Visualizing QuICC Outputs with VAPOR 3

Adam Harris

March 3, 2021

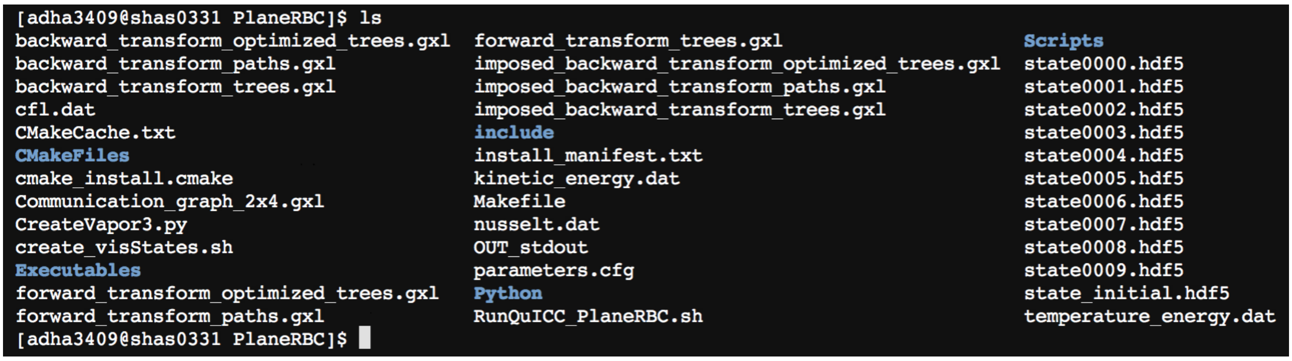
1 Installing and Locally Preparing VAPOR 3

At this point, VAPOR 3 proves simpler to operate locally than on Summit. The following strategy is specifically for MacOS users, however the strategy can be adapted to other operating systems. The following also assumes that you have successfully **Run a simulation** as described by **https://github.com/QuICC/QuICC/wiki/tutorial-compile** .

Navigate to **https://github.com/NCAR/VAPOR/releases/tag/3.3.0** and download the appropriate asset for your system (for MacOS, VAPOR3-3.3.0-Darwin.dmg). Follow the usual steps to install the software to your computer. Once the software is installed, identify the local path to the application. It may be convenient for future steps to copy the package contents of the application to your Desktop or a familiar folder. I copied the entire application to my Desktop for future use.

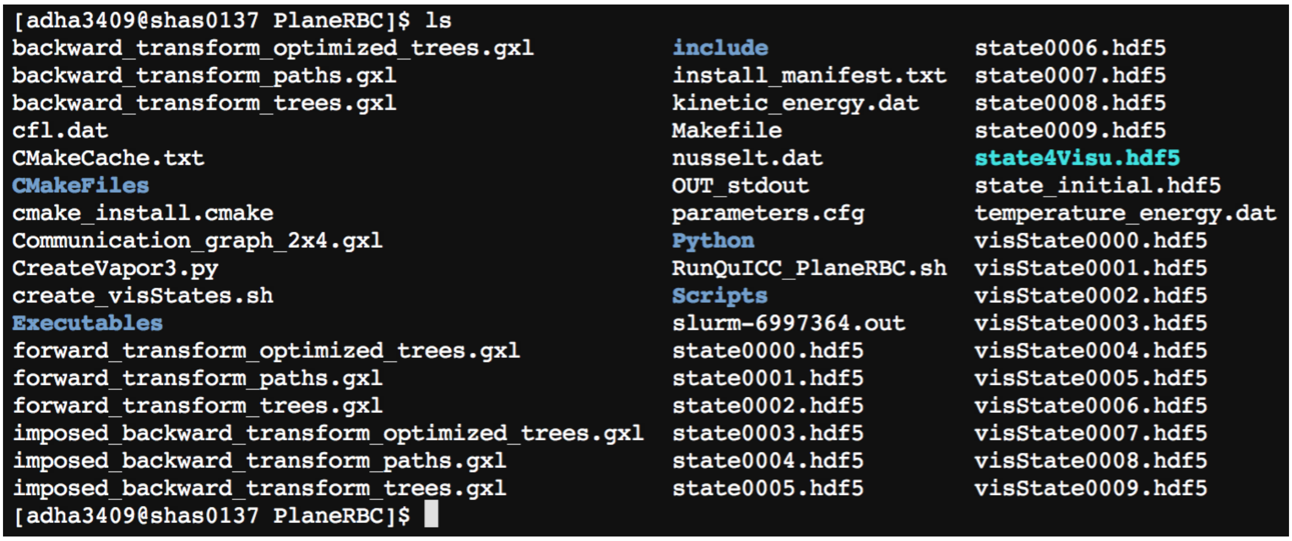
2 Generating VAPOR 3 Readable Files from QuICC

