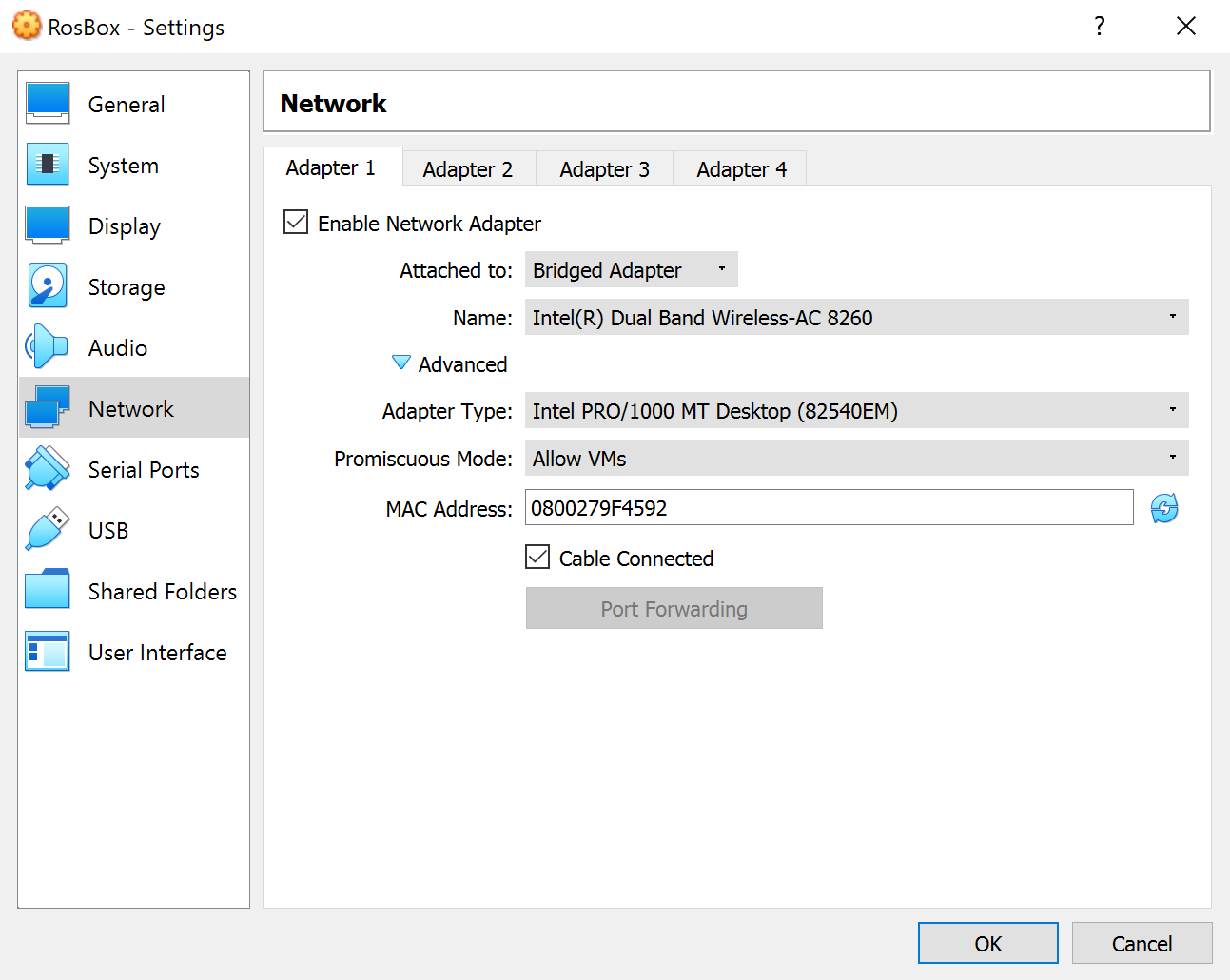
**Sending and receiving data**

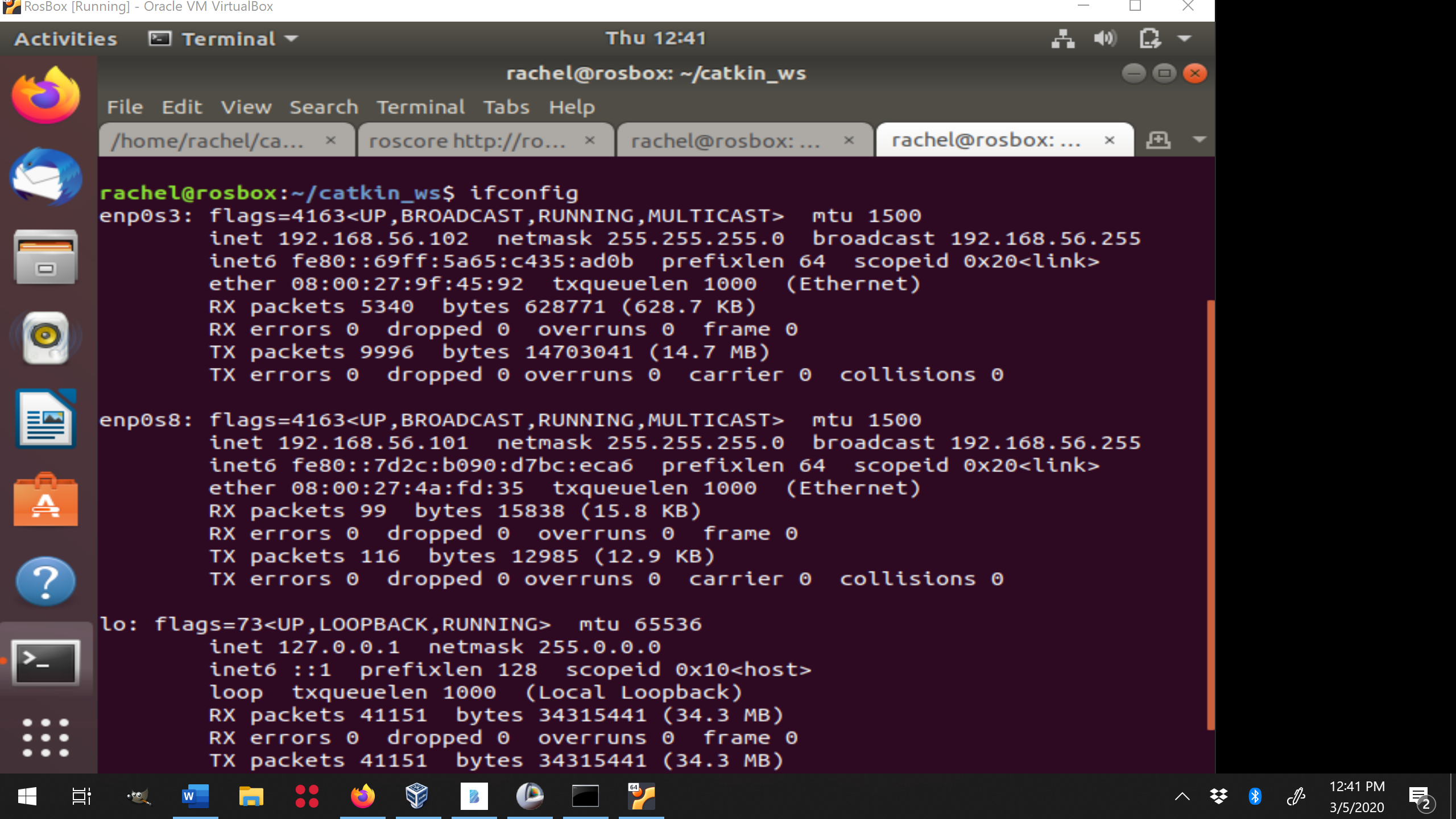
To create a two-way connection between ROS and 3DSlicer we need slightly different configurations

A host-only adapter will create an TCP/IP connection between your ROS virtual machine and your local host. It will only enable your local machine to detect this connection, ensure you set your promiscuous mode to all vms. However, this will disable your ability to connect to the internet.

