

ANSYS on the CFAM Cluster

May 26, 2017

There are two methods by which the CFAM cluster can be accessed from a PC. The first is through a terminal emulator and the second is through a graphical user interface (GUI). If you aren't comfortable with a command line interface, the GUI may be more familiar.

You can also access the cluster with a command line interface via a terminal window on a Mac or Linux machine.

See Lynda.com for tutorials regarding the command line interface.

1 Terminal Emulator

KiTTY and PuTTY are two popular terminal emulators through which you can access the cluster. No GUI is available through these.

Both are available for download for free. See kitty.9bis.com or putty.org.

After downloading, enter `cfam-cluster.engr.unr.edu` for the Host Name and 22 for the Port. A terminal window will open and ask for your login. Enter your NetID and hit enter, then enter your password and hit enter

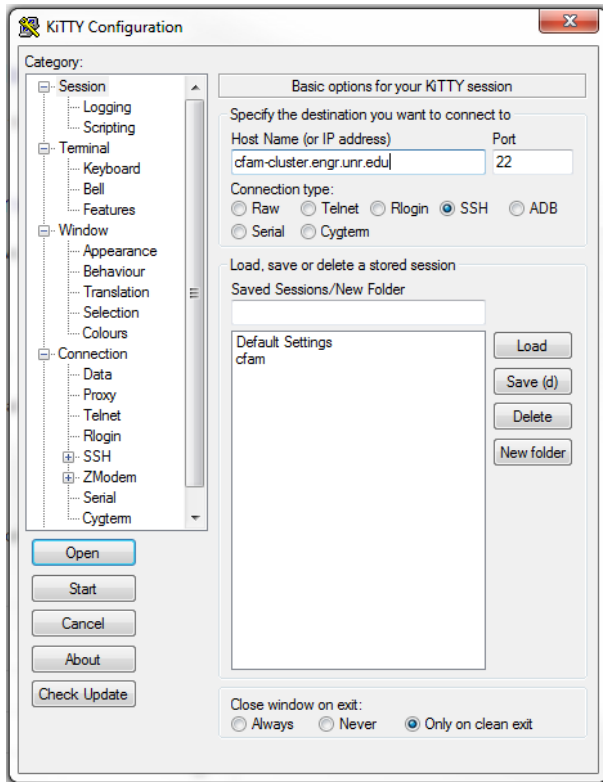


Figure 1: Kitty Configuration Window

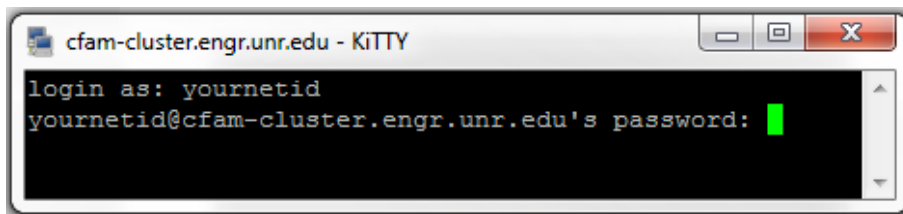


Figure 2: Kitty Terminal Window

At this point you will have access to the cluster via the command line. See below for some basic command line instruction or try Lynda.com for in-depth instruction.

2 Graphical User Interface

Through X2Go client, the cluster can be accessed with a GUI. This allows you to open the same interfaces to which you're accustomed with the power of the cluster's computational power.

You can find X2Go, free to download online. See wiki.x2go.org or simply search "X2Go." After downloading, open the client and start a new session. For the host, enter `cfam-cluster.engr.unr.edu`, for the login, enter your NETID, for SSH port, enter 22 and for the Session type, select XFCE.

