

Table of Contents

Get Started.....	2
Preliminary info.....	2
How to transfer files	2
Create new directory in gpfs/scratch.....	3
How to install python (or another software) in your own project folder.....	4
How to load a module:.....	5
How to use our node in interactive mode	5
How to run matlab, rstudio, python	6
How to use cluster and software modules	7
About the Scheduler	9
Partitions in CCR (ub-hpc) Cluster.....	9
An example job script	9
Using faculty partition.....	10
Practical example of running MATLAB script.....	11
How to load and use Tigramite package.....	12

Get Started

Go to ondemand CCR (<https://ondemand.ccr.buffalo.edu>) and login with your UB credentials.
(You can launch batch or iterative apps) BEST BROWSER IS CHROME

Alternative option:

On campus or UB VPN, use secure shell (ssh) -> login node vortex.ccr.buffalo.edu

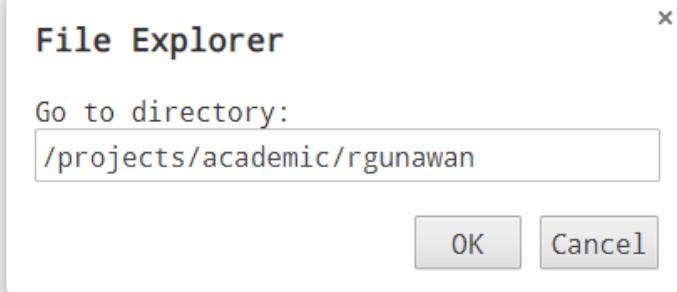
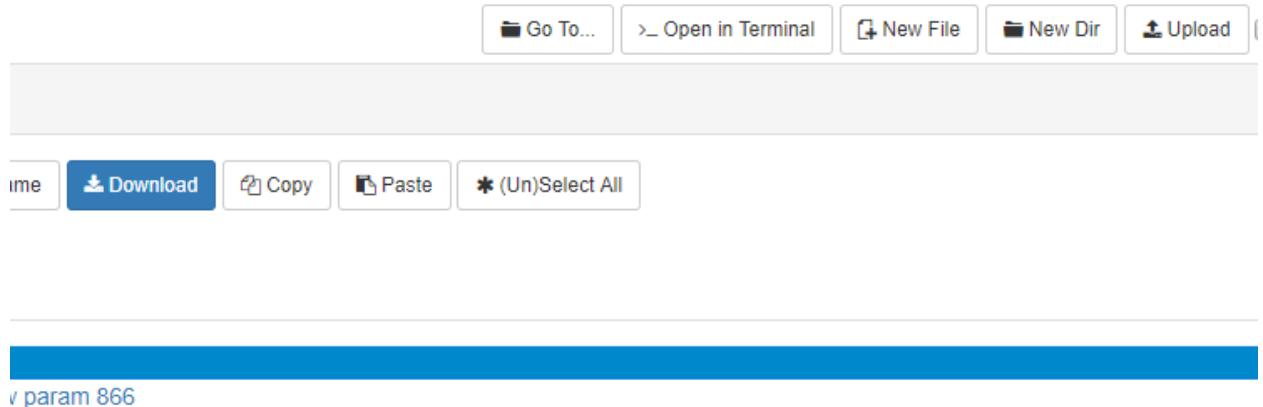
Preliminary info

