Introduction to Cuda Visualization

The CUDA programming paradigm is NVidia's development tool which is used to enable advanced computer processing on their GPGPU (General Purpose graphics Processing Units) architecture. Due to the advanced capabilities of these cards it can enable us to carryout out advanced visualization at fast rates using single or multiple GPGPU configuration on a system. This tutorial will carry through the different steps needed to configure the Nvidia K40 on Clemson's palmetto cluster to carry out visualization on your machine.

Graphical Application Tunnelling on Palmetto

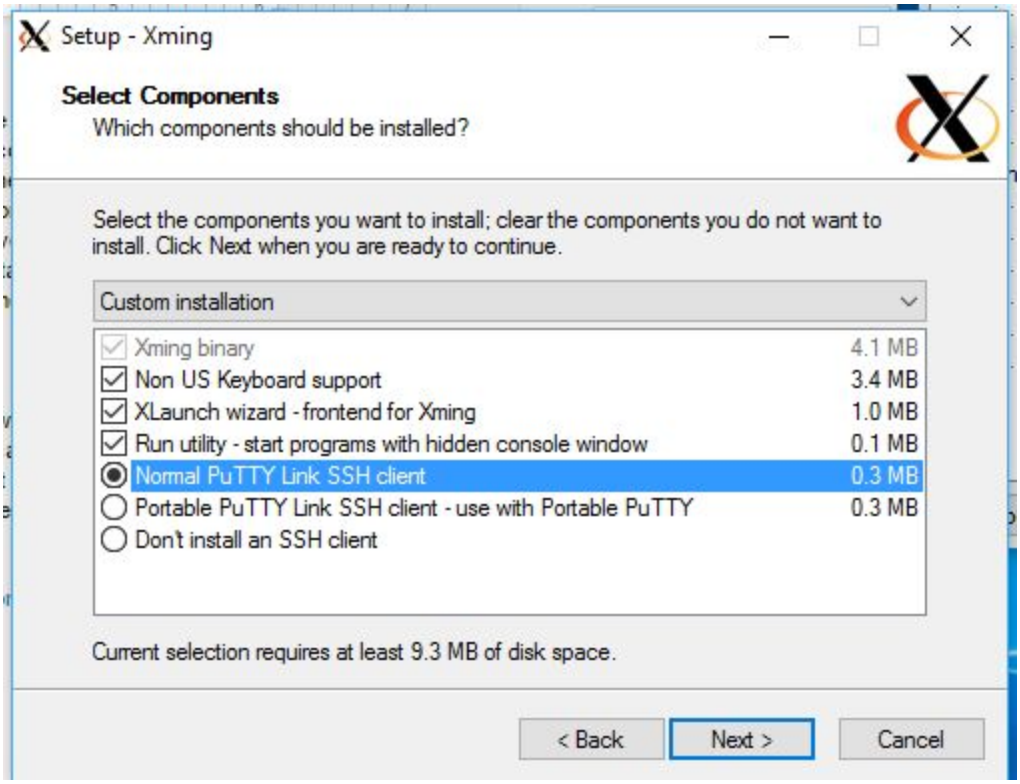
In order for us to use palmetto to run these graphic visualizations we need to enable x11 tunneling to provide the user with the visual results of the executed code or applications. This method “forwards’ or “tunnels” graphical data to your system from the palmetto cluster by starting a “x-server” that pulls the visual data onto your system. In order to enable this tunnelling via X-server; this tutorial will provide different setup instructions derived from the Palmetto user guide for Mac, Windows and Linux systems. If you have already installed tunneling from the palmetto user guide; you can skip ahead to Cuda software Requirements.

Windows 10 x11 Setup:

Unlike Mac and Linux systems, windows by default does not come with any built in x11 or ssh client to connect to palmetto (Using the newly integrated bash you can ssh but there are stability issues as of when this documentation is being created.). As a result of this you will need to install a x11 tunnelling client; and the recommended software by palmetto is Xming as listed below with url.

