado  
Anschutz Medical Campus

# LMOD R

- Please use only the version **4.4.0** to install your packages!
- Note that some advance package installation might require some dependencies. Please open a ticket at [hpcsupport@cuanschutz.edu](mailto:hpcsupport@cuanschutz.edu) if you are stuck!



University of Colorado  
Anschutz Medical Campus

# LMOD R

Use LMOD R if:

- You will run heavy job on Alpine, such as high memory computation jobs.
- You don't need to use a GUI interactively.
- You need to run distributed computations, such as with gnu parallel or MPI.



University of Colorado  
Anschutz Medical Campus

# Slurm script

- We use “module load” as seen last week.



```
#!/bin/bash

#SBATCH --partition=amilan      #
#SBATCH --job-name=Rscript_test
#SBATCH --output=Rscript_test.%j.out   :
#SBATCH --error=Rscript_test.%j.err    :
#SBATCH --account=amc-general # Account
#SBATCH --qos=normal  # Quality of service
#SBATCH --nodes=1    # Total number of nodes
#SBATCH --ntasks=4    # Total number of tasks
#SBATCH --mail-type=ALL # Begin, End and Error
#SBATCH --mail-user=kevin.fotso@cuanschi.de
#SBATCH --time=00:00:01

# Load the compiler that R depends on.
module load R/4.4.0

# Run the R command
Rscript R_test.
```

# Slurm script

- Call “Rscript” to run your R software.

```
#!/bin/bash

#SBATCH --partition=amilan      #
#SBATCH --job-name=Rscript_test
#SBATCH --output=Rscript_test.%j.out   :
#SBATCH --error=Rscript_test.%j.err    :
#SBATCH --account=amc-general # Account
#SBATCH --qos=normal   # Quality of service
#SBATCH --nodes=1     # Total number of nodes
#SBATCH --ntasks=4     # Total number of tasks
#SBATCH --mail-type=ALL  # Begin, End and Error
#SBATCH --mail-user=kevin.fotso@cuanschi.de
#SBATCH --time=00:00:01

# Load the compiler that R depends on.
module load R/4.4.0

# Run the R command
Rscript R_test.
```



University of Colorado **Anschutz Medical Campus**

# Part III- Miniforge

# Miniforge R

What is miniforge/anaconda R?

- It gives you the possibility to install many R related packages through miniforge.
- Note that your miniforge R located in your conda environment will be very distinct from LMOD R!



University of Colorado  
Anschutz Medical Campus

# Miniforge R

Use miniforge/anaconda R if:

- You are a frequent anaconda user.
- If you have followed this guide already:  
[https://github.com/kf-cuanschutz/CU-Anschutz-HPC-documentation/blob/main/miniforge\\_migration.md](https://github.com/kf-cuanschutz/CU-Anschutz-HPC-documentation/blob/main/miniforge_migration.md)
- You can also refer to our last week's workshop on how to use miniforge.



University of Colorado  
Anschutz Medical Campus

# Miniforge R

Use miniforge/anaconda R if:

- If you plan to share the environment you are using with the rest of your lab on Alpine. For example, one can export a miniforge environment as yml.
- Consider this r-irkernel guide if you plan on using Jupyterlab:

[https://curc.readthedocs.io/en/latest/open\\_ondemand/jupyter\\_session.html#creating-your-own-custom-jupyter-kernel](https://curc.readthedocs.io/en/latest/open_ondemand/jupyter_session.html#creating-your-own-custom-jupyter-kernel)

# R installation

We first access a compute node with acompile.



```
[...@xsede.org@login-ci1 ~]$ acompile --ntasks=4 --time=12:00:00
acompile: submitting job... salloc --nodes=1 --partition=acompile --ntasks=4 --time=12:00:00 --mem=acompile --bell --oversubscribe srun --pty /bin/bash
salloc: Granted job allocation 17847185
salloc: Nodes c3cpu-a2-u32-3 are ready for job
[...@xsede.org@c3cpu-a2-u32-3 ~]$ module load miniforge
(base) [...]@xsede.org@c3cpu-a2-u32-3 ~]$ conda create --name R_workshop r-base r-essentials -y
Channels:
- conda-forge
Platform: linux-64
Collecting package metadata (repodata.json): done
Solving environment: done

==> WARNING: A newer version of conda exists. <==
    current version: 24.11.3
    latest version: 25.7.0
```

# R installation

We load miniforge.

```
[... @xsede.org@login-ci1 ~]$ acompile tasks=4 --time=12:00:00  
acompile: submitting job... salloc --nodes=1 --ntasks=4 --partition=acompile --time=12:00:00 --mem=16G  
=acompile --bell --oversubscribe srun --name=acompile /bin/bash  
salloc: Granted job allocation 17847185  
salloc: Nodes c3cpu-a2-u32-3 are ready for job  
[... @xsede.org@c3cpu-a2-u32-3 ~]$ module load miniforge  
(base) [...]@xsede.org@c3cpu-a2-u32-3 ~]$ conda create --name R_workshop r-base r-essentials -y  
Channels:  
- conda-forge  
Platform: linux-64  
Collecting package metadata (repodata.json): done  
Solving environment: done  
  
==> WARNING: A newer version of conda exists. <==  
current version: 24.11.3  
latest version: 25.7.0
```

# R installation

We create a new R environment with R v.4.4.3

```
[...@xsede.org@login-ci1 ~]$ acompile tasks=4 --time=12:00:00  
accompile: submitting job... salloc --nodes=4 --ntasks=4 --partition=accompile --time=12:00:00 --mem=16G  
=accompile --bell --oversubscribe srun --pre=/bin/bash  
salloc: Granted job allocation 17847185  
salloc: Nodes c3cpu-a2-u32-3 are ready for job  
[...@xsede.org@c3cpu-a2-u32-3 ~]$ module load miniforge  
(base) [...]@xsede.org@c3cpu-a2-u32-3 ~]$ conda create --name R_workshop r-base r-essentials -y  
Channels:  
- conda-forge  
Platform: linux-64  
Collecting package metadata (repodata.json): done  
Solving environment: done  
  
==> WARNING: A newer version of conda exists. <==  
current version: 24.11.3  
latest version: 25.7.0
```

# R installation

We activate the ENV.

```
(base) [1]: @xsede.org@c3cpu-a2-u32-  
(R_workshop) [1]: @xsede.org@c3cpu-a  
Channels:  
- conda-forge  
Platform: linux-64  
Collecting package metadata (repodata...): done  
Solving environment: done
```



```
$ conda activate R workshop  
2-3 ~]$ conda install conda-forge::r-seurat
```

```
==> WARNING: A newer version of conda exists. <==  
    current version: 24.11.3  
    latest version: 25.7.0
```

Please update conda by running

```
$ conda update -n base -c conda-forge conda
```

# Let's install Seurat!

We install Seurat using the conda-forge channel.

```
(base) [1] > conda activate R_workshop  
(R_workshop) [1] > conda install conda-forge::r-seurat
```

Channels:

- conda-forge

Platform: linux-64

Collecting package metadata (repodata.json): done

Solving environment: done

==> WARNING: A newer version of conda exists. <==

  current version: 24.11.3

  latest version: 25.7.0

Please update conda by running

```
$ conda update -n base -c conda-forge conda
```



University of Colorado **Anschutz Medical Campus**

# THANK YOU