Visualizing Data on Cooley with ParaView

Joseph Insley
Lead, Visualization & Data Analysis
Argonne Leadership Computing Facility

Silvio Rizzi
Assistant Computer Scientist
Argonne Leadership Computing Facility

Janet Knowles
Principal Software Engineering Specialist
Argonne Leadership Computing Facility

Software for visualization hands on session

If you would like to follow along the visualization hands on activities, we recommend that you download and install ParaView version 5.8 in advance.

Please point your browser to https://www.paraview.org/download/

- **Windows**

Download and install `ParaView-5.8.0-Windows-Python3.7-msvc2015-64bit.exe`
Note: if you find and error about VCOMP140.DLL, there is a possible solution here
https://discourse.paraview.org/t/missing-dll/3650

- **Linux**

Download and untar `ParaView-5.8.0-MPI-Linux-Python3.7-64bit.tar.gz`

- **macOS**

Download and install `ParaView-5.8.0-MPI-OSX10.12-Python2.7-64bit.dmg`

Where to get it…

ParaView

– www.paraview.org

A version of this tutorial

– www.alcf.anl.gov/user-guides/vis-paraview-red-blood-cell-tutorial

Download data

– web.alcf.anl.gov/visinternal/MISC/BLOODFLOW_ANIMATION_DATA.tar.gz (272MB/390MB)

– web.alcf.anl.gov/visinternal/MISC/BLOODFLOW_ANIMATION_DATA_SMALL.tar.gz (136MB/195MB)

– Available on Cooley:

- /lus/theta-fs0/projects/Comp_Perf_Workshop/visualization/DATA/BLOODFLOW_TUTORIAL_DATA

More Information

Online Help – F1

The ParaView User’s Guide

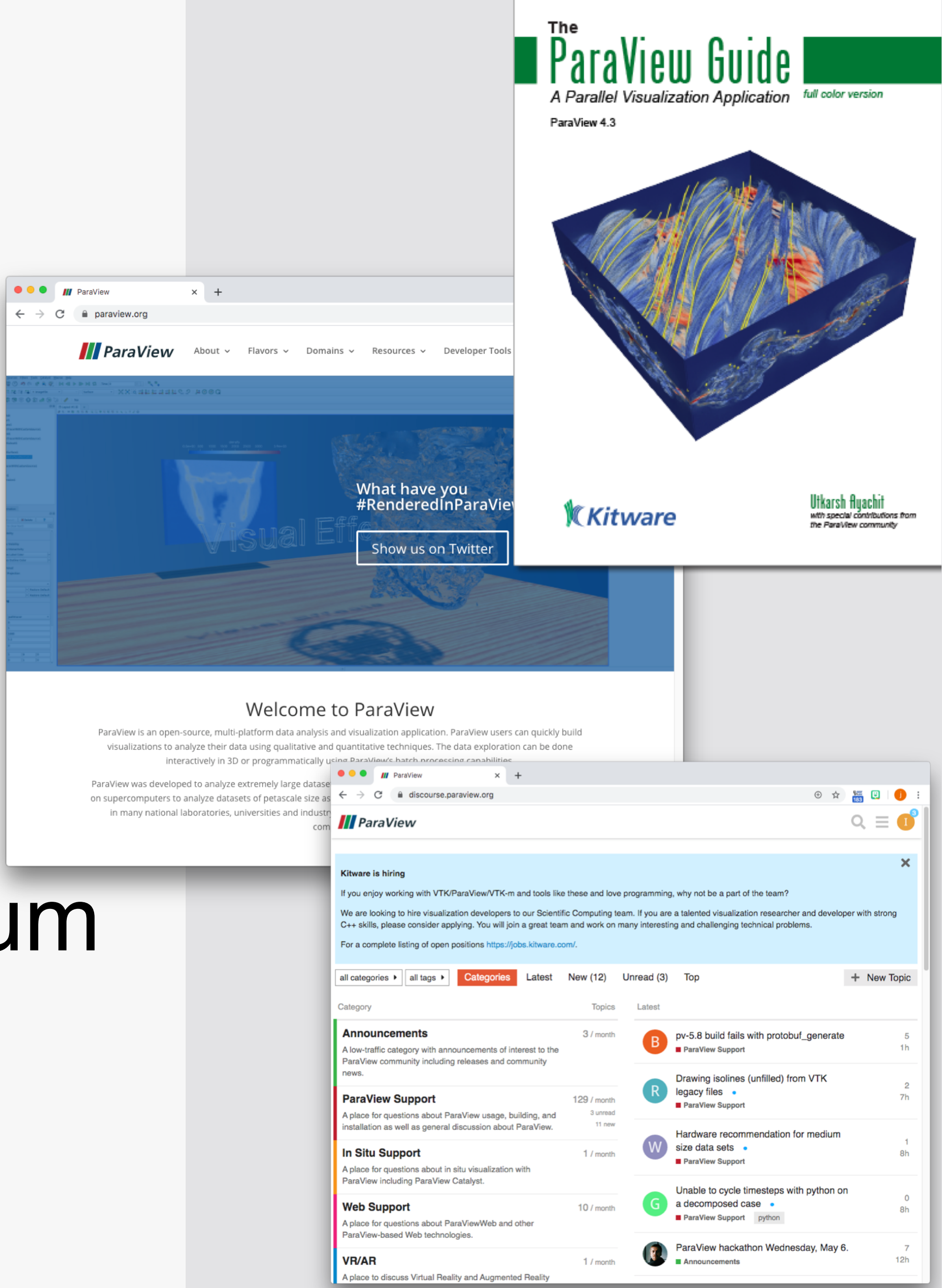
–www.paraview.org/paraview-guide

The ParaView web page

– www.paraview.org

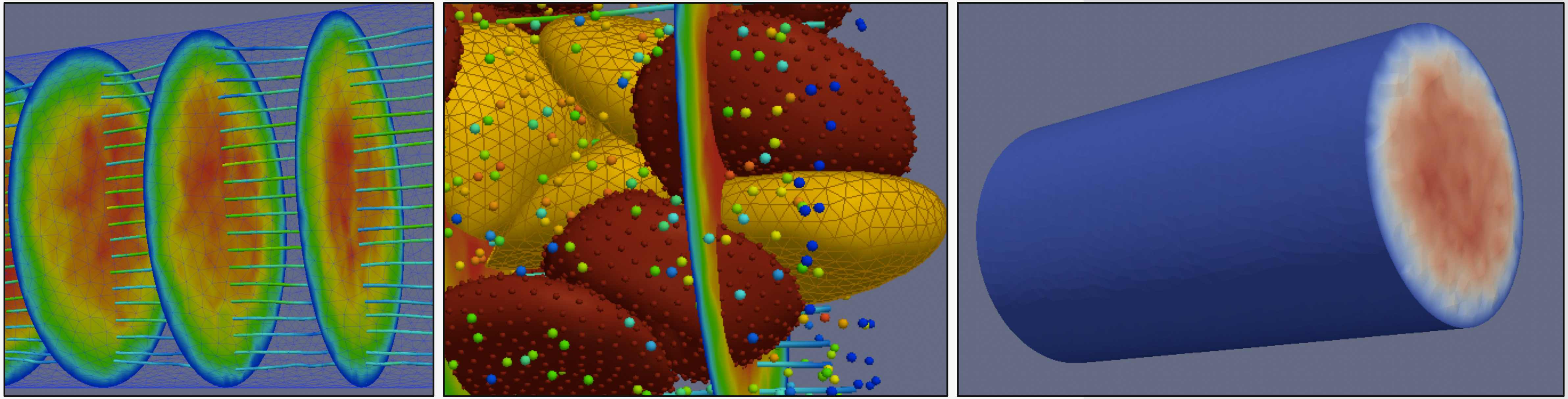
ParaView Discourse Support Forum

– discourse.paraview.org



Goals

- Tour of ParaView
- Launch ParaView in Client/Server mode on Cooley
- Show range of visualization methods
- Feel for ParaView “way”
- Scripting and running in batch mode



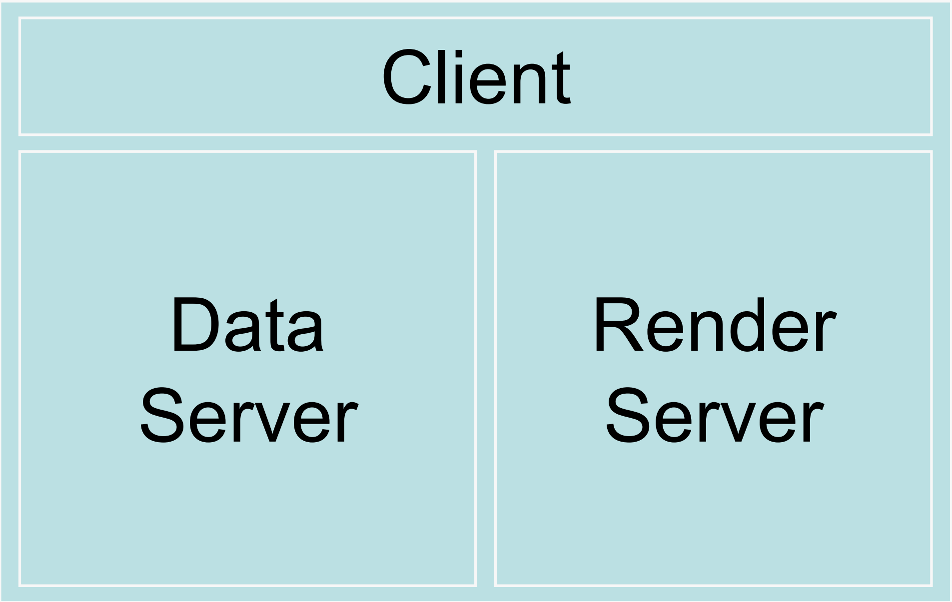
Data

- Blood flow simulation data
- Multiple data types
  - Continuum data field (unstructured mesh, tetrahedral)
  - Particle data (unstructured points)
  - Red Blood Cells (RBC, polygonal mesh, triangles)
- Generated using NekTar/LAMMPS simulation code
- Courtesy of George Karniadakis and Leopold Grinberg of Brown University

ParaView Architecture

- Three tier
  - Data Server
  - Render Server
  - Client

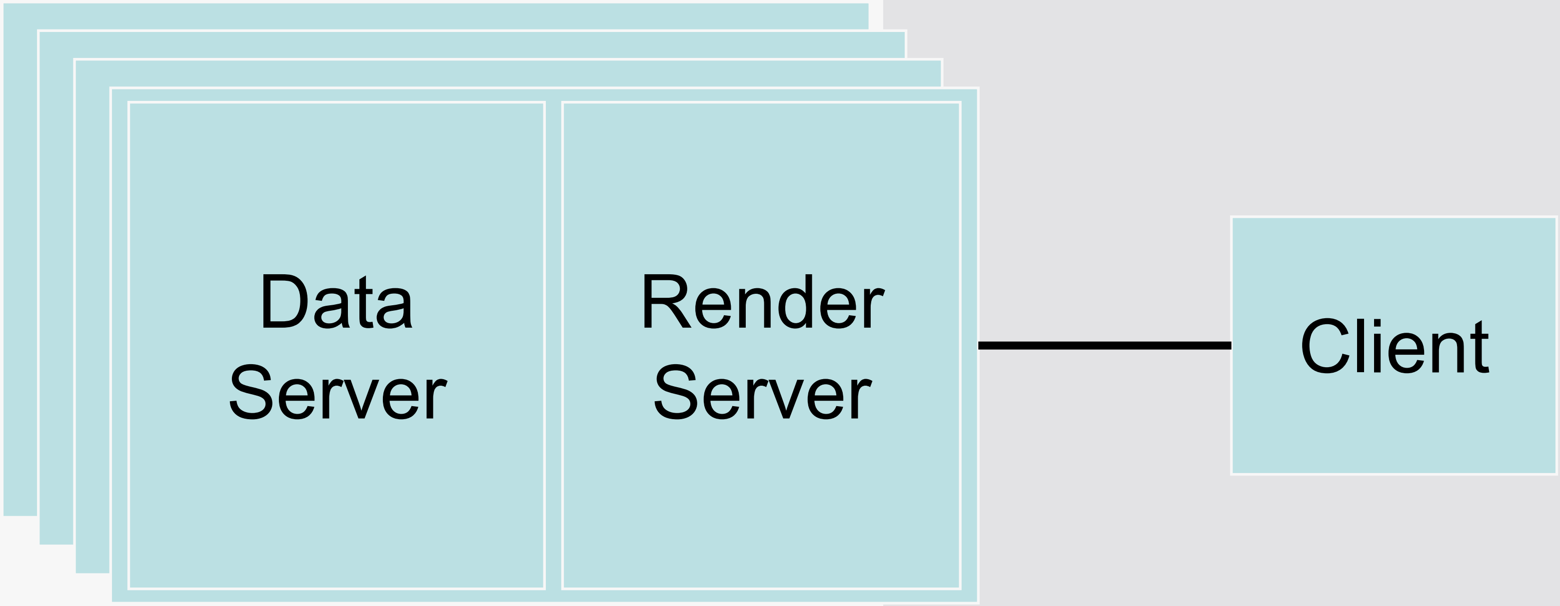
Standalone



ParaView Architecture

- Three tier
  - Data Server
  - Render Server
  - Client

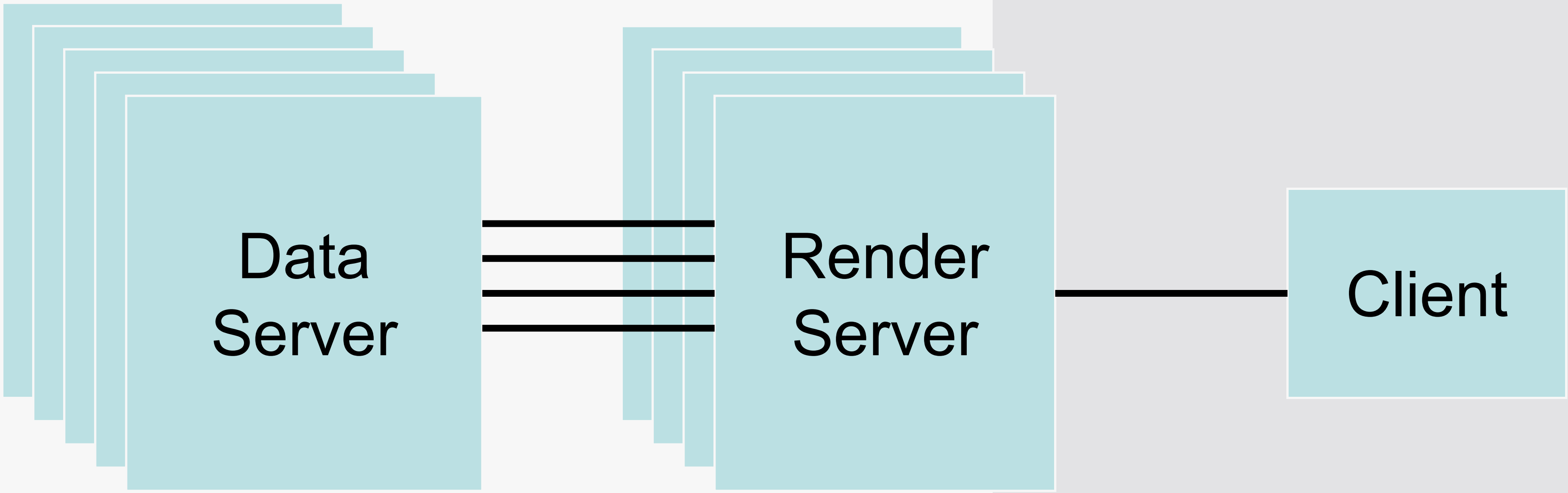
Client-Server



ParaView Architecture

- Three tier
  - Data Server
  - Render Server
  - Client

Client-Render Server-Data Server



ParaView on Cooley / Theta

Version 5.8.0 on Cooley

Version 5.7.0 on Theta

(Client and Server versions must match)