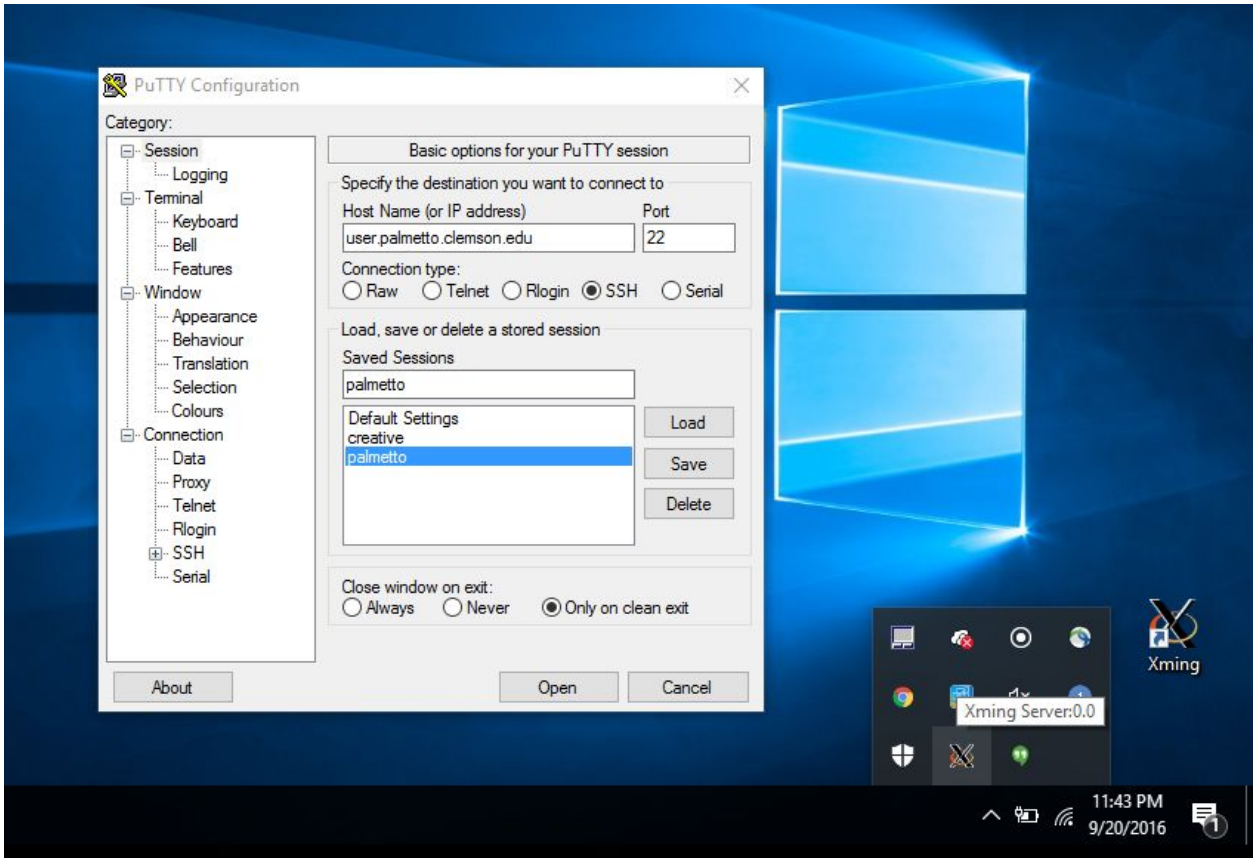
- Xming, Windows-based X-server (http://www.straightrunning.com/XmingNotes and select "Xming" under Public Domain Releases)

The following are screenshots of installing Xming with the selection of installing the putty ssh tool. Based on a few installations on different systems this combination of putty and xming has been the most stable on windows 10 at the time of writing this tutorial.