Click the New Session icon to enter your password

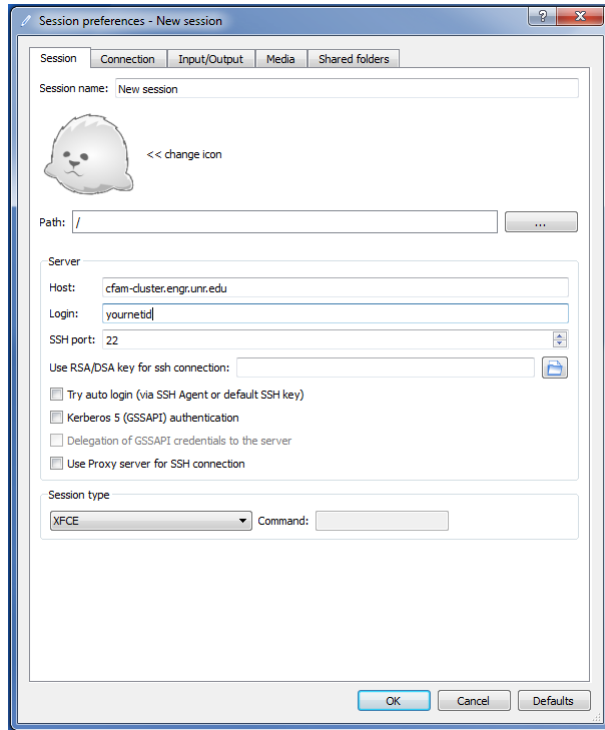


Figure 3: X2Go Session Startup Window

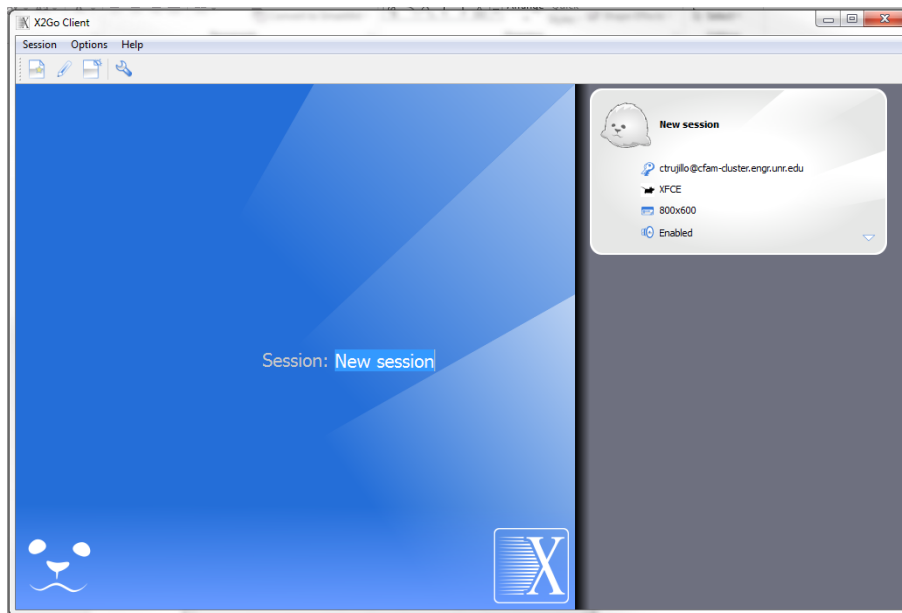


Figure 4: X2Go Startup

This will open a GUI window where you can access any files or programs on the Cluster.