- On Cooley, add the following to your .soft.cooley file (before @default):

  @paraview-5.6.1

ParaView on Cooley / Theta

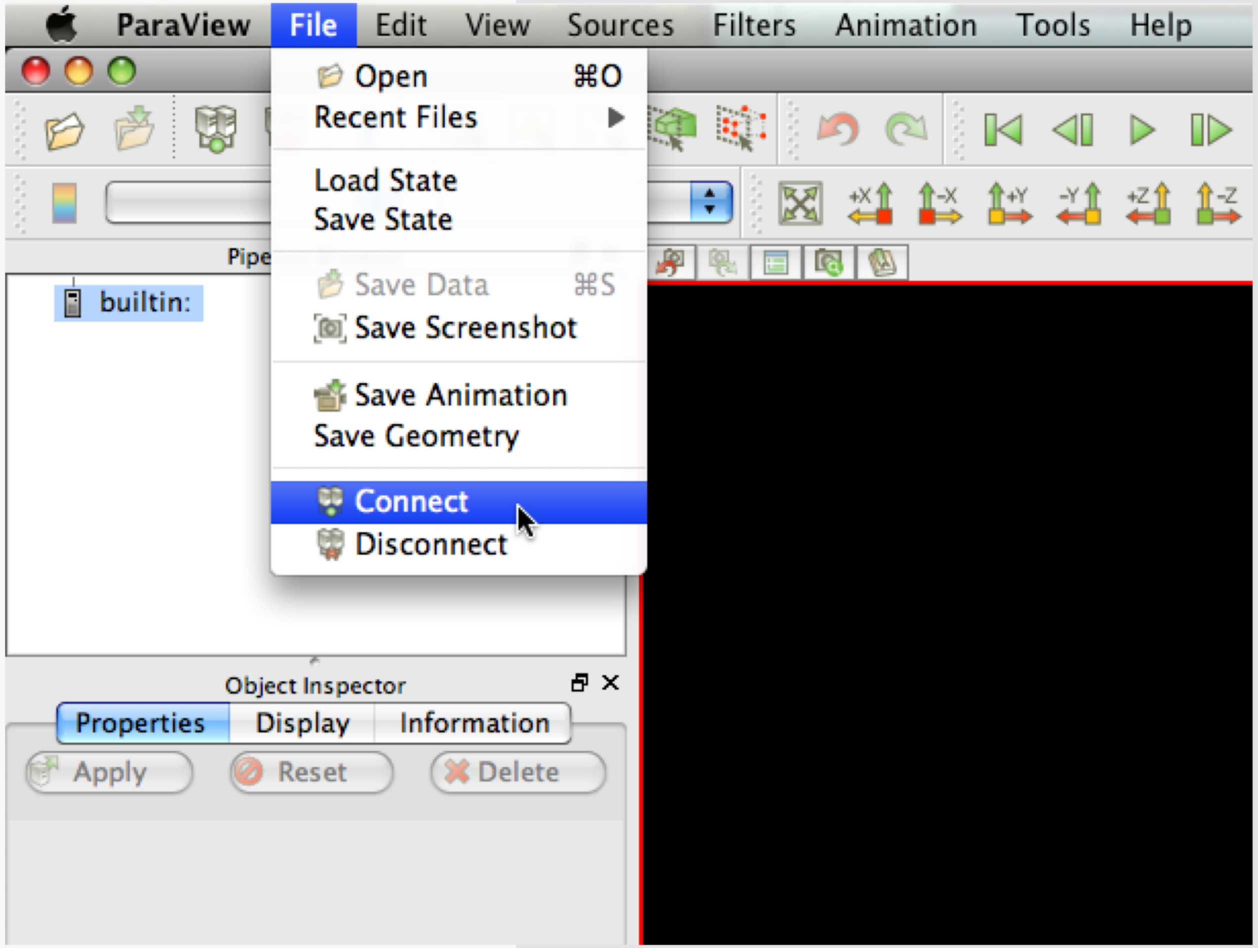
Version 5.8.0 on Cooley

Version 5.7.0 on Theta

(Client and Server versions must match)

After launching client locally

– Connect
– Fetch servers (first time only)
– Fetch Cooley configuration
– Connect
– Configure server settings
– Connecting: Enter Password
– Open File



ParaView on Cooley / Theta

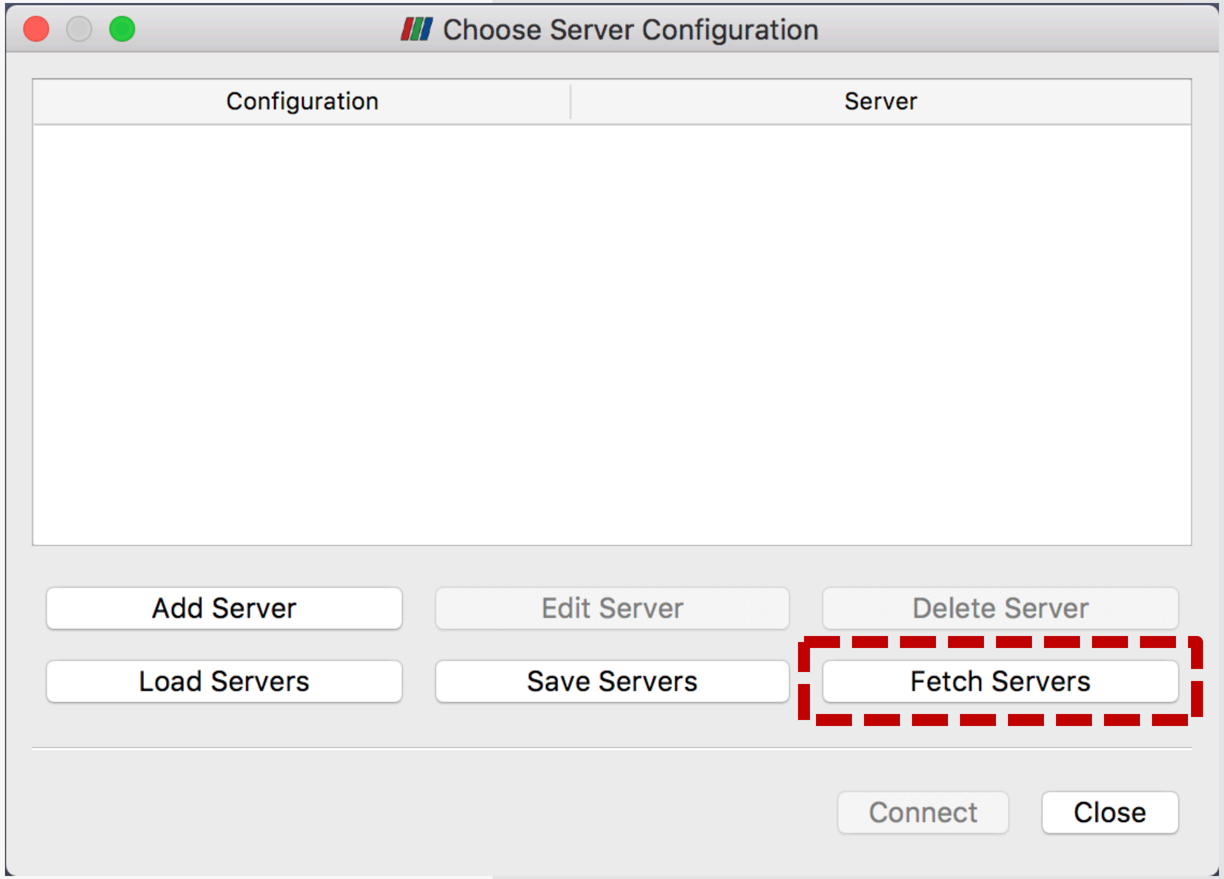
Version 5.8.0 on Cooley

Version 5.7.0 on Theta

(Client and Server versions must match)

After launching client locally

– Connect
– Fetch servers (first time only)
– Fetch Cooley configuration
– Connect
– Configure server settings
– Connecting: Enter Password
– Open File



ParaView on Cooley / Theta

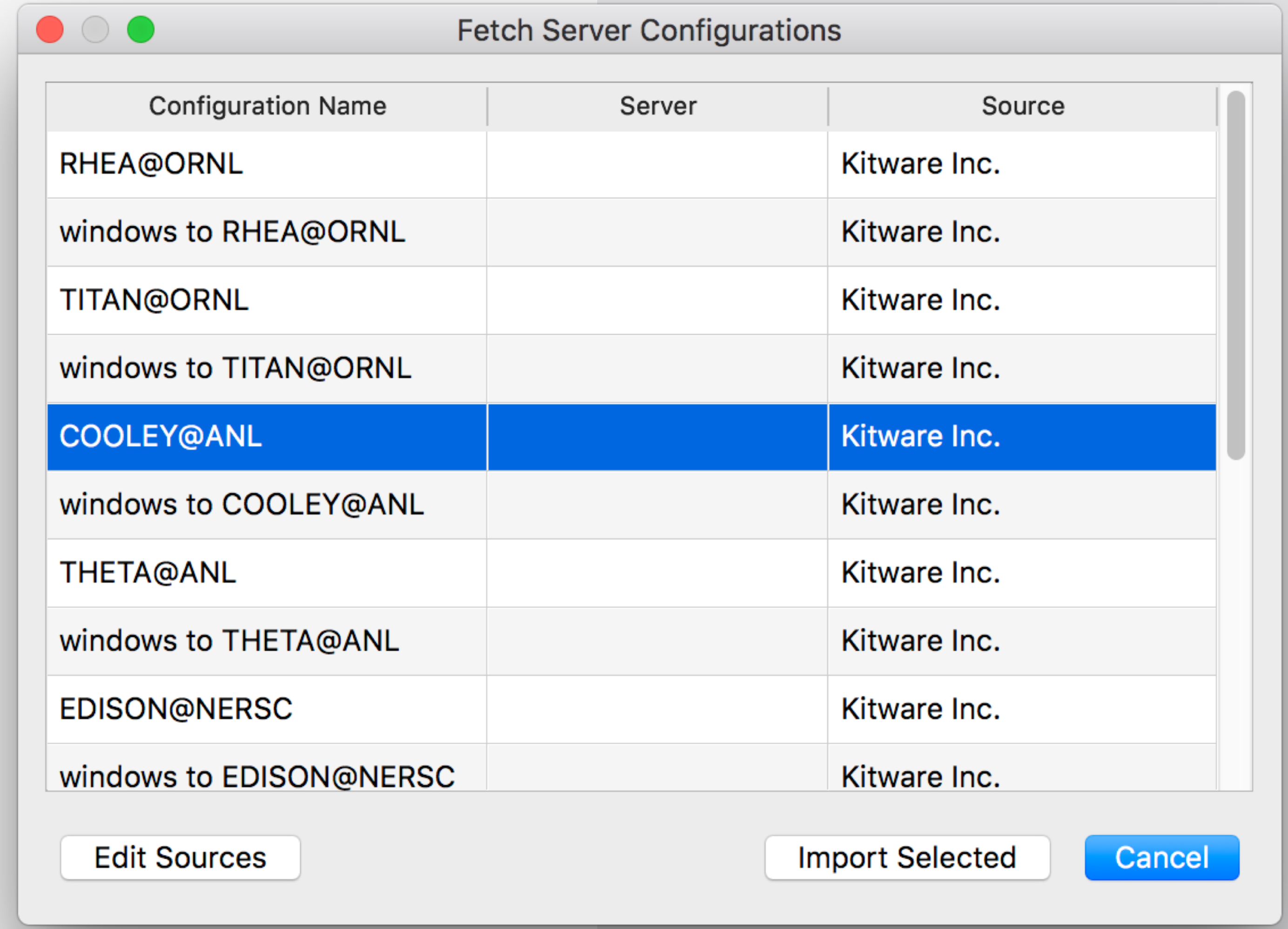
Version 5.8.0 on Cooley

Version 5.7.0 on Theta

(Client and Server versions must match)

After launching client locally

– Connect
– Fetch servers (first time only)
– Fetch Cooley configuration
– Connect
– Configure server settings
– Connecting: Enter Password
– Open File



