

# July 2022 User Conference Call

## Remote Visualization with VNC

Benjamín Hernández  
AI & Analytics Methods at Scale Group  
OLCF

July 27th, 2022



ORNL is managed by UT-Battelle LLC for the US Department of Energy



# Andes

## CPU nodes (batch partition)

- 704 compute nodes
- AMD EPYC 7302 - 32 cores per node
- 256 GB RAM

## GPU nodes (gpu partition)

- 9 GPU nodes
- Intel Xeon E5-2695 - 28 cores / 56 threads
- 2x NVIDIA K80 - 4 GPUs per node
- 1 TB RAM

# TurboVNC

- Users can interact with a remote desktop (xfce) on Andes
- TurboVNC use
  - RFB (remote frame buffer protocol) for optimal keyboard and mouse event and frame buffer delivery.
- The desktop's frame buffer is compressed using JPEG.

TurboVNC: andes333.olcf.ornl.gov:1 (benjha) [Tight + JPEG IX Q95 + CL 1]

Applications MATLAB R2020a Terminal

File Edit View Terminal Tabs Help

```
bash-4.4$ module load matlab
bash-4.4$ matlab
MATLAB is selecting SOFTWARE OPENGL rendering.
```

MATLAB R2020a

HOME PLOTS APPS

New Script New Live Script New Open Compare Import Data Save Workspace Open Variable Favorites Analyze Code Run and Time Clear Workspace Clear Commands Layout Set Path Add-Ons Help Learn MATLAB RESOURCES

FILE VARIABLE CODE ENVIRONMENT RESOURCES

Current Folder Editor: /gpfs/alpine/stf218/proj-shared/benjha/code/cmns-edge/gvs\_paper/analysis\_code/gLine\_Switching\_Analyzer\_v06.m

```
1 %Takes in data collected in Gmode, performs FastSVD (PCA) and saves them
2 %extracted data as well as results
3
4 - clear all;
5 - close all
6 - clc
7 - format long g;
8 - econRAM = 1; % clear non-critical variables asap
9
10 %% Select File
11
12 rootPol = '/gpfs/alpine/proj-shared/stf218/benjha/code/cmns-edge/gvs_paper/';
13 addpath([rootPol 'raw_data']);
14 addpath([rootPol 'analysis_code']);
15
16 %[filename,out_path,FilterIndex] = uigetfile({'*.dat'});
17 %temp = strfind(FileName,'_');
18 %out_path = out_path(1:temp-1);
19 in_path = '/gpfs/alpine/proj-shared/stf218/benjha/code/';
20 out_path = '/gpfs/alpine/proj-shared/stf218/benjha/code/';
21 file_base_name = 'GLine_BV_10kHz_256x256_0001';
22 %file_base_name = FileName(1:(temp(end)-1));
23 clear(['FileName','FilterIndex','temp']);
24 fprintf('working on data set: %s\n',in_path);
25
```

Details Select a file to view details Ready

TurboVNC: andes-gpu5.olcf.ornl.gov:1 (benjha) [Tight + JPEG IX Q95 + CL 6]

Applications VMD 1.9.3 OpenGL Display VMD Main Terminal

File Edit View Terminal Tabs Help

```
bash-4.4$ module load vmd
bash-4.4$ vglrun vmd
```

VMD 1.9.3 OpenGL Display

VMD Main

File Molecule Graphics Display Mouse Extensions Help

ID T A D F Molecule Atoms Frames Vol

slurm-145364.o ut keys

algebra.log temp multiprocessing\_test.py dash-worker-space

a.out summit cmb124 repodata.json

slurm-114426.o test\_sympy.py rheia isaac

peak blazingsql\_on\_summit.txt load\_WarpX.sh activate.patch protobuf.bak

Ready

## Running Matlab on a CPU node

## Running VMD on a GPU node

# When to use... ?

	TurboVNC (CPU)	TurboVNC (GPU)
Availability	Regular nodes	GPU nodes
Use case	Non graphics intensive apps, e.g. 2D graphics, any user interface (matlab, performance tools, editors, etc.)	GPU accelerated 3D graphics, apps with no client/ server architecture e.g. VMD, yt's 3D visualization, USC Chimera, custom viz. tools, CUDA+OpenGL, Latest Paraview binary from Kitware

Note compute nodes does not have access to the Internet.

