

Getting Started with OmniSci resources on Harvard's Cannon Computation Cluster

Contents

Introduction	1
Getting started	1
Start OmniSci or PostGIS instance on FASRC	2
Create VPN Tunnel to new Instance	5
Accessing OmniSci Immerse from your browser via the tunnel	9
Running scripts - general	11
Loading Hourly CGA-SBG Geotweet CSVs to OmniSci	12
Filter geotweets on country:.....	13
Useful Links	13

Introduction

This work was funded by OmniSci Technologies and by NSF grants 1841403 and 2027540.

This is a guide for researchers interested in using OmniSci resources on Harvards' Cluster. Additional details are available here: https://github.com/cga-harvard/GIS_Apps_on_HPC/wiki.

These instructions are Windows oriented. We expect Linux users will not have trouble applying these instructions, but if needed we will add Linux specifics.

Note: In addition to the tools CGA has installed OmniSci and PostGIS, the cluster has a wide range of powerful data processing tools available including GDAL, OpenCV, and VTK:
<https://portal.rc.fas.harvard.edu/p3/build-reports/>.

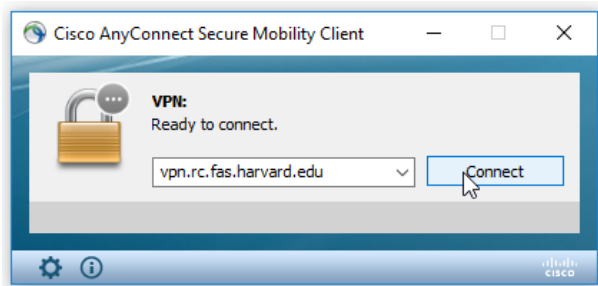
Getting started

You will need a Harvard account to start. Please follow instructions here for obtaining an FASRC account:

<https://docs.rc.fas.harvard.edu/kb/how-do-i-get-a-research-computing-account/>

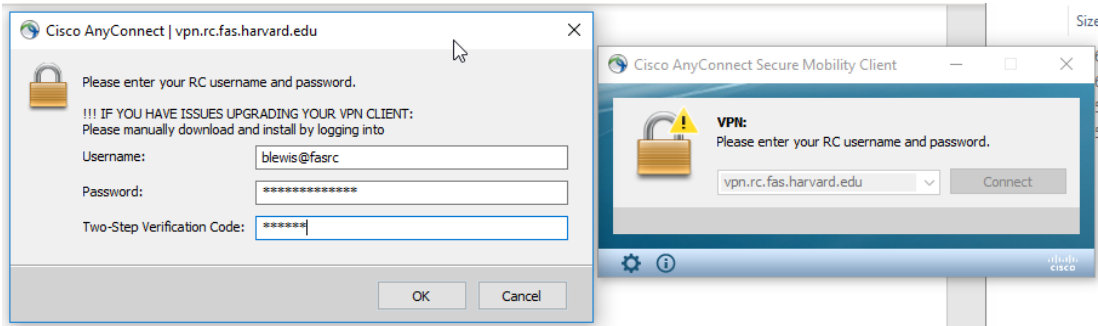
Once you have a Harvard account (using Globus it may be possible to provide access without a Harvard account):

Open VPN connection to FASRC



Use FASRC username, password, and two-step verification code.

Note user name is your-user-name@fasrc.



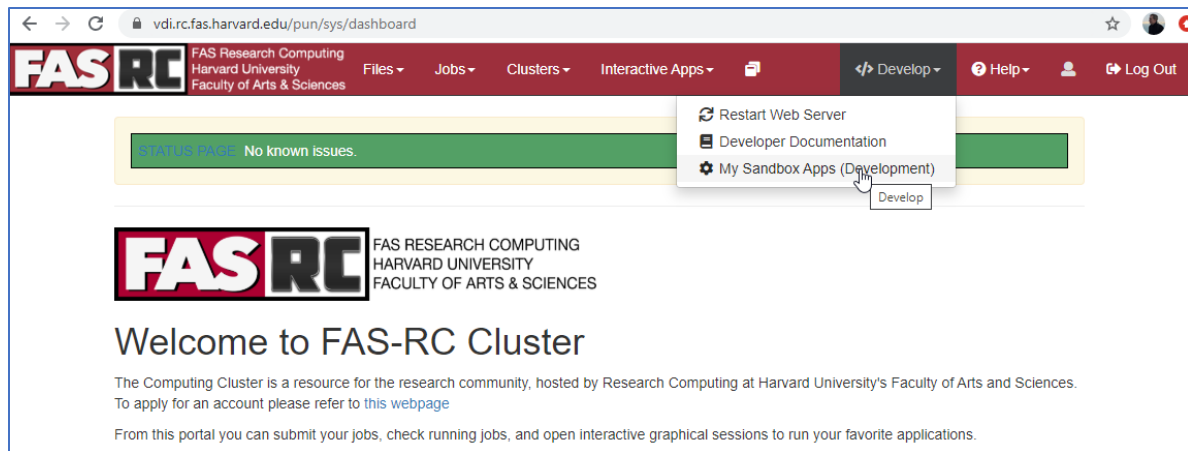
Start OmniSci or PostGIS instance on FASRC