ParaView on Cooley / Theta

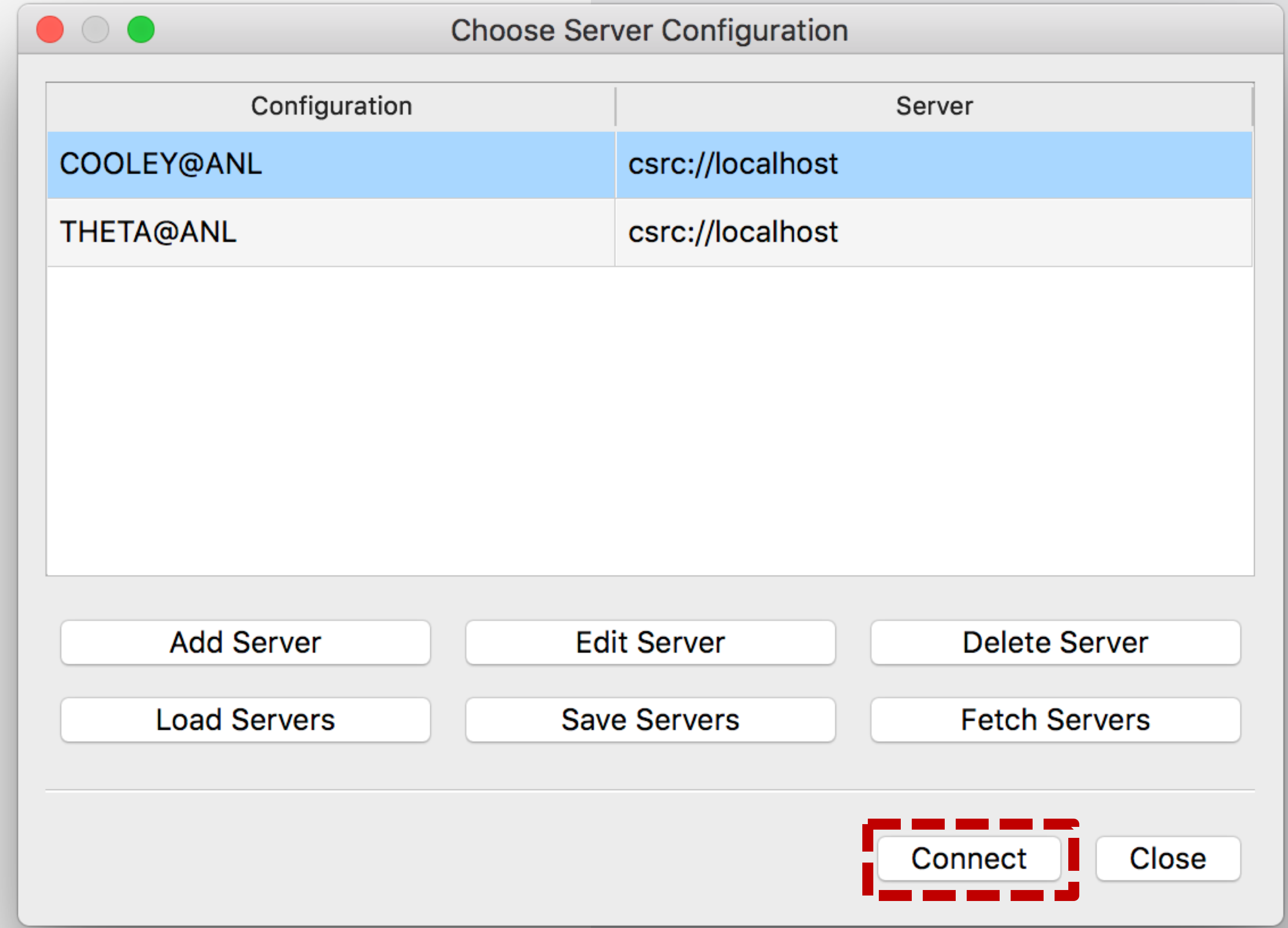
Version 5.8.0 on Cooley

Version 5.7.0 on Theta

(Client and Server versions must match)

After launching client locally

– Connect
– Fetch servers (first time only)
– Fetch Cooley configuration
– Connect
– Configure server settings
– Connecting: Enter Password
– Open File



ParaView on Cooley / Theta

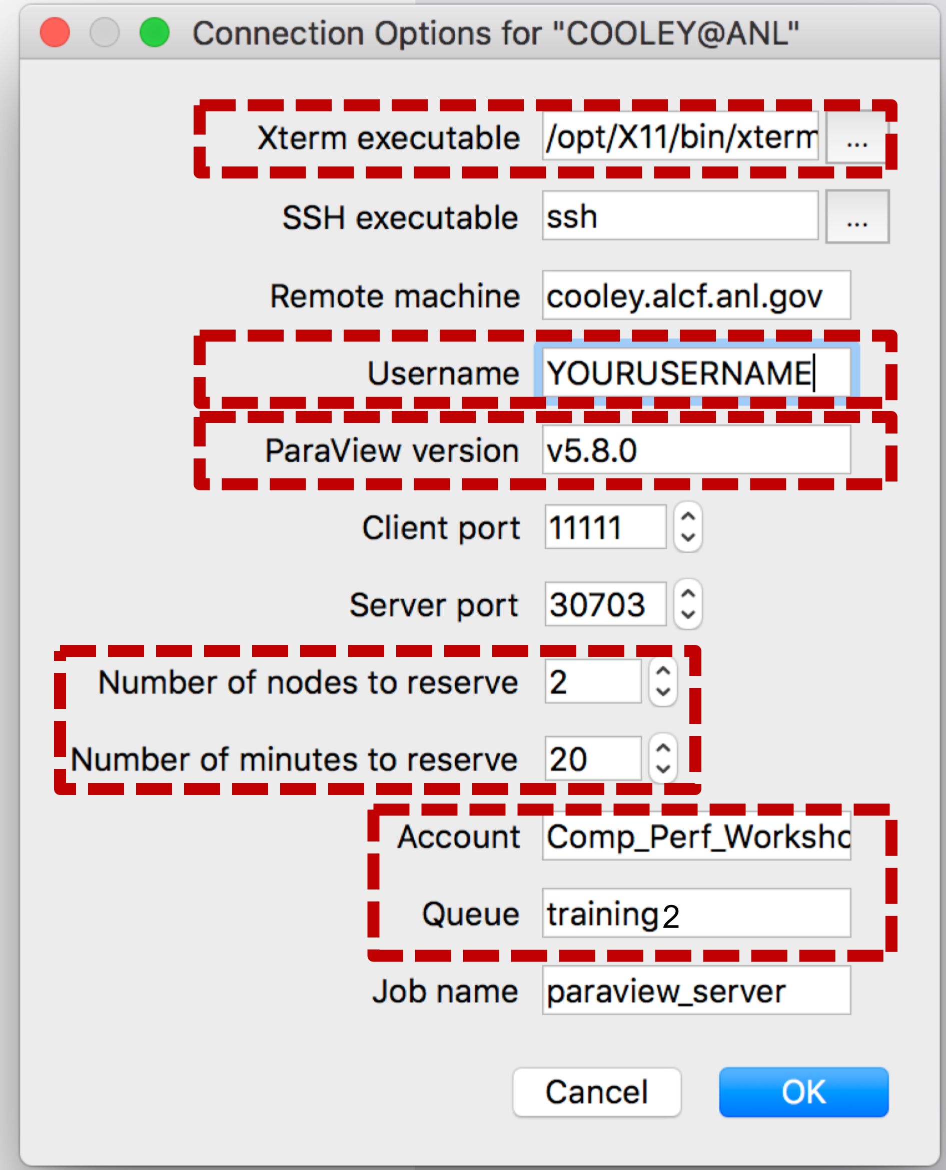
Version 5.8.0 on Cooley

Version 5.7.0 on Theta

(Client and Server versions must match)

After launching client locally

- Connect
- Fetch servers (first time only)
- Fetch Cooley configuration
- Connect
- Configure server settings
- Connecting: Enter Password
- Open File



Argonne NATIONAL LABORATORY

ParaView on Cooley / Theta

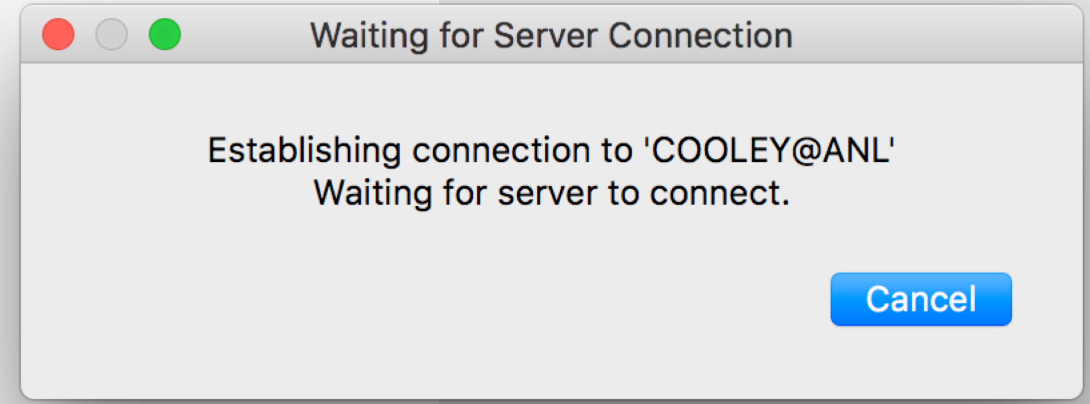
Version 5.8.0 on Cooley

Version 5.7.0 on Theta

(Client and Server versions must match)

After launching client locally

- Connect
- Fetch servers (first time only)
- Fetch Cooley configuration
- Connect
- Configure server settings
- Connecting: Enter Password
- Open File



ParaView on Cooley / Theta

Version 5.8.0 on Cooley

Version 5.7.0 on Theta

(Client and Server versions must match)

After launching client locally

– Connect

– Fetch servers (first time only)