# Pre-requisites

TurboVNC Server v2.2.5 is installed on Andes:

- Get vncviewer v2.2.5 and install it in your local system

<https://sourceforge.net/projects/turbovnc/files/2.2.5/>

- (Linux) If not installed, install Java JRE

```
$ sudo apt install openjdk-11-jre
```

- (Linux) vncviewer will be available under

/opt/TurboVNC/bin

# Running Turbo VNC

## TurboVNC Regular nodes

### Terminal 1 (run VNC server)

```
local$ ssh -X userid@andes.olcf.ornl.gov  
andes-login$ salloc -A abc123 -N 1 -t 0:30:00 --x11=batch  
bash$ source ./run-vnc.sh
```

## TurboVNC GPU nodes

### Terminal 1 (run VNC server)

```
local$ ssh -X userid@andes.olcf.ornl.gov  
andes-login$ salloc -A abc123 -N 1 -t 0:30:00 -p gpu  
--x11=batch  
bash$ source ./run-vnc-gpu.sh
```

### Terminal 2 (tunneling)

```
local$ ssh -L 5901:localhost:5901  
username@andes.olcf.ornl.gov  
andes$ ssh -L 5901:localhost:5901 andes683.olcf.ornl.gov
```

### Terminal 2 (tunneling)

```
local$ ssh -L 5901:localhost:5901  
username@andes.olcf.ornl.gov  
andes$ ssh -L 5901:localhost:5901 andes683.olcf.ornl.gov
```

### Terminal 3 (run vncviewer)

```
local$ /opt/TurboVNC/bin/vncviewer localhost:5901
```

### Terminal 3 (run vncviewer)

```
local$ /opt/TurboVNC/bin/vncviewer localhost:5901
```

# run-vnc.sh

```
#!/bin/sh

HOST=$(hostname)
USER=$(whoami)

echo "Starting vncserver"
/opt/TurboVNC/bin/vncserver :1 -geometry 1920x1080 -depth 24
echo
echo
echo ****
echo "Instructions"
echo
echo "In a new terminal, open a tunneling connection with $HOST and port 5901"
echo "example:"
echo "    localsystem: ssh -L 5901:localhost:5901 $USER@andes.olcf.ornl.gov "
echo "    andes: ssh -L 5901:localhost:5901 $HOST "
echo
echo ****
echo
export DISPLAY=:1
```

# run-vnc-gpu.sh

```
#!/bin/sh

HOST=$(hostname)
USER=$(whoami)

echo "Starting X"
xinit &
sleep 5

echo "Starting vncserver"
/opt/TurboVNC/bin/vncserver :1 -geometry 1920x1080 -depth 24
echo
echo
echo ****
echo "Instructions"
echo
echo "In a new terminal, open a tunneling connection with $HOST and port 5901"
echo "example:"
echo "    localsystem: ssh -L 5901:localhost:5901 $USER@andes.olcf.ornl.gov "
echo "    andes: ssh -L 5901:localhost:5901 $HOST "
echo
echo ****
echo
export DISPLAY=:1
```