Enter this URL from your browser, rather than clicking on link.

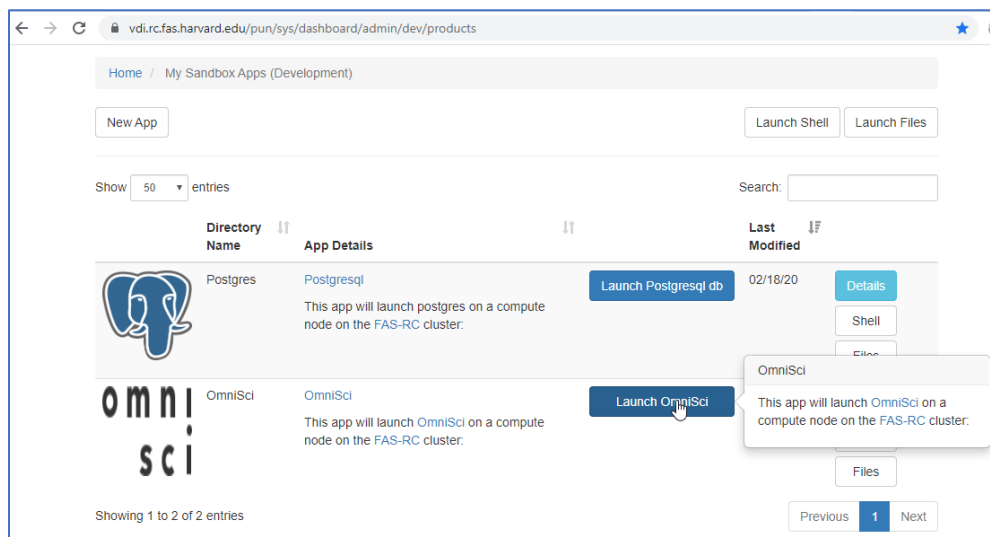
<https://vdi.rc.fas.harvard.edu/>.

If you are prompted for a user name and password, use your base FASRC user name. In this example it would be “blewis”, and enter the same password you used for making your VPN connection.

You will be redirected to this page. Go to “My Sandbox Apps”.



Which is this URL: <https://vdi.rc.fas.harvard.edu/pun/sys/dashboard/admin/dev/products>
 Here you will see two options: Postgres and OmniSci.



To start an a new OmniSci instance, click “Launch OmniSci”.

Configure your OmniSci Instance using the following parameters to start with these (if you want to ask for more, try, but you might wait a while). These resources are sufficient to handle 100 million geotweets.

- For Partition choose “gpu”
- For Memory Allocation choose max of 256 GB
- For number of cores choose max of 2, 1 may be quicker.
- For Number of GPUs choose 1.
- For allocated time choose 1 week of hours.
- Leave rest of fields as they are.

Special requests can be made to FASRC.