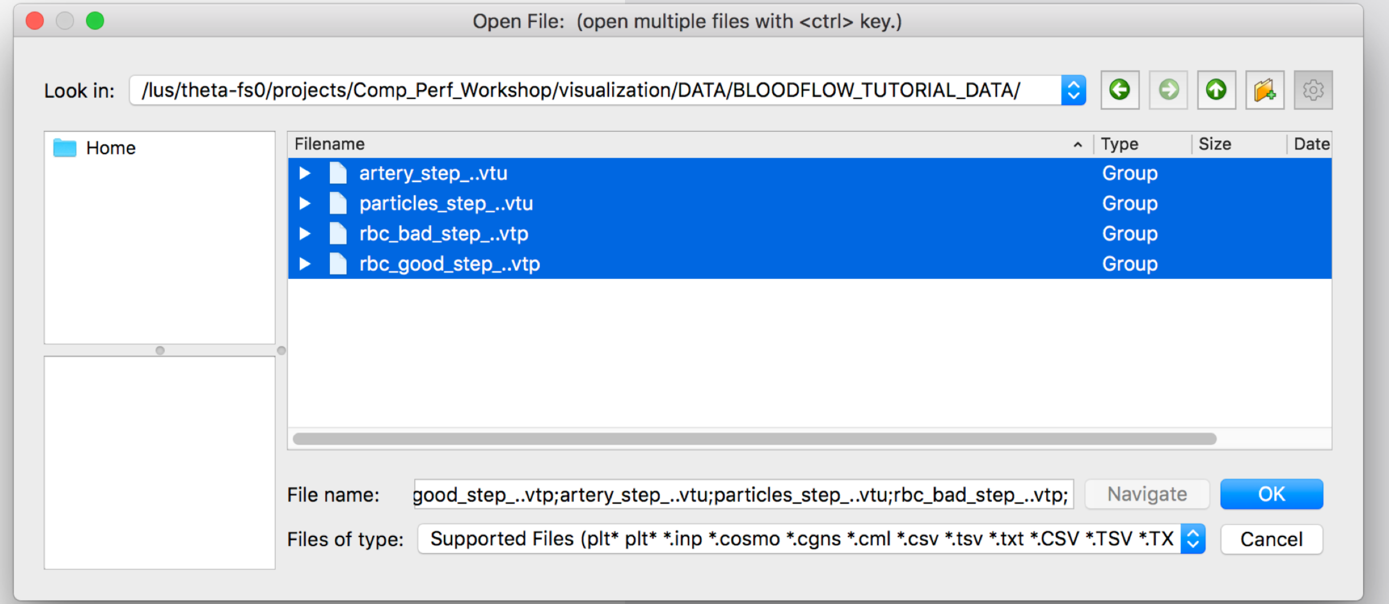
– Fetch Cooley configuration

– Connect

– Configure server settings

– Connecting: Enter Password

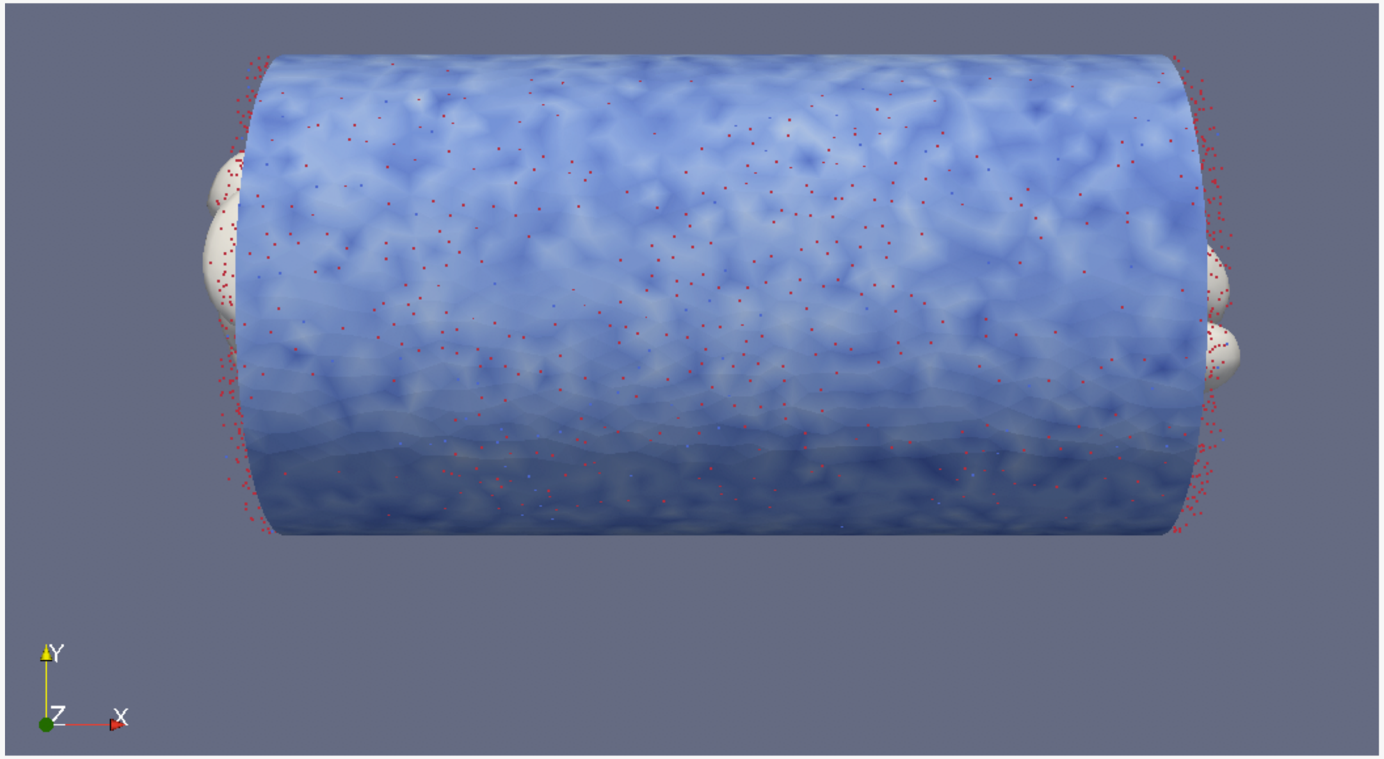
– Open File

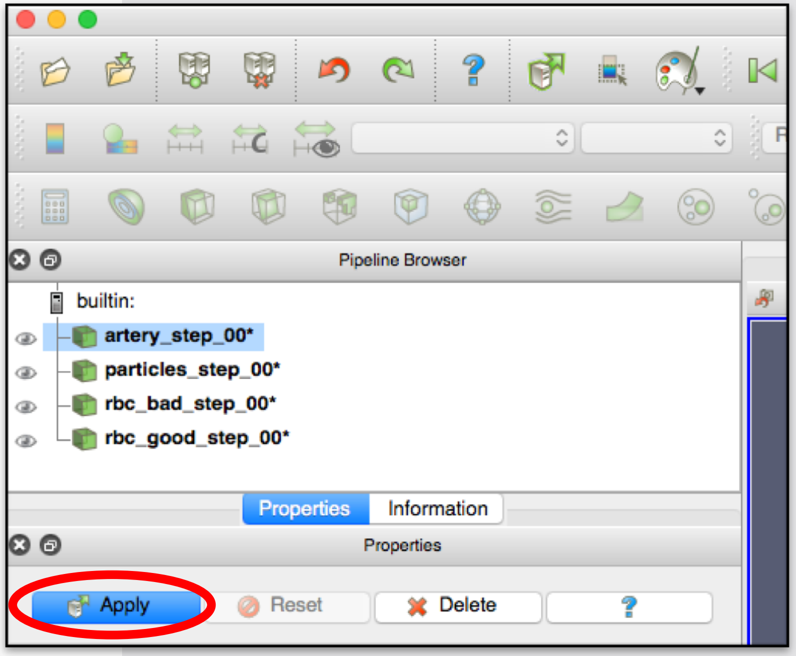


Path: */lus/theta-fs0/projects/Comp_Perf_Workshop/visualization/DATA/BLOODFLOW_TUTORIAL_DATA*

Load Multi-component Dataset

- Pipeline Browser
  - Click Apply
- Default values will result in:





Argonne NATIONAL LABORATORY

User Interface

Menu Bar

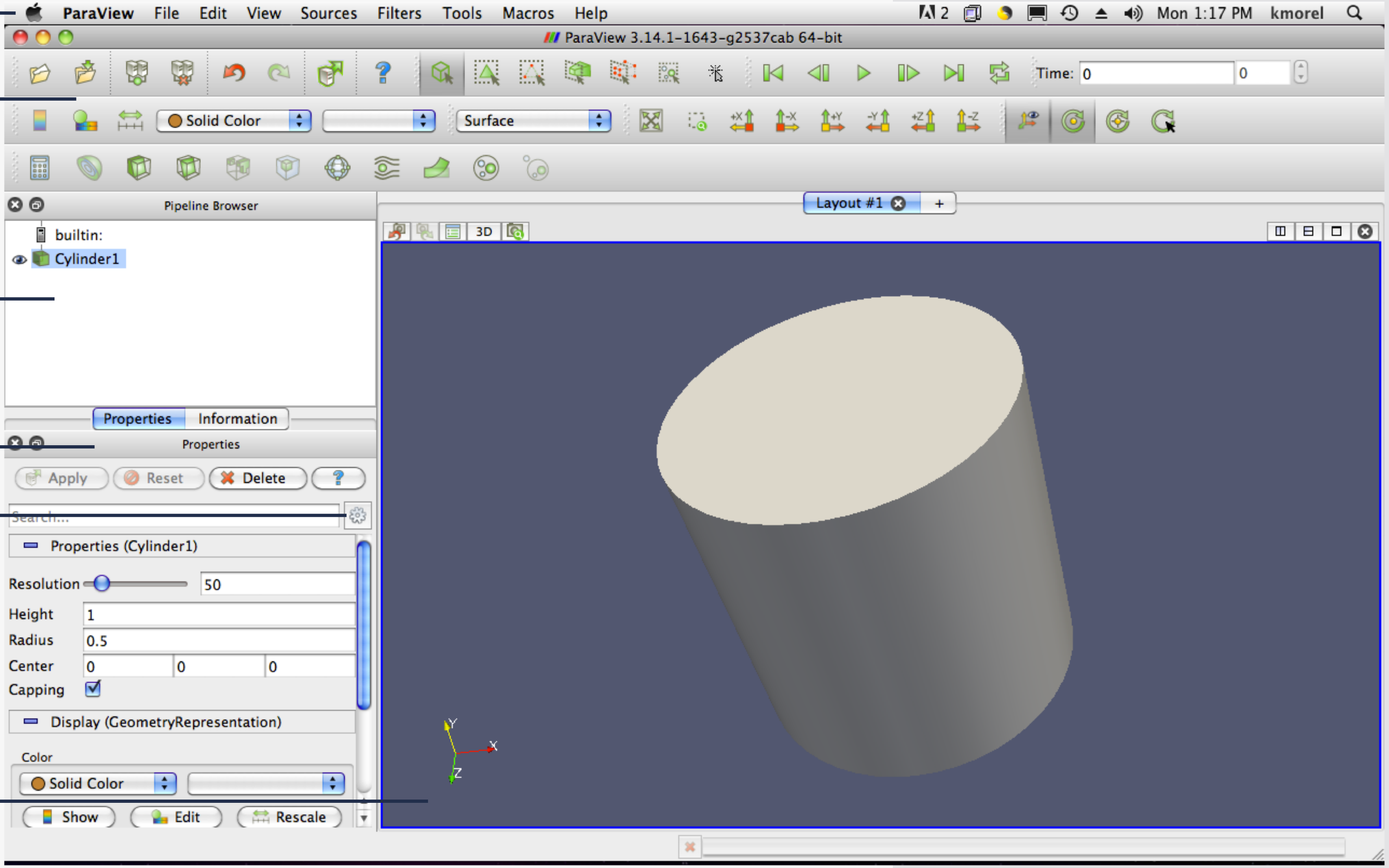
Toolbars

Pipeline Browser

Properties Panel

Advanced Toggle

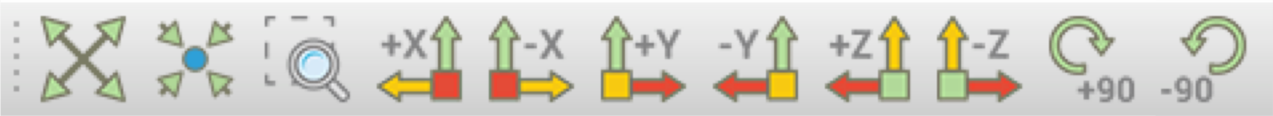
3D View



Argonne NATIONAL LABORATORY

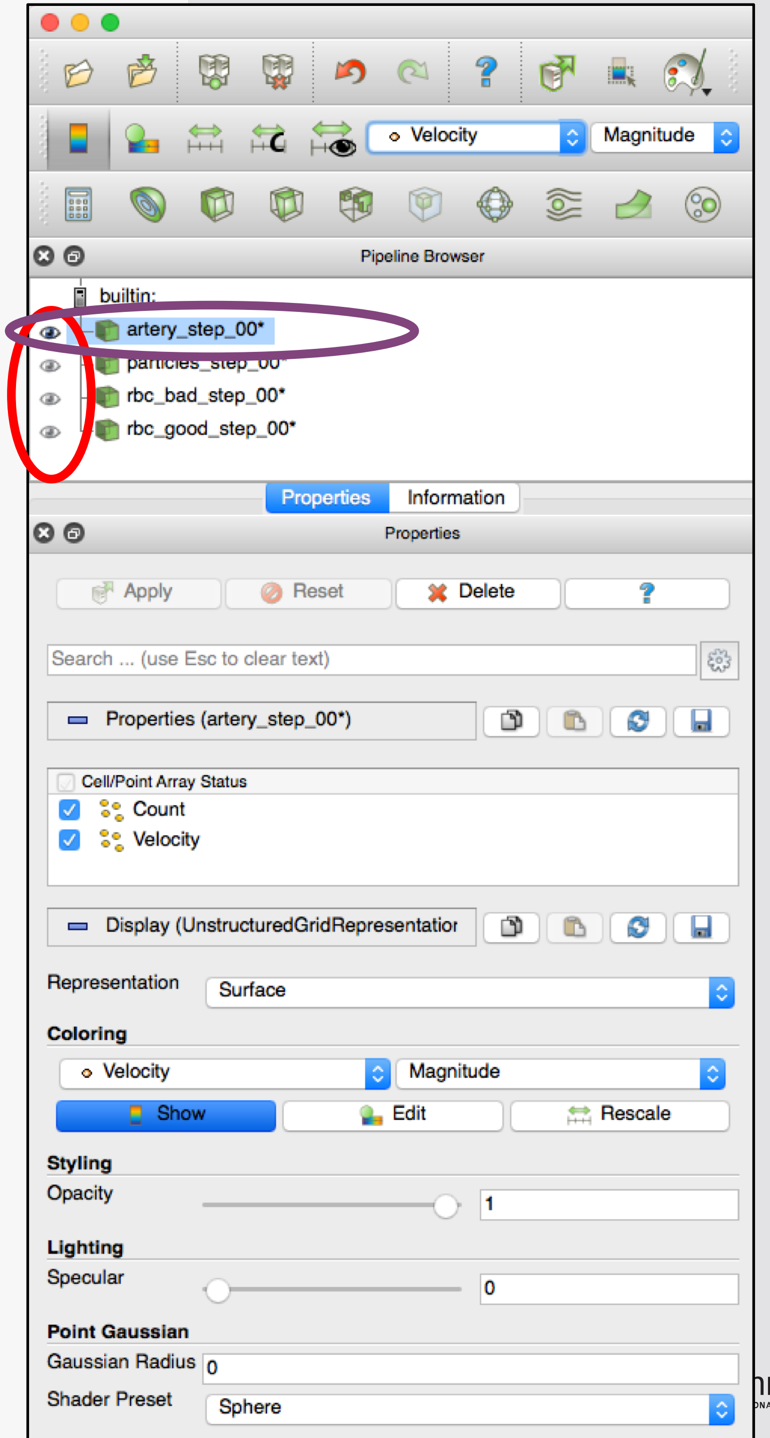
Simple Camera Manipulation

- Drag left, middle, right buttons for rotate, pan, zoom.
  - Also use Shift, Ctrl modifiers.
  - Hold down x, y, or z key to constrain rotation to a specific axis

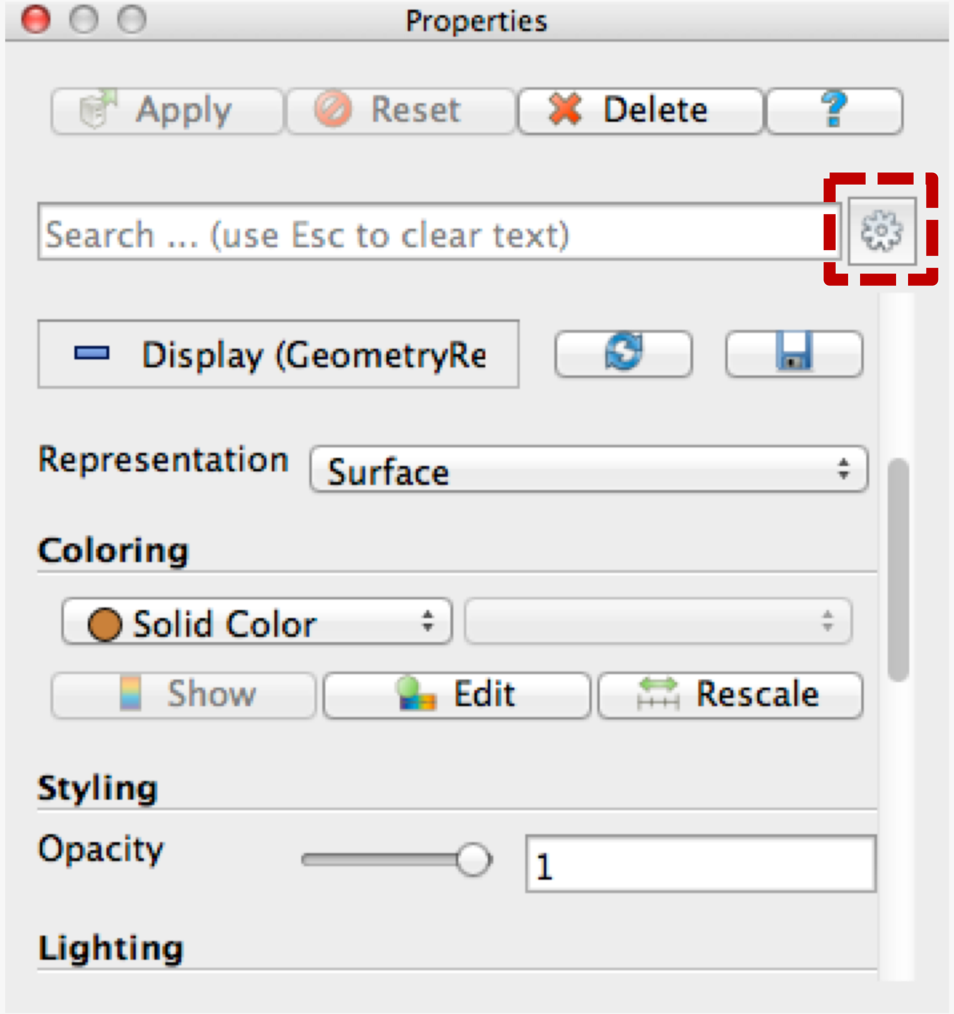


Select which data to view

- Hide data sets with Eyeball icon
  - Black (Open) == visible
  - Grey (Closed)== hidden
- Select artery_step_000* in the Pipeline Browser
  - Click on the name to highlight it
- When manipulating appearance or applying filters, these always affect the selected data set