## vncserver

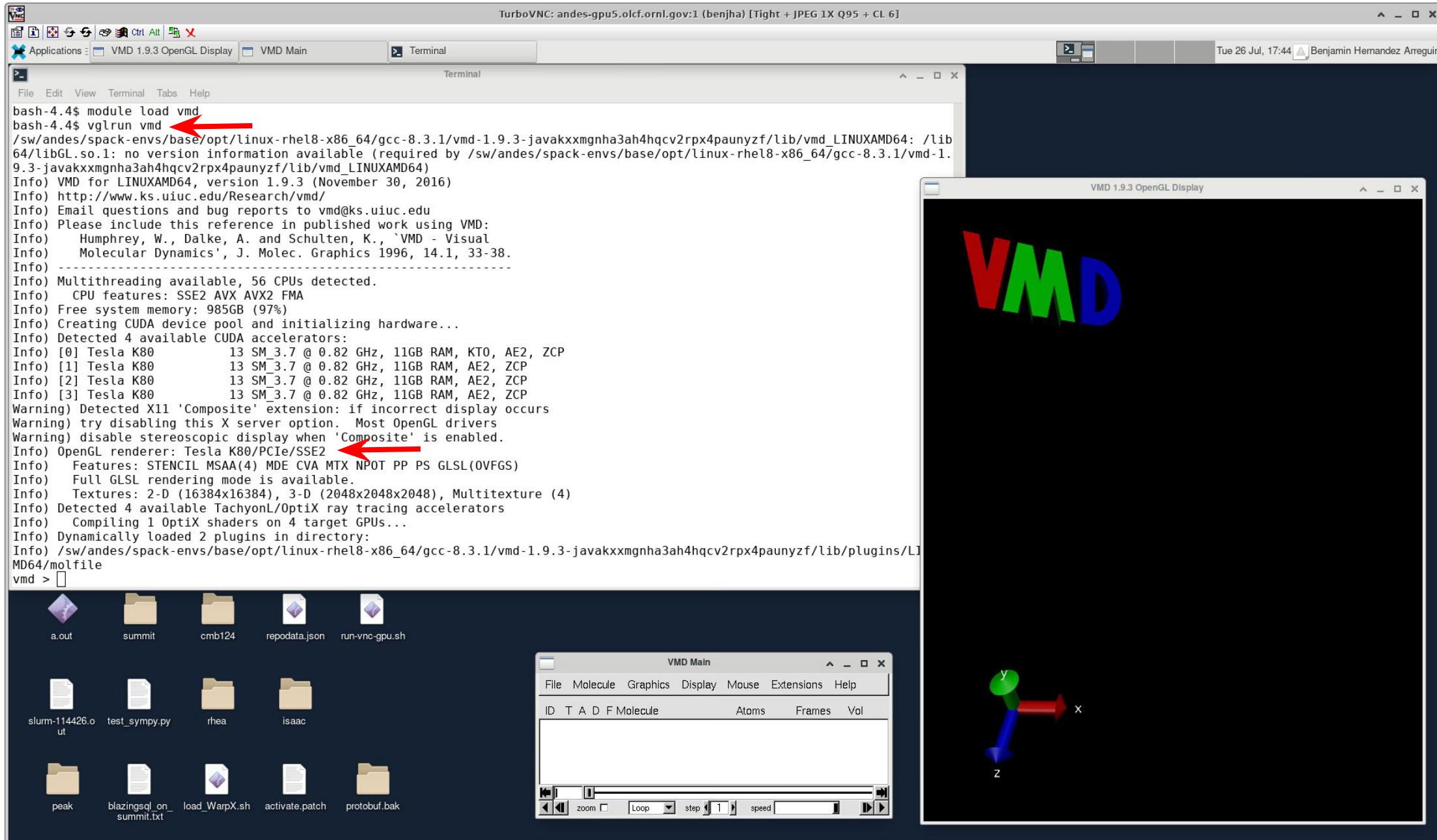
```
/opt/TurboVNC/bin/vncserver :1 -geometry 1920x1080 -depth 24
```

The diagram illustrates the command line options for starting a VNC server. The command is `/opt/TurboVNC/bin/vncserver :1`. It is followed by two options: `-geometry 1920x1080` and `-depth 24`. A bracket under `-geometry` is labeled "desktop resolution". Another bracket under `-depth` is labeled "color depth". A bracket under `:1` is labeled "display".