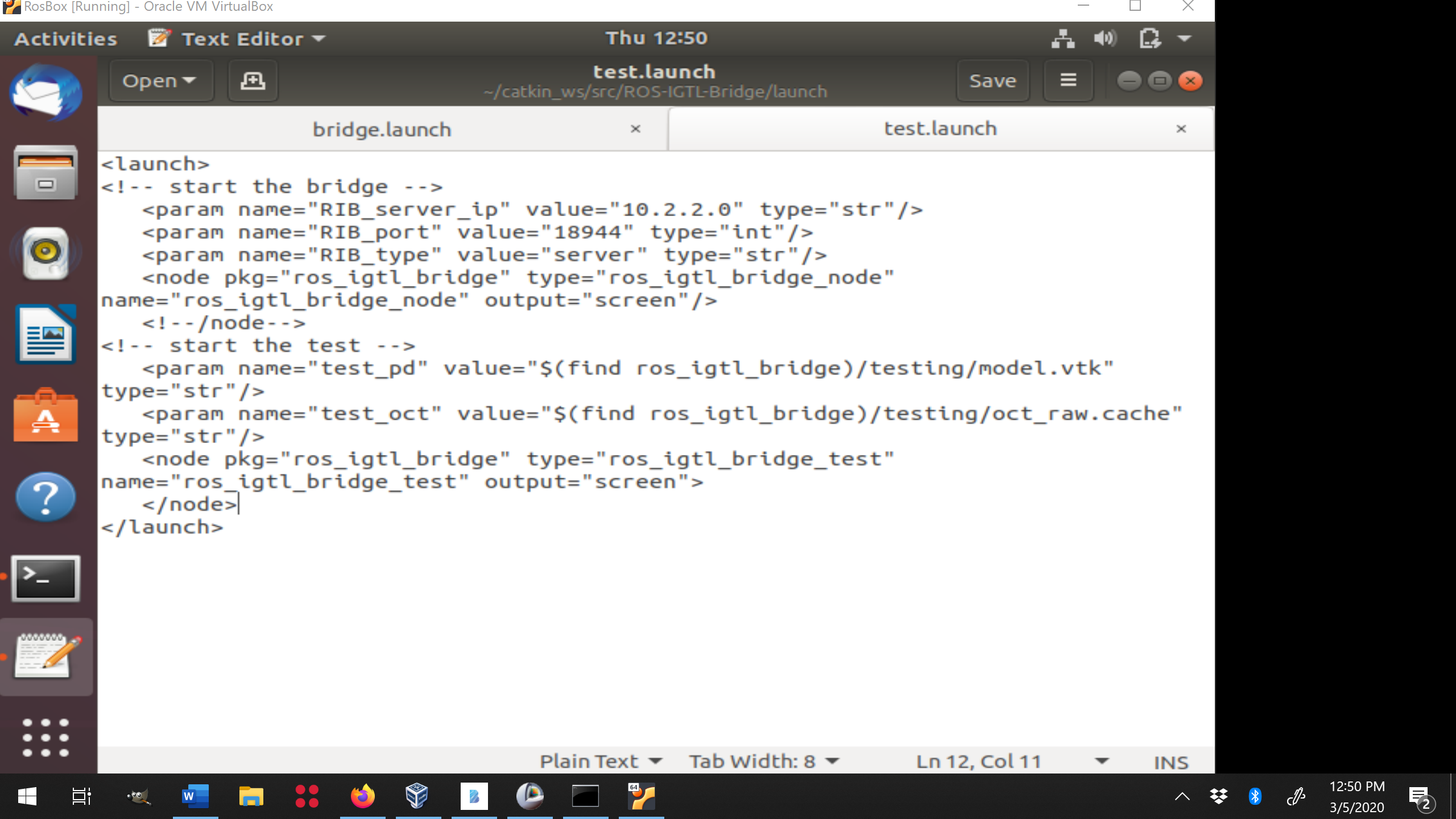
Alternatively, set a Bridge Adapter to enable your ROS virtual machine to connect to the internet and your local host.