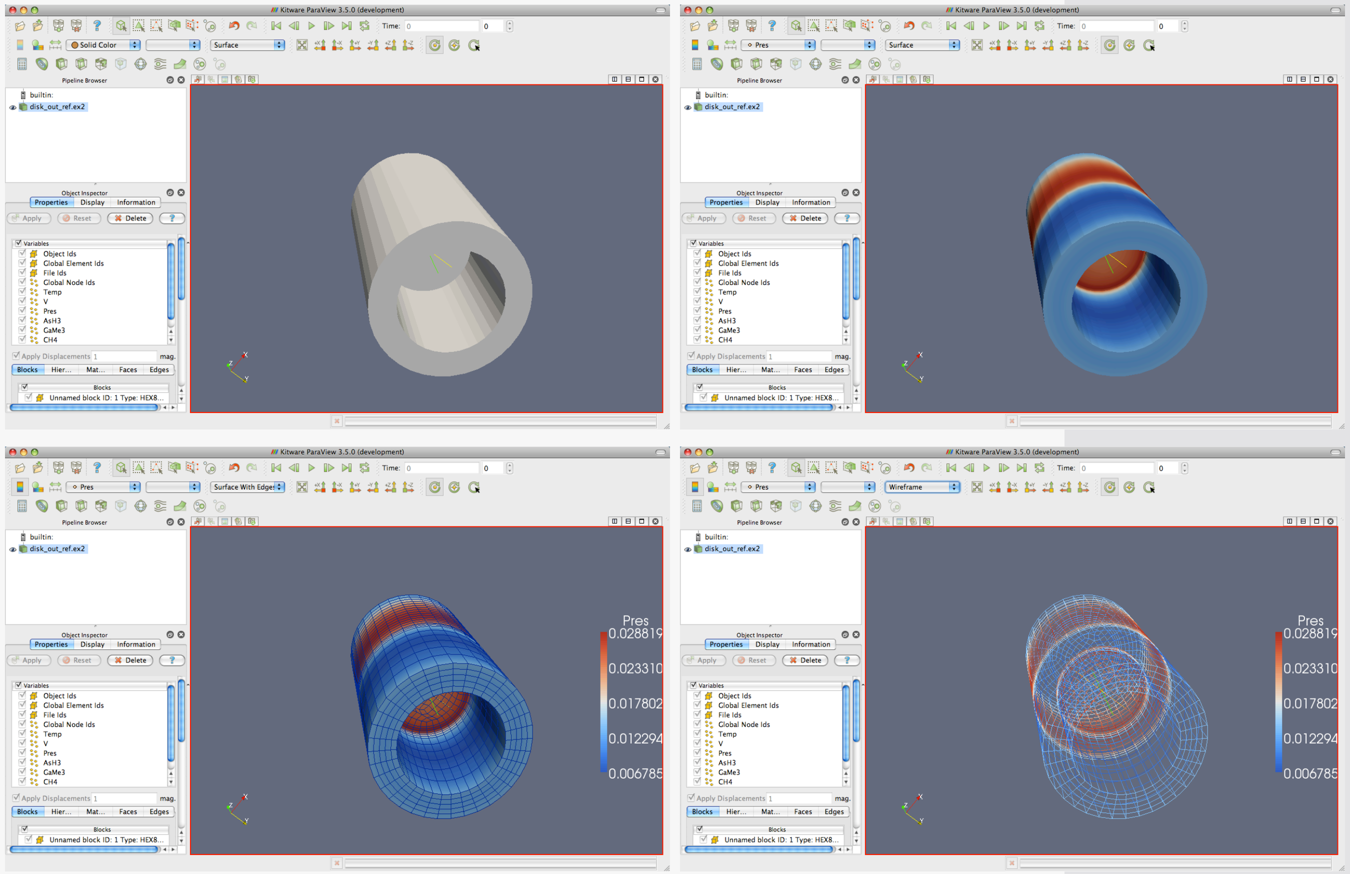


Display Properties



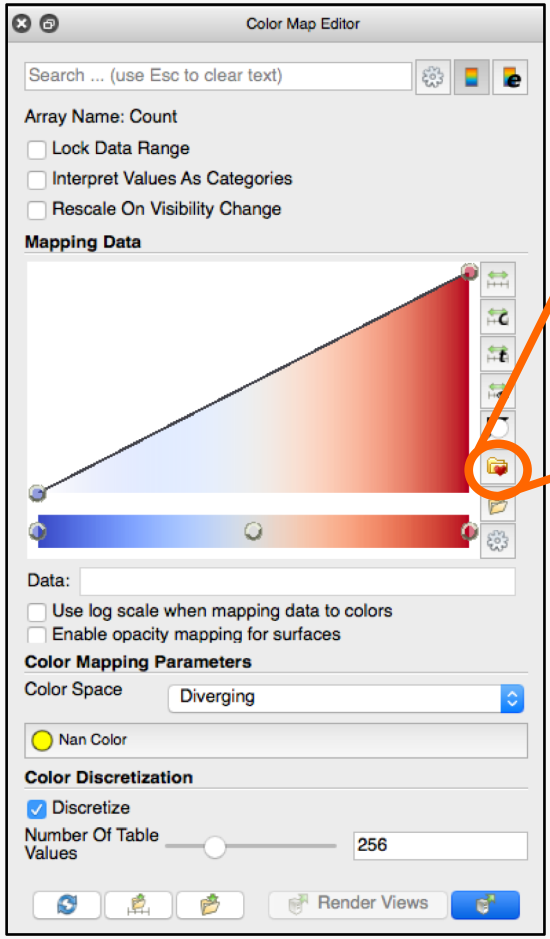
Advanced Properties

Data Representation

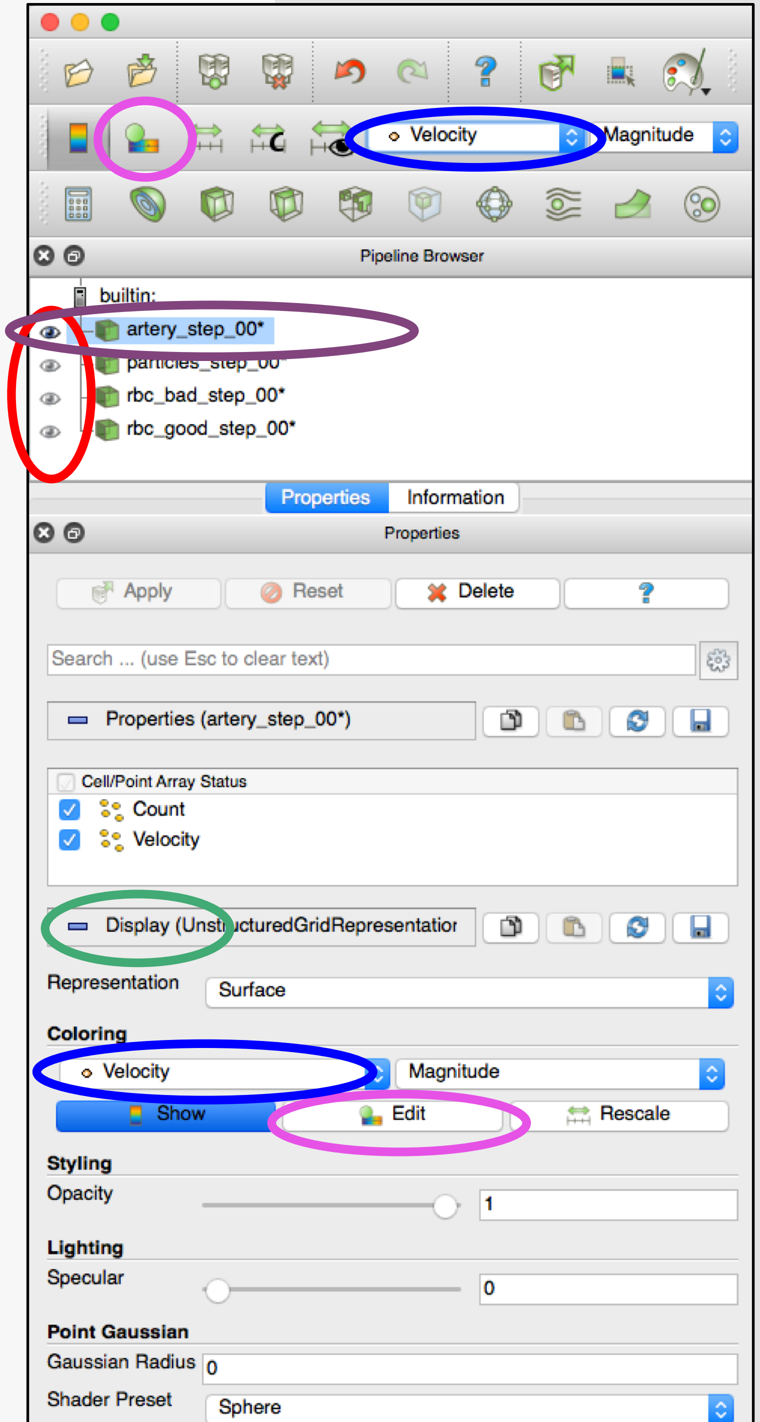


Manipulating the color map

- Under Coloring click Edit



- Click favorites icon
- Select a Preset
  - Default / All dropdown
- Click Apply



