

# Paraview Client & Server

Timothy Brown  
June 25th 2015

Paraview can be run in a client-server mode.

In this tutorial we will run the Paraview GUI (client) on your desktop and `pvserver` (server) on Yeti.

We will be using a ssh tunnel for the communication between the Paraview client and server.

Running the Paraview GUI (client) on your desktop means all the rendering is still done using your desktop graphics processor. This will be a bottleneck if you are trying to visualize huge datasets.

# Yeti

1. Log in to Yeti.

```
laptop ~$ ssh yeti.cr.usgs.gov
```



2. Start a compute job.

```
yeti-login01 ~$ sinteractive -A training \
                    -p prod           \
                    -t 01:00:00 -N 1  \
                    --exclusive
```



### 3. Start the paraview server.

```
compute80 ~$ module load viz/paraview-4.3.1-50  
compute80 ~$ module load mpi/mpich-x86_64  
compute80 ~$ mpiexec -np 12 pvserver \  
--use-offscreen-rendering
```

The pvserver will bind to port 11111 by default.

#### 4. Create a tunnel.

```
laptop ~$ ssh -f -N -L 11111:compute80:11111 \  
yeti.cr.usgs.gov
```



The tunnel takes the form of

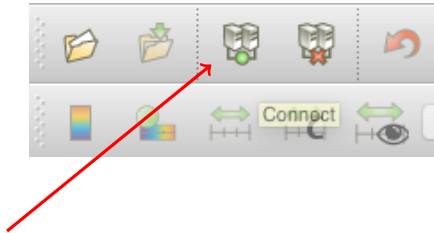
```
-L local_port:remote_host:remote_port
```

So we are forwarding the default Paraview port (11111) on our laptop to the compute node port 11111, through the login node.

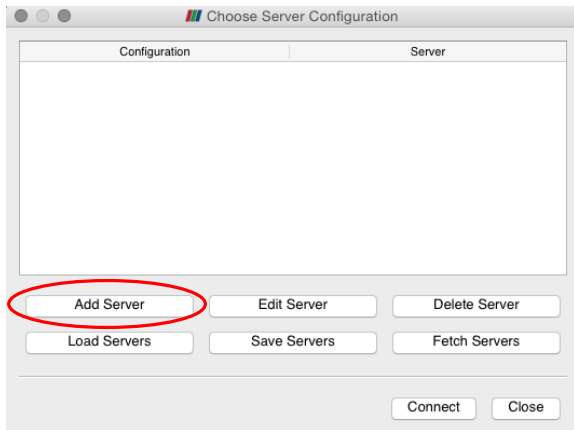
5. Start paraview locally.

```
laptop ~$ paraview
```

6. Create a new connection.



## 7. Add a new server.





## 8. New server definitions.

8.1 It can have any unique name.

8.2 Use the localhost (loop-back)

8.3 Use the local port in the ssh tunnel we just defined.

A screenshot of a dialog box for defining a new server. The dialog contains the following fields and values:

- Name:** local
- Server Type:** Client / Server (dropdown menu)
- Host:** localhost (indicated by a red arrow)
- Port:** 11111 (indicated by a red arrow)

At the bottom right of the dialog are two buttons: **Configure** and **Cancel**.

9. Use the manual startup procedure, since pvserver is already running on the compute node.

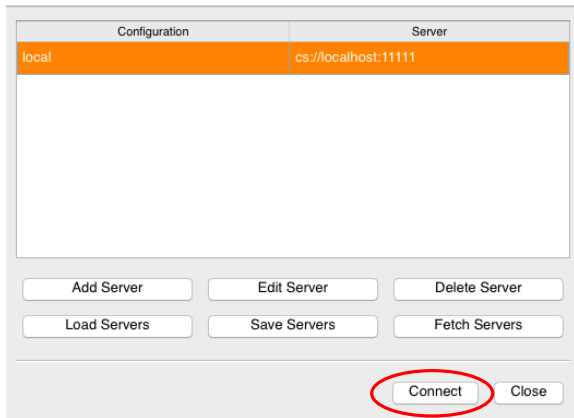
Configure server foobar (cs://foobar)

