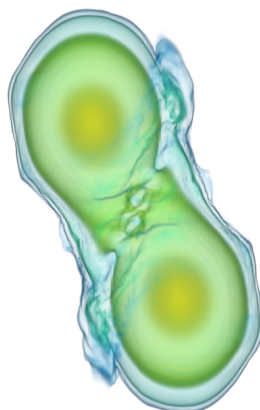


Simulating and visualizing a double neutron star binary inspiral

Roland Haas & the Einstein Toolkit Maintainers

Jul 21 2014



1 Downloading, compiling and running the code

This section provides a simple guide for compiling and running an example parameter file for the Einstein Toolkit.

One complexity of high performance computing is the heterogeneous environments presented by different resources with different core software installations and local setups. For example, Einstein Toolkit codes require compiling with MPI and then running the resulting application in a batch queue. This complexity is hidden as much as possible in Cactus and accompanying tools such as the Simulation Factory.

The example worked through below assumes that you are using the SDSC resource Gordon. We will simulate a pair of neutron stars inspiralling towards each other, colliding and forming a hypermassive neutron star. You can find this simulation in the gallery at <http://einstein toolkit.org/about/gallery/>.

In the following, whenever you need to enter a command, we will use a monospaced font indicate input. When input is too long to fit a single line, we will break the line and indicate the line breaking point by `\`. When entering such a line, please remove both the `\` and the line break.

1.1 Preparations

You will need to use `ssh` to log into Gordon. At a command line prompt, type

```
ssh YOUR_LOGIN@gordon.sdsc.edu
```

to log into Gordon. From now on *all* input should be entered into the ssh session on Gordon.

The login shell in Gordon defaults to `tcsh`, if you are instead familiar with `bash` (the default shell for Linux) then you may want to change the login shell. You can change it permanently by pointing your browser to <https://passive.sdsc.edu/> and submit a change request. Gordon is not explicitly listed, however the change will take place if you select **Trestles** as the machine. This change may take some time to become effective. Instead you can also enter

```
exec bash -l
```

which will start a new bash as if you had just logged into it. Exiting this shell will log you out of Gordon.

We will put all downloaded source code into a directory `gordon` inside of your `$HOME` directory:

```
mkdir gordon
cd gordon
```

1.2 Download

Download `GetComponents` to checkout all the components of the Einstein Toolkit, and ensure it is executable:

```
wget --no-check-certificate \
https://raw.githubusercontent.com/gridaphobe/CRL/ET_2014_05/GetComponents
chmod 755 GetComponents
```

Checkout the Einstein Toolkit using anonymous authentication (this may take a few minutes):

```
./GetComponents --parallel http://www.tapir.caltech.edu/~rhaas/ET/NsNsToHMNS.th
```

This checks out Cactus, the Einstein Toolkit thorns, the Simulation Factory and example parameter files into a directory named Cactus.

Download the parameter file describing the simulation to Cactus

```
wget -O Cactus/par/NsNsToHMNS.par \
http://www.tapir.caltech.edu/~rhaas/ET/NsNsToHMNS.par
```

1.3 Configuring the Simulation Factory

The Simulation Factory (`simfactory` for short) needs to know a little bit about you in order to properly submit simulations to the queuing system on Gordon.

```
cd Cactus
nano simfactory/etc/defs.local.ini
```

starts up a simple editor and lets you enter the required information. Please enter

```
[gordon]
allocation = ddp182
runscript = /home/rhaas/gordon/Cactus/simfactory/mdb/runscripts/gordon.run
envsetup = <<EOF
source /etc/profile
module unload mpich2_ib
module unload mvapich2_ib
module unload mv2profile_ib
module unload openmpi_ib
module load mvapich2_ib
module load gsl
EOF
basedir = /oasis/projects/nsf/ddp182/YOUR_LOGIN/simulations
[default]
email = YOUR@EMAIL.ADDRESS
user = YOUR_LOGIN
```

replacing

- **YOUR_LOGIN:** with your user name on Gordon
- **YOUR@EMAIL.ADDRESS:** with your usual email address.

Do *not* change `/home/rhaas/gordon/Cactus/simfactory/mdb/runscripts/gordon.run` since the file `gordon.run` that ships with `simfactory` by default will not work and instead we use a modified file. Exit `nano` by pressing `CTRL-o`, then `Enter` to save your changes and `CTRL-X` to quit. Now create an output directory for our simulations

```
mkdir /oasis/projects/nsf/ddp182/$USER/simulations
```

1.4 Obtaining data files for initial data

We use initial data provided by the LORENE code developed by the group in Meudon, France.

```
mkdir /oasis/projects/nsf/ddp182/$USER/LoreneID
cd /oasis/projects/nsf/ddp182/$USER/LoreneID/
wget -O G2_I12vs12_D4R33T21_45km.tar.gz \
ftp://octane.obspm.fr/BinaryNS/GR/Polytrope/gamma2/Irrotation/EqualMass/equal_12vs12/\
G2_I12vs12_D4R33T21_45km.tar.gz
mkdir G2_I12vs12_D4R33T21_45km
tar xf G2_I12vs12_D4R33T21_45km.tar.gz -C G2_I12vs12_D4R33T21_45km
ln -s G2_I12vs12_D4R33T21_45km/resu.d ./G2_I12vs12_D4R33T21_45km.resu
```

You will have to let `simfactory` know where to find the initial data files. We use shell environment variables for this, and if you are using `bash` please do

```
export LORENE_DATA=$PWD || setenv LORENE_DATA $PWD
echo "export LORENE_DATA=$PWD" >>$HOME/.bashrc
echo "setenv LORENE_DATA $PWD" >>$HOME/.cshrc
```

to record the current directory as the initial data location (the second line adds the setting to your `.bashrc` and `.cshrc` files so that you don't have to re-do this step each time you log in).

1.5 Compiling executable

Enter

```
cd $HOME/gordon/Cactus
mkdir exe
./simfactory/bin/sim build --thornlist thornlists/NsNsToHMNS.th --virtual \
--virtual-executable /home/rhaas/gordon/Cactus/exe/cactus_sim
```

to use a pre-compiled version of the code.

If you'd rather compile it yourself instead, please do

```
cd $HOME/gordon/Cactus
./simfactory/bin/sim build --thornlist thornlists/NsNsToHMNS.th
```

1.6 Starting the simulation

You are now ready to submit your simulation to the queuing system. Enter

```
simfactory/bin/sim create --parfile par/NsNsToHMNS.par NsNsToHMNS
simfactory/bin/sim submit --procs 48 --walltime 2:00:00 NsNsToHMNS
```

to first create a new simulation `NsNsToHMS` and then to ask the queueing system to run it for 2 hours using 48 processor cores (3 nodes).