Undo Redo

Undo

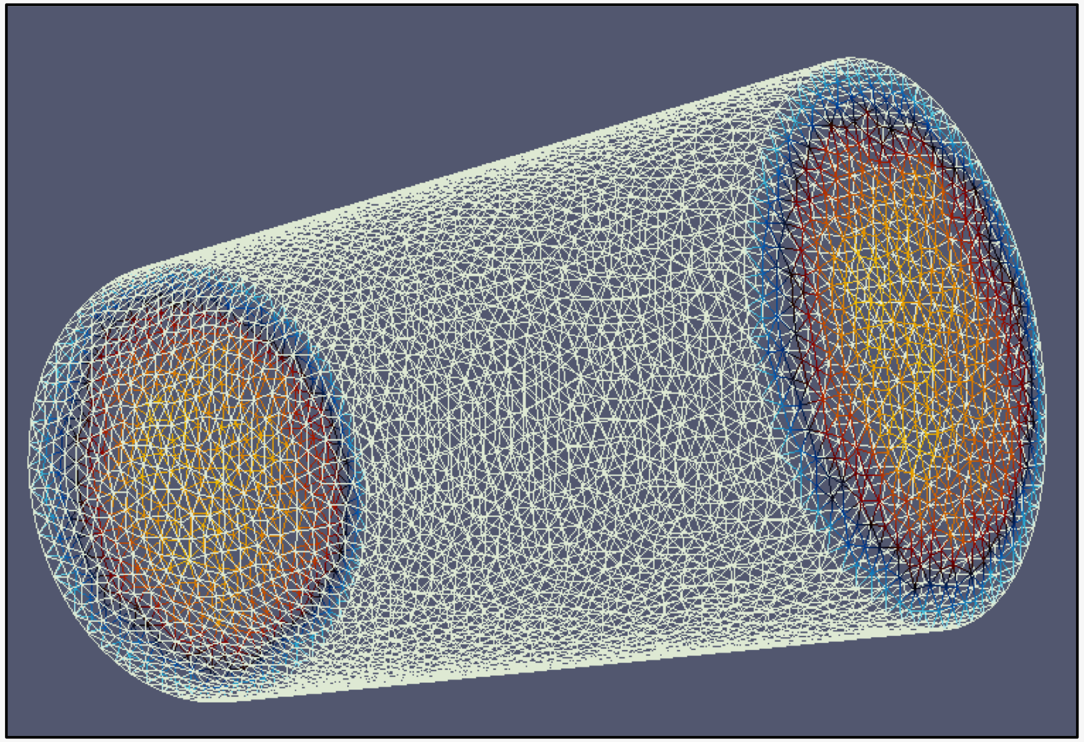
Redo

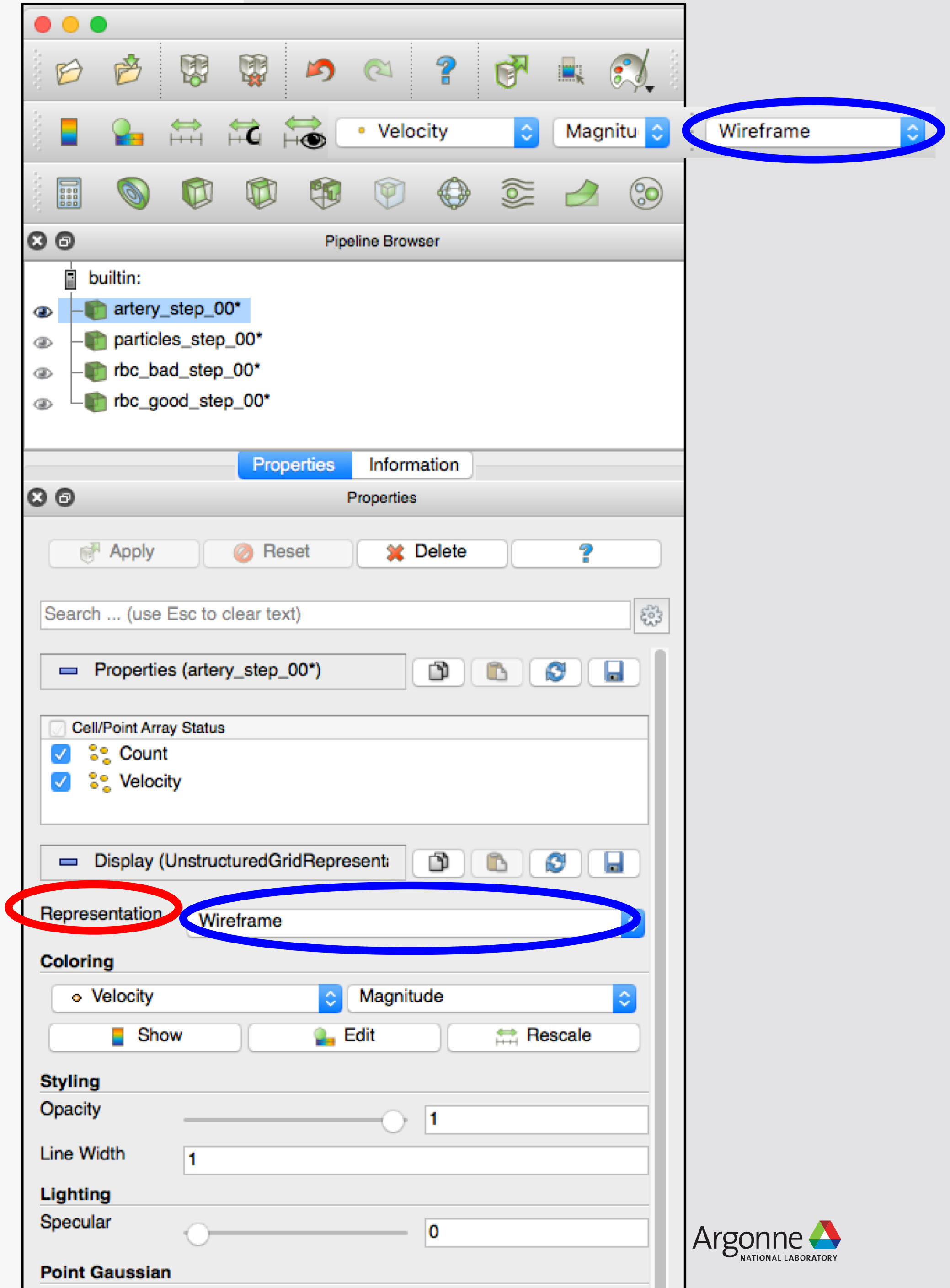
Camera Undo

Camera Redo

Data Representation

- Under Properties: Representation dropdown, select Wireframe



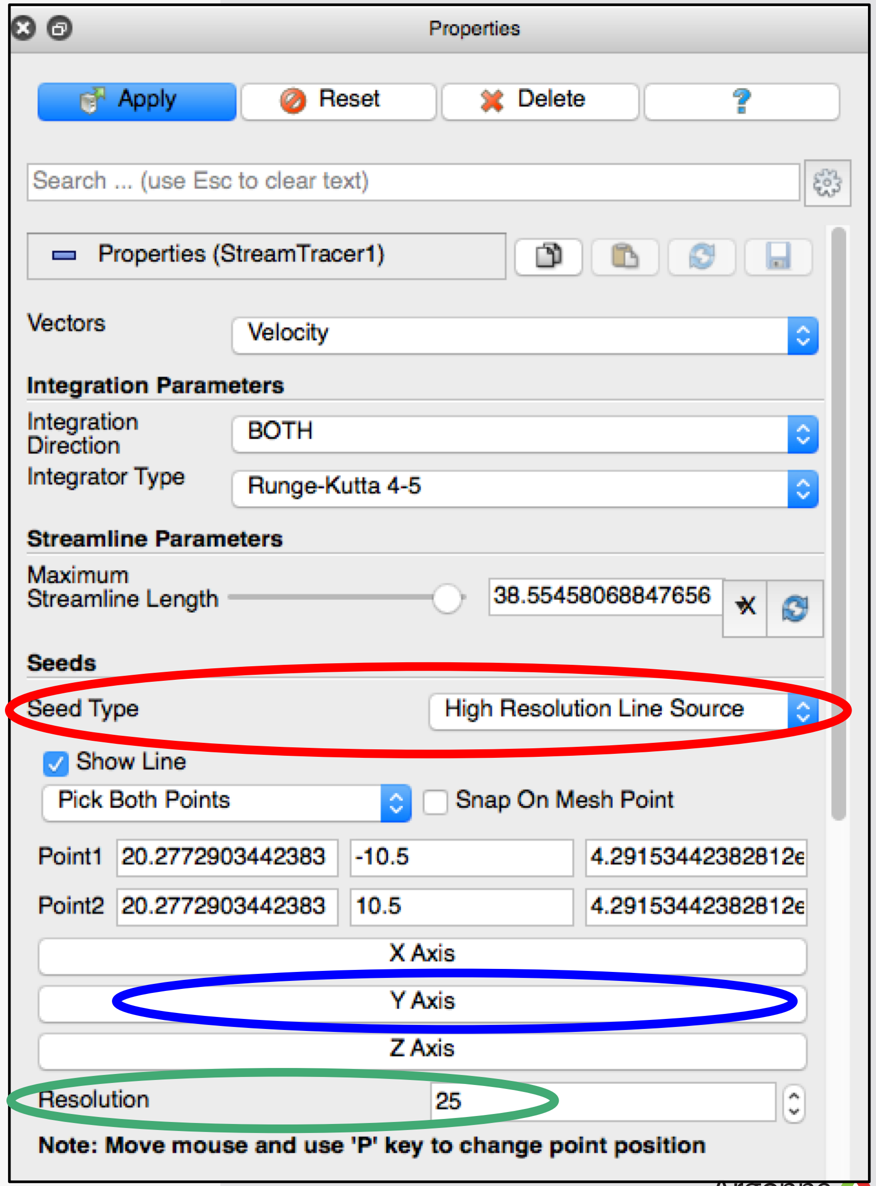


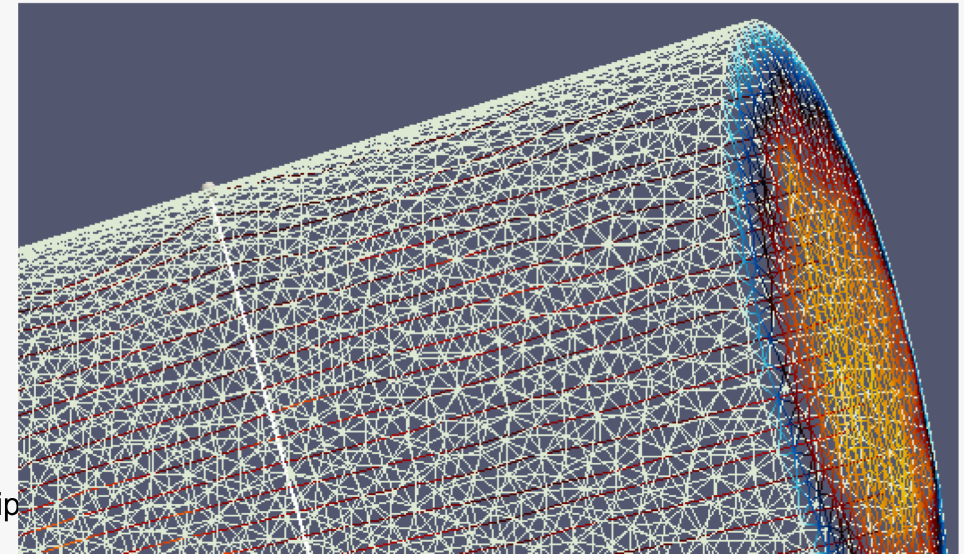
Common Filters

- Calculator
- Contour
- Clip
- Slice
- Threshold
- Extract Subset
- Glyph
- Stream Tracer
- Warp (vector)
- Group Datasets
- Extract Level

Generate Streamlines

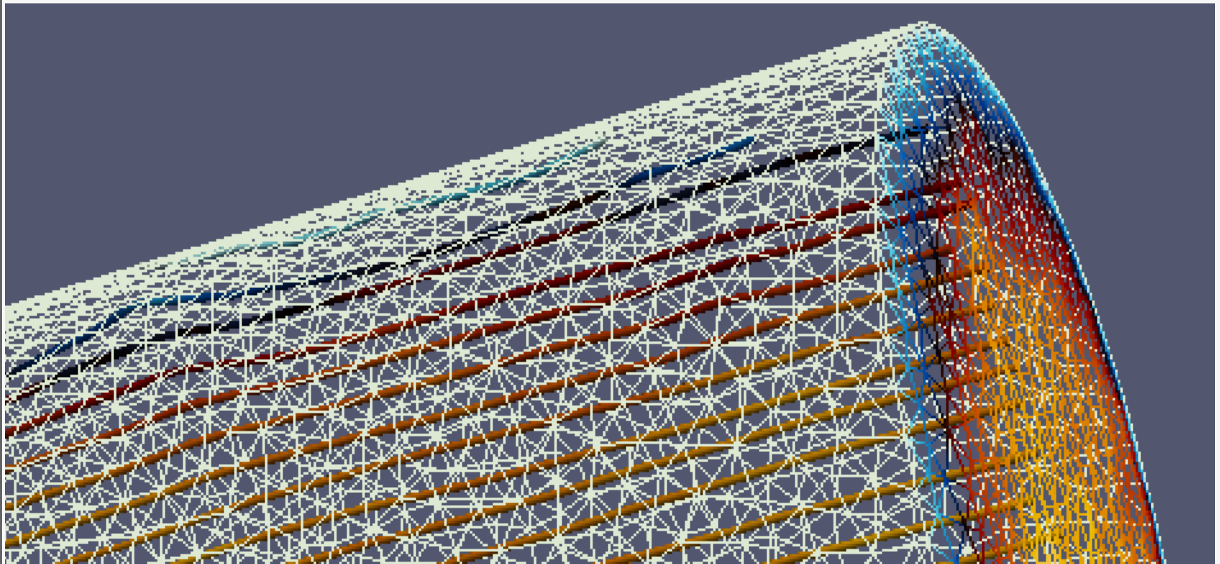
- Make sure artery_step_00* is selected in the Pipeline Browser
- Main menu: Filters-> Alphabetical->Stream Tracer
- Seeds: Seed Type to High Resolution Line Source.
- Click the Y Axis button
- Resolution: 25.
- Apply

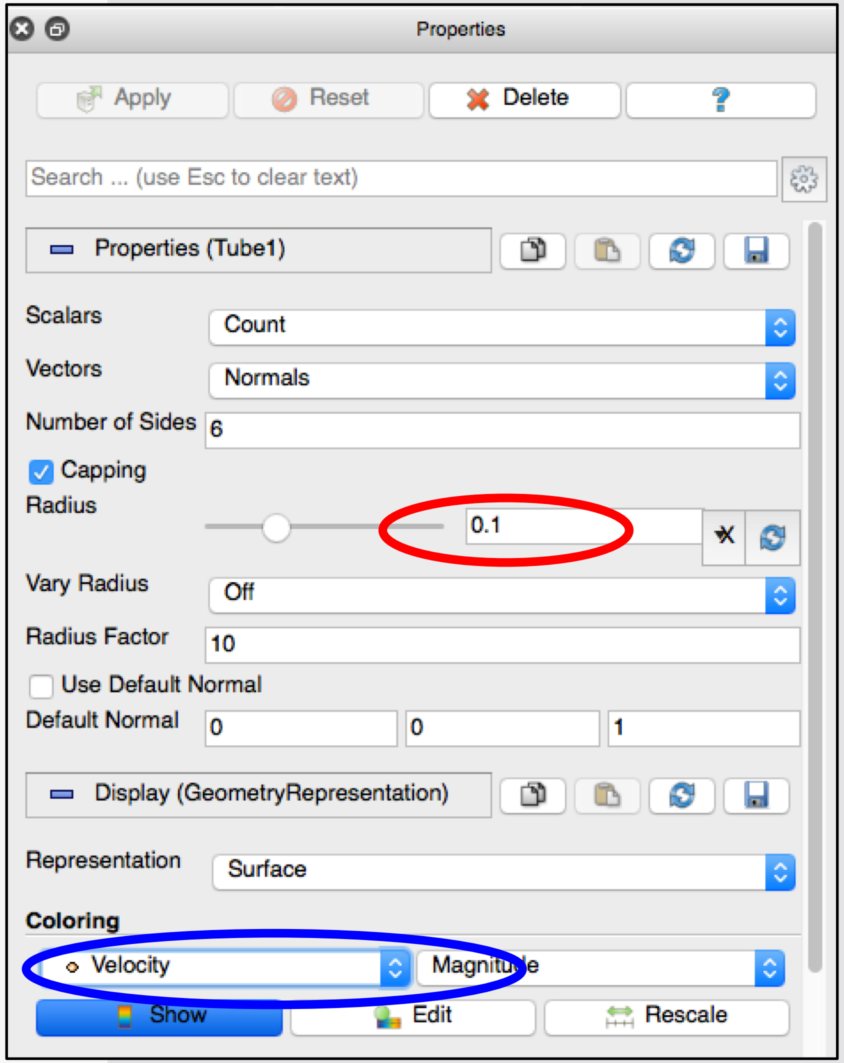




Streamlines as Tubes

- With StreamTracer1 selected: Filters->Alphabetical->Tube
- Radius: 0.1
- Apply
- Coloring: Velocity