Interactive Apps

Desktops

FAS-RC Remote Visualization

FAS-RC Remote Desktop

Containerized FAS-RC Remote Desktop

FAS CGA

OmniSci

Postgresql db

FAS Informatics

Jupyter Lab (scipy-notebook)

RStudio Server (Bioconductor + tidyverse)

GUIs

Desktop Environment for Totalview

Matlab

Slurm

Servers

JBrowse

Jupyter Lab

Jupyter notebook

Rstudio Server

TensorBoard

Interactive Apps [Sandbox]

FAS CGA

OmniSci

Postgresql db

OmniSci

This app will launch OmniSci on a compute node on the FAS-RC cluster:

Partition

Memory Allocation In GB

Number of cores

Number of Cpus to allocate

Number of GPUs

Number of GPUs to allocate. Available only on GPU enabled partitions

Allocated Time (expressed in MM , or HH:MM:SS , or DD-HH:MM).

location to map omnisci-storage

This is the folder location that will be mapped to omnisci-storage. It should contain the subfolders (Datasets, omnisci-storage) (default: /scratch/\$USER/\$SLURM_JOB_ID)

script to be executed before starting OmniSci

This will be executed before starting the container outside the container

☐ I would like to receive an email when the session starts

email address for status notification

Reservation

Slurm Account

If you are not in multiple labs please leave this blank.

Launch

The instance generally takes a few minutes to launch. Once it is launched, go here to see your instance(s) running:

https://vdi.rc.fas.harvard.edu/pun/sys/dashboard/batch_connect/sessions

OmniSci (51411901)

1 node | 4 cores | Running

Host: [aagk80gpu52.rc.fas.harvard.edu](#)

Delete

Created at: 2020-04-06 11:07:36 EDT

Time Remaining: 167 hours and 57 minutes

Session ID: 9f867037-aea8-435c-b049-6d184ed392c1

Connect to Omnisci

For the time being the proxy does not work for this application.
You can connect by tunneling via the login nodes:

1. ssh -NL 8192:[aagk80gpu52.rc.fas.harvard.edu:8192](#) [blewis@login.rc.fas.harvard.edu](#)
2. open [http://localhost:8192](#) in your browser link

Note your instance name and port: **aagk80gpu52.rc.fas.harvard.edu:8192**.

You will use them to create a tunnel to access your new server running OmniSci Immerse via a Putty client and a browser.

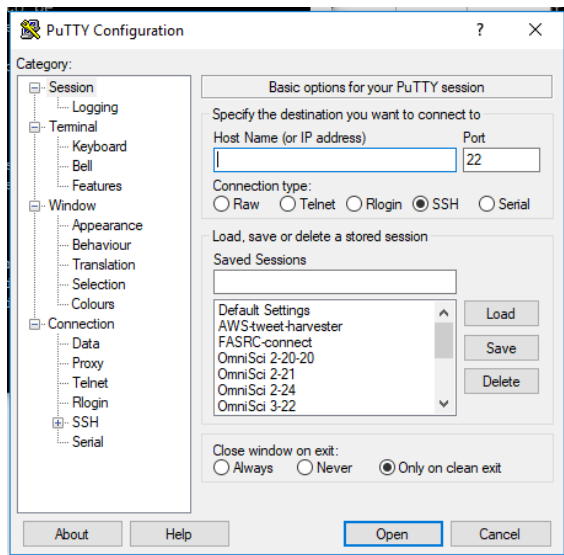
Create VPN Tunnel to new Instance

You will use this information to create a tunnel using putty Windows (Putty) or Linux command line `ssh -NL aagk80gpu52.rc.fas.harvard.edu:8192blewis@login.rc.fas.harvard.edu`

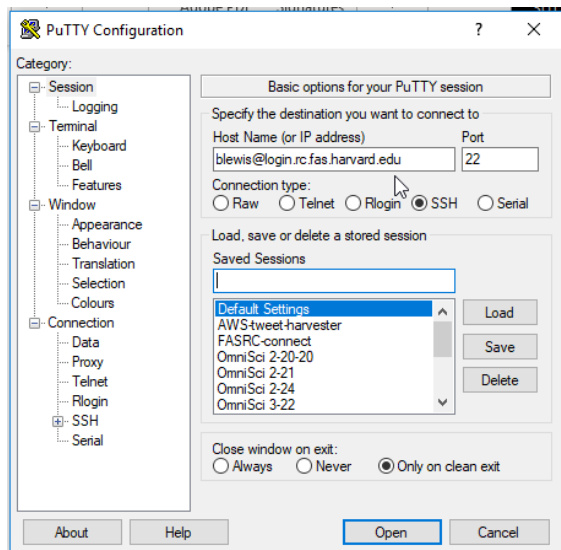
and this information to access Immerse once you have created the tunnel
open [http://localhost:8192](#) your browser [link](#)