To check the status of the simulation, use

```
./simfactory/bin/sim list-simulations
```

or the more low level

```
qstat -u $USER
```

Once the simulation starts, you can read its log output via

```
./simfactory/bin/sim show-output --follow NsNsToHMNS
```

(press `CTRL-C` to stop watching the output).

2 Visualizing your results

Gordon provides the **VisIt** (<https://wci.llnl.gov/simulation/computer-codes/visit/>) visualization package which can read the output files produced by **Cactus**.

First you will have to download the **VisIt** client to your local machine to serve as a user interface. We will remotely connect to Gordon to load the files without having to copy them to your local machine.

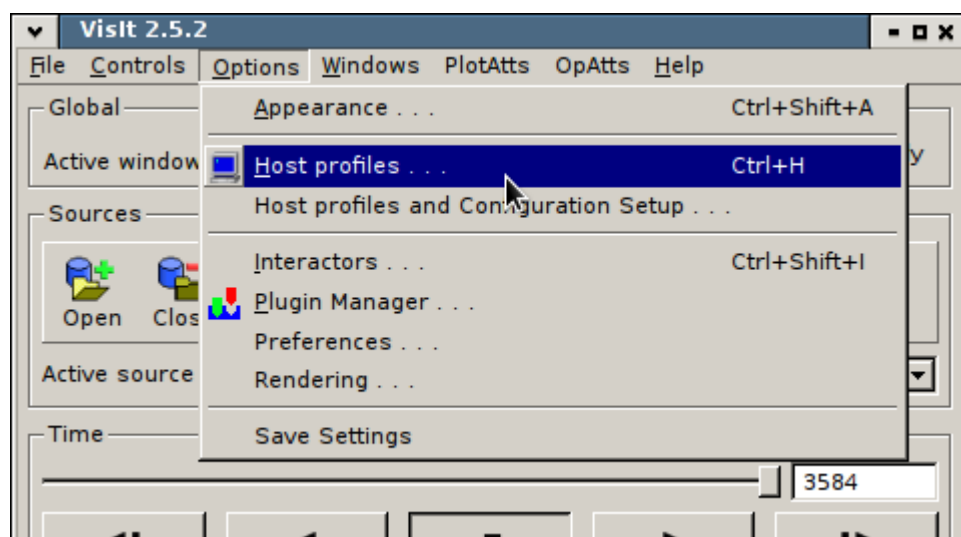
You can download pre-compiled executables of **VisIt** for Linux, OSX and Windows from the **VisIt** website at <https://wci.llnl.gov/simulation/computer-codes/visit/executables>. Please note that the version that you download *MUST* match the version installed on Gordon exactly. Currently the version installed on Gordon is version 2.5.2. Thus if you are running OSX please download http://portal.nersc.gov/svn/visit/trunk/releases/2.5.2/VisIt-2.5.2-x86_64-installer.dmg if you are running Ubuntu-Linux, please download http://portal.nersc.gov/svn/visit/trunk/releases/2.5.2/visit2_5_2.linux-x86_64-ubuntu11.tar.gz and if you are running RedHat-Linux please download http://portal.nersc.gov/svn/visit/trunk/releases/2.5.2/visit2_5_2.linux-x86_64-rhel5.tar.gz. For those of you running Windows, please download <http://portal.nersc.gov/svn/visit/trunk/releases/2.5.2/visit2.5.2.exe>.

Once you have downloaded and installed **VisIt** please make sure that you still have a login to Gordon active as this will save some headache and then start **VisIt**.

2.1 Setting up VisIt

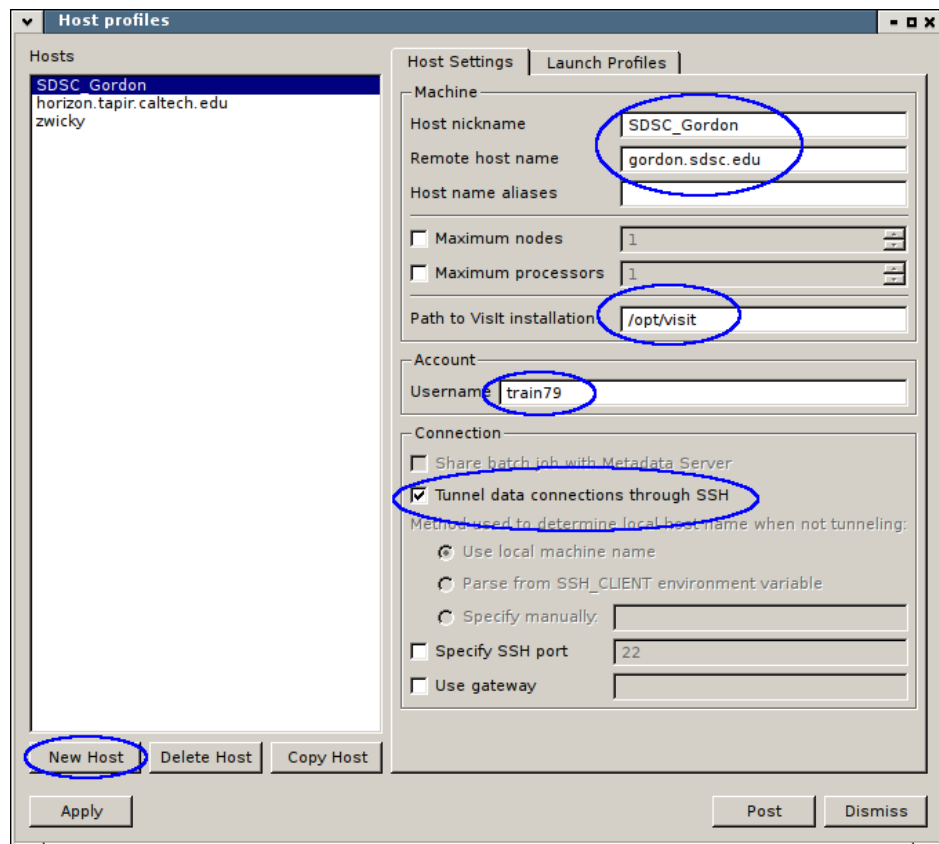
SDSC provides instructions on how to start **VisIt** on one of the compute nodes at http://www.sdsc.edu/us/resources/gordon/docs/gordon_visit.html. We will instead run on the head node, which is a shared resource, so please be considerate when using it.