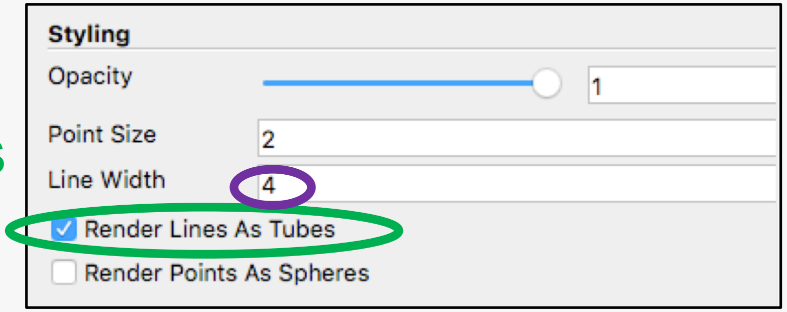


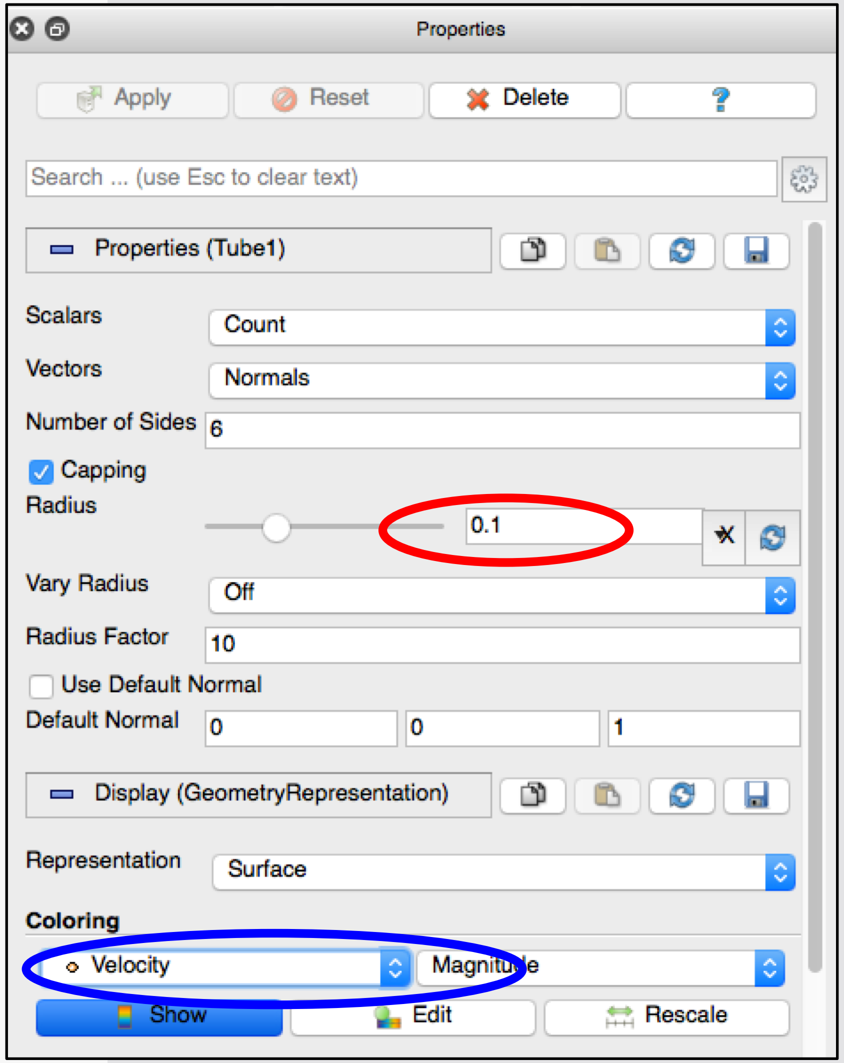
Argonne NATIONAL LABORATORY

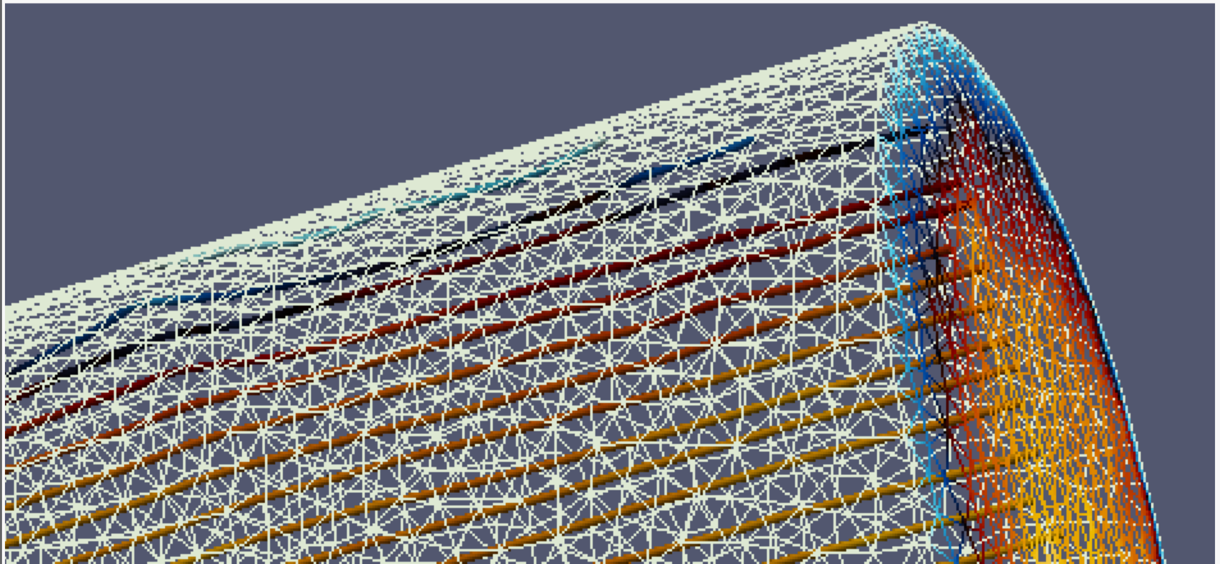
Streamlines as Tubes

- With StreamTracer1 selected: Filters->
  Alphabetical->Tube
- Radius: 0.1
- Apply
- Coloring: Velocity

- Render Lines As Tubes
- Line Width: 4

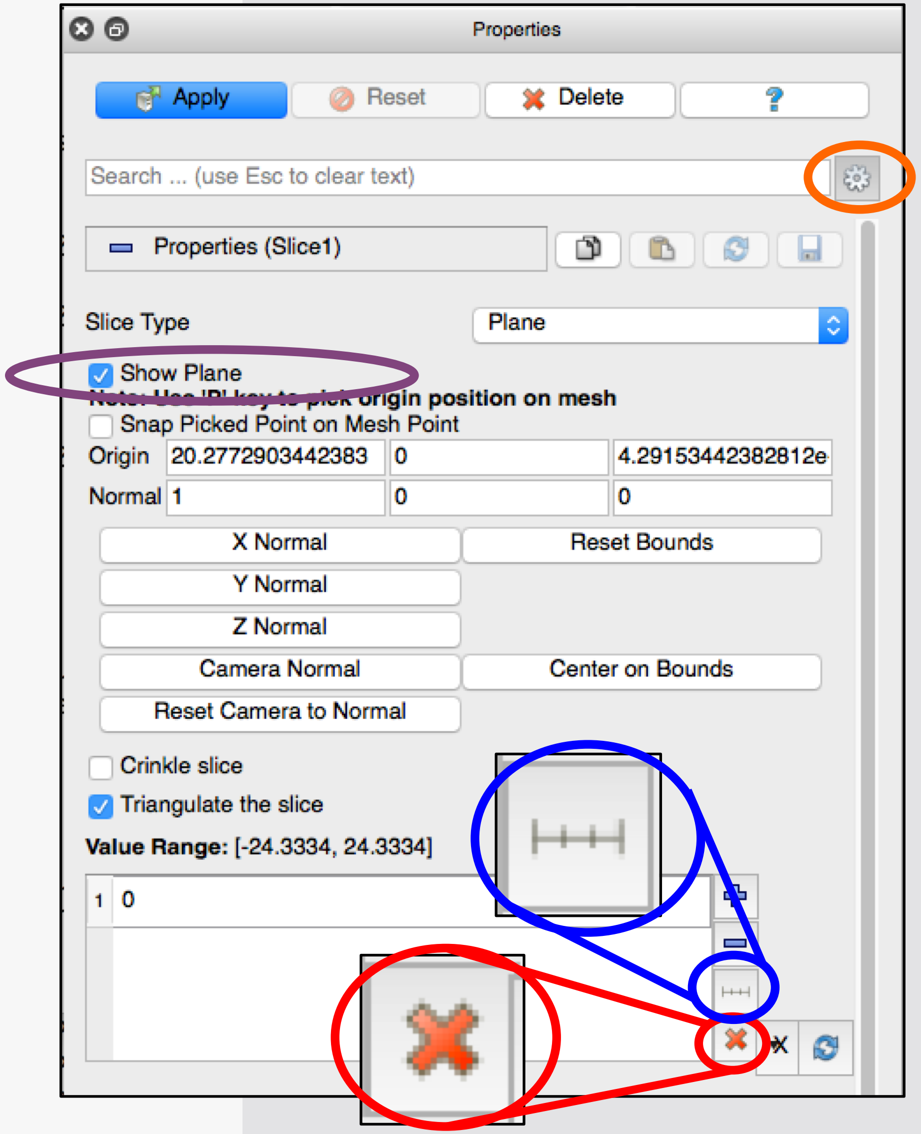


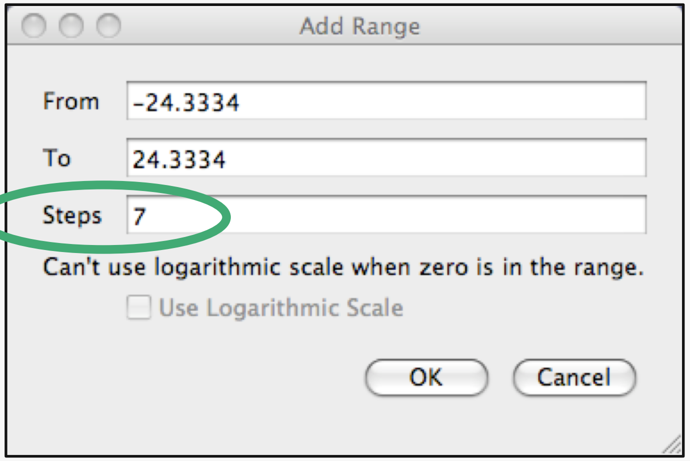




Cutting Planes (Slices)

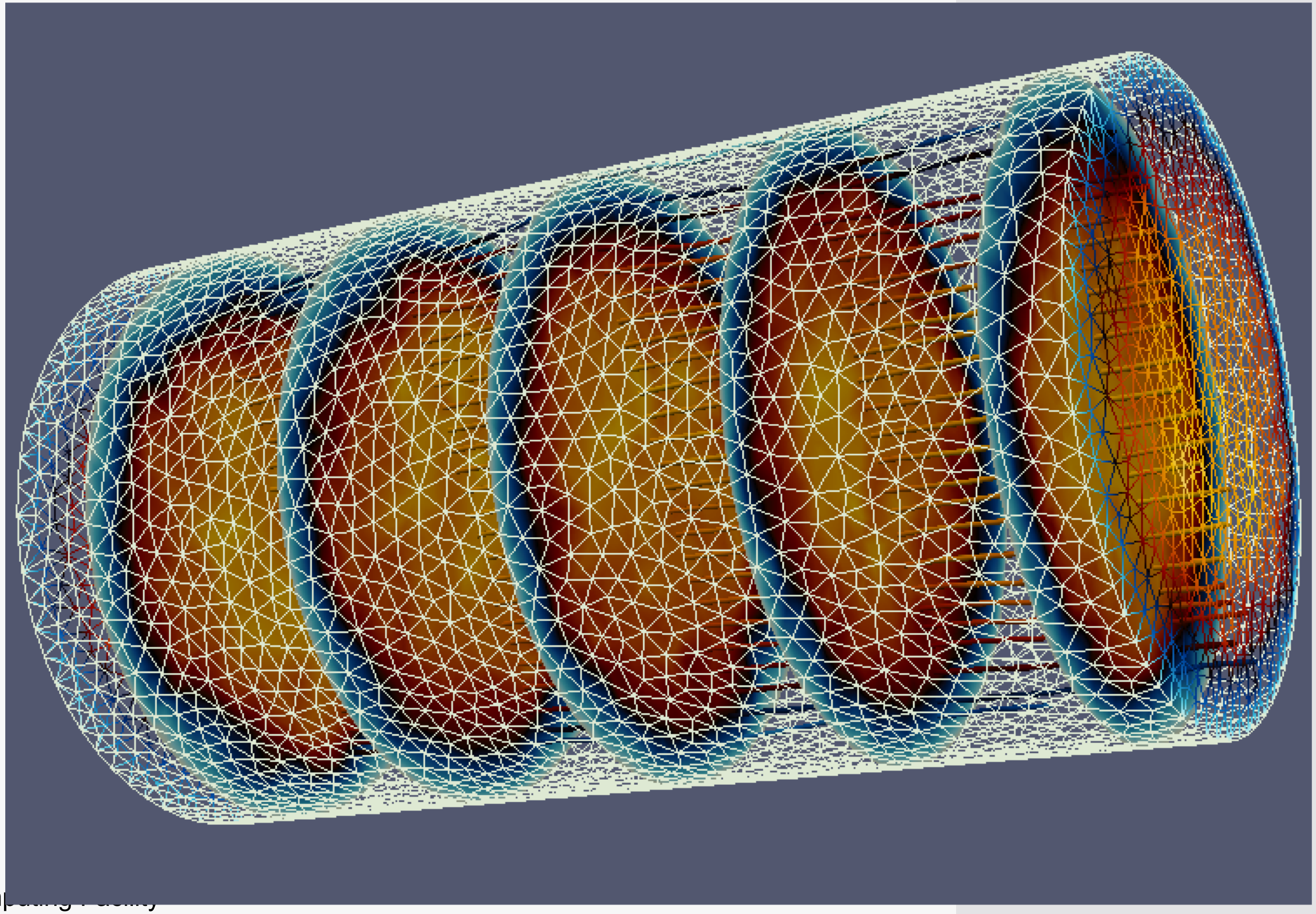
- artery_step_000* selected in the Pipeline Browser
- Filters->Alphabetical->Slice
- Click Gear Icon
- Value Range: Delete All
- New Range
- Steps: 7
- Uncheck: Show Plane