## vncserver

- vncserver uses ports 59xx for communication between the client
  - For this case, vncserver is using \$DISPLAY :1
  - Therefore port 5901 is used

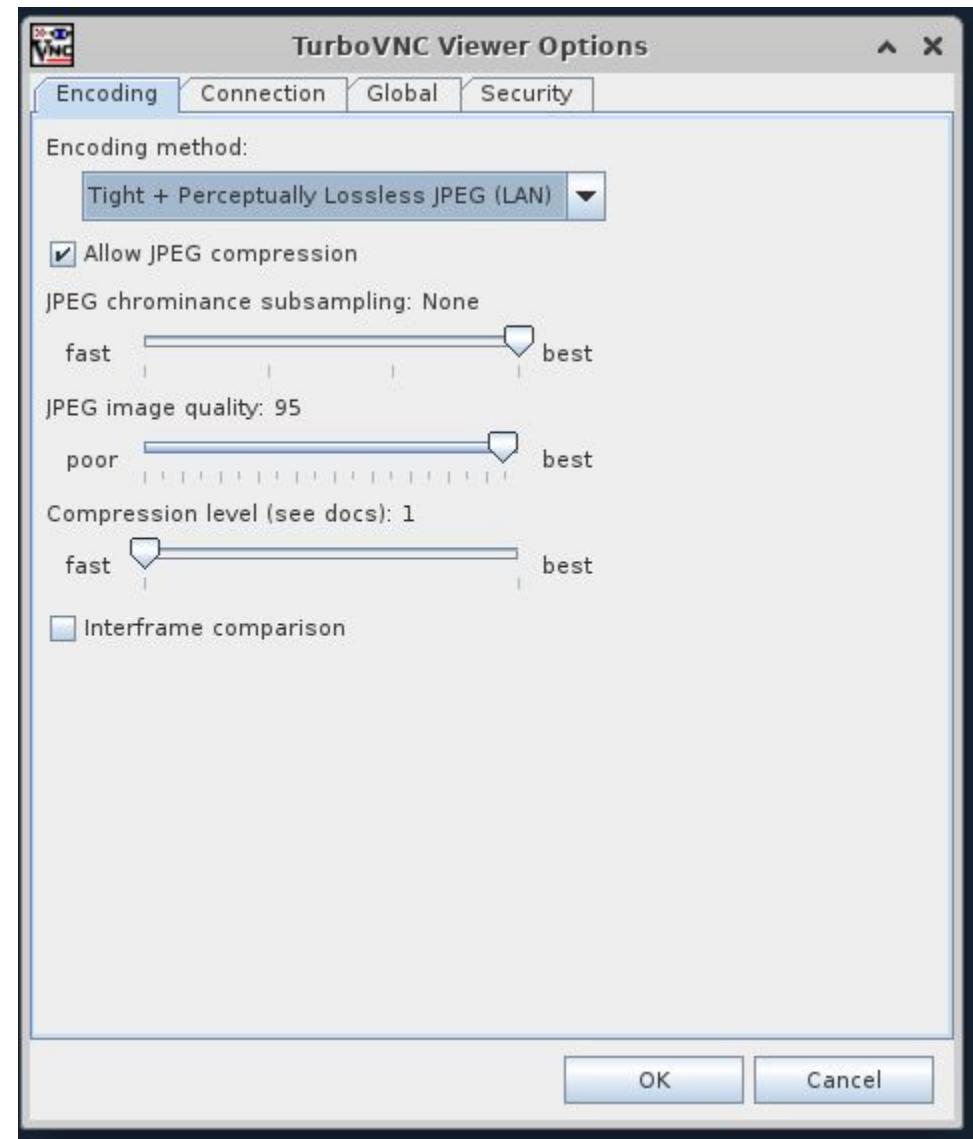
# On GPU nodes use `vglrun <program>` to get hardware accelerated graphics!



# vncviewer

- When running vncviewer for the first time, it will ask for a password. Set the password and reuse it for future sessions
- The VNC password is stored inside of the `~/.vnc/passwd` file. This directory exist in your `$HOME` directory in Andes

# vncviewer options to improve streaming



# Shutting down vncserver CPU and GPU nodes

```
bash$ /opt/TurboVNC/bin/vncserver -kill $DISPLAY
Killing Xvnc process ID 1717153
bash$ pkill xinit
bash$ exit
exit
salloc: Relinquishing job allocation 249539
andes-login$ _
```

Questions and feedback

[help@olcf.ornl.gov](mailto:help@olcf.ornl.gov)