First you will have to set up a host profile for Gordon in **VisIt**. Choose **Host** profiles from the **Options** menu in the main **VisIt** window:



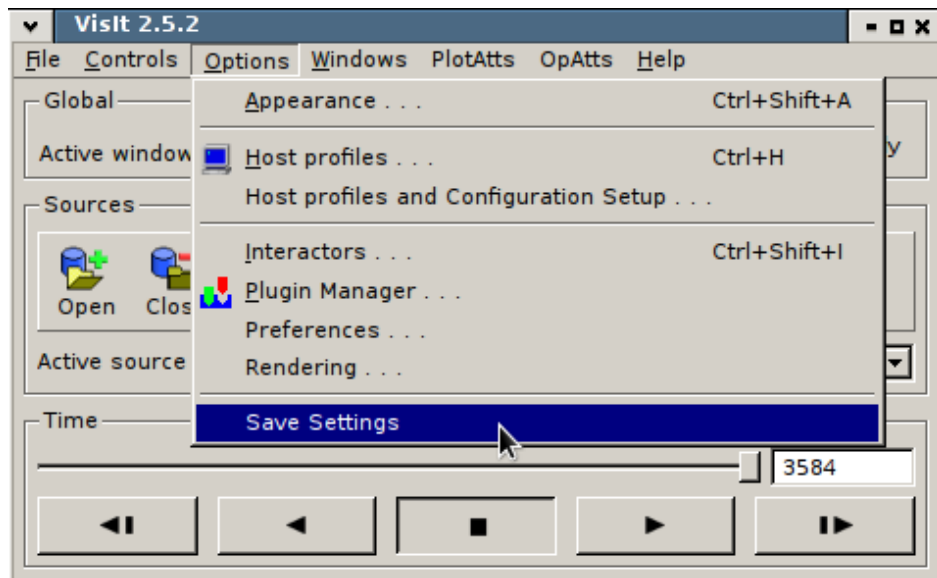
In the **Host** profiles dialog box, click on **New Host** in the lower left of the window. Enter the following in the **Host** Settings tab:

- Host nickname: SDSC_Gordon
- Remote host name: gordon.sdsc.edu
- Path to VisIt installation: /opt/visit
- Username: YOUR_LOGIN
- Tunnel data connections through SSH: Check box to enable



Click Apply on the lower left of the window and then click Dismiss.

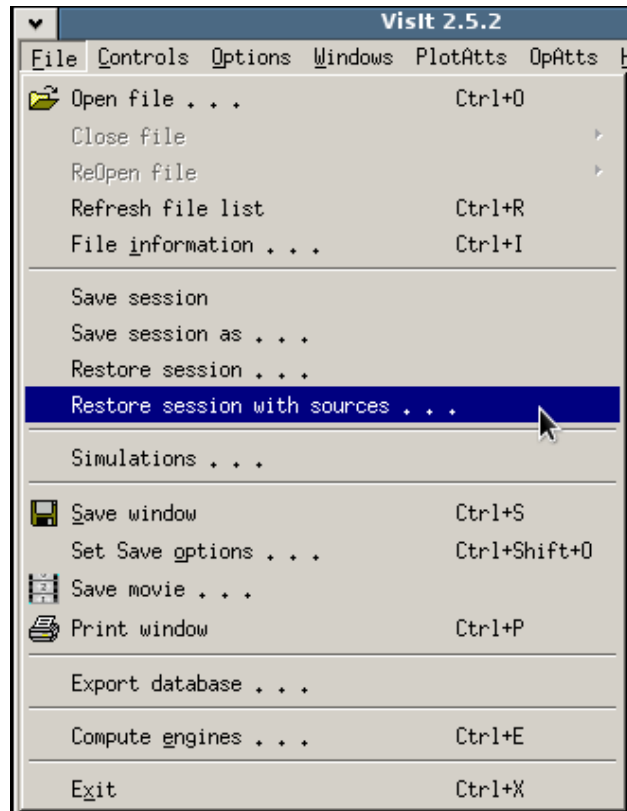
CAUTION: Your settings are not yet saved so the next step is very important and easily missed by many users. From the **Options** menu, choose **Save Settings**. If you miss this step, your host profile will not be saved!



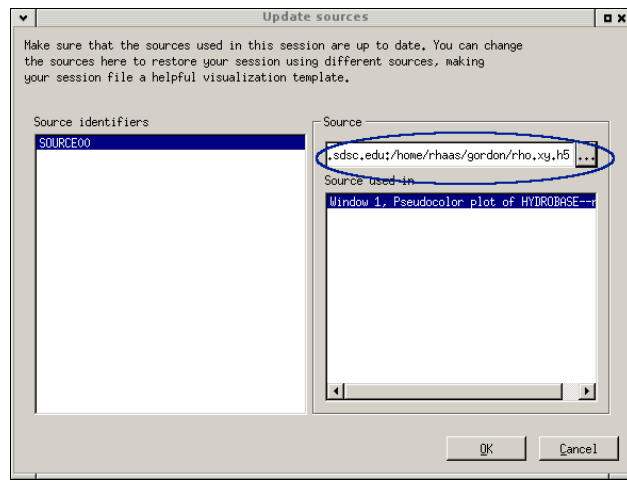
2.2 Loading your data into VisIt

To simplify things, we provide a preset VisIt session that already sets up the plots as desired. Using a webbrowser download the file <http://www.tapir.caltech.edu/~rhaas/ET/NsNsToHMNS.session> and save it to your local machine.

Start VisIt and select **Restore session with sources** from the File menu



and in the dialog box, open the file `NsNsToHMNS.session` that you have just downloaded. A new dialog box `Update sources` opens in which you need to select your data file. Otherwise the session file will use the data generated by Roland.