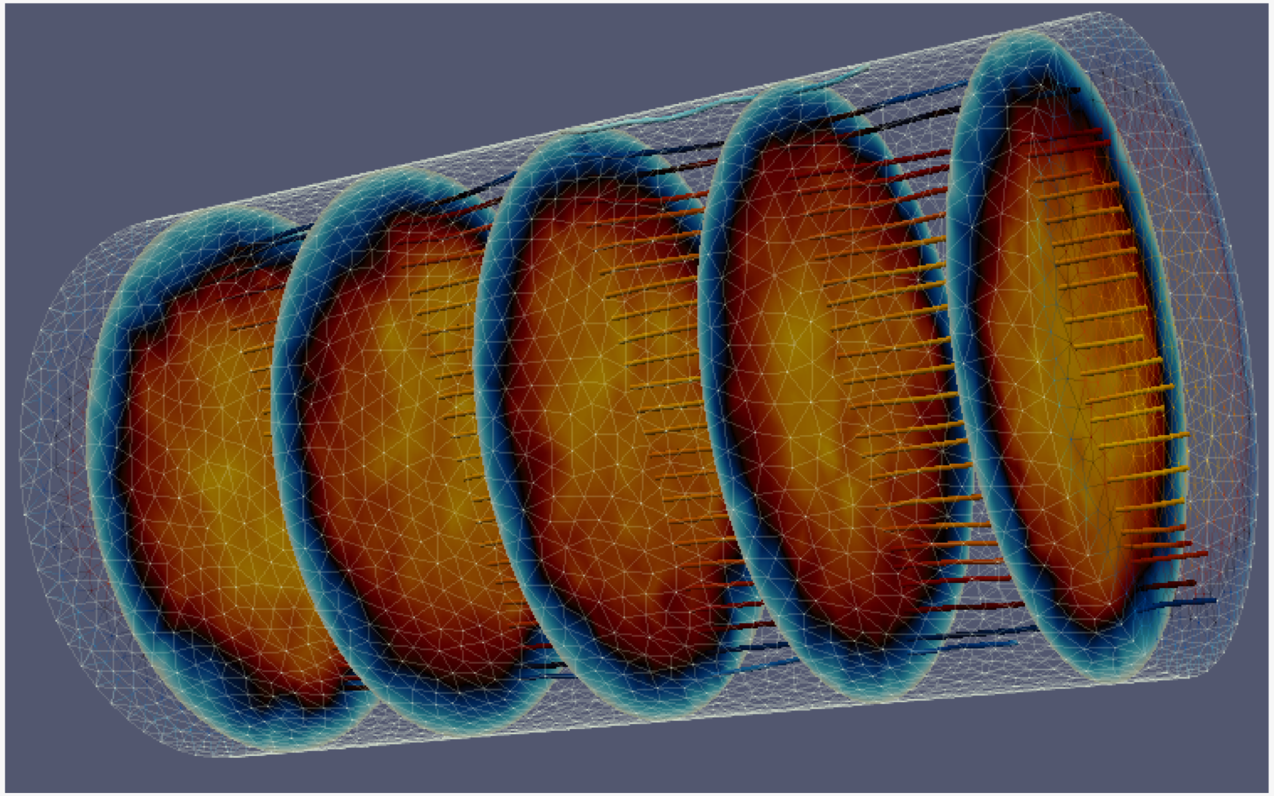
Cutting Planes (Slices)

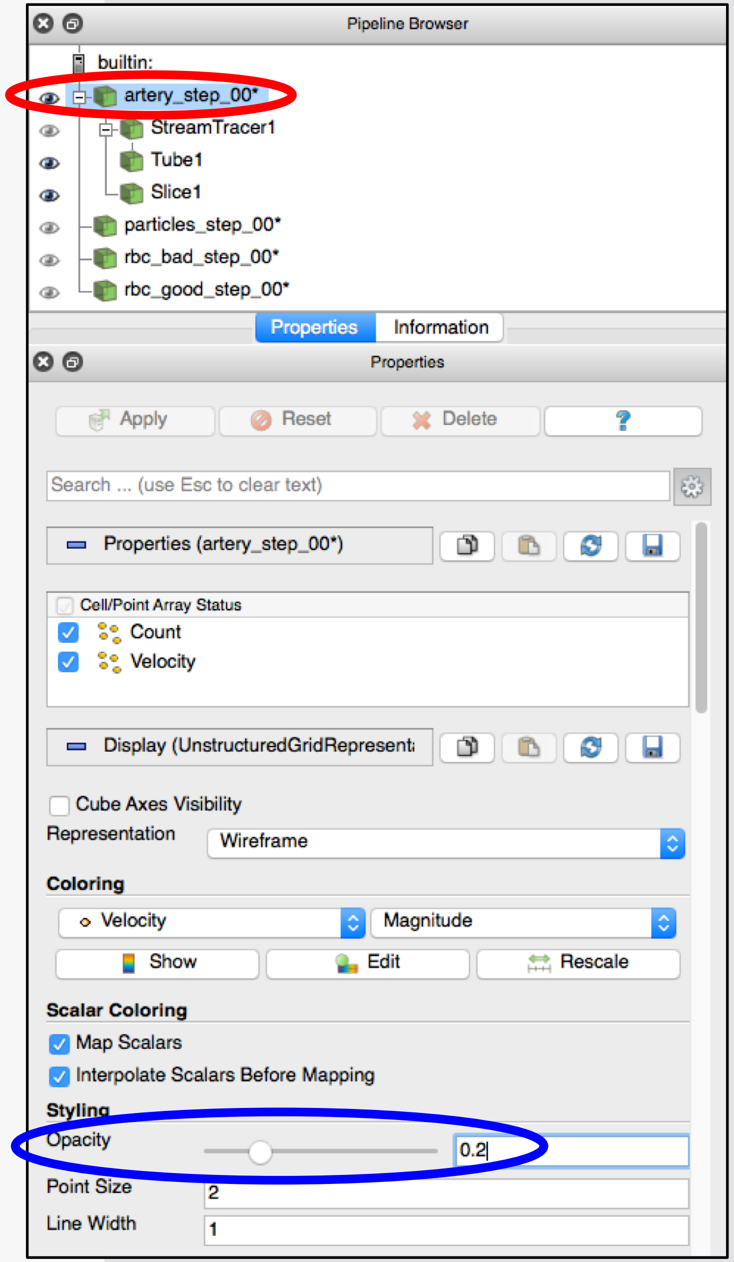
- Color by: Velocity



Data representation: Opacity

- artery_step_000* selected in the Pipeline Browser
- Properties: Styling
- Opacity: 0.2

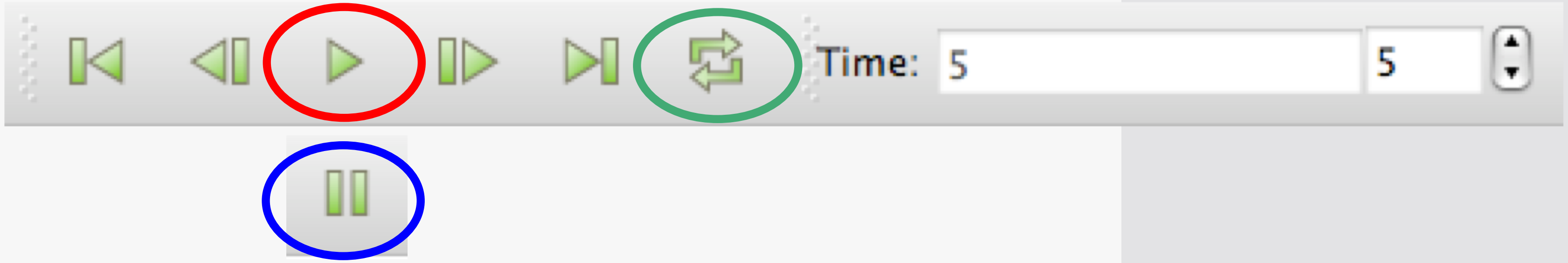




Argonne NATIONAL LABORATORY

Animating Simulation Data

- Play button on the animation bar at the top of the GUI
- Pause to stop
- Loop: Repeat animation until stopped

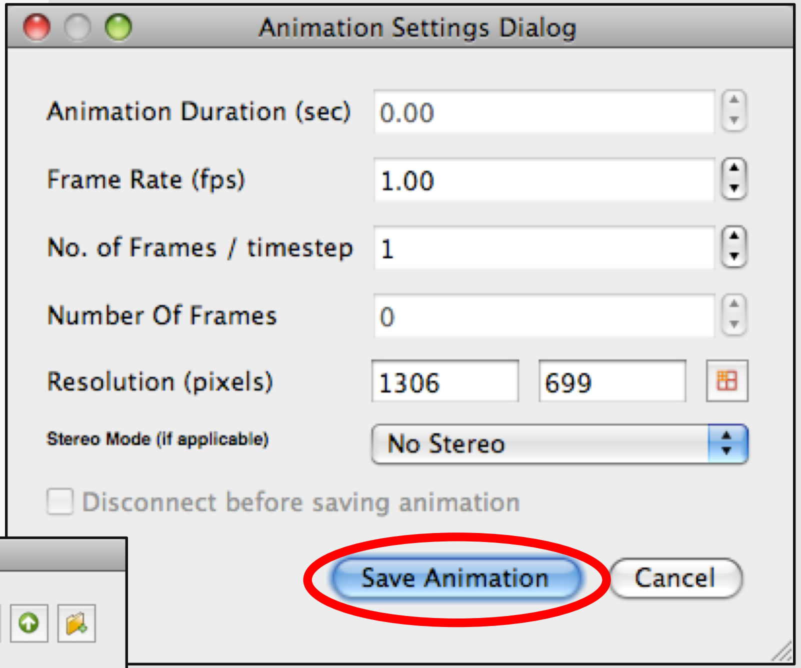


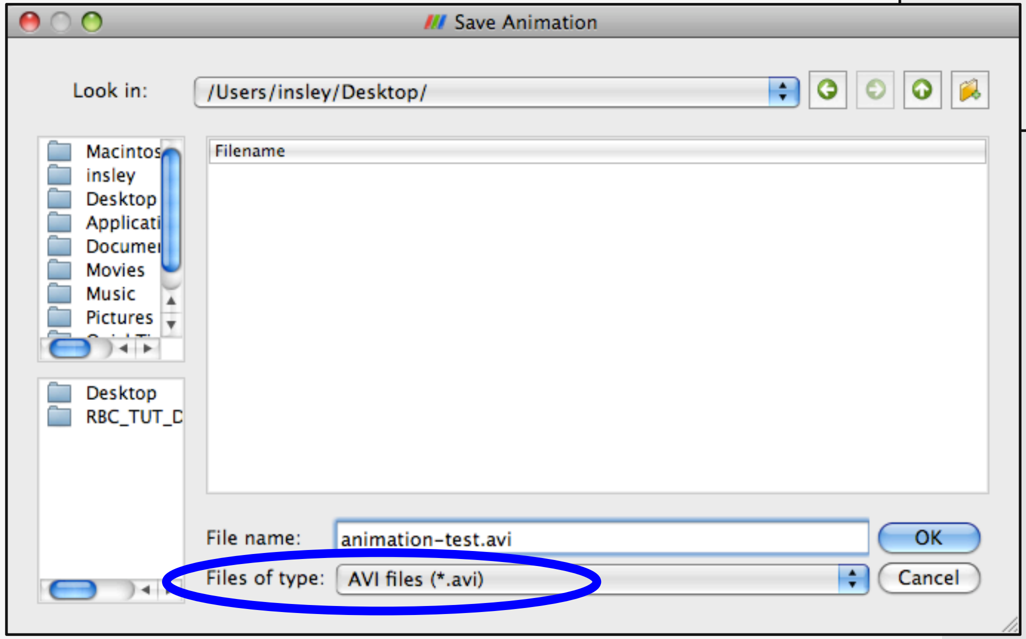
Save Screenshot

1. Choose File → Save Screenshot...
2. Complete the following dialogs.

Save Animation

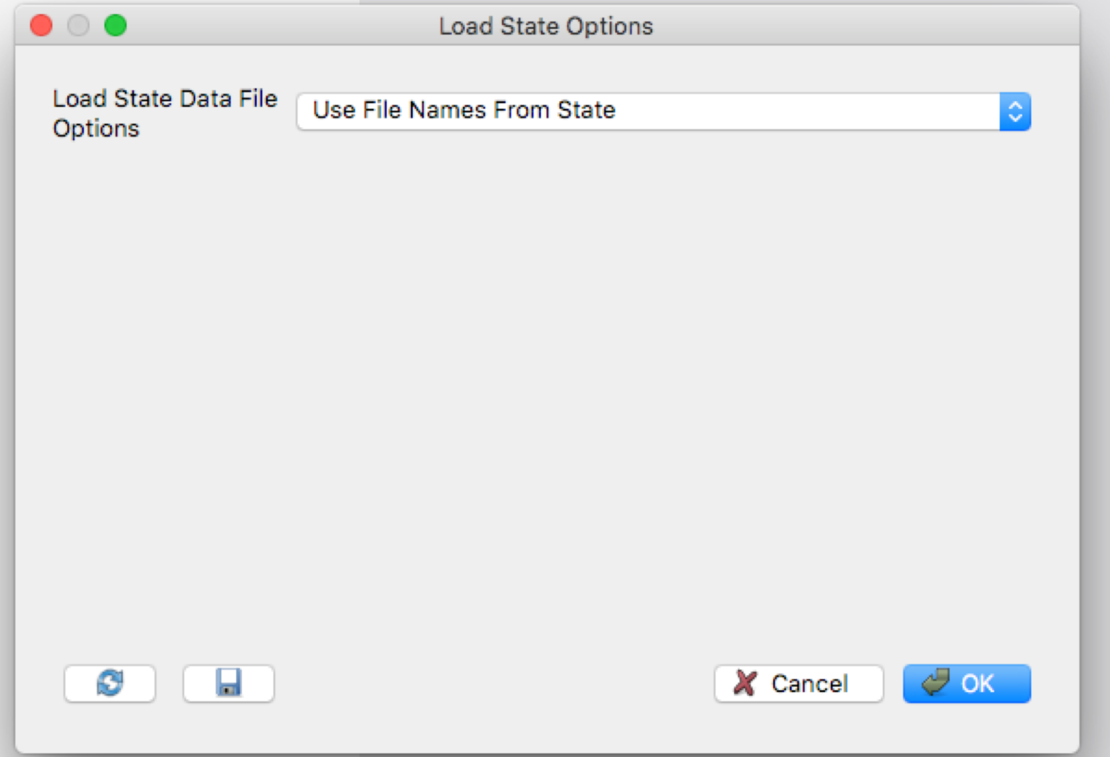
1. File-> Save Animation
2. Animation Settings Dialog: Save Animation
3. Files of type:
   - AVI
   - PNG





Saving ParaView State