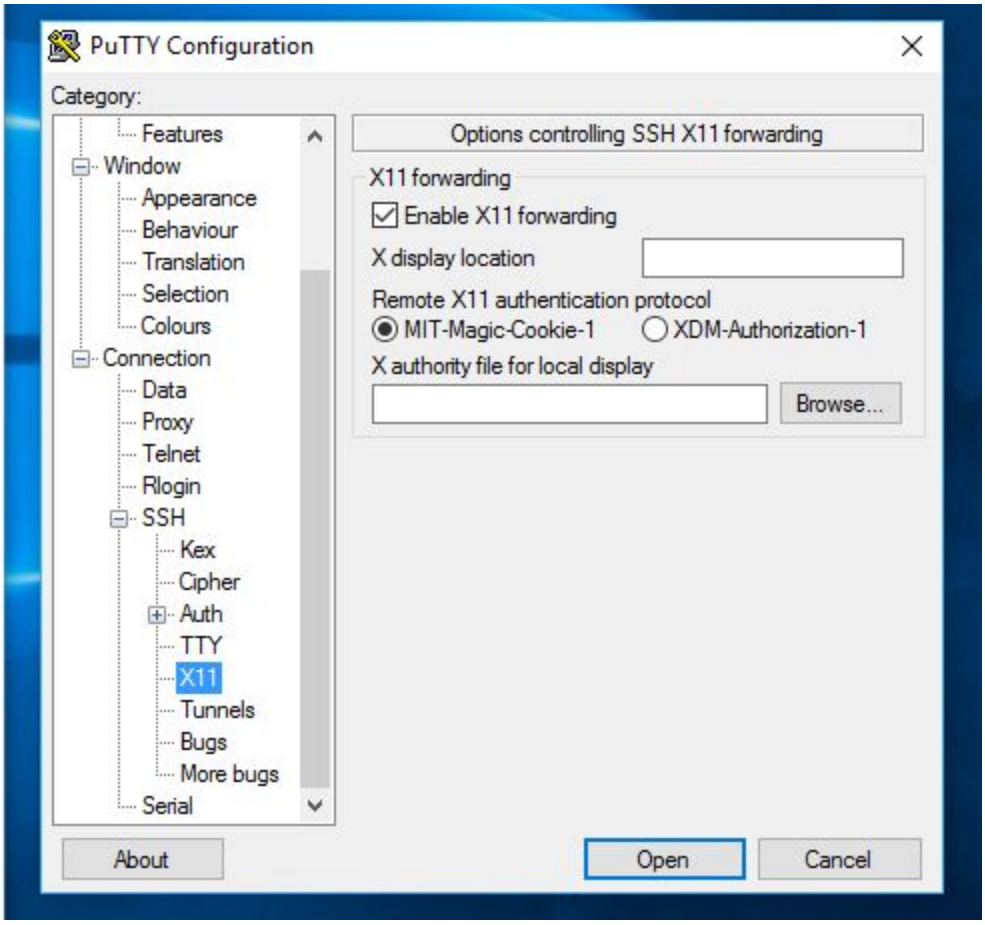


After install both these clients you must first start up xming by selecting the icon in your start menu or on desktop and an icon stating xming server will be displayed, indicating that it is running.