Creating tunnel using Putty

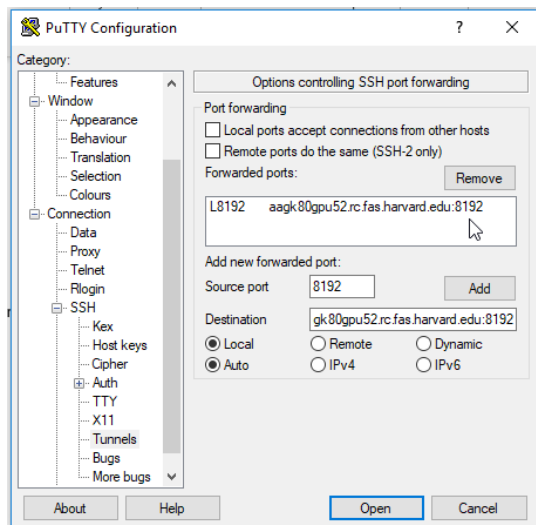
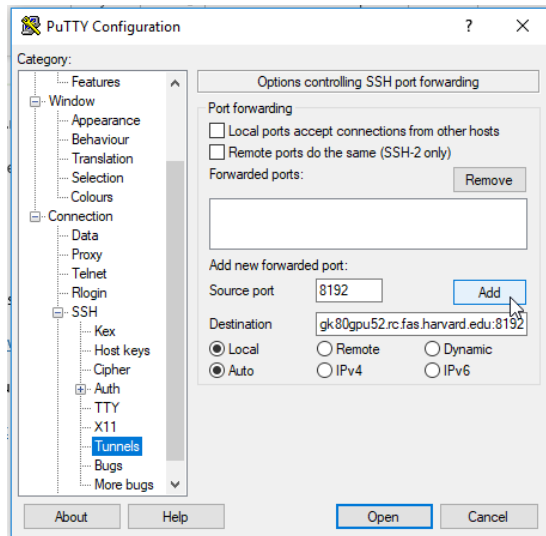
Open Putty:



Add user name/host name, and port:

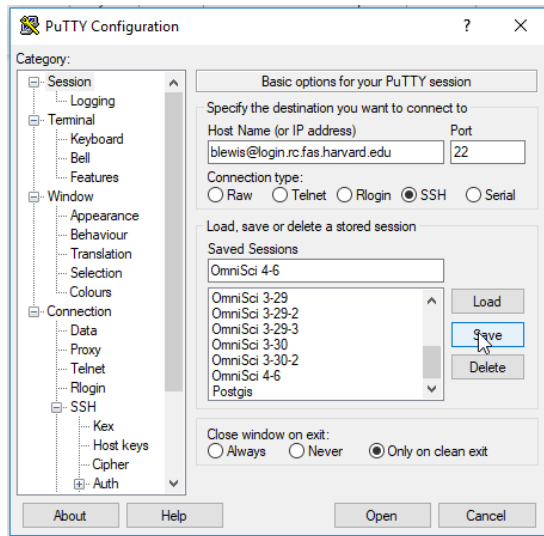


Go to SSH, then Tunnel. Past in your instance name and port, click Add:



Save your settings. Go to Sessions and give a name for the session, click Save.

As long as this instance exists, you can quickly create a tunnel to it using Putty and your saved settings.



The default life of an instance is one week. It is possible to extend an instance for a longer period by making a request to FASRC admins. It also may be possible to request machines with more RAM and CPUs.

Clicking Open on your Putty configuration will open a terminal and you will be prompted for your Harvard PIN password. Then you will be prompted for your third-party authentication code.