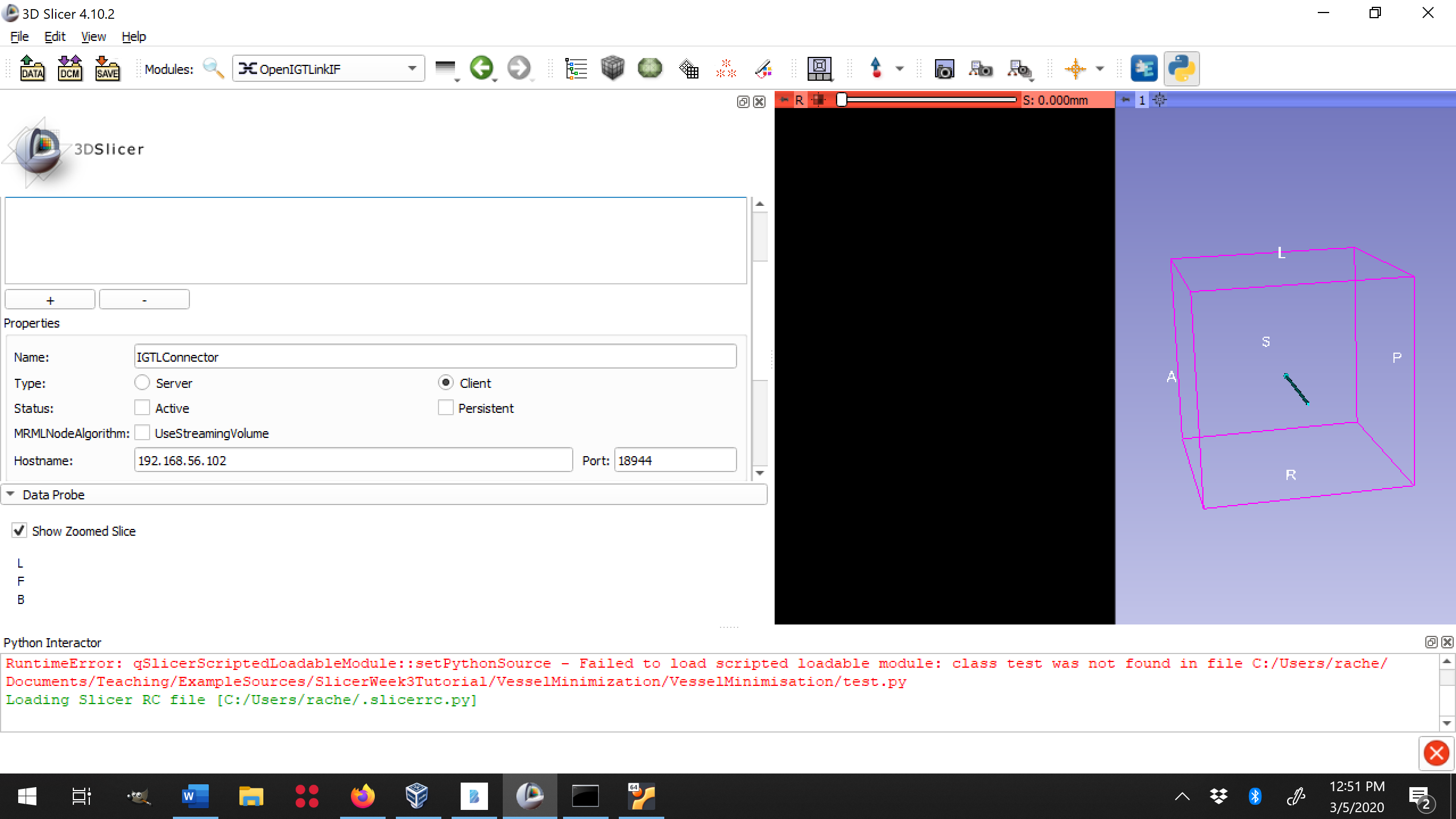
To identify the IP address of your virtual machine use “ifconfig”