- 5 Gb for each user in the home directory
- Up to 1 TB of group space

How to transfer files

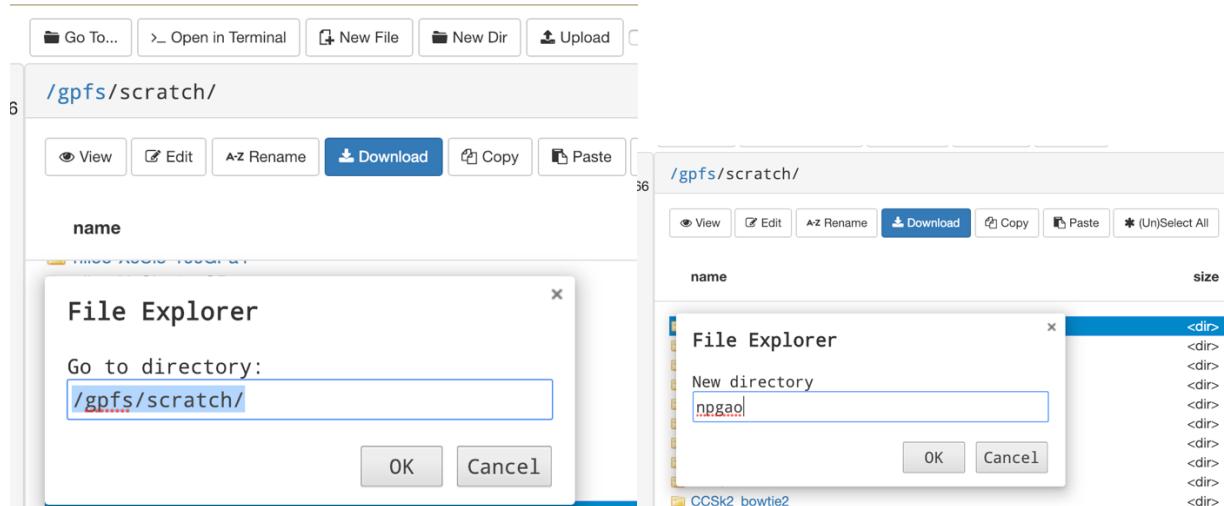
Files -> home directory -> go to -> /projects/academic/rgunawan and go to single-cell for example

The screenshot shows the OnDemand web interface. At the top, there is a blue header bar with navigation links: 'Files', 'Jobs', 'Clusters', 'Interactive Apps', 'My Interactive Sessions', 'Help', 'Logged in as npgao', and 'Log Out'. Below the header, a message for new users reads: 'NEW USERS: Please run this script in a terminal window before submitting jobs to the cluster: /util/ccr/bin/ssh_no_password.sh'. The main content area features the 'OnDemand' logo with a red circular arrow icon and the word 'OnDemand' in bold black letters. A sub-headline states: 'OnDemand provides an integrated, single access point for all of your HPC resources.' The rest of the page is mostly blank white space.



Create new directory in gpfs/scratch

Goto /gpfs/scratch/ and press “New Dir”



How to install python (or another software) in your own project folder

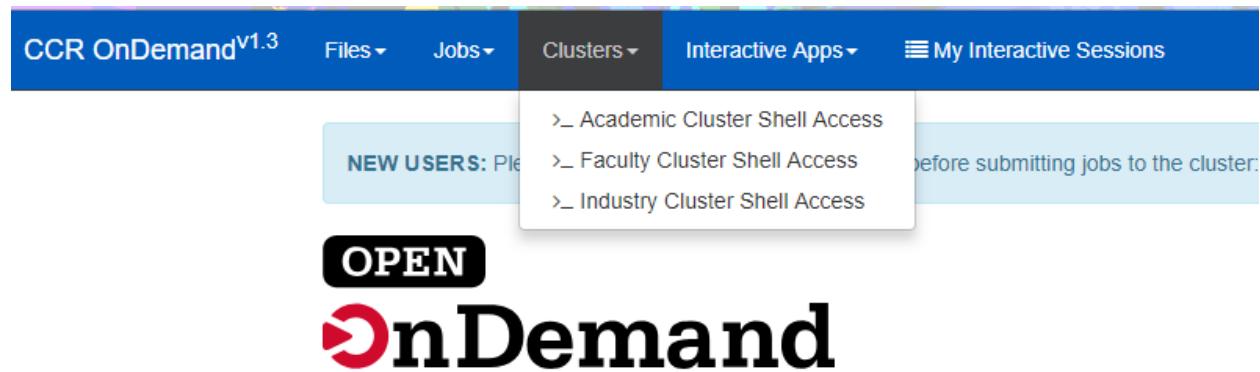
FIRST get the installation file if the version you need is not already available in
/projects/academic/rgunawan/software

OPTION1: download it and transfer to /projects/academic/rgunawan/software/python

OPTION2: “go to” /gpfs/scratch/cdc (FROM CYNTHIA) and copy the file to
/projects/academic/rgunawan/software/python

SECOND

Open terminal from cluster -> academic_cluster_shell..