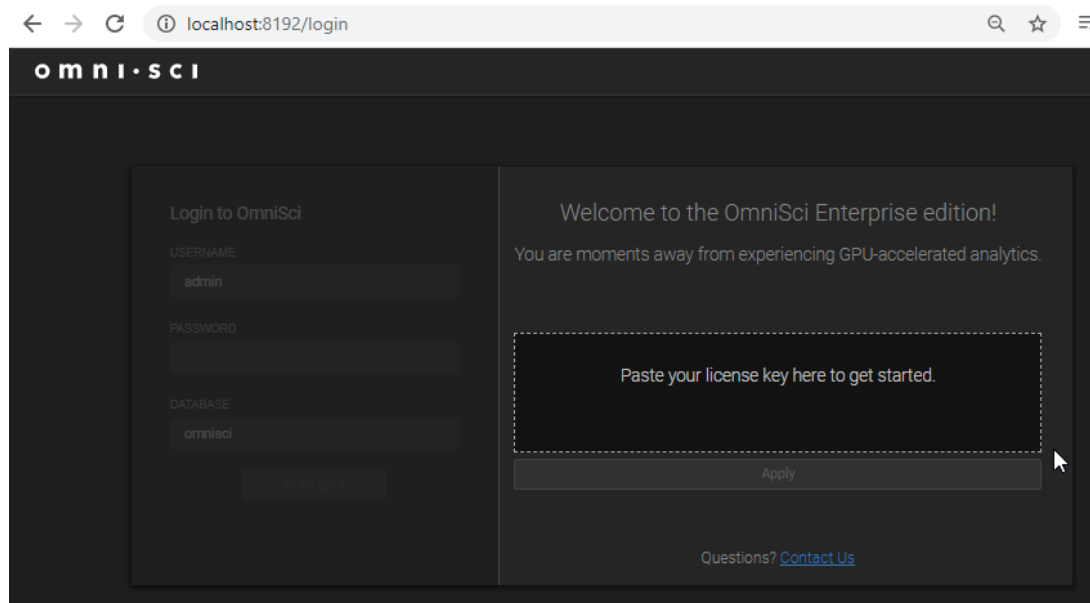
```
Using username "blewis".
Using keyboard-interactive authentication.
Password:
Using keyboard-interactive authentication.
Verification code:
```

Now you should be in.

NOTE: Do not perform computations on the home node you arrive on.

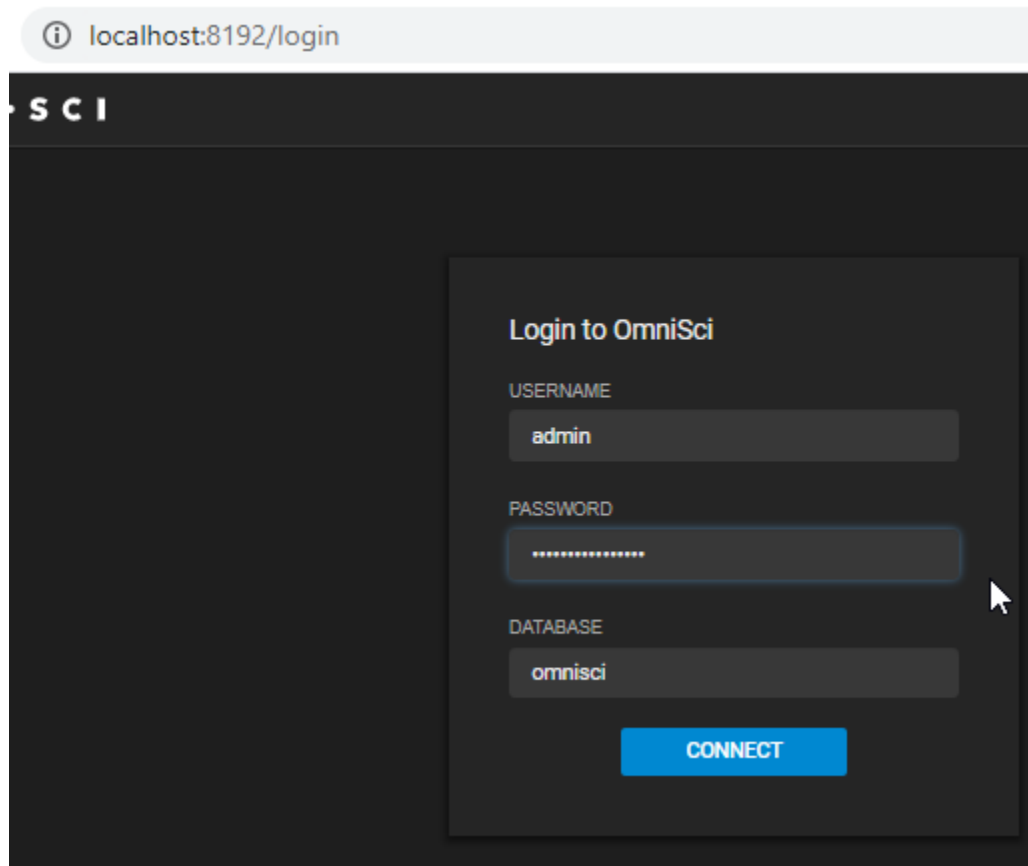
For any compute work you must SSH to the machine that you set up, i.e. `aagk80gpu52.rc.fas.harvard.edu`.