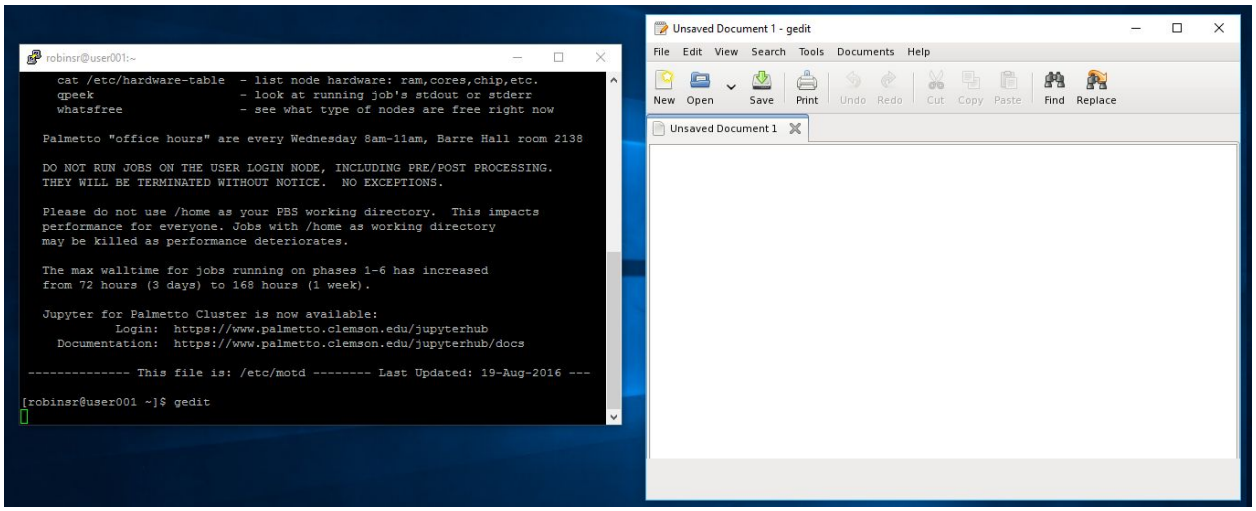
Next you can open up your putty.exe and enter the connection settings as displayed below to enable connection to the palmetto cluster.