To identify your port information look at your launch file

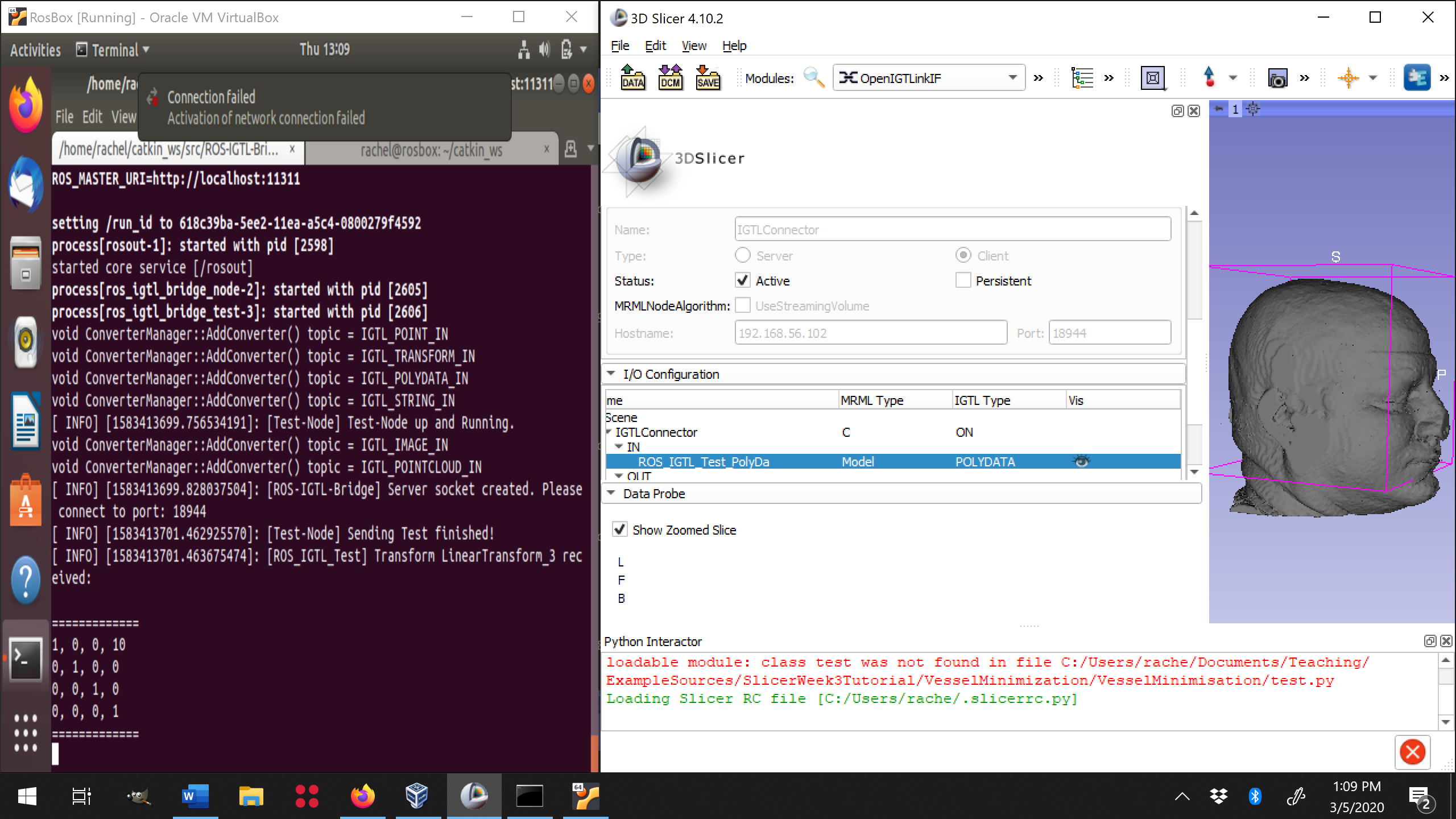


Set corresponding information in Slicer OpenIGTLinkIF

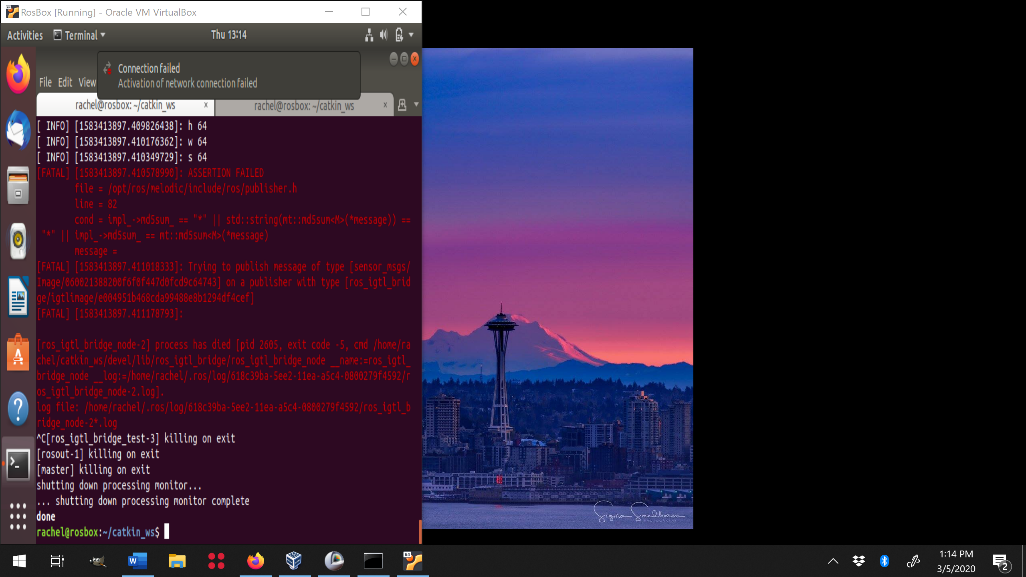


Turn connection to active.

Launch ros\_igtl node. You should see data appear in slicer and data received by the ROS bridge.



Limitations

1. LabeledImages cannot be added to output connections in OpenIGTLinkIF
2. Pushing images to ROS\_IGTLink will cause problems unless you properly set up your own publisher
3. Fiducials are not default passable in OpenITGLinkIF

The test bride node only sends only one polydata node called test\_pd. I have provide a modification of ROS\_IGTL\_BRIDGE on the class repository (https://gitlab.com/rsparks/7mri0070/-/tree/master/Modified-ROS-IGTL-Bridge) with a slicer\_models.launch file to pass vessels, cortex, and hippocampus.vtk to slicer. Please update your ROS\_IGTL\_BRIDGE folder with this repository and use catkin\_make to build the package.

Note that if you pass the \*ROS.vtk files to slicer they are oriented properly, but if you pass the \*Slicer.vtk files they are not properly oriented, this is do a difference in the coordinate frames of the two systems. In the following two weeks we will be setting a transformation protocol to correct this problem.