Visualization

- Visualization Consulting
- Visual Applications like Matlab
- VPN - Virtual Private Network
- Ansys Workbench
- GLX - Useful for VisIt

Visualization Consulting

UA HPC offers consulting services in visualization.

hpc-consult@list.arizona.edu: Request Technical Support from consultants

Paraview and VisIt are available on Ocelote as modules

Visual Applications like Matlab

You can utilize the GUI version of an application like Matlab using the Ocelote Desktop version of OnDemand.

Web Services#OpenOnDemand

When you choose Ocelote Desktop from the Interactive Apps dropdown at the top of the page and then choose the resources you need (like cores and wall time hours) you will get a session that is started on one of the GPU provisioned nodes. Click on the Launch noVNC in New Tab button and you will get a Desktop screen.

From there you could choose Applications, System Tools, Terminal. For an example do "module load matlab", "matlab" and then wait for all the busy Matlab things to happen.
VPN - Virtual Private Network

A VPN service (Virtual Private Network) is available for HPC, primarily for applications that cannot navigate the bastion host for visualization, and because the performance is frequently better than tunneling through the Bastion host. This is separate from the UA VPN.

Ansys Workbench

This is a GUI used to launch Ansys projects.

1. ssh -X netid@hpc.arizona.edu
2. ocelote -X
3. qsub -I -X -N jobtest -W group_list=hpcteam -q windfall -l select=1:ncpus=28:mem=168gb -l walltime=1:0:0
4. module load ansys
5. runwb2

GLX - Useful for VisIt

The functionality of GLX requires synchronization between your workstation and the Ocelote nodes. Keep in mind that on Ocelote you will first access the login nodes which do not have a graphics.

These steps provide a practical example for using the Visualization software called VisIt

1. ssh -X netid@hpc.arizona.edu
2. echo $DISPLAY  # should show localhost
3. ocelote -X
4. qsub -I -X -N jobtest -W group_list=hpcteam -q windfall -l select=1:ncpus=28:mem=168gb -l walltime=1:0:0
5. module load cuda91  # This is key as it has support for GLX although you do not need to run on a node with a GPU.
6. glxinfo
   # You might get “Bad Value” error on the Mac command line. Fixed with entering this command on the Mac: defaults write org.macosforge.xquartz.X11 enable_iglx -bool true
   # On a linux workstation you might need to create a file "/etc/X11/xorg.conf" with the following section:

   Section "ServerFlags"
     Option "AllowIndirectGLX" "on"
     Option "IndirectGLX" "on"
   EndSection

7. glxspheres64  # this is a good test of rendering back on the workstation
8. module load visit
9. visit