TYPE:

```
cd /project/academic/rgunawan/software/python/
```

```
ls
```

```
bash Anaconda-2018.12-linux etc.
```

Type yes

Now we are in /user/saberma/anaconda3 then type:

```
/project/academic/rgunawan/software/python/anaconca3-2018.12 for installation
```

Meantime open another window (Open terminal from cluster -> academic_cluster_shell..

```
cd /project/academic/rgunawan/software/python/
```

```
)
```

```
Type cd /project/academic/rgunawan/software/python/
```

```
ls -l
```

To see that python is installing!

Then we have to create the module for python:

OPTION2: “go to” /gpfs/scratch/cdc (FROM CYNTHIA) and copy the file to
/projects/academic/rgunawan/module

Now open the file.lua e.g. python37.lua we should type:

Whatis([[python37]])

Change second line path to :

/projects/academic/rgunawan/software/python/Anaconda3-2018.02

How to load a module:

Make sure the .lua file is present in module directory, i.e.

/projects/academic/rgunawan/module/test.lua

```
$module avail python
$module use /projects/academic/rgunawan/module
$module load test
$module list (now you should be able to see that your module is loaded)
$which python (shows your python directory)
$python -V (shows your python version)
```

How to use our node in interactive mode

Go to Interactive Apps -> faculty clusters desktop

The screenshot shows the CCR OnDemand v1.3 interface. At the top, there is a blue header bar with the text "CCR OnDemand^{v1.3}" and navigation links for "Files", "Jobs", "Clusters", "Interactive Apps", and "My Interactive Sessions". Below the header, there is a message for "NEW USERS": "Please run this script to set up your environment." A large "OPEN" button is prominently displayed. To the right, under the "Interactive Apps" heading, there is a list titled "Desktops" with several options: "Academic Cluster (UB-HPC) Desktop", "OpenGL Desktop", "StarCCM Desktop", "CUDA Desktop", and "Faculty Clusters Desktop".

Then enter:

Partition

rgunawan



Please select a partition from the drop-down menu

QOS

rgunawan

Usually the same as partition unless using the 'supporters' QOS on the ub-hpc cluster

Number of hours

1

Number of nodes

1

— — — — —
And launch (in interactive sessions we can use only one node).

How to run matlab, rstudio, python

In interactive mode: File system -> File -> open terminal
Otherwise from clusters -> academic cluster shell access

The screenshot shows the CCR OnDemand v1.3 interface. At the top, there is a blue header bar with the title "CCR OnDemand v1.3" and navigation links for "Files", "Jobs", "Clusters", "Interactive Apps", and "My Interactive Sessions". The "Clusters" link is currently active, as indicated by a dark grey background. A dropdown menu is open under "Clusters", listing three options: "Academic Cluster Shell Access", "Faculty Cluster Shell Access", and "Industry Cluster Shell Access". To the left of the dropdown, a light blue box contains the text "NEW USERS: Please log in before submitting jobs to the cluster". In the center, there is a large "OPEN" button with the "OnDemand" logo.

Type:

```
module avail matlab  
module load matlab/R2017b  
matlab -nosoftwareopengl
```

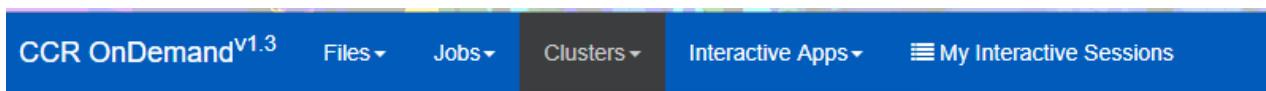
```
module load R  
rstudio
```

(PLEASE TERMINATE YOUR JOB ONCE YOU FINISHED!!!!!!)

This screenshot shows the details of a session named "Faculty Clusters Desktop (1650886)". The session was created at "2019-02-06 12:58:01 EST" and has "1 node | 32 cores | Starting". It has a "Delete" button. The session ID is "1677732d-ef74-44e0-9721-a2662fce33b2". A message at the bottom states: "Your session is currently starting... Please be patient as this process can take a few minutes."