Accessing OmniSci Immerse from your browser via the tunnel

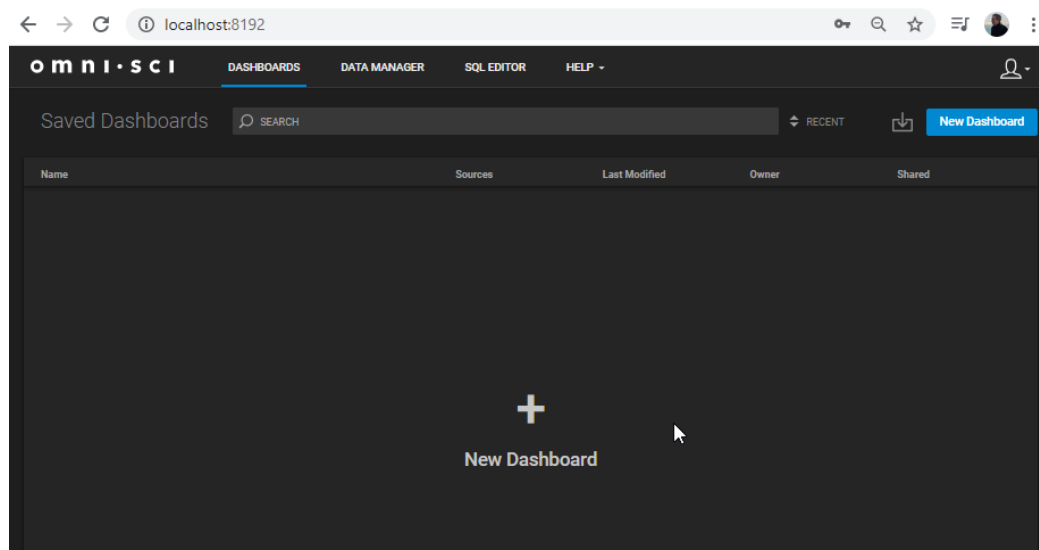


Paste in License Key and click Apply.

Login using user name and password



Now you can now start loading data and building dashboards



Running scripts - general

Copy data you want to work with to /n/holyscratch01/cga/<use name>/data/

- NOTE: storage on /n/holyscratch01/ is a special advantage of FASRC. Data is guaranteed to remain here for 90 days after you load it and it may remain longer. Read/write is *much* faster here than working from the directory on the local node (/n/cga/data) for example.

For any script you run which accesses your omnisci instance you will need to update the correct port in the script to use to access the OmniSci backend for loading data. This backend TCP port you will obtain by clicking on the Session ID.

OmniSci (51411901) 1 node | 4 cores | Running

Host: [_aagk80gpu52.rc.fas.harvard.edu](#) Delete

Created at: 2020-04-06 11:07:36 EDT

Time Remaining: 167 hours and 57 minutes

Session ID: 9f867037-aea8-435c-b049-6d184ed392c1

[Connect to Omnisci](#)

For the time being the proxy does not work for this application.
You can connect by tunneling via the login nodes:

- ssh -NL 8192: [aagk80gpu52.rc.fas.harvard.edu:8192](#) [blewis@login.rc.fas.harvard.edu](#)
- open [http://localhost:8192](#) in your browser link

Which will take you to this page of information about the instance you just created. Click on “output.log” and click “Download”. Open log file.