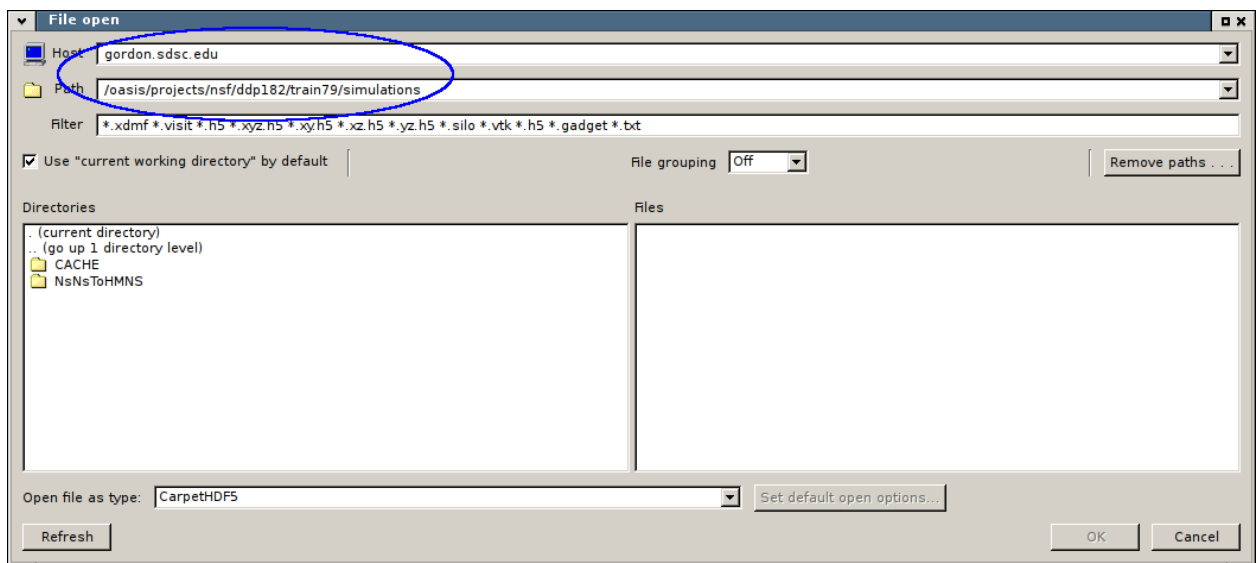


To do so, click on the button labelled by an ellipsis in the upper right corner of the window and `Open` dialog appears.

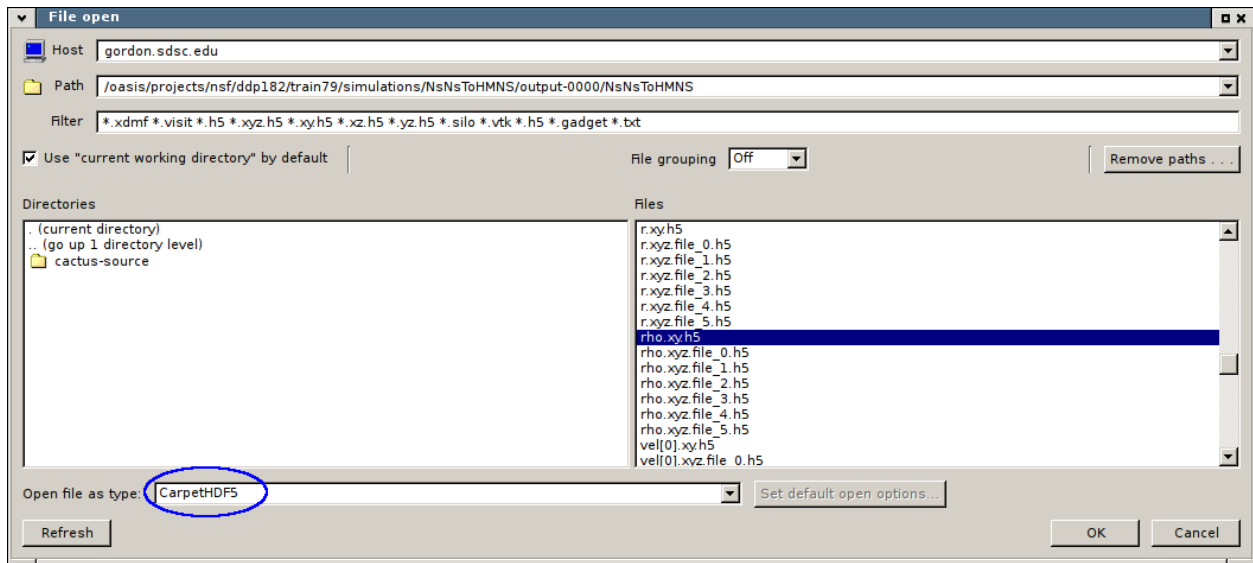
Enter

`/oasis/projects/nsf/ddp182/YOUR_LOGIN/simulations`

into the `Path` entry box, where `YOUR_LOGIN` is your user name on Gordon. Press `Enter` to have `VisIt` actually change into the directory.



then go into the directory `NsNsToHMNS/output-0000/NsNsToHMNS` where you will find a file `rho.xy.h5`.



Make sure to select **CarpethDF5** as the file type in the dropdown box at the bottom left of the window. Open the file by clicking on **Ok**.

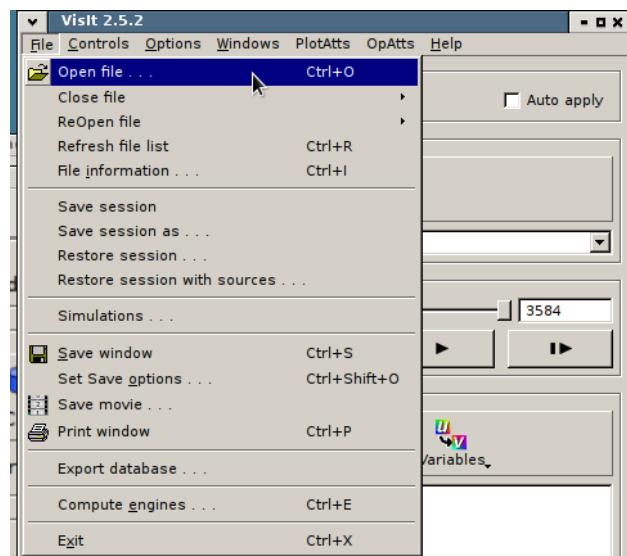
Now click on **Ok** in the **Update sources** dialog and **VisIt** will load and display the data file. You can play around with the time slider and plot options in **VisIt**. Double click on a plot or one of the operators in the main **VisIt** window to change their settings.

