Paraview

ParaView is an open-source, multi-platform data analysis and visualization application. ParaView users can quickly build visualizations to analyze their data using qualitative and quantitative techniques. The data exploration can be done interactively in 3D or programmatically using ParaView's batch processing capabilities.

ParaView was developed to analyze extremely large datasets using distributed memory computing resources. It can be run on supercomputers to analyze datasets of exascale size as well as on laptops for smaller data.

For more information see http://www.paraview.org.

Usage

First you need to decide on the version of the software you want to use. Use

```
module avail paraview
```

to check what versions are available. We normally recommend using the latest version available. For example, to load the 4.3.1 version of ParaView use

```
module load paraview/4.3.1
```

For more details on using modules see our modules help guide.

If you want to use several CPUs to do the rendering we recommend the following procedure:

```
# 1) Start an interactive PBS job, for example like this:

qsub -I -X -lwalltime=2:00:00,mem=30GB,ncpus=16

# wait for a prompt. This may take some time when raijin is busy. You may want to use express queue.

# When the prompt appears,

module load paraview/4.3.1
cd your-working-directory

# 2) Start pvserver

mpirun pvserver &

# wait for the pvserver to print "Waiting for connection" or a similar message

# 3) Work with paraview.

paraview

# This should start paraview X-window.

# In the paraview X-window, go to the file menu and click connect. The "connected" message should appear in the
original ssh window.
# In the paraview X-window, go to the file menu choose a file to open and then work as you normally would.
# Note that this approach will open an additional empty 16 windows, just ignore them
# If you need to use more than 16 CPUs, the first two steps will look differently. For example, for 128 CPUs:

# 1)

qsub -I -X -lwalltime=2:00:00,mem=250GB,ncpus=128

# wait for a prompt. This may take some time when raijin is busy. You may want to use express queue.

# When the prompt appears,

module load paraview/4.3.1
cd your-working-directory

# 2) Start pvserver

mpirun /apps/paraview/4.3.1a/bin/pvserver &

# This command with start pvserver build with OSMesa. Continue with paraview as above.
```