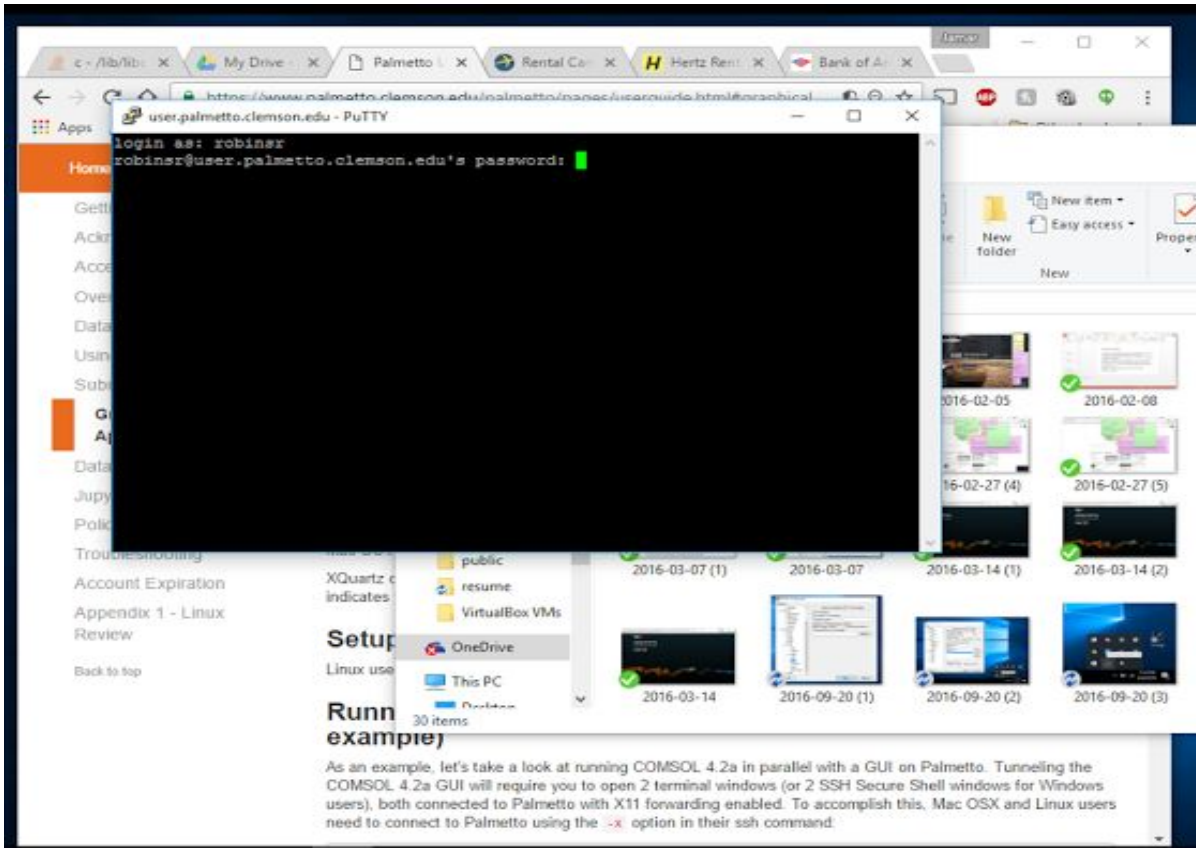


You also need to enable X11 Forwarding in the ssh menu as displayed below in the image.



After completing the Putty set up, click on open and enter your palmetto username at the login prompt and password and the palmetto login screen will be displayed. To test whether your tunneling is working correctly, enter the geddit command. If a window appears as seen below you have correctly configured your x11 tunneling.





MAC OS X setup:

Mac OS X users will need to install XQuartz (http://xquartz.macosforge.org).

XQuartz can be launched from Applications > Utilities. When you see the XQuartz "X" icon on your dock, that indicates that it's running.

Linux Setup:

Linux users will not normally have to do any additional setup on their local machines.

VNC VIEWER

After connecting to Palmetto using your preferred ssh program on windows or a terminal on Mac OS-X/Linux, submit an interactive job with inclusion of -X tunneling command

```
qsub -I -X -l select=1:ncpus=4:ngpus=1:gpu_model=k40,walltime=02:00:00
```

*For a 2 hour VNC session as seen in the walltime variable

Next we need to start a VNC server on this terminal by executing the following command:

```
LANG=C && /opt/TurboVNC/bin/vncserver
```

It will then ask you to set up a password (I use a simple password such as “visual”) and follow question which you can enter “n” when prompted for a view only password.
The screenshot below shows these steps.

```
[robinsr@user001 ~]$  qsub -I -X -l select=1:ncpus=4:ngpus=1:gpu_model=k40,wallt
ime=02:00:00
qsub (Warning): Interactive jobs will be treated as not rerunnable
qsub: waiting for job 5561924.pbs02 to start
qsub: job 5561924.pbs02 ready

[robinsr@node0512 ~]$  LANG=C && /opt/TurboVNC/bin/vncserver

Desktop 'TurboVNC: node0512:1 (robinsr)' started on display node0512:1

Starting applications specified in /home/robinsr/.vnc/xstartup.turbovnc
Log file is /home/robinsr/.vnc/node0512:1.log

[robinsr@node0512 ~]$
```

Next take note of the node and port number which format is “nodeXXXX:Y” where XXXX is the node number and port is Y.

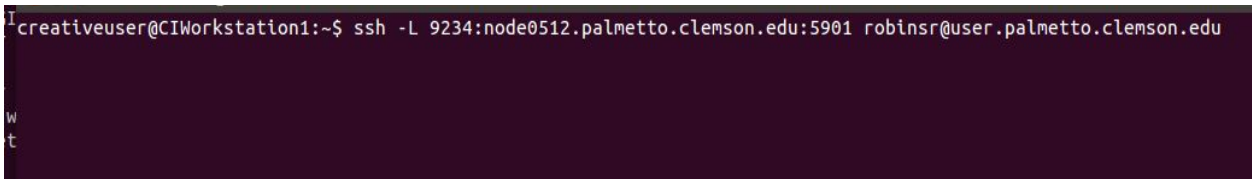
Eg. node0512:1 as shown in the example screenshot below

```
Desktop 'TurboVNC: node0512:1 (robinsr)' started on display node0512:1
```