3 In detail instructions

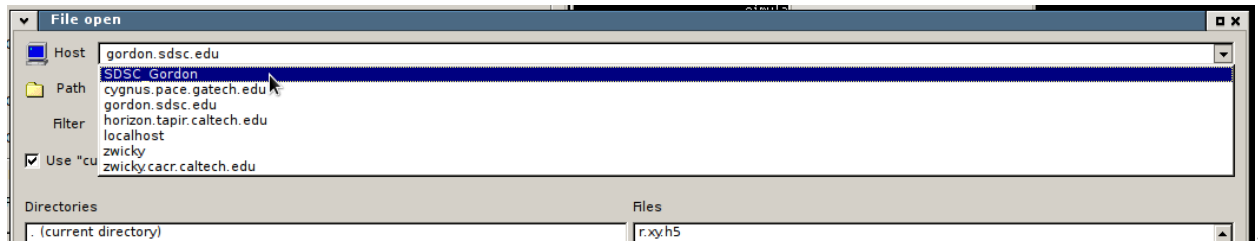
This section describes the steps required to set up the plots you used in the previous section. Only work through it if you have extra time.

3.1 Loading data files into VisIt

Open the **File open** dialog by clicking on **Open file** in the **File** menu



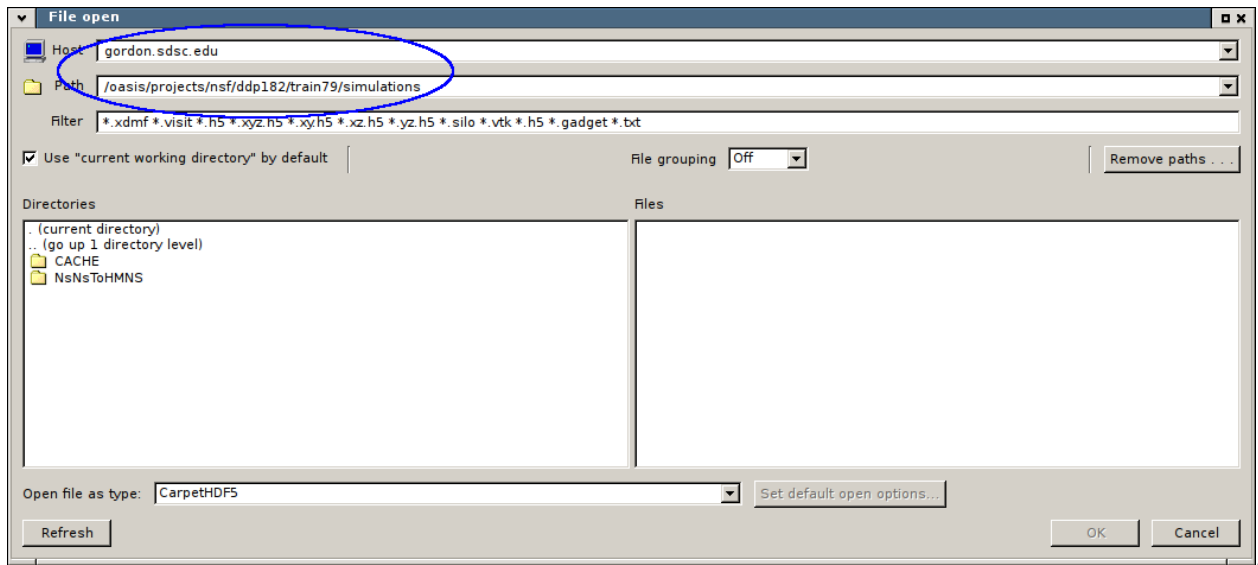
then select `SDSC_Gordon` in the Host dropdown box



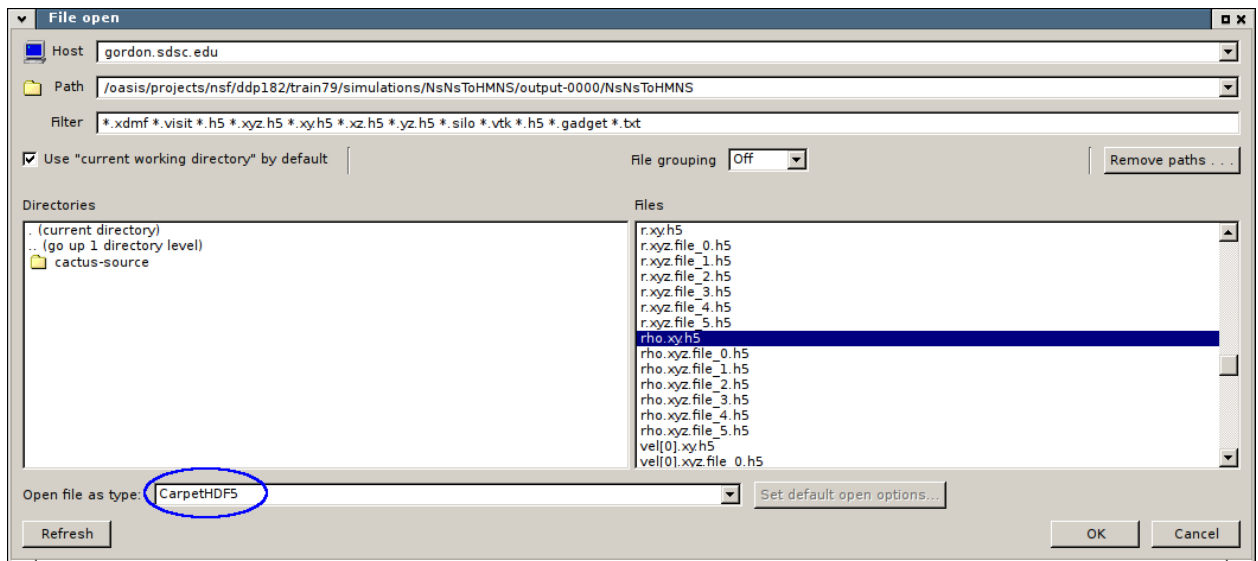
Enter

/oasis/projects/nsf/ddp182/YOUR_LOGIN/simulations

into the Path entry box, where YOUR.LOGIN is your user name on Gordon. Press Enter to have VisIt actually change into the directory.