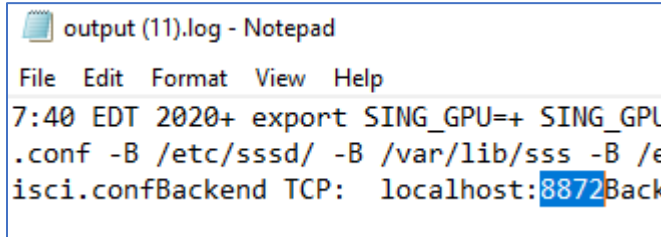
File Explorer Go To... Open in Terminal New File New Dir Upload Show Dotfiles Show Owner/Mode

[/n/home09/blewis/fasrc/data/sys/dashboard/batch_connect/dev/OmniSci/output/e7cbc5da-afdb-47a8-a1df-adedf29b3797/](#)

View Edit Rename/Move Download Copy Paste (Un)Select All Delete

name	size	modified date
..	<dir>	
after.sh	442b	02/18/2020
before.sh	2.44kb	06/24/2020
connection.yml	77b	06/24/2020
job_script_content.sh	3.09kb	06/24/2020
job_script_options.json	508b	06/24/2020
output.log	2.65kb	06/24/2020
script.sh	1.27kb	06/24/2020
script_original.sh	1.19kb	06/24/2020
user_defined_context.json	326b	06/24/2020

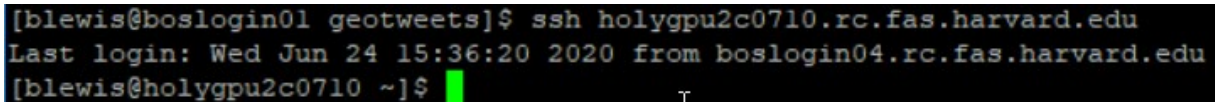
Scroll to bottom and you will find the Backend TCP. In this case it is 8872.



Again, before running processes you must ssh to your GPU instance after logging in to FASRC. Example:

```
ssh aagk80gpu46.rc.fas.harvard.edu
```

Or another example from the actual command line:



- Connect to your compute node
- For scripts involving OmniSci, PostGIS, and geotweets, Load Conda

```
module load Anaconda3/5.0.1-fasrc02
```

- Create environment, activate env and install libraries

```
conda create -n geotweets python=3.6
source activate geotweets
pip install pandas
pip install geopandas
pip install numpy
pip install shapely
pip install pymapd
```

- Change the input file name to your CSV and run the script

```
python3 /n/holyscratch01/cga/dkakkar/scripts/geotweets.py
```

- For more details please see Wiki https://github.com/cga-harvard/GIS_Apps_on_HPC/wiki.

Loading Hourly CGA-SBG Geotweet CSVs to OmniSci

- Move latest geotweets from cga-geotweets.rc.fas.harvard.edu to Archive at /n/cga/data/geotweets/cga-sbg/2020/

- Copy tweets you want to work with to /n/holyscratch01/cga/<use name>/data/geotweets/
 - NOTE: storage on /n/holyscratch01/ is a special advantage of FASRC. Data is guaranteed to remain here for 90 days after you load it and it may remain longer. Read/write is *much* faster here than working from the directory on the local node (/n/cga/data) for example.
- Edit your script to point to the Backend TCP port (see above for how to find it)
- Edit your script to point to your data source at /n/holyscratch01/cga/dkakkar/data/geotweets/
- Activate screen

screen

- Ssh to Omnisca compute node
- Load Conda

```
module load Anaconda3/5.0.1-fasrc02
source activate geotweets
```

- Edit the script to include your file path and omnisca port just like gdelt script
- Run the script (example)

```
python3 /n/holyscratch01/cga/dkakkar/scripts/geotweets_omnisca_geom.py
(EXAMPLE)
```

Filter geotweets on country

- First, upload the geotweets tables (see above)
- Run the country script by changing name of country and table name

```
screen
module load Anaconda3/5.0.1-fasrc02
source activate geotweets
  • Example
python3
/n/holyscratch01/cga/dkakkar/scripts/geotweets_omnisca_countries.py
```

Useful Links

- Intro to the Cannon Cluster <https://www.rc.fas.harvard.edu/wp-content/uploads/2019/12/Intro-to-Cannon.pdf>
- FASRC Quick Start Guide <https://docs.rc.fas.harvard.edu/kb/quickstart-guide/>
- Create Account and Access FAQ <https://docs.rc.fas.harvard.edu/kb/access-and-login/>
- How to Run Jobs <https://docs.rc.fas.harvard.edu/kb/running-jobs/>
- SLURM Commands <https://docs.rc.fas.harvard.edu/kb/convenient-slurm-commands/>
- HUIT Security Policy <https://security.harvard.edu/>
- Research Data Security Policy <https://vpr.harvard.edu/pages/harvard-research-data-security-policy>