Next open a new terminal to setup a connection to this node with the following command:

ssh -L 9234:node1978.palmetto.clemson.edu:5901 username@user.palmetto.clemson.edu

The following screenshot shows the command



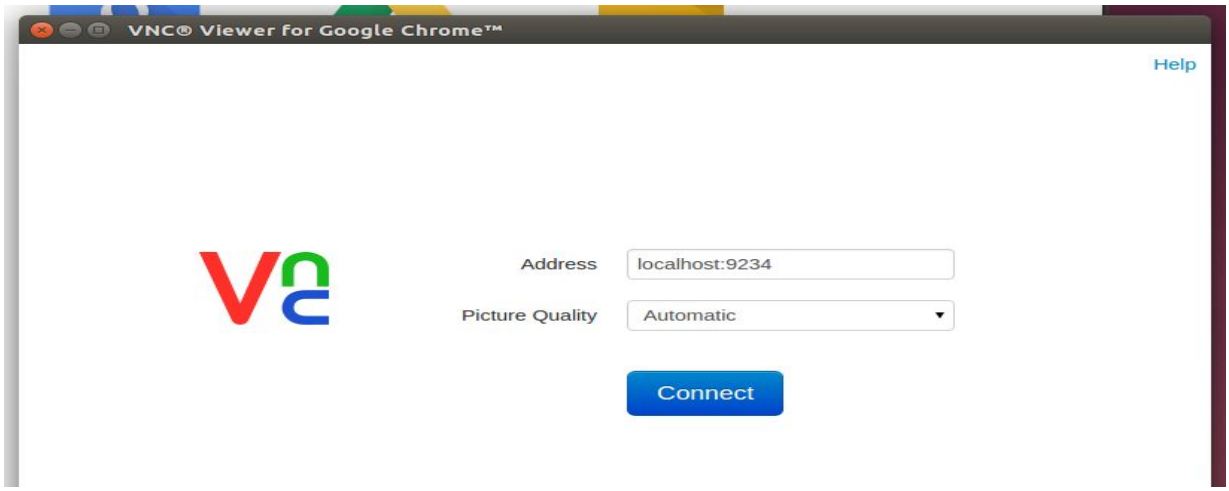
- Where 9234 is a user defined number which is usually large to ensure that port is not taken by another node on palmetto
- Node0512 is based on the nodeXXXX we obtain in the first terminal where we established a VNC server
- 5901: represents a default 5900 that you add the Y port number we obtained from the server
- username@user.palmetto.clemson.edu: is your palmetto username connection.

Next you need to obtain a VNC server client such as TigerVNC or VNCVIEWER if you use the google chrome browser.