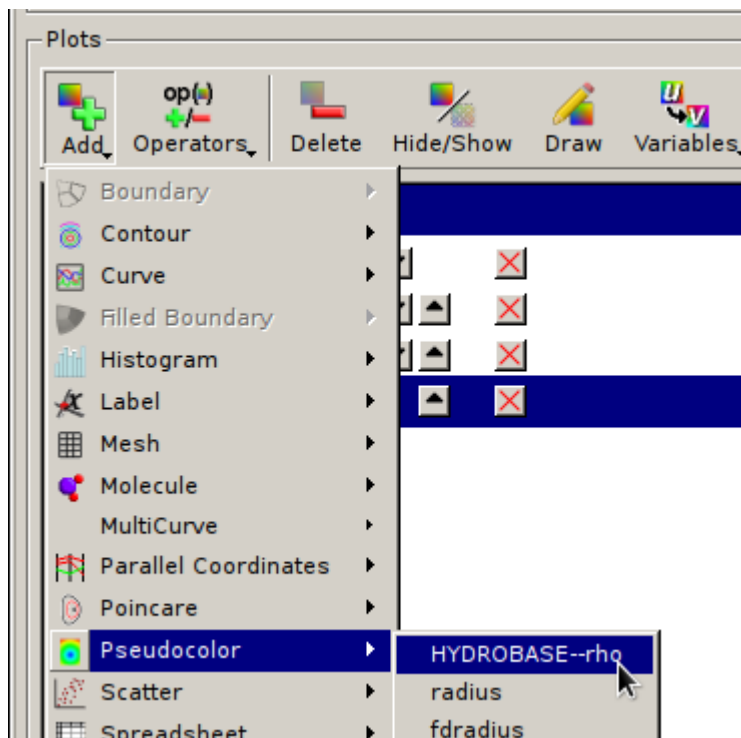
then go into the directory NsNsToHMNS/output-0000/NsNsToHMNS where you will find a file rho.xy.h5.



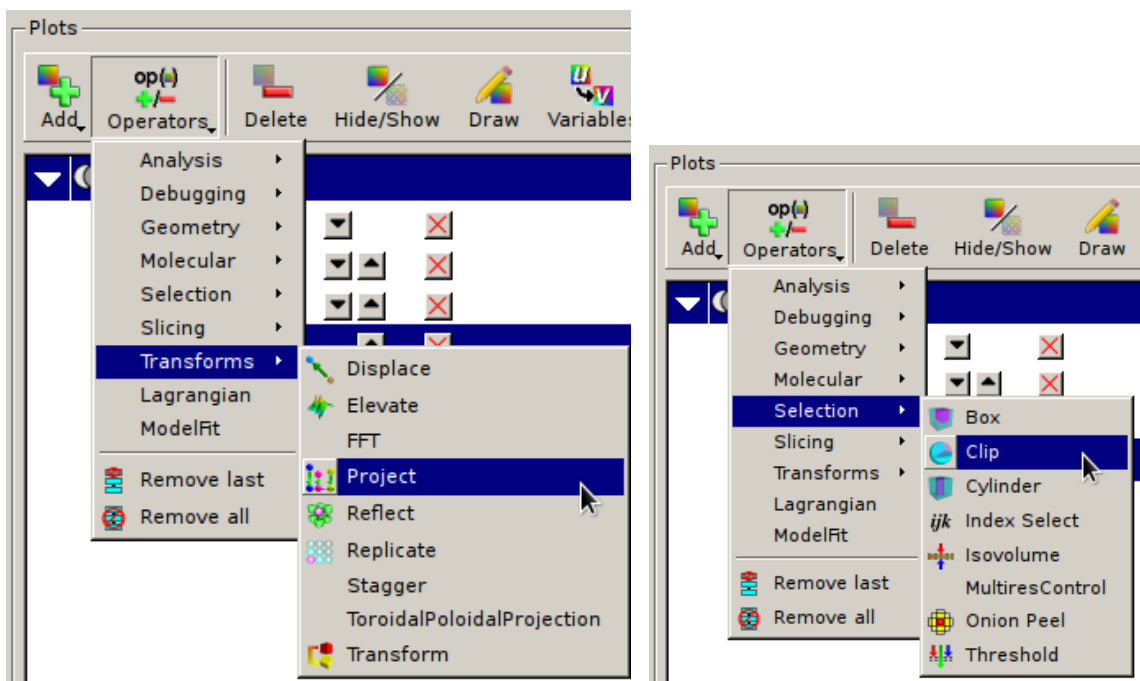
Make sure to select **CarpethDF5** as the file type in the dropdown box at the bottom left of the window. Open the file by clicking on **Ok**.

3.2 Making a plot

Using the Add/Pseudocolor menu item, add Pseudocolor plot for the variable HydroBase--rho