This step begins on Summit and ends on your local computer. Navigate to your directory on Summit where you ran the Model executable for your simulation. Your directory should look something like this:



```
[adha3409@shas0331 PlaneRBC]$ ls
backward_transform_optimized_trees.gxl  forward_transform_trees.gxl                     Scripts
backward_transform_paths.gxl            imposed_backward_transform_optimized_trees.gxl  state0000.hdf5
backward_transform_trees.gxl            imposed_backward_transform_paths.gxl            state0001.hdf5
cfl.dat                                 imposed_backward_transform_trees.gxl            state0002.hdf5
CMakeCache.txt                          include                                         state0003.hdf5
CMakeFiles                              install_manifest.txt                            state0004.hdf5
cmake_install.cmake                     kinetic_energy.dat                              state0005.hdf5
Communication_graph_2x4.gxl             Makefile                                        state0006.hdf5
CreateVapor3.py                         nusselt.dat                                     state0007.hdf5
create_visStates.sh                     OUT_stdout                                      state0008.hdf5
Executables                             parameters.cfg                                  state0009.hdf5
forward_transform_optimized_trees.gxl   Python                                          state_initial.hdf5
forward_transform_paths.gxl             RunQuICC_PlaneRBC.sh                            temperature_energy.dat
[adha3409@shas0331 PlaneRBC]$ 
```

Load the appropriate modules then compile and run the Visu executable as described in the QuICC Wiki's **Visualize state files** section. If you have multiple stateXXXX.hdf5 files and you desire multiple visStateXXXX.hdf5 files, it is convenient to run the **create_visStates.sh** job script after generating the Visu executable. Now, you should have the visStateXXXX.hdf5 file(s) in your simulation's directory on Summit, and your directory should look something like this:



```
[adha3409@shas0137 PlaneRBC]$ ls
backward_transform_optimized_trees.gxl          include               state0006.hdf5
backward_transform_paths.gxl                    install_manifest.txt  state0007.hdf5
backward_transform_trees.gxl                    kinetic_energy.dat    state0008.hdf5
cfl.dat                                         Makefile              state0009.hdf5
CMakeCache.txt                                  nusselt.dat           state4Visu.hdf5
CMakeFiles                                      OUT_stdout            state_initial.hdf5
cmake_install.cmake                             parameters.cfg        temperature_energy.dat
Communication_graph_2x4.gxl                     Python                visState0000.hdf5
CreateVapor3.py                                 RunQuICC_PlaneRBC.sh  visState0001.hdf5
create_visStates.sh                             Scripts               visState0002.hdf5
Executables                                     slurm-6997364.out     visState0003.hdf5
forward_transform_optimized_trees.gxl           state0000.hdf5        visState0004.hdf5
forward_transform_paths.gxl                     state0001.hdf5        visState0005.hdf5
forward_transform_trees.gxl                     state0002.hdf5        visState0006.hdf5
imposed_backward_transform_optimized_trees.gxl  state0003.hdf5        visState0007.hdf5
imposed_backward_transform_paths.gxl            state0004.hdf5        visState0008.hdf5
imposed_backward_transform_trees.gxl            state0005.hdf5        visState0009.hdf5
[adha3409@shas0137 PlaneRBC]$ 
```

To get the files into the format we need for VAPOR 3, run the **CreateVapor3.py** in your simulation's directory. The python script must be followed by the list of visStateXXXX.hdf5 files that you wish to use. The command looks like this:

python CreateVapor3.py visState0000.hdf5 visState0001.hdf5 visState0002.hdf5 ...

or to convert all of your visStateXXXX.hdf5 files conveniently, the command is:

python CreateVapor3.py visState0*

You should now have a single file named **vapor.nc** in your simulation's directory. Determine and take note of the path on Summit to this file.