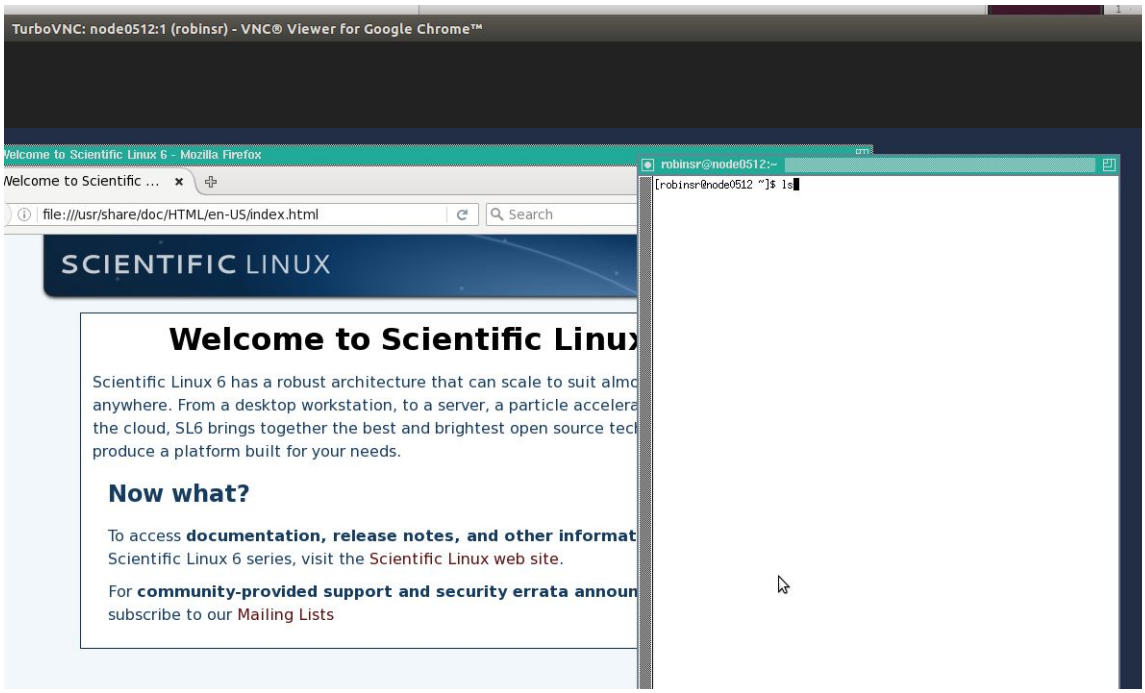
Once the VNC Viewer is running enter the following in the connection text box where 9234 is the user defined number after ssh -L as mentioned above:

Localhost:9234

Screenshot of VNCVIEWER connection on Google Chrome



After the connection is established and the window is present with a terminal and the location of your files on Palmetto. As shown below:

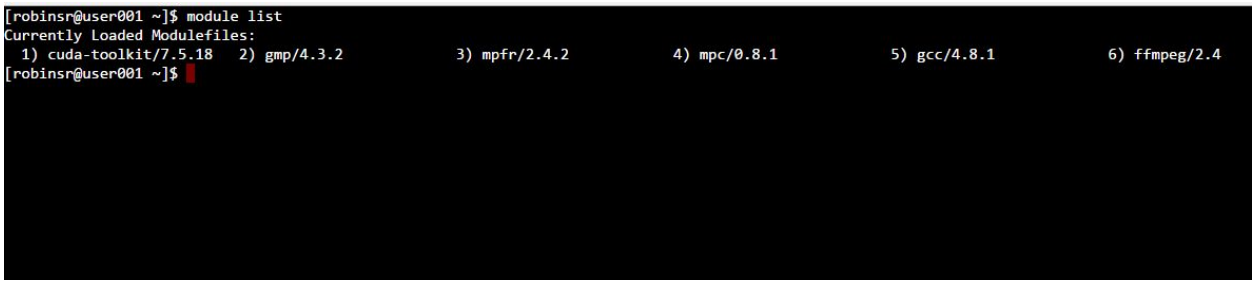


Next navigate to where you saved your sample folder and navigate to 2_graphics/Mandllebrut/. Execute the command DFLT_PATH=Lib64 make to make the sample folder.
Next execute the Mandlebrut executable by vglrun ./Mandlebrut
A window should now be displayed with the visualization example.

Cuda Software Visualization Example

Cuda Beginner Example Case

In order to enable the Cuda programming paradigm you need to add the cuda programming modules upon logging onto palmetto. To do this use the following command module add cuda-toolkit/7.5.18 and using the module list to ensure it has been loaded as shown below;



In order to get the sample cuda examples to execute and learn basic concepts of designing and executing a simplegl visualization you need to use the command

cp /software/cuda-toolkit/7.5.18/samples/2_Graphics/ .

This will copy the sample graphics software to your home directory.

After copying the file you need to execute the make command to build the samples in the file.

Mandelbrot example

1. cd to the 2_graphics directory and go to the Mandelbrot folder
2. Execute the command DFLT_PATH=/usr/lib64 make (this will make the executable)
3. Execute the command vglrun ./Mandelbrot to start the application

The following screenshots shows the above steps and result.

Mandelbrot Running