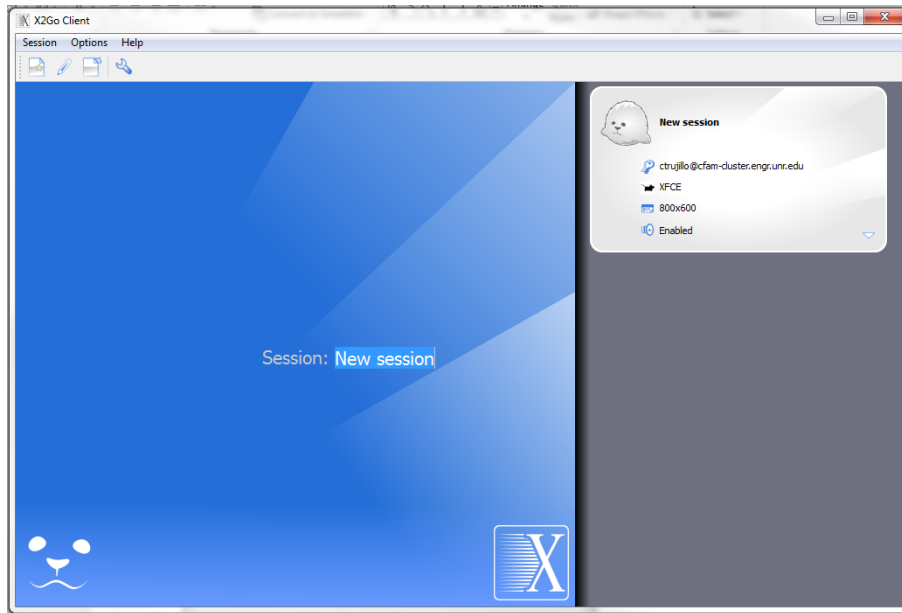


Figure 5: Password Entry

3 ANSYS on Linux

Running ANSYS on the HPC cluster requires at least a bit of background knowledge working with the Linux command line. As mentioned, the command line must be accessed through a terminal emulator - either with PuTTY, KiTTY or with the terminal emulator available on the Desktop of the GUI when accessing the cluster through X2Go.

To start open a terminal window. With X2Go, click the icon on at the bottom of the desktop. This will open a terminal window from which you can enter commands to run the programs you need.



Figure 6: Terminal Emulator Icon

One of the basic concepts to master in Linux is navigating the directory (or folder) structure. From the terminal window, the `ls` command displays the contents of the current directory. The `cd` command allows you to change the directory. To enter a directory within your current directory just type `cd <name_of_directory>`. It's simpler if directories don't

have spaces in their names. To navigate up a directory, use `cd ../` and to navigate up two directories, `cd ../../` can be used.

4 Submitting Jobs

Jobs can be submitted through a terminal window either in X2Go or through the terminal emulator on your PC. You can submit Fluent jobs or Workbench jobs. If you need the Fluent and Workbench GUIs, you must access the cluster via X2Go.

Be sure to check your usage limitations before running a job. The first place to look is Ganglia, which shows which nodes of the cluster are being used. This can be accessed by typing `cfam-cluster.engr.unr.edu` into your browser window and clicking the “Ganglia Cluster Monitor” link.

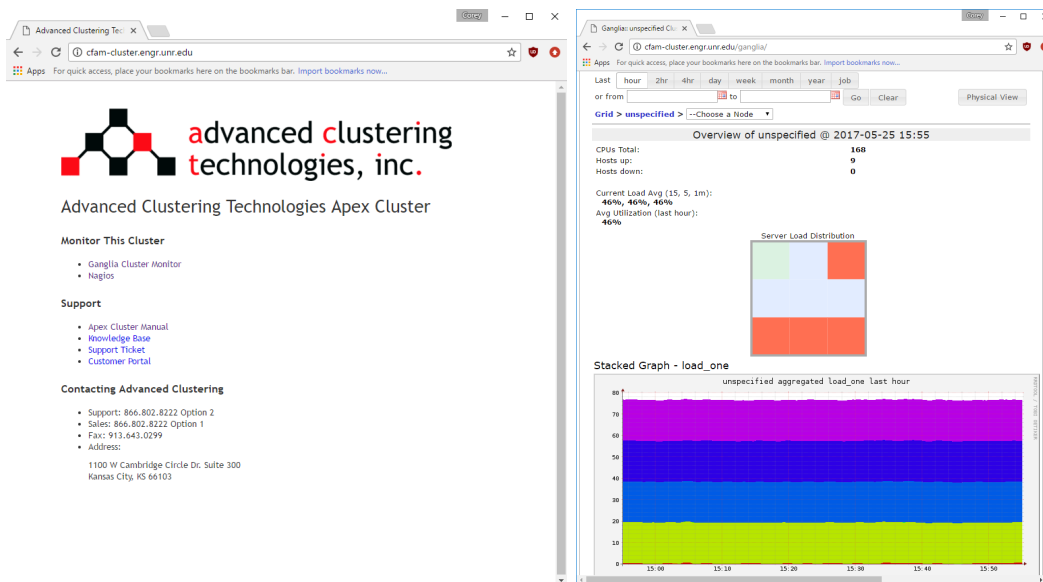


Figure 7: Accessing Ganglia

You will also want to check with the system administrator to verify what your permissions and priority are.