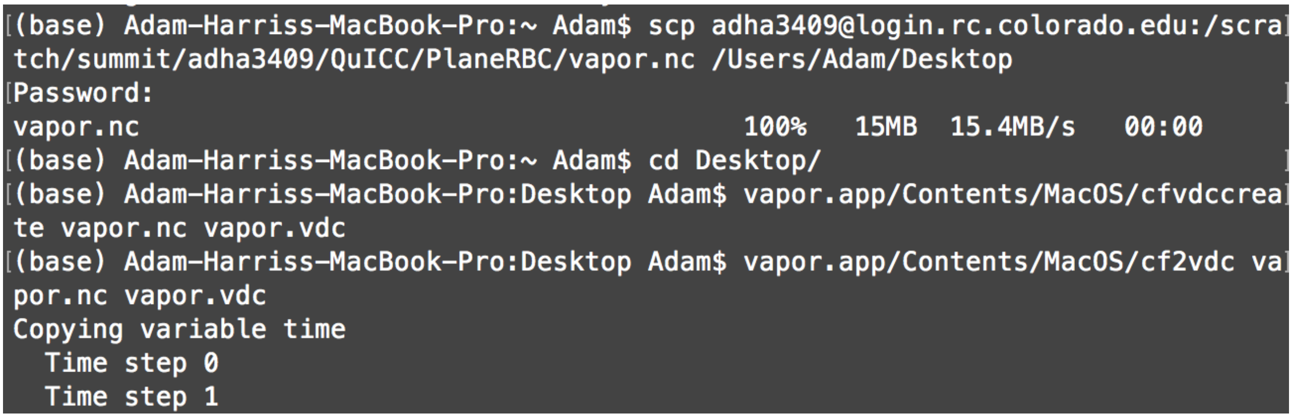
Now we are done with Summit. Exit Summit and open a Terminal on your local computer. We need to copy the **vapor.nc** file from Summit to our local computer, we do this with an scp command. For me, the command is something like this:

scp adha3409@login.rc.colorado.edu:/scratch/path/vapor.nc /Users/Adam/Desktop

The file **vapor.nc** should now be on your Desktop. Still in your local Terminal, navigate to your Desktop's directory. We need to convert the .nc file into a .vdc file using the executables included in the VAPOR 3 application's package contents. The following commands are for my specific setup, you will need to adjust these commands slightly. The following commands create the .vdc file and populate the .vdc file. Enter the likeness of the following commands serially in your local Terminal:

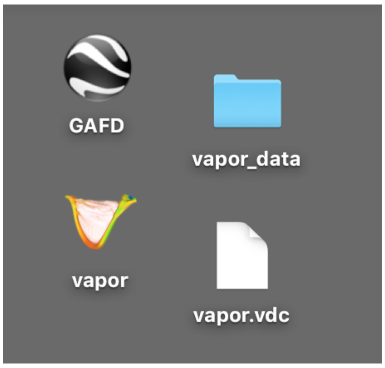
vapor.app/Contents/MacOS/cfvdccreate vapor.nc vapor.vdc

