

Introduction to UM CCS and Pegasus supercomputer

Natalie Perlin

(nperlin@rsmas.miami.edu; hpc@ccs.miami.edu)

*Center of Computational Science,
Rosenstiel School of Marine and Atmospheric Sciences,
University of Miami
September 14, 2018*

Outline

- Center of Computational Science (CCS), University of Miami
- Advanced Computing (High Performance Computing) group:
computing, storage and database services: ccs.miami.edu/ac
- Pegasus cluster basic information
- Requesting the CCS and Pegasus account:
<https://portal.ccs.miami.edu/>
- Joining or creating a Project
- Accessing the Pegasus cluster
- Linux and Home environment. Software modules
- Data and file transfer
- Compilers, MPI, and Interpreters
- Job scheduling with LSF. Running Interactive jobs
- HPC Documentation. Help via RT tickets: hpc@ccs.miami.edu

Center of Computational Science (CCS), University of Miami

<http://ccs.miami.edu/>

The screenshot shows the CCS website on a Mac OS X desktop. The browser window title is "The Center for Computational Science". The address bar shows the URL "ccs.miami.edu". The page header features the University of Miami "U" logo and the text "UNIVERSITY OF MIAMI CENTER for COMPUTATIONAL SCIENCE". Below the header, there's a "Focus Areas" section with a grid of images related to advanced computing, big data, and other scientific fields. A "Research Highlights" section follows, displaying small thumbnail images of various projects. At the bottom, there's a brief summary about prognostic models in coronary artery disease.

"Prognostic models in coronary artery disease: Cox and network approaches"
There may be characteristic factors involved in prognostic stratification whose complexity suggests an exploration beyond the analysis provided by the still fundamental Cox approach.

Several groups and focus areas :

- Advanced Computing
- Big Data Analysis and Data Mining
- Climate and Environmental Hazards
- Computational Biology and Bioinformatics
- Drug Discovery
- Engagement and Outreach
- Social Systems Informatics
- Software Engineering
- Smart Cities
- Visualization

Advanced Computing group

<http://ccs.miami.edu/ac>

<http://ccs.miami.edu/ac/docs>

-  Maintains one of the largest centralized academic High Performance Computing (HPC) cyber infrastructures in the country
-  Compute resources, data storage, and database services for UM students, faculty, and staff: over 1500 CCS users
-  Pegasus supercomputer: Linux/CentOS 6.5 batch/interactive compute cluster, ~ 5,500 cores based on IBM iDataPlex systems
-  Visualization clusters, and dedicated graphical nodes on Pegasus
-  Integrated storage environment: high-capacity dedicated storage and high-speed storage optimized for performance, all supported by the IBM SpectrumScale (former GPFS - general parallel file system)
-  Systems and Cloud Storage maintenance

Pegasus cluster info

- Location: independent network provider, Verizon Terremark NAP of the Americas Datacenter in Miami
- Based on IBM iDataPlex dx360 M4 systems
- Computational nodes: 16 cores (two 8-core SandyBridge Intel 2.6 GHz processors)
- Total of 5500+ computational cores
- Node memory: 32GiB (2 GiB per core) and several larger node for memory-intense computations
- Node interconnect: Mellanox FDR InfiniBand fiber optic cables
- Login and management nodes
- CentOS 6.5 Operating System
- LSF (Load Sharing Facility) job scheduler
- Dedicated high-volume DDN storage (diskless)
- IBM SpectrumScale (GPFS) filesystem, designed for massively parallel work
- Large suite of compilers, MPI, standard scientific and optimized libraries
- Software modules with most demanding applications

Requesting and obtaining the CCS and Pegasus account

- NB: All the HPC resources including the portal web site are accessed **ONLY ONLY ONLY** via secure campus network (wired or wireless “SecureCanes”) or VPN connection!
- Register at the CCS/Pegasus User Portal: <https://portal.ccs.miami.edu/>



- Register as CCS User:
 - Enter your University CaneID to proceed
 - Must have a valid UM e-mail address to register
 - Follow the instructions sent to your UM e-mail to confirm your identity, affiliation with the UM
- Use User Portal:
 - Login, view or change your account info
 - Join or create a Project, for scratch (work) space allocation, software requests
 - Reset Password

My Pegasus

Home / My Pegasus

Edit Profile
My Projects
My Tasks
My Requests
My Resources

Account settings and resource management

Edit Profile

CCS account settings and password reset

CCS users can update account settings (including default shell and e-mail address) and reset passwords here.