4.1 Submitting Interactive Fluent Jobs

The first step is to navigate to the folder where your case and data files are saved. Then you will need to submit the job

You need several nested commands to run a job using the Fluent GUI. The first is `salloc`, which allows another command to be executed on a reserved node or group of nodes. The form of the `salloc` command is as follows:

```
salloc --exclusive -N# -w nodelist command
```

The input arguments we'll use are as follows:

- `--exclusive` mandates that only your job can be on the nodes that you specify with `-w`. This means you can't share nodes with other jobs.
- `-N#` Specifies the number of nodes where `#` is the number. For example, `-N2` would specify two nodes
- `-w nodelist` tells the cluster which nodes to reserve. For example `-w highmem01,node01` would specify high memory node 1 and standard node 1. The number of nodes specified with `-N` should match the number of nodes in your nodelist.
- For the `command`, we'll use will be `srun` with the Fluent executable which will be explained below.

The `srun` command allows interactive jobs to be run. This means that your job won't run passively but you can specify changes in the Fluent GUI while once your job allocation is granted. The form of the `srun` command is as follows:

```
srun --x11 -N# <Executable>
```

- the `--x11` argument activates a plugin that allows a GUI to be used
- `-N#` specifies the number of nodes (`#`). This must be specified in the `salloc` as well as `srun`
- The `<Executable>` in our case will be the Fluent command-line executable, described below.

The current location for the Fluent executable is `/opt/ansys17-2/v172/fluvent/bin/fluvent`. If you enter only that line into the terminal emulator , the Fluent Launcher should appear:

