Cloud Computing and Data Storage

ACE 592 SAE

So, you hit the memory limit…

We’ve gone over many ways to make sure you don’t hit the memory limit of your machine.

There are times at which this is inevitable, however.

What should you do then?

Your Options

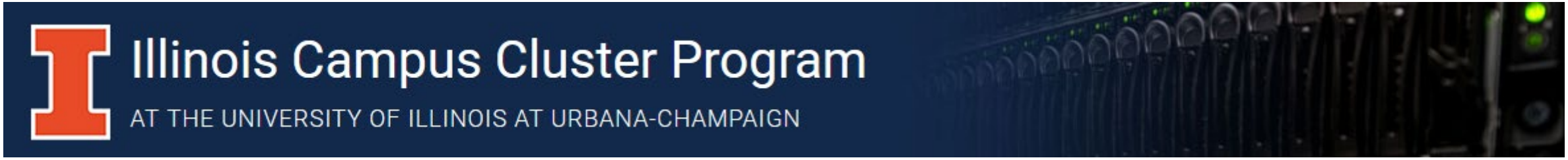
The Option	Amount of Typical Memory	Use this when…
Your computer	8-16 GB	You are doing smaller operations or testing out how to scale something
The computer lab	64 GB	You need a bit more memory ***but the job takes less than 6 hours.***
The Campus Cluster	Up to 384 GB	You need ***a lot of memory*** and ***the job will take more than 6 hours.***
Cloud Computing Services (Google Cloud, AWS, Azure)	As much as you want to pay for.	You need a very specific computing need and have some cash lying around.

Computer Labs

Computer Labs are a common resource

- Do NOT be the person that abuses that resource and messes up everyone else's work!
- To prevent you from overgrazing, the lab will automatically kick you off if your job runs more than 24 hours.
- In general, a job that takes more than 6 hours to run is ***NOT appropriate for the computer lab.***

Then what should you do?



What is the Campus Cluster?

- A campus wide resource for doing jobs that require a lot of memory or CPUs.
- The resources are pooled, but an admin system queues your job until it has the available resources. For this reason, you must “wait in line.”
- The upshot is that you can use an incredibly large resource.
- A node has been purchased for use in ACES, but not sure about other colleges.

Using the Cluster

1. Sign up for an account.
2. SSH into the node.
3. Move any files using Globus.
4. Load your environment.
5. Test the job before running.
6. Run the batch job.

1. Sign up for an account.

- Use this link to sign up for a user account: https://campuscluster.illinois.edu/new_forms/user_form.php
- Fill out all the information.
- Under “primary queue,” choose the relevant option.

    If you are in ACES, use ***ACES***

    If not, talk to your IT admin to see if your department has access.

2. Use SSH to access the node.

Ok, what is ssh?

Secure Shell Protocol (ssh) is a network protocol for logging into other computers. It mostly works through the ***terminal*** or ***command prompt***, which is the only way to access the Campus Cluster.

There is no GUI on the Campus Cluster, meaning you must use the terminal to do all of your work.

2. Use SSH to access the node.

The typical security system for ssh is by generating a ***private key***. Your ssh key is a very long hash which has one part on your computer and another on the server.

Clients for ssh:

- SSH Secure Shell (Windows)
- PuTTY (all platforms)
- Terminal (Mac)

2. Use SSH to access the node.

Example

3. Move files using Globus

- In your /home folder, you are given about 4 GB of storage.
- In your /scratch folder, you have 10 TB of storage, but it get's purged every 30 days.

The easiest way to move files, I have found, is using Globus:

- Log in using their website: https://www.globus.org/
- Navigate to "File Manager."
- Upload your file by selecting the "endpoints" of your transfer.

3. Move files using Globus

- In your /home folder, you are given about 4 GB of storage.
- In your /scratch folder, you have 10 TB of storage, but it get's purged every 30 days.