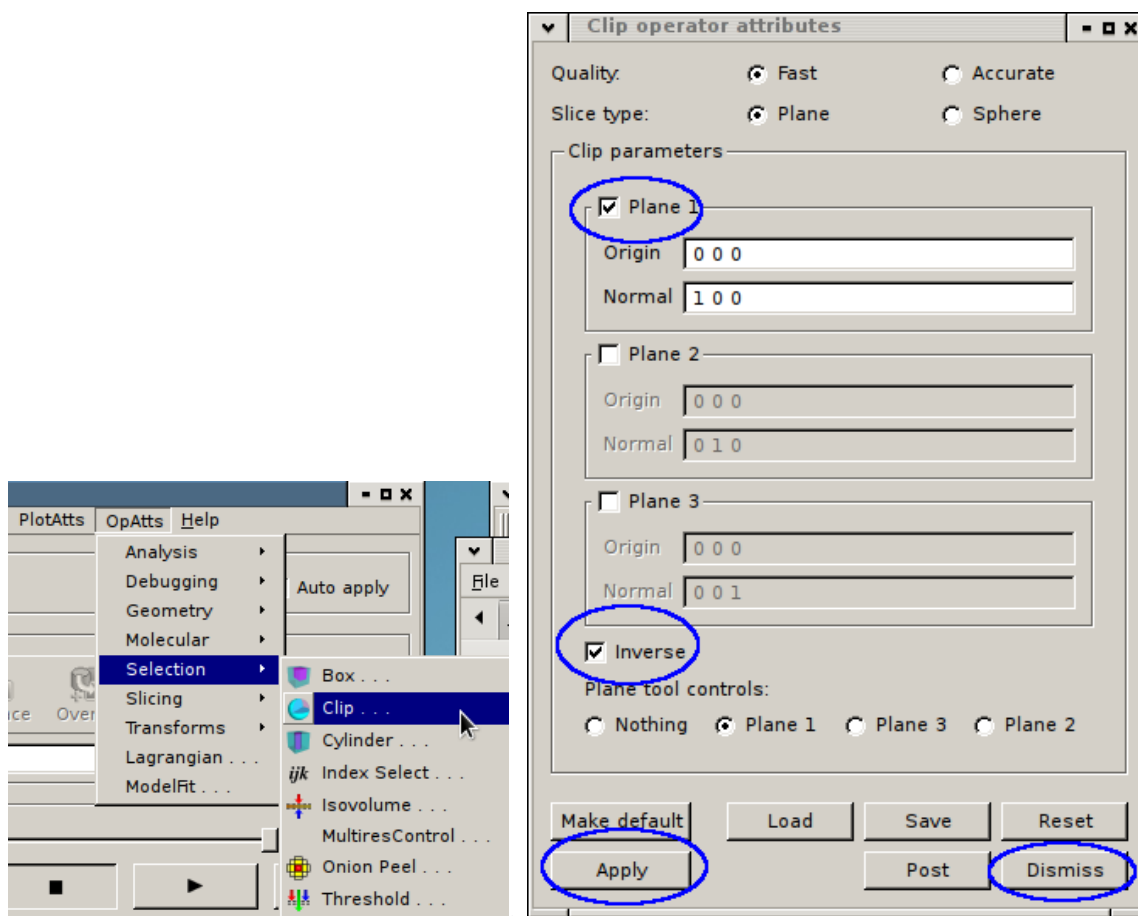


Next, add (*in this order!*) four operators: a Project operator from the Transforms list of operators then a Clip operator from the Selection list of operators,



then a Reflect operator from the Transforms list of operators and finally a Box operator from the Selection list of operators.

Bring up the Clip operator option dialog by clicking on OpAtts/Selection menu



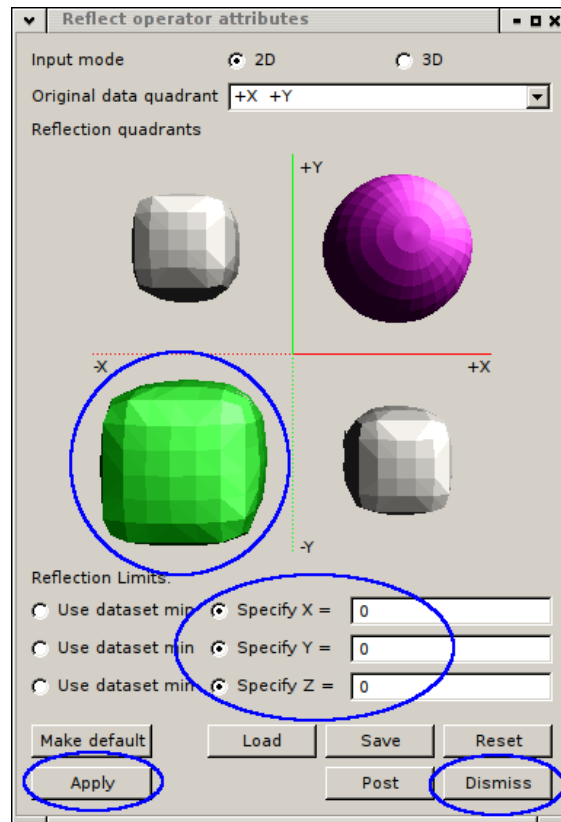
and change these entries

Plane 1 Select the checkbox

Inverse Select the checkbox.

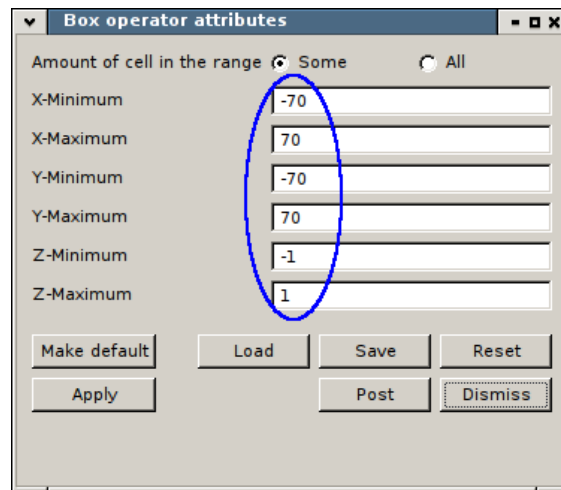
Then click on **Apply** and **Dismiss** to apply the new settings of the operator.

Bring up the **Reflect** operator's attributes dialog by clicking on **OppAtts/Transforms** or **OperatorAttributes/Transforms** and select the bottom left and top right quadrants, changing the origin of the reflect operation to (0,0,0).



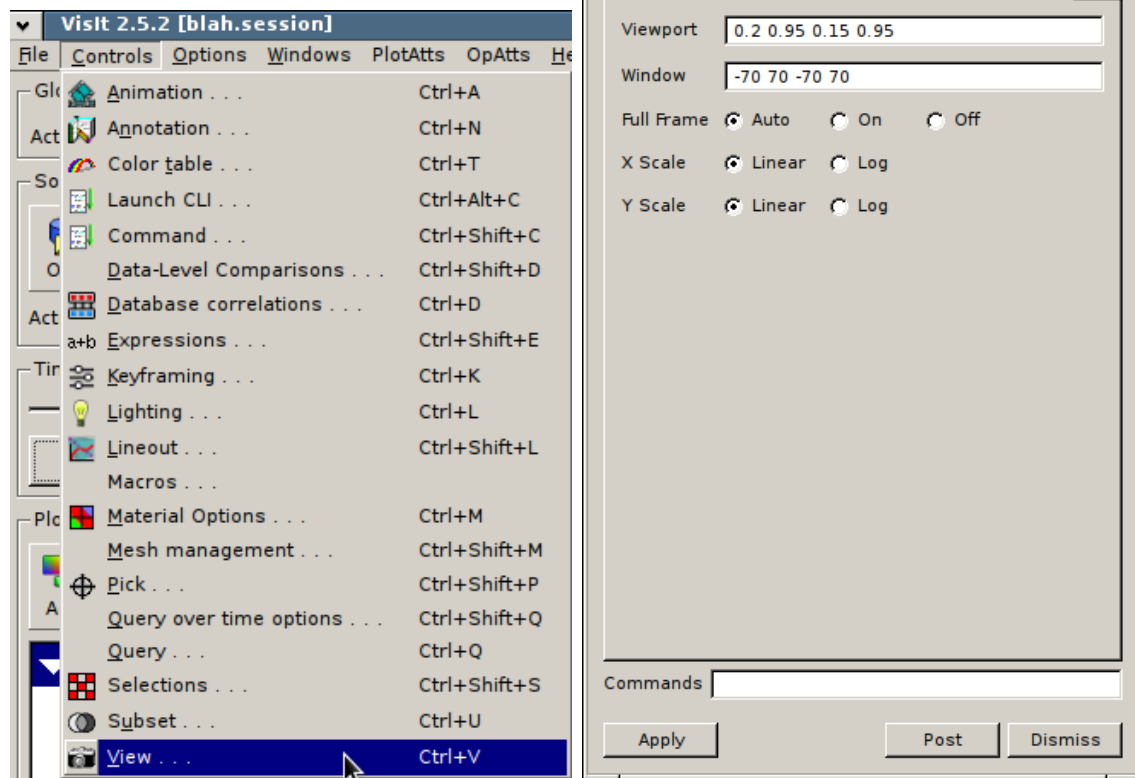
Click on **Apply** and **Dismiss** to apply the new settings of the operator.