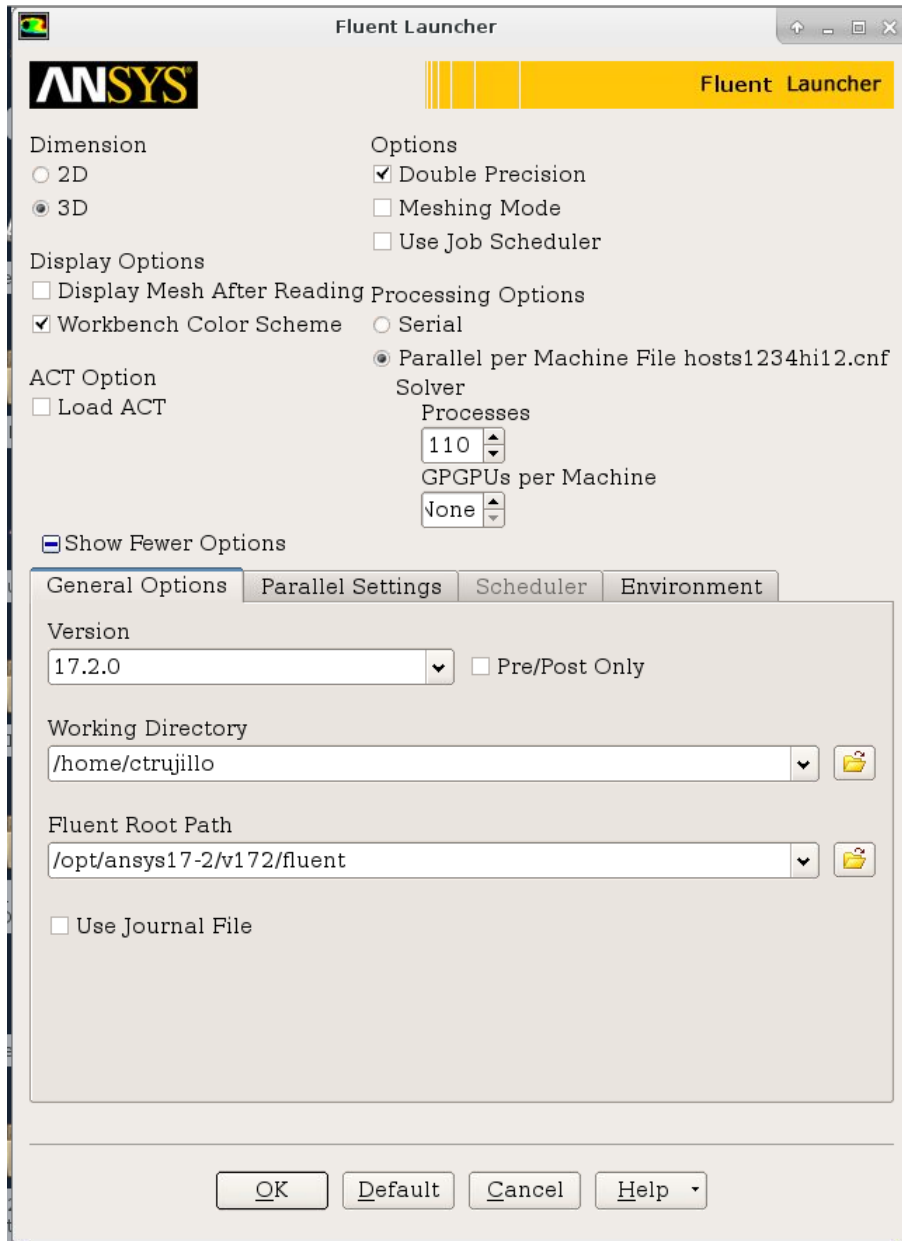


Figure 8: Fluent Launcher

However, you cannot submit a job through the launcher with `salloc` - you must bypass the Fluent Launcher to submit a Fluent job successfully. To do this, you must specify all the settings in the command line that you would usually specify in the Launcher. To see all available options, type `/opt/ansys17-2/v172/fluent/bin/fluent -help` in the terminal window.

A typical Fluent Executable with arguments might look like: `/opt/ansys17-2/v172/fluent/bin/fluent -r17.2.0 3ddp -t36 -pinfiniband -mpi=pcmpi -cnf=highmem01,node03 -nm -ssh`

- `-r17.2.0` specifies the version
- `3ddp` specifies a 3-dimensional simulation with double precision

- `-t36` specifies the number of cores. This number will be evenly divided among reserved nodes. Each node has ten cores, but they're hyper-threaded so you can specify up to double the number of cores. It is recommended that you don't specify more than 18 cores here since you'll need at least one core leftover as the host.
- the interconnect: use `-pinfiniband`, which will specify that the fastest available connections between nodes be used
- `-mpi=pcmpi` specifies the MPI which dictates how data is passed between nodes
- `-cnf=highmem01,node03` specifies to use high-memory node 1 and standard node 3.
- `-nm` allows Fluent to load faster by not displaying the mesh
- `-ssh` specifies to Fluent the kind of Linux cluster being used.

For example, you may enter the following in a single line in the terminal emulator of X2Go:

```
salloc --exclusive -N2 -w highmem01,node03 srun --x11 -N2
/opt/ansys17-2/v172/fluent/bin/fluent -r17.2.0 3ddp -t36 -pinfiniband -mpi=pcmpi
-cnfn=highmem01,node03 -nm -ssh
```

Where:

- The `salloc` command allocates node resources to your job
 - The `--exclusive` argument ensures that no other jobs will share the nodes you specify
 - The first `-N2` argument specifies the number of nodes to the `salloc` command
 - `-w highmem01,node03` specifies which nodes will be used for the `salloc` command
 - The `srun` command allows an interactive job to be run
 - * The `--x11` argument allows the GUI to be used
 - * The second `-N2` argument specifies the number of nodes to the `srun` command
 - * `/opt/ansys17-2/v172/fluent/bin/fluent` launches Fluent
 - `-r17.2.0` specifies the Fluent Version
 - `3ddp` specifies a 3D and double precision simulation
 - `-t36` specifies 36 cores are used
 - `-pinfiniband` specifies the fastest interconnect between nodes
 - `-mpi=pcmpi` specifies the MPI which dictates messaging between nodes
 - `-cnfn=highmem01,node03` specifies to Fluent which nodes are used. This was already specified to the cluster with the `-w` command but it must be re-specified to Fluent.
 - `-nm` loads Fluent without displaying the mesh, which saves time.
 - `-ssh` specifies to Fluent the kind of Linux cluster being used.

4.2 Submitting Passive Jobs through the Command Line

4.3 Submitting Workbench Jobs with the GUI