The easiest way to move files, I have found, is using Globus:

- Log in using their website: https://www.globus.org/
- Navigate to "File Manager."
- Upload your file by selecting the "endpoints" of your transfer.

4. Load your environment

By default, no software is loaded into your session. You must load the "modules" yourself at the start of the session.

Currently it supports:

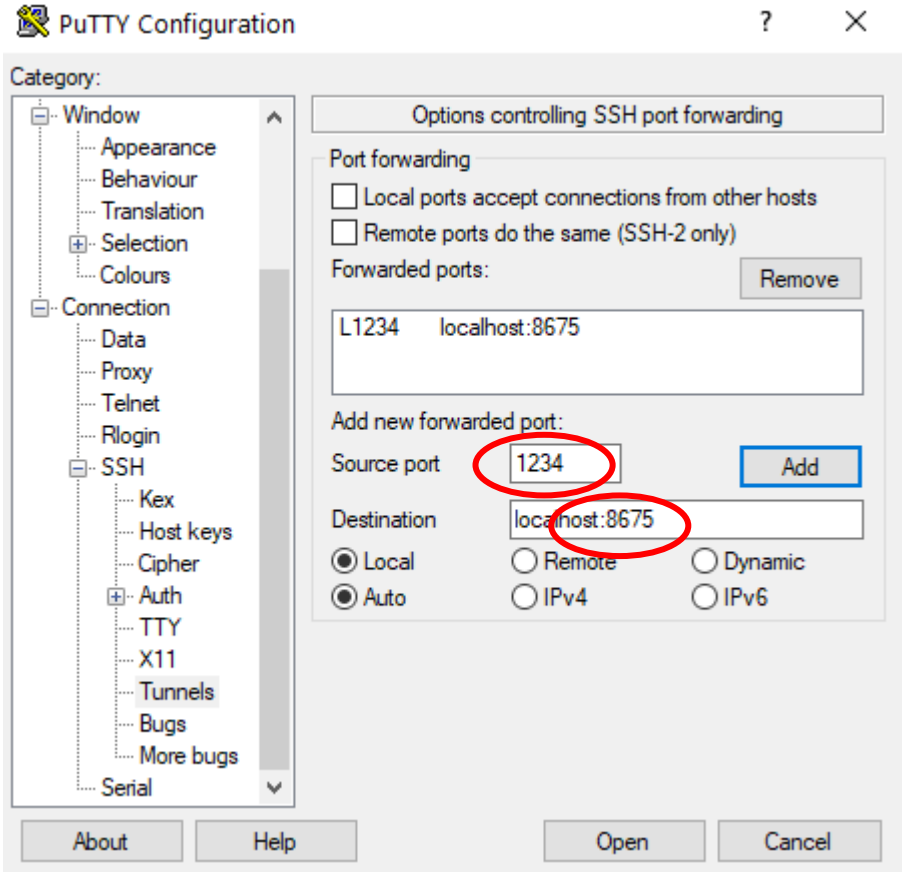
- R 3.5.1 – 4.0.2
- Python 2-3, Anaconda 2-3
- Texlive
- Matlab
- Open MPI
- Various things for compiling: GCC, C++, Java,

5. Test Your Job

There is a little memory and CPU available for you to test your code before running it.

If you want to use Jupyter Notebook, you need to ***port forward:***

In this example, I'm creating a port on my computer "1234" that will access a port on the server at "8675"



5. Test Your Job

Once you open the connection, type this into the prompt:

“jupyter notebook --no-browser –port=8675” (since our port is 8675)

