

# Startup guide for using Stampede2 ECCO Summer School 2019

Timothy Smith, An T. Nguyen

May 23, 2019

This document is designed to provide instructions for

1. setting up a computing environment on the Stampede2 supercomputer, e.g. for running the MITgcm
2. getting started with the available analysis tools

If you have not created a TACC account please navigate [here](#) to do so. Once you have an account, please try logging in to our web portal [here](#), and contact me at [tsmith@oden.utexas.edu](mailto:tsmith@oden.utexas.edu) with any issues. If you run into permissions issues in any of the following operations, please contact me with your TACC user ID and a brief description of the problematic directories.

For any other problems, please open up a ticket through the web portal [here](#) through the [web portal](#) and briefly explain your problems to the TACC folks.

We currently have analysis tools in python through Jupyter notebooks or in MATLAB through our web portal. Each of these resources will be running on a single skylake compute node on TACC's Stampede2 super computer. This will provide the user with 48 cores and 192GB of memory. For more information regarding these processors, please see the [Stampede2 User Guide](#). See below for quick start guides to using python and MATLAB on these nodes.

## 1 Basic setup

Here we'll discuss the location of public (or community) data and the essential repositories which are located there for running on the supercomputer. First, log in to the [web portal](#). Then, navigate to "My Dashboard" after logging in by clicking on your username in the top right corner. From there, you can find the "Community Data" directory from the Data Files tab on the left hand side. You should see something like what is shown in Fig. [1.1](#) below. This mirrors the directory on Stampede2:

```
/work/projects/aci/ECCO/community
```

which I will refer to as `$communitydir` in this document (this environment variable is set in our `bashrc` file, see next section for where this is located).

One can log on to Stampede2 from a terminal as follows:

```
$ ssh <username>@stampede2.tacc.utexas.edu
```

and find the community directory above. For now, the easiest way to log in is from a local machine. We hope to be able to access the Stampede2 file system with a terminal enabled directly through the web portal, but this is currently in development.

ECCO Data Transfer Guide Contact 🔔 tsmith

Dashboard Data Files Applications Notifications Search

Search [ ] [Q] [Rename] [Move] [Copy] [Preview] [Compress] [Extract] [Download] [Move to Trash]

**COMMUNITY DATA**

Name	Size	Last Modified	Permissions
■ ASTE	4.0 kB	5/7/19 9:47 AM	All
■ atnguyen_example	4.0 kB	5/8/19 11:52 AM	All
■ computing	4.0 kB	5/7/19 6:38 PM	All
📄 dir_permissions.txt	385.0 bytes	5/9/19 9:16 AM	Read/Write
■ ECCO	4.0 kB	5/9/19 9:24 AM	All
■ grids	4.0 kB	5/7/19 6:44 PM	All
■ GRISO-X	4.0 kB	5/7/19 11:15 AM	All
■ input_ecco_v4r3	4.0 kB	5/8/19 11:48 AM	All
■ matlab	4.0 kB	5/9/19 2:30 PM	All
■ permissions_example	4.0 kB	5/9/19 9:08 AM	All
■ porta_Lowned	4.0 kB	5/9/19 8:56 AM	All
■ python	4.0 kB	5/9/19 12:41 PM	All
■ SWOT	4.0 kB	5/7/19 11:15 AM	All
■ test_pem	4.0 kB	5/9/19 12:42 PM	All

My Data My Projects Google Drive **Community Data**

Figure 1.1: Navigate to “Community Data”

Note also that your personal work directory, `$WORK/./` on Stampede2, is mirrored on the portal under the tab “My Data”. Note that `$WORK` takes you to a directory like: `/work/####/username/stampede2` so you can choose to either use this directory or the one directly above it in the following instructions.

## 1.1 Environment: bash scripts, optfiles etc.

Within the community directory, you’ll see the `computing/` directory which is a clone [our group’s git repo on BitBucket](#). Here you will find:

- the `bashrc` script for stampede2 at

```
/work/projects/aci/ECCO/community/computing/env/bashrc_stampede2
```

which has important environment variables for running python scripts and compiling the MITgcm. Be sure to source our `bashrc` file by adding this line to your own `~/.bashrc`

```
source /work/projects/aci/ECCO/community/computing/env/bashrc_stampede2
```

Alternatively, you can concatenate this file to your personal `bashrc` by typing the following command (If you choose this route, you only have to do it the one time):

```
cat /work/projects/aci/ECCO/community/computing/env/bashrc_stampede2 >> ~/.bashrc
```