Please configure the startup procedure to be used when connecting to this server:

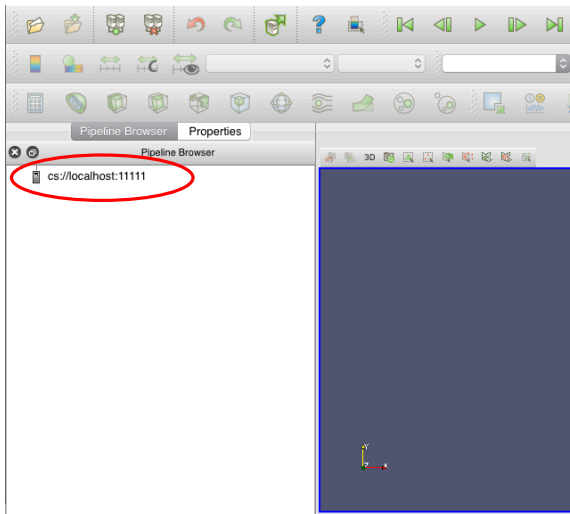
Startup Type:

Manual Startup - no attempt will be made to start the server. You must start the server manually before trying to connect.

## 10. Connect to the new server.



11. The **Pipeline Browser** should now show this connection.



# Possible Issues

## 1. Client and server version mismatch.

```
ERROR: In /Users/kitware/Dashboards/MyTests/NightlyMaster/ParaViewSuperbuild-Release-
Python27/paraview/src/paraview/ParaViewCore/ClientServerCore/Core/vtkTCPNetworkAccessManag
e.r.cxx, line 330
vtkTCPNetworkAccessManager (0x7fe548c76b80): Failed to connect to localhost:11111

ERROR: In /Users/kitware/Dashboards/MyTests/NightlyMaster/ParaViewSuperbuild-Release-
Python27/paraview/src/paraview/ParaViewCore/ClientServerCore/Core/vtkTCPNetworkAccessManag
e.r.cxx, line 340
vtkTCPNetworkAccessManager (0x7fe548c76b80):
*****
Connection failed during handshake. This can happen for the following reasons:
1. Connection dropped during the handshake.
2. vtkSocketCommunicator::GetVersion() returns different values on the
   two connecting processes (Current value: 100).
3. ParaView handshake strings are different on the two connecting
   processes (Current value: paraview.4.3).
*****
ERROR: In /Users/kitware/Dashboards/MyTests/NightlyMaster/ParaViewSuperbuild-Release-
Python27/paraview/src/paraview/ParaViewCore/ServerManager/Core/vtkSMSessionClient.cxx, line
261
vtkSMSessionClient (0x7fe54c2909b0): Some error in socket processing.
```

## 2. Firewalls

### 2.1 The login server does not allow port forwarding to the compute nodes.

- ▶ Select a **reverse connection** under the server type on slide 9.
- ▶ Start the pvserver with the additional options

`--reverse-connection --client-host=AA.BB.CC.DD`

with the appropriate hostname or IP address of your client.

### 2.2 The client is behind a firewall as well.

- ▶ Create a reverse tunnel, step 4 on slide 6.  
`ssh -f -N -R 11111:localhost:11111 yeti.cr.usgs.gov`
- ▶ Create a tunnel from the compute node back to the login node.

`ssh -f -N -L 11111:localhost:11111 yeti.cr.usgs.gov`

- ▶ Select a **reverse connection** under the server type on slide 9.
- ▶ Start the pvserver with the additional options

`--reverse-connection --client-host=AA.BB.CC.DD`

with the appropriate hostname or IP address of your client.

# License

© University of Colorado Boulder, 2015

This work is licensed under the Creative Commons Attribution 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>

When attributing this work, please use the following text:  
“Paraview Client & Server”, Research Computing, University of Colorado Boulder, 2015. Available under a Creative Commons Attribution 4.0 International License.