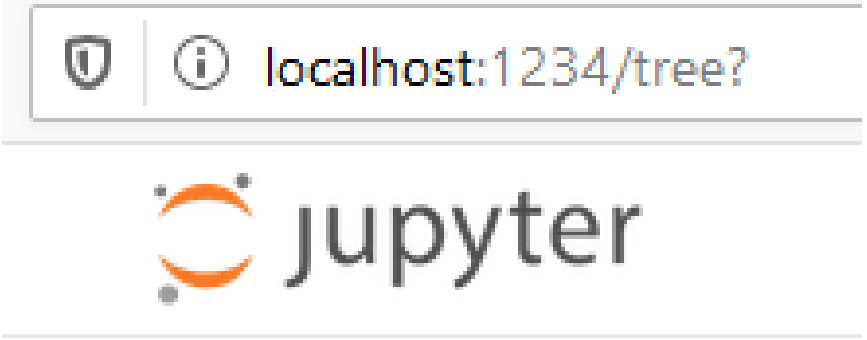
```
[jhtchns2@golubh3 ~]$ module load anaconda/3
[jhtchns2@golubh3 ~]$ jupyter notebook --no-browser --port=8675
[I 17:10:28.859 NotebookApp] JupyterLab beta preview extension loaded from /usr/
local/anaconda/5.2.0/python3/lib/python3.6/site-packages/jupyterlab
[I 17:10:28.859 NotebookApp] JupyterLab application directory is /usr/local/anac
onda/5.2.0/python3/share/jupyter/lab
[I 17:10:28.869 NotebookApp] Serving notebooks from local directory: /home/jhtch
ns2
[I 17:10:28.869 NotebookApp] 0 active kernels
[I 17:10:28.869 NotebookApp] The Jupyter Notebook is running at:
[I 17:10:28.869 NotebookApp] http://localhost:8675/?token=25b935d188c328016070fc
1440572150f3ed9853252370db
[I 17:10:28.869 NotebookApp] Use Control-C to stop this server and shut down all
 kernels (twice to skip confirmation).
[C 17:10:28.873 NotebookApp]

    Copy/paste this URL into your browser when you connect for the first time,
    to login with a token:
        http://localhost:8675/?token=25b935d188c328016070fc1440572150f3ed9853252
370db&token=25b935d188c328016070fc1440572150f3ed9853252370db
```

5. Test Your Job

Finally, go to your browser and navigate to:

"localhost:1234"



The first time it may ask you to type in the token found here:

```
[C 17:10:28.873 NotebookApp]

    Copy/paste this URL into your browser when you connect for the first time,
    to login with a token:
        http://localhost:8675/?token=25b935d188c328016070fc1440572150f3ed9853252
370db&token=25b935d188c328016070fc1440572150f3ed9853252370db
```

6. Run the Batch Job

To run jobs, you need to write a shell script and evaluate it using the command “sbatch.” Here’s an example where I ran an R script:

```
  GNU nano 2.3.1

#!/bin/bash
#SBATCH --time=05:00:00
#SBATCH --nodes=1
#SBATCH --ntasks=1
#SBATCH --mem-per-cpu=246000
#SBATCH --partition=aces
#SBATCH --mail-user=jhtchns2@illinois.edu
#SBATCH --mail-type=BEGIN,END

module load R/3.6.2

Rscript 70_r_robustness_check.r
```

Specifies:

- The memory per CPU you want.
- The number of cores.
- The partition (in this case ACES).
- A place they can email me when it’s done.

6. Run the Batch Job

Example

Now you can run jobs on the cluster!

- These skills translate to cloud computing services, in that the workflow is almost exactly the same.
- The main difference will be that in cloud computing you can create your server with the specs you need and are charged by the minute.

Bonus Suggested Assignment: *If you are in ACES, sign up for an account and practice running a very simple job.*

Then you will be almost entirely prepared to run a job on the cluster!

Data Storage

- Other than hard disk storage, there are a number of cloud storage options:
    - Google Cloud
    - Dropbox
    - Box

The most useful ones are the ones who have the most cross-platform support.

Databases

- If you want to avoid reading all the data in at once, a SQL database can be helpful.
- Relational databases are useful for storage of several data sets which are linked by various keys.
- Requires a server where you can host it.

Google Big Query

- Big Query is Google's version of a SQL database.
- Can host it on their servers if you pay for it.
- Uses the exact same query language as SQL.