Once you have done either of these, be sure to re-run your `bashrc` by typing the command:

```
$ source ~/.bashrc
```

(or log out and back in)

## 2 Instruction for re-running ECCOv4r3

Please find the following necessary files for re-running ECCOv4 release 3:

- The main directory with all inputs etc.:

```
$communitydir/ECCO/ECCOv4/Release3/input.ecco_v4r3
```

- There are two sets of instructions for running ECCOv4r3. First, we have a set of instructions which are specific for running ECCOv4r3 on Stampede2, including specifics like which compile time option files to use for this machine. Find this [here on the summer school website](#), [here on Overleaf](#), or on stampede2 at:

```
$communitydir/ECCO/ECCOv4/Release3/input.ecco_v4r3/reproducing_ECCOv4r3_TACC.pdf
```

Second, a general set of instructions for running ECCOv4r3 that applies to all machines, including getting the right checkout of the MITgcm, can be found [here on the summer school website](#) or [here on the JPL server](#) (link initiates download).

- All ECCO output can be found at

```
$communitydir/ECCO/ECCOv4
```

See within for release 1 and release 3 output.

## 2.1 Copying files to \$communitydir

Instructions for copying files to Stampede2 can be found from our web portal, [here](#), which includes many methods for doing so. Users comfortable with the command line and prefer `scp` or `rsync` over GUIs should skip to the bottom of the page.

**IMPORTANT:** whenever you copy files to the directory, you must change permissions so that they are accessible. Because this directory is mirrored on the web portal, there are a few more commands than normal. Please perform the following commands e.g. for a new directory:

1. Change the group number as follows:

```
$ chgrp -R G-820213 new_dir/
```

2. Give portal account full access via ACL

```
$ setfacl -R -m u:wma_prtl:rwX new_dir/
```

3. Ensure read, write, execute permissions for users, read and execute for the group:

```
$ chmod -R u=rwX,g=rX new_dir/
```

These instructions can also be found at `$communitydir/dir_permissions.txt`

Ian Fenty has also created a bash script to automate this at `$communitydir/give_permissions.sh`, which, from within the new directory, updates all permissions of its contents. Thanks Ian!

## 3 Python

Jupyter notebooks with python 3 kernels can be accessed through the stampede2 terminal as follows:

1. Connect to Stampede2 through a terminal

```
$ ssh <username>@stampede2.tacc.utexas.edu
```

2. Be sure to source our bashrc file by adding this line to your own `~/.bashrc`

```
source $communitydir/computing/env/bashrc_stampede2
```

Once you have done this, be sure to re-run your bashrc by typing the following command into your terminal

```
$ source ~/.bashrc
```

or log out and back in.

3. Make a directory to launch the job from, we suggest this to be in your personal work directory. From here link the submission

```
$ cd $WORK/  
$ cd ../ #this should now be /work/####/username  
$ mkdir python_launch  
$ cd python_launch/
```

4. Grab the following files

```
$ cp $communitydir/python/submit.sh .  
$ cp $communitydir/python/test_notebook.ipynb .
```

5. The script **submit.sh** has some parameters at the top which modify the job which will run your Jupyter notebook on Stampede2. These are shown here:

```
#SBATCH -p "skx-normal"  
#SBATCH -J "myjob"  
#SBATCH -o "out.txt"  
### N is the number of node, need to be 1  
#SBATCH -N 1  
### n is the number of cpus per node, you can ask for 48cpus  
#SBATCH -n 48  
#SBATCH -t "06:00:00"  
#SBATCH -A "Polar-Project"  
#SBATCH --reservation=ECC019School
```

The main argument you may want to change is the time limit which is set by -t HH:MM:SS. The default here is 6 hours.

6. Launch the submit script

```
$ sbatch submit.sh
```

7. This generates a file `out.txt` in your directory, grab the port number and password for your notebook at the bottom of this file

```
$ tail out.txt
```

until you see something like

“Your notebook is now running at <http://stampede2.tacc.utexas.edu:PPPP> with password XXXX”

8. In your browser on your local machine, navigate to that website and copy+paste the given password

9. If you have an “Invalid Credentials” error, you may have to modify line 36, which looks like:

```
LOGIN_IPY_PORT=8890
```

Modify this number by adding one or two (i.e. set it to 8889, 8891, etc) until something works. Each time you modify the number, please cancel your current job before re-submitting.

