Using and Installing Custom Software on HARDAC
Part 2 - Conda Environments
Goals of Class

Understand Terms

• conda environment, conda channel, conda package

Be able to

• Create a conda environment
• Install software into a conda environment
• Use a conda environment
Class Requirements

Access to the HARDAC cluster

• Email gcb-help@duke.edu to request access if necessary

Complete previous gcb-academy courses

• High Performance Computing (HPC) / SLURM Best Practices for HARDAC

• HARDAC Software: Environment Modules
Setup Directories for Class

Create a directory named "envs" in your lab's data directory:

```bash
$ mkdir -p /data/<LabName>/envs
```

Create a directory to hold files for this class

```bash
$ mkdir -p software_class
```

Copy example fasta data files into your new directory

```bash
$ cp /data/common/software_class/*.*.fasta software_class
```

Change to the exercises directory

```bash
$ cd software_class
```
Installing Cluster Software Challenges

• Troubleshooting Compilation Errors
• Gathering and Configuring Dependencies

Conda

✓ Pre-Compiled Software
✓ Bundled Dependencies
Conda

- Software package manager
- Not just python - R, Java, C, and more
- Large variety of software
Conda Setup

Problem: Conda can fill up your home directory

[Errno 122] Disk quota exceeded

By default conda saves downloaded packages in a ".conda" directory in your home directory. This can fill up your home directory and fail.

Fix: Add lines for conda cache directory in your ~/.condarc file

```
pkgs_dirs:
  - /data/<LabName>/conda_cache
```
Cconda Environments

A conda environment is a special directory where conda packages can be installed. A conda environment can be activated so you can use the installed software from a shell environment.

• Requires Anaconda3 module - module load Anaconda3/...
• Create a conda environment - conda create ...
• Activate a conda environment - conda activate ...
• Install software into the active environment - conda install ...
• Deactivate the current conda environment - conda deactivate
Create/Activate a conda environment

Load Anaconda3 module for the conda command

$ module load Anaconda3/2019.10-gcb02

Create a lab specific conda environment

$ conda create --prefix /data/<LabName>/envs/conda

NOTE: We use the --prefix option so conda will store the software in a lab specific directory instead of your home directory.

Activate the environment

$ conda activate /data/<LabName>/envs/conda
Search for conda packages
https://anaconda.org/search

BLAST+ is a new suite of BLAST tools that utilizes the NCBI C++ Toolkit.

Installers

To install this package with conda run one of the following:

conda install -c bioconda blast
Install a conda package

Install the "blast" conda package

```bash
$ conda install -c bioconda blast=2.10.1
Collecting package metadata (current_repodata.json): done
...
The following NEW packages will be INSTALLED:
...
  blast            bioconda/linux-64::blast-2.10.1-p1526he19e7b1_2
...
Proceed ([y]/n)? y

Preparing transaction: done
Verifying transaction: done
Executing transaction: done
```
Verify conda environment and cleanup

verify command is working as expected

```
$ blastp -h
USAGE
  blastp [-h] [-help] [-import_search_strategy filename]
...
DESCRIPTION
  Protein-Protein BLAST 2.10.1+
...
```

cleanup - deactivate environment/purge modules

```
$ conda deactivate
$ module purge
```
# sbatch script with conda environment

**Make a file called blastpconda.sh**

```bash
#!/bin/bash
module load Anaconda3/2019.10-gcb02
conda activate /data/<LabName>/envs/conda
blastp -query query.fasta -subject subject.fasta -out result2.txt
```

**Run it using the sbatch command**

```
$ sbatch blastpconda.sh
Submitted batch job 23231383
```
List contents of a conda environment

```
$ module load Anaconda3/2019.10-gcb02
$ conda list --prefix /data/<LabName>/envs/conda

...  # Name               Version                    Build    Channel
...  blast              2.10.1                      p1526he19e7b1_2  bioconda
  brotlipy            0.7.0                      py39h27cfd23_1003
...
```
pip – python package manager

pip is the default package manager for python

Rule for pip in a conda environment

"Use pip only after conda"

aka. DO NOT use conda install after using pip
Real World Example - Process

• Review the software installation instructions
• Look for conda packages that contain the software or it's requirements
• Install Software
• Verify Software
Real World Example - Seurat Documentation

Installation Instructions for Seurat

1. Install R, version 3.6 or greater
2. Recommended: Install R Studio

Install from CRAN

Seurat is available on CRAN for all platforms. To install, run:

```R
# Enter commands in R (or R studio, if installed)
install.packages('Seurat')
library(Seurat)
```
Real World Example - Requirements

https://anaconda.org/search

Is there a seurat conda package?

Yes: conda install -c conda-forge r-seurat
Real World Example - Install Seurat

Install r-seurat conda package

```
$ module load Anaconda3/2019.10-gcb02
$ conda activate /data/<LabName>/envs/conda
$ conda install -c conda-forge r-seurat=3.2.2
```
Real World Example - Verify Seurat

Try first few lines from a Seurat tutorial

```r
library(dplyr)
library(Seurat)
library(patchwork)

# Load the PBMC dataset
pbmc.data <- Read10X(data.dir = "../data/pbmc3k/filtered_gene_bc_matrix")
```
Tips for Installing Software

Installing into empty conda environments are more reliable

When no conda package found for your needed software

- Look for conda packages for your software requirements
- Follow instructions for installing your software
Helpful Resources

HARDAC WIKI
https://wiki.duke.edu/display/HAR/

Requesting Help or Video Feedback
gcb-help@duke.edu