My Projects

Projects owned, administered, and joined

Access to computing and storage resources managed by CCS Advanced Computing is granted through projects. Click 'Read More' to access the project page.

Project owners can assign an administrator to each project. Both project owners and administrators can manage project membership and request new resource allocations via the CCS Portal. To request updates to existing resource allocations, contact the resource owner directly. To manage membership, see the 'Members' section of each project page.

Member management

On the project page, click the EDIT button under 'Members'.

To add a member, type a CCS username into the text box that appears then click the ADD button.

To remove a member, click the 'x' next to the CCS username on the Members list.

When finished modifying the Members list, click the SAVE button to commit your changes (or the CANCEL button to exit editing without saving any changes). If you encounter an error removing a CCS account from your project, please contact CCS AC for assistance: hpc@ccs.miami.edu

My Tasks

Pending resource requests

Resource owners and administrators can approve, edit, and remove resource allocations here. This includes project requests, project resource allocation requests, and software requests. Previously approved project allocation requests are also listed.

My Requests

Resource request status and history

Resource requestors can see their request statuses and histories here, including: project requests, project resource allocation requests, and software requests.

My Resources

Approved and active resources

See all active project resource allocations related to your CCS account here. Project owners and administrators can also request updates.

Contact Us

Full Name:
Natalie Perlin

Email:
nperlin@rsmas.miami.edu

Comment:

[Submit](#)

Joining or creating a Project

- Login to <https://portal.ccs.miami.edu/> with your CCS User and password credentials
- Top menu: choose “**Forms and Access**”, left menu: “**Project**”
- **To Create a new project:** fill the form *Project Request Detail* .
Note: approval of the projects is done by the allocation committee, and the projects requesting Scratch Size > 2TB or/and CPU hours > 50,000 hours could take longer time to be reviewed and approved (> 1 month).
- **To join an existing Project:** contact your RSMAS advisor or faculty member!
 - Project Users have higher priority in job queues
 - Project Users get access to the project scratch space shared by the project users, in addition to their \$HOME directory 250G allocation

\$HOME	/scratch/projects
250 G	2 T *
permanent, not purged	high-speed, purged
/nethome/uid	/scratch/projects/projectid

* - Project members have read and write access to this space. Subfolder as needed!

* - Data older than 3 weeks are subject to purge

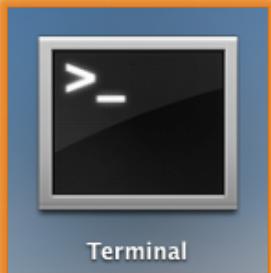
Accessing Pegasus cluster

with Secure Shell (SSH)



SSH – Linux and Mac OS X

- Use the included SSH in Terminal

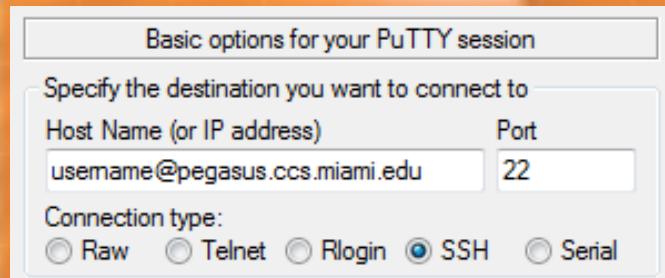


```
$ ssh username@pegasus.ccs.miami.edu
```

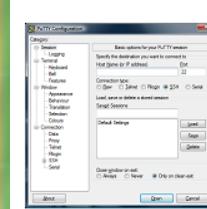


SSH – MS Windows

- Use an SSH client like PuTTY



- Download from:
<http://www.putty.org>



Download PuTTY

PuTTY is an SSH and telnet client, developed originally by Simon Tatham for the Windows platform. PuTTY is open source software that is available with source code and is developed and supported by a group of volunteers.