10. Give `test_notebook.ipynb` a whirl!
11. IMPORTANT: Please be sure to end your job after you have exited the jupyter notebook. Do this one of two ways.

Simply delete the file titled `delete_me_to_end_session.txt`, which should be in the directory opened in the Jupyter environment, or

with the following commands:

command	explanation
<code>squeue -u yourusername</code>	check the status of your job
<code>scancel yourjobid</code>	kill your job

Table 3.1

## 4 MATLAB

One can analyze ECCO data directly from the portal without having to log in through a terminal. Simply navigate to the web portal and go through the following steps

1. Make a directory in your My Data location from the “Data Files” tab, all within the portal, as in Fig. 4.1

Name	Size	Last Modified	Permissions
■ .APPDATA	4.0 kB	5/7/19 8:44 AM	All
■ .slurm	4.0 kB	10/28/15 4:07 PM	All
■ archive	4.0 kB	4/17/19 1:45 PM	All
■ bash-envy	4.0 kB	1/4/18 3:22 PM	All
■ computing	4.0 kB	10/3/18 4:48 PM	All
□ delete_me_to_end_session	35.0 bytes	5/8/19 1:08 PM	Read/Write
■ ecco-data	4.0 kB	9/2/18 6:02 AM	All
■ ECCOv4-py	4.0 kB	3/21/19 2:22 PM	All
■ gcm-contrib	4.0 kB	5/6/19 2:32 PM	All
■ gcmfaces	4.0 kB	4/17/19 2:20 PM	All
■ grids	4.0 kB	5/6/19 11:47 AM	All
□ jra55	26.0 bytes	8/23/18 10:32 AM	All
■ jupyter	4.0 kB	5/8/19 10:07 AM	All
■ jupyter-examples	4.0 kB	3/21/19 2:23 PM	All
□ jupyter_notebook_config.py	29.0 kB	5/8/19 1:08 PM	Read/Write
■ labsea-exp-design	4.0 kB	8/23/18 2:09 PM	All