How to use cluster and software modules

Go to cluster -> Academic Cluster Shell Access



NEW USERS: Please

- >_ Academic Cluster Shell Access
- >_ Faculty Cluster Shell Access
- >_ Industry Cluster Shell Access

before submitting jobs to the cluster:

OPEN
OnDemand

CCR supports 300+ software packages, installed as modules

module avail (what's installed?)

module load (load modules)

module unload

University at Buffalo The State University of New York

Engineering	Machine Learning	Hadoop/MapReduce	Math/Physics	Image Processing	Utilities
ABAQUS	Caffe	Hadoop	BandUP	EMAN2	7zip
ANSYS	Caffe2	HBase	CrystFEL	FFMPEG	Amt
AVL FIRE	Keras	Hive	FEniCS	OpenCV	DMAKE
CFX	MikeNet	Pig	GAP	OpenEXR	DMTCP
COMSOL	TensorFlow	Spark	Gmsh	OpenSlide	gnu-parallel
CPLEX	Theano	ZooKeeper	Gurobi	Environmental	HDF/HDFS
DAKOTA	Torch	GPU Programming		kmers	Cartes
Iso2mesh	Data Analytics		CUDA	GRASS	Izml
LSDYNA	edgar	Intel-OpenCL	VMTK	HYSPLIT	Mono
NETGEN	Jags	OpenCL	Visualization		netCDF, NCO, COO
OpenFOAM	MCL	PyCUDA	decaff	MODFLOW	phipic2
OpenSees	ms	Editors		NCL	NaSt3DGP
pyFormex	R	Emacs	ParaView	puffin	TauBench
QUCS	SAS	Idle	TecPlot	TELEMAC	tmux
SALOME	Stan	NetBeans	VTK	TITAN	xclip
StarCCM	STATA	Vim	Xmgrace	WRF-WPS	xfig

About the Scheduler

SLURM (workload manager)

Useful scheduler commands:

sbatch submit a job script

squeue check the status of a job

scancel delete a job

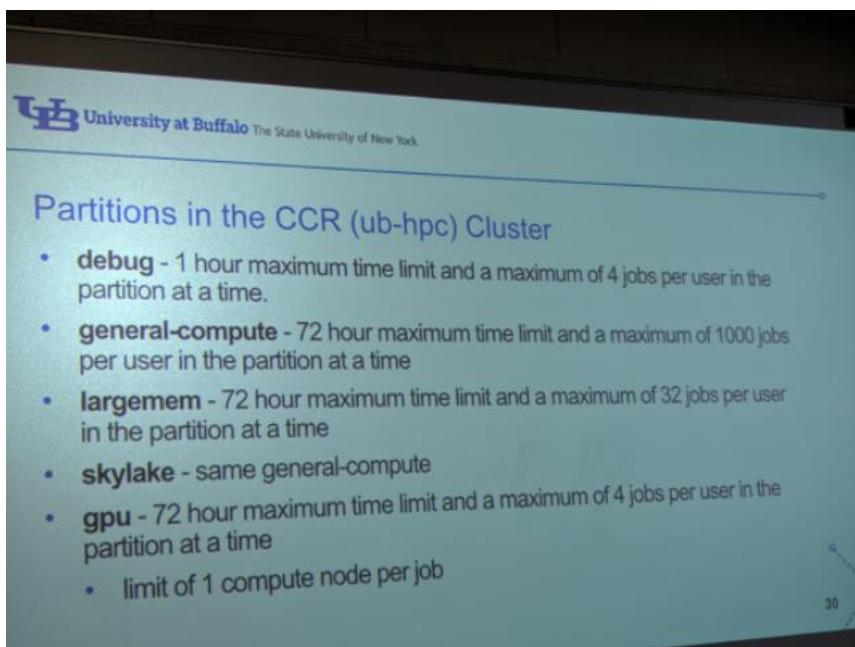
snodes show node info and status

fisbatch submit an interactive job

salloc request cluster resource for interactive job

Partitions in CCR (ub-hpc) Cluster

(long story short: skylake is the best one)



NOTE: use the “**snodes**” command to use the details of the compute nodes in partition.

An example job script

Go to the directory of interest to submit the script!



University at Buffalo The State University of New York

An Example Job Script

```
#!/bin/sh
#SBATCH --partition=general-compute
#SBATCH --qos=general-compute
#SBATCH --time=00:15:00
#SBATCH --nodes=1
#SBATCH --ntasks-per-node=1
#SBATCH --mem=48000
#SBATCH --job-name="test"
#SBATCH --output=test-%j.out
##SBATCH --mail-user=user@buffalo.edu
##SBATCH --mail-type=END
```

Job scripts begin with a list of SBATCH directives. These directives inform the scheduler about the resources needed for the job along with who to contact when the job completes and what to do with output that would normally be printed on the screen.