Bring up the **Box** operator's attributes dialog by clicking on **OppAtts/Selection** and restrict the range to display to $-70 \leq x \leq 70$, $-70 \leq y \leq 70$, $-1 \leq z \leq 1$.

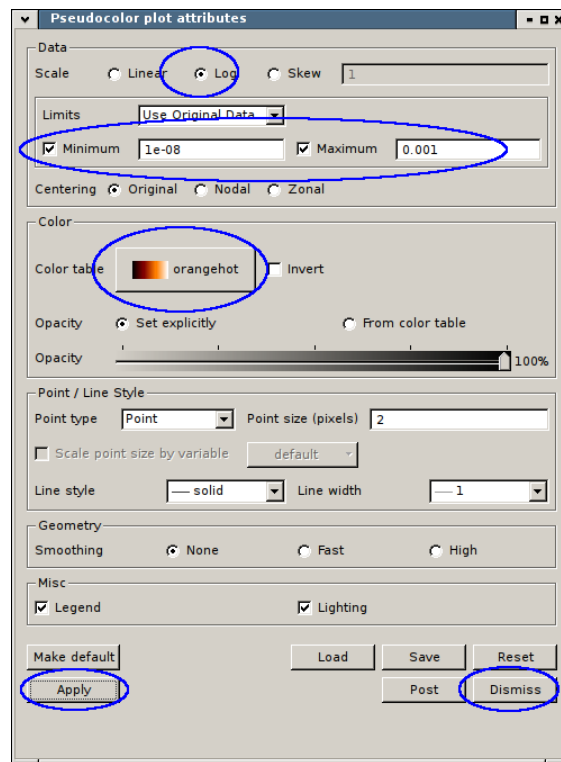


Click on **Apply** and **Dismiss** to apply the new settings of the operator.

Click on the **Controls/View** menu item to open the **View** dialog box, and in the **2D view** tab set **Window** to **-70 70 -70 70** to restrict the view range to that of the **Box** operator.

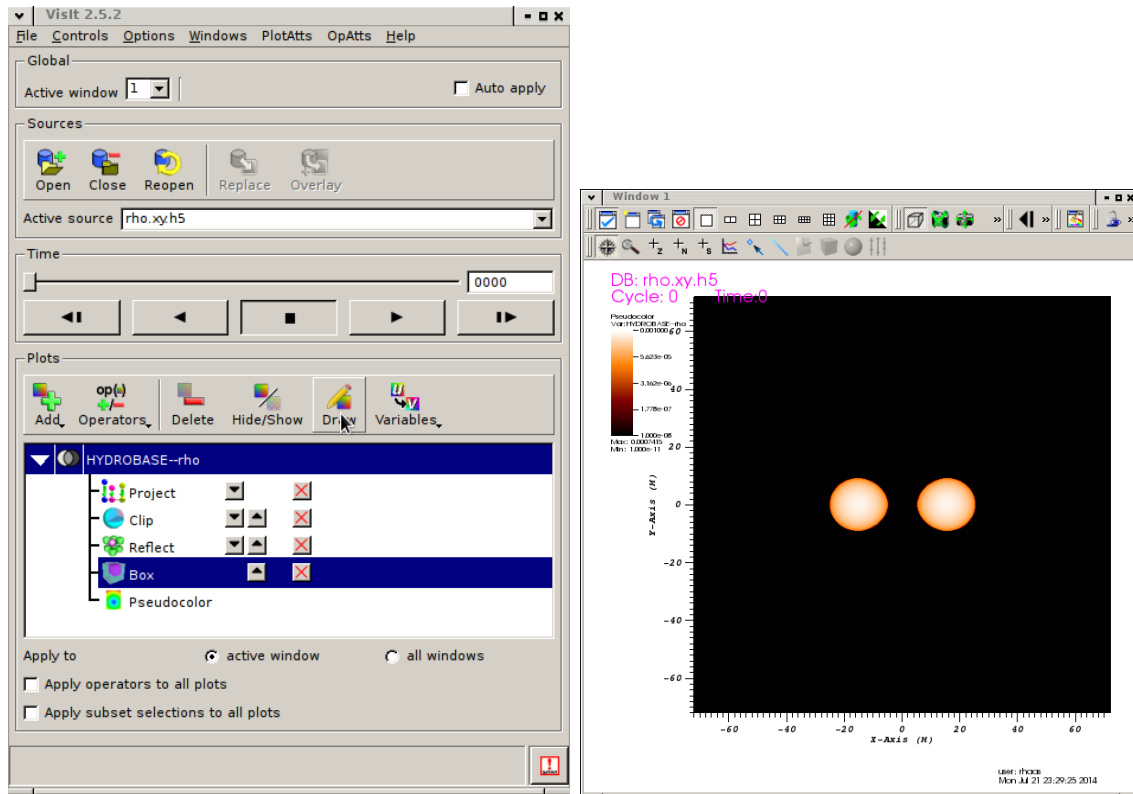


Bring up the Pseudocolor plot's attributes dialog by clicking on PlotAtts/PseudoColor and change the settings to enable a logarithmic color scale, set the minimum value to 10^{-8} , the maximum to 10^{-3} and select the orangehot color table.



Click on **Apply** and **Dismiss** to apply the new settings of the plot.

Finally click on the Draw button to have VisIt render the first frame.



You can play around with the time slider and plot options in VisIt. Double click on a plot or one of the operators in the main VisIt window to change their settings.