You can download PuTTY [here](#).

- Choose the SSH client

For Windows on Intel x86

PuTTY: [putty.exe](#)

(or by [FTP](#))

Accessing Pegasus cluster

Enable graphics, graphical display forwarding with X11



SSH – Linux:

- Set “**X11Forwarding yes**” in SSH config: in **/etc/ssh/ssh_config**
- Use –X flag: **\$ ssh -X username@pegasus.ccs.miami.edu**



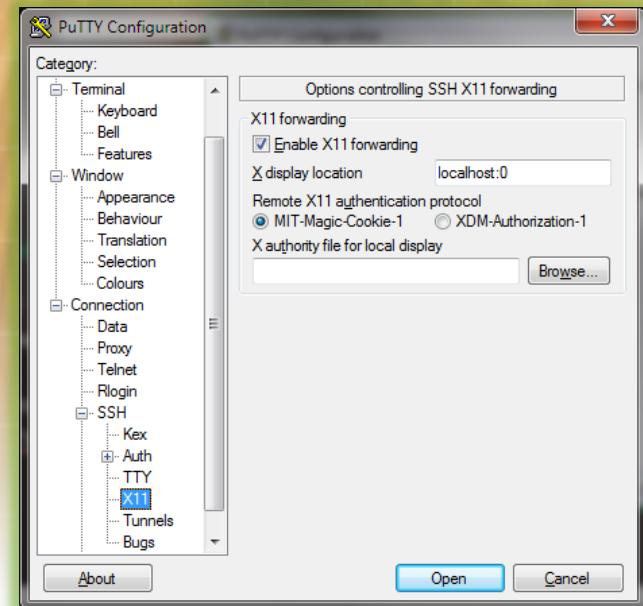
SSH – Mac OS X:

- Install XQuartz X11 created by Apple as a community project to continue support X11 on OS X:
<http://xquartz.macosforge.org/>
- Use –X flag: **\$ ssh -X username@pegasus.ccs.miami.edu**



SSH – MS Windows

- Install the Xming X server for Windows:
<http://sourceforge.net/projects/xming/>
- Launch Xming **before** launching PuTTY
- Configure PuTTY to use X11:
 - From the left Configuration menu, find **Connection-> SSH -> X11**
 - Check the box “**Enable X11 forwarding**”
 - Add “**localhost:0**” to X display connection
 - Save Session settings in PuTTY to remain



Linux and Home environment.

Software modules

Home environment is **Linux bash shell** (or other Linux/Unix shell of your choice), and customized by environmental variables

- view them by typing “env” or “env | sort”
- Your home directory is env. variable \$HOME=/nethome/**uid** , \$USER=**uid**
- check a particular env. variable setting with “echo \$**VARIABLE**”, for instance, “echo \$SHELL”

Software is accessible by loading the appropriate **Modules**

- **module avail** (gives you a list of available modules)
- **module list** (gives you a list of modules loaded into your current environment)
- **module load modulename** (loads the requested module environment)
- **module display modulename** (displays module environment and software installation information)
- **module unload modulename** (removes the environment associated with the module from your profile)

Request a new software at <https://portal.ccs.miami.edu/>

Forms and Access (top menu) -> **Software** (left menu)

- We only globally install software when we have multiple requests for the software

Install a new software in your own home directory if needed

Data and file transfer to/from Pegasus

<http://ccs.miami.edu/ac/docs/#fts>

Command line programs in a terminal:

- **scp**
- **rsync**
- **sftp**

SCP file to Pegasus:	<code>bash\$ scp /local/file.txt uid@pegasus.ccs.miami.edu:/nethome/uid/file.txt</code>
SCP directory to Pegasus (recursive):	<code>bash\$ scp -r /local/localdirectory uid@pegasus.ccs.miami.edu:/nethome/uid/</code>

GUI interface: *FileZilla* (multi-platform)

- Download a specific client:
 - <https://filezilla-project.org/download.php?type=client>
 - choose a specific client from “**Show additional download options**”



Download FileZilla Client for Mac OS X

The latest stable version of FileZilla Client is 3.36.0

Please select the file appropriate for your platform below.

Mac OS X X



This installer may include bundled offers. Check below for more options.