University at Buffalo The State University of New York

An Example Job Script continued

```
echo "SLURM_JOBID=$$SLURM_JOBID
echo "SLURM_JOB_NODELIST=$$SLURM_JOB_NODELIST
echo "SLURM_NNODES=$$SLURM_NNODES
echo "SLURMTMPDIR=$$SLURMTMPDIR
echo "working directory = $$SLURM_SUBMIT_DIR
ulimit -s unlimited
```

The next step in the job script is to configure the desired computing environment by loading required modules and setting relevant variables.

We typically also print (echo) various useful SLURM variables, like the job id, the nodelist, and the working directory. These can come in handy if there is a need to troubleshoot a completed job.

Using faculty partition

slimits to see permissions
sbatch --clusters+mae
squeue -M mae -p rgunawan
snodes all mae/rgunawan to see who's using out nodes
sinfo -M mae -p rgunawan

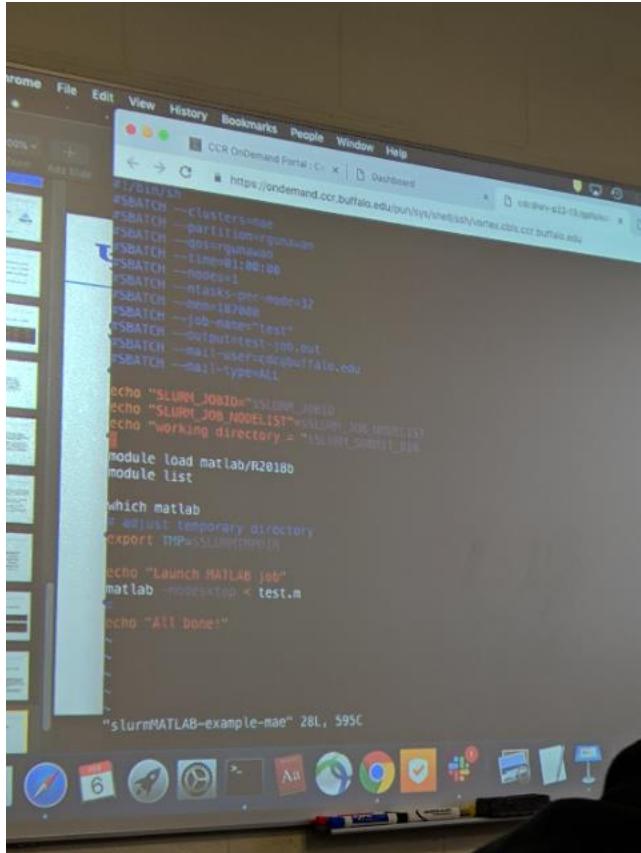
squeue -M mae --nodelist cpn-u12-12 to see the scavenger muahahah

```
sinfo -M mae -p rgunawan
```

Practical example of running MATLAB script

```
cd /gpfs/scratch/cdc
```

copy the entire folder matlab-example in your home
modify slurmMATLAB-example-mae



Cd to the matlab-example folder in you directory.

```
cd ./matlab-example
```

Now run it on the terminal with : **sbatch slurmMATLAB-example-mae**

```
[npgao@vortex1:~/matlab-example]$ sbatch slurmMATLAB-example-mae
Submitted batch job 1650996 on cluster mae
[npgao@vortex1:~/matlab-example]$ squeue -M mae -p rgunawan
CLUSTER: mae
      JOBDID PARTITION      NAME      USER ST          TIME  NODES NODELIST(REASON)
      1650996  rgunawan      test    npgao R        0:41       1 cpn-u12-11
[npgao@vortex1:~/matlab-example]$ ssh cpn-u12-11
```

more test.m (check matlab file)

more test-job.out

How to load and use Tigramite package

Start an interactive session on OnDemand and open Terminal.

```
$module use /projects/academic/rgunawan/module
```

```
$module load tigramite          (or tigramite-parallel in case you want to run in parallel)
```

```
$export XDG_RUNTIME_DIR=/projects/academic/rgunawan/scratch
```

```
$spyder
```