Figure 4.1: Make a directory from the portal

2. Navigate to "Community Data/matlab" and copy the files

- `startup.m`
- `establish_mygrid.m`
- `test_script.m`

to your newly made directory, see Fig. 4.2 below

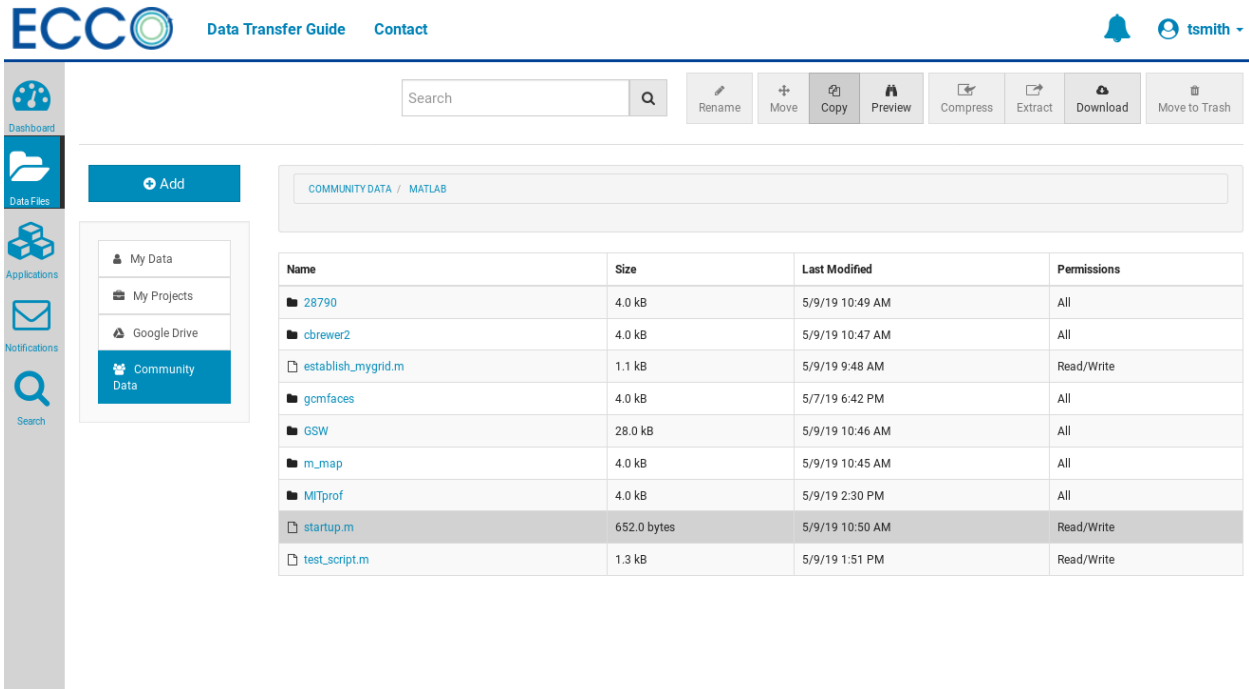


Figure 4.2: Open up MATLAB from the portal

3. Open up MATLAB from the Applications -> Data Processing tab, as in Fig. 4.3

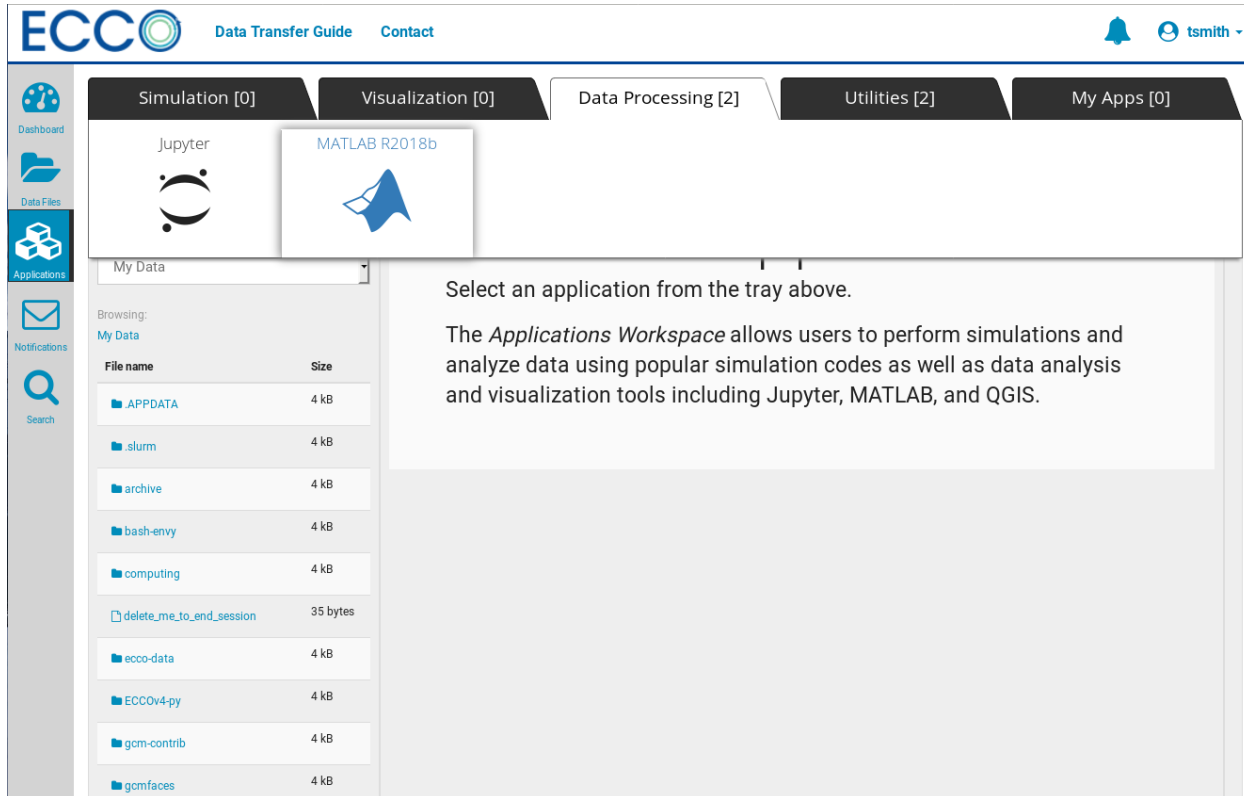


Figure 4.3: Open up MATLAB from the portal



4. Select your directory to launch the job from and give the job a useful name
5. Before trying any gcmfaces operations, run the startup script
6. Enjoy the power of the skylake nodes... :)