vapor.app/Contents/MacOS/cf2vdc vapor.nc vapor.vdc

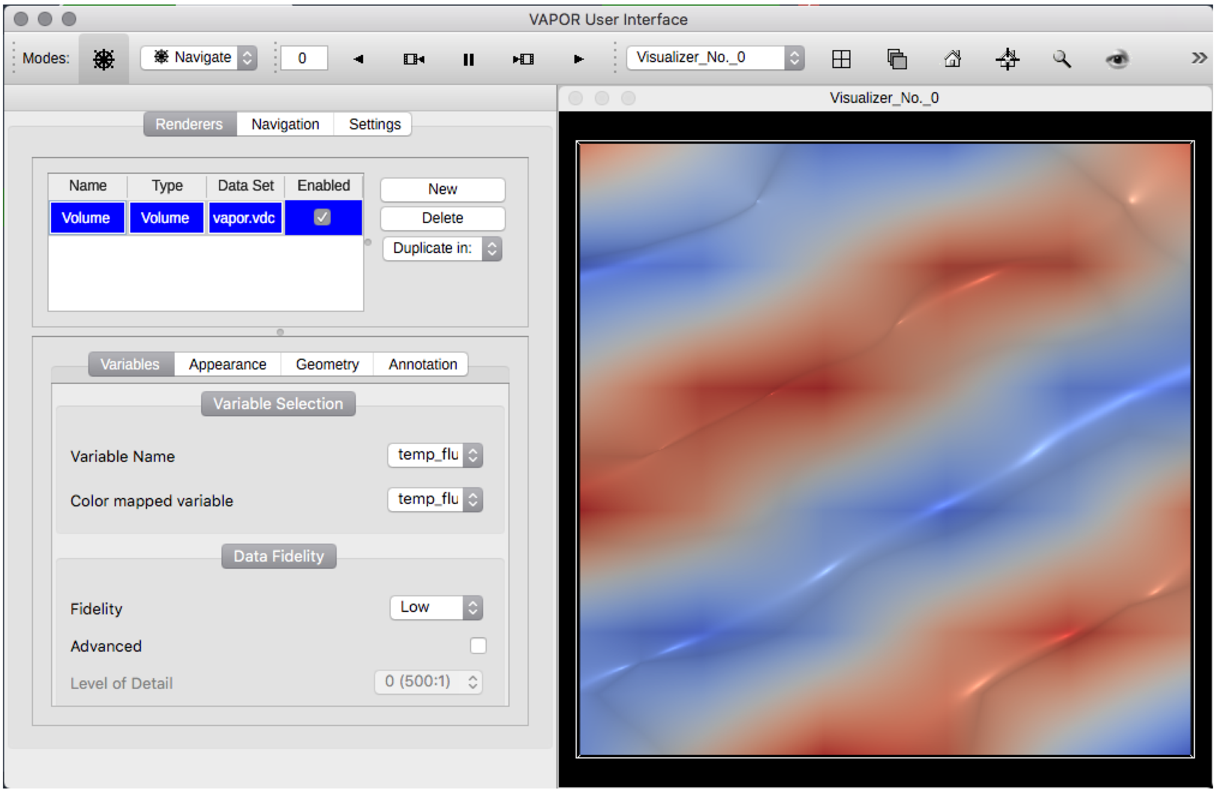


```
[(base) Adam-Harriss-MacBook-Pro:~ Adam$ scp adha3409@login.rc.colorado.edu:/scra]
 tch/summit/adha3409/QuICC/PlaneRBC/vapor.nc /Users/Adam/Desktop
[Password:                                                                      ]
 vapor.nc                                      100%   15MB  15.4MB/s   00:00
[(base) Adam-Harriss-MacBook-Pro:~ Adam$ cd Desktop/                            ]
[(base) Adam-Harriss-MacBook-Pro:Desktop Adam$ vapor.app/Contents/MacOS/cfvdccrea]
 te vapor.nc vapor.vdc
[(base) Adam-Harriss-MacBook-Pro:Desktop Adam$ vapor.app/Contents/MacOS/cf2vdc va]
 por.nc vapor.vdc
 Copying variable time
   Time step 0
   Time step 1
```

You should now have a **vapor.vdc** file and a **vapor_data** folder on your Desktop.



These two items MUST be paired, and always stored in the same location in order to work. From here, you should be good to go! Open the VAPOR 3 application on your computer, select File → Open VDC then navigate to where you have **vapor.vdc** and **vapor_data**, and open the VDC file. Now click on "New" in the right panel of the VAPOR 3 GUI, and select "Volume" from the list, then check the box under "Enabled". You should see this:



Name	Type	Data Set	Enabled
Volume	Volume	vapor.vdc	☑

Variable Name temp_flu

Color mapped variable temp_flu

Fidelity Low

Advanced □

Level of Detail 0 (500:1)

Change fidelity to high. You can click and drag the image around to different orientations, right-click and drag to zoom the image in and out, and use the play/pause/forward/back buttons on the top to cycle through timesteps. You can also select different variables from the drop down list.