Requires OS X 10.9 or newer

More download options

Other platforms:

Not what you are looking for?

Show additional download options



MS Windows: other programs of your choice

- PSFTP (PuTTY FTP)
- Cygwin FTP

Compilers, MPI, and language interpreters

- GNU compilers : **gcc/4.4.7** (default for gcc) module, version 4.4.7 20120313
 - **f77, f95/gfortran**, - Fortran compilers
 - **gcc** - C and C++ compiler
- Intel compiler suite : **intel** module, version composer_xe_2013.2.146
 - **icc, icpc** - Intel C , C++ compilers
 - **ifort** - Fortran compiler
- Portland Group's PGI Accelerator : **pgi/13.10** module, version 13.10
 - **pgf77, pgf90** - PGI Fortran compilers
 - **pgcc, pgCC/pgc++** - PGI C, C++ compilers
- MPI builds:
 - **impi** - Intel-built MPI module
 - **openmpi/1.7.5** module built with Intel ifort, icc compilers
 - **openmpi-gcc/1.6.2** module (default for openmpi-gcc), build with GNU compilers
- Language interpreters:
 - **python** - modules **python/2.7.3** (default for python), **python/3.3.1**
 - **perl** - module **perl/5.18.1**
 - **R** - modules **R/2.15.2, R/3.0.0, R/3.0.1, R/3.1.2**

Job scheduling with LSF (batch jobs)

- The computational jobs are scheduled to run on processing nodes in using the LSF (Load Sharing Facility) resource manager, LSF docs: <http://www.ccs.miami.edu/hpc/lsh/9.1.1/>
- Create a script with all the commands to launch your application as a bash or csh script.
- Supply the header with the information for the LSF (see examples).
- Submit your job : **[login3]\$ bsub < myjob.sh**
- Common LSF commands:
 - **bjobs** - displays your running and pending jobs
 - **bkill jobID** - terminates/cancels a job identified by jobID number

Example of a serial (1-cpu) job script header:

```
#!/bin/bash
#BSUB -J serialjob
#BSUB -n 1
#BSUB -q general
#BSUB -W 3:00
#BSUB -R "rusage[mem=512] span[hosts=1]"
#BSUB -o %J.out
#BSUB -e %J.err
#
myjob.exe
```

Example of an MPI job script header (with Intel mpi):

```
#!/bin/bash
#BSUB -J mpijob
#BSUB -n 32
#BSUB -q general
#BSUB -W 5:00
#BSUB -R "rusage[mem=28000]"
#BSUB -R "span[ptile=16]"
#BSUB -o %J.out
#BSUB -e %J.err
#
# Run an MPI job with the "hydra" MPI job starter
mpiexec.hydra myjob.exe
```

Running interactive jobs

- Running interactive (non-graphical) job, starting on regular compute nodes:

```
[login3]$ bsub -Is $(your_job_and_options)
```

```
[login3]$ bsub -Is matlab -nodisplay
```

- There is an interactive queue for starting interactive graphical jobs:

```
[login3]$ bsub -q interactive -Is -XF matlab
```

```
Job <4427514> is submitted to queue <interactive>.
```

```
<<ssh X11 forwarding job>>
```

```
<<Waiting for dispatch ...>>
```

```
<<Starting on n002>>
```

- Upon exiting the interactive job, you are returned to your login node.
- Make sure you have the graphical display forwarding (X11) set-up and enabled on your local machine/terminal if running a graphic applications.

HPC Resources and Documentation

-  Policies: <http://ccs.miami.edu/ac/policies/>
-  Documentation: <http://ccs.miami.edu/ac/docs>
-  Pegasus Linux User Guides: <http://ccs.miami.edu/ac/docs-linux/>
-  CCS User Portal: <https://portal.ccs.miami.edu/>
-  Support and FAQ: <http://ccs.miami.edu/ac/support/>
-  Help and questions : hpc@ccs.miami.edu



Dear HPC, ...

Sending a mail to hpc@ccs.miami.edu creates an RT (Request Tracking) ticket and assigns a ticket number. Further communication including this number will ensure the problem is tracked till it solved, and then the ticket could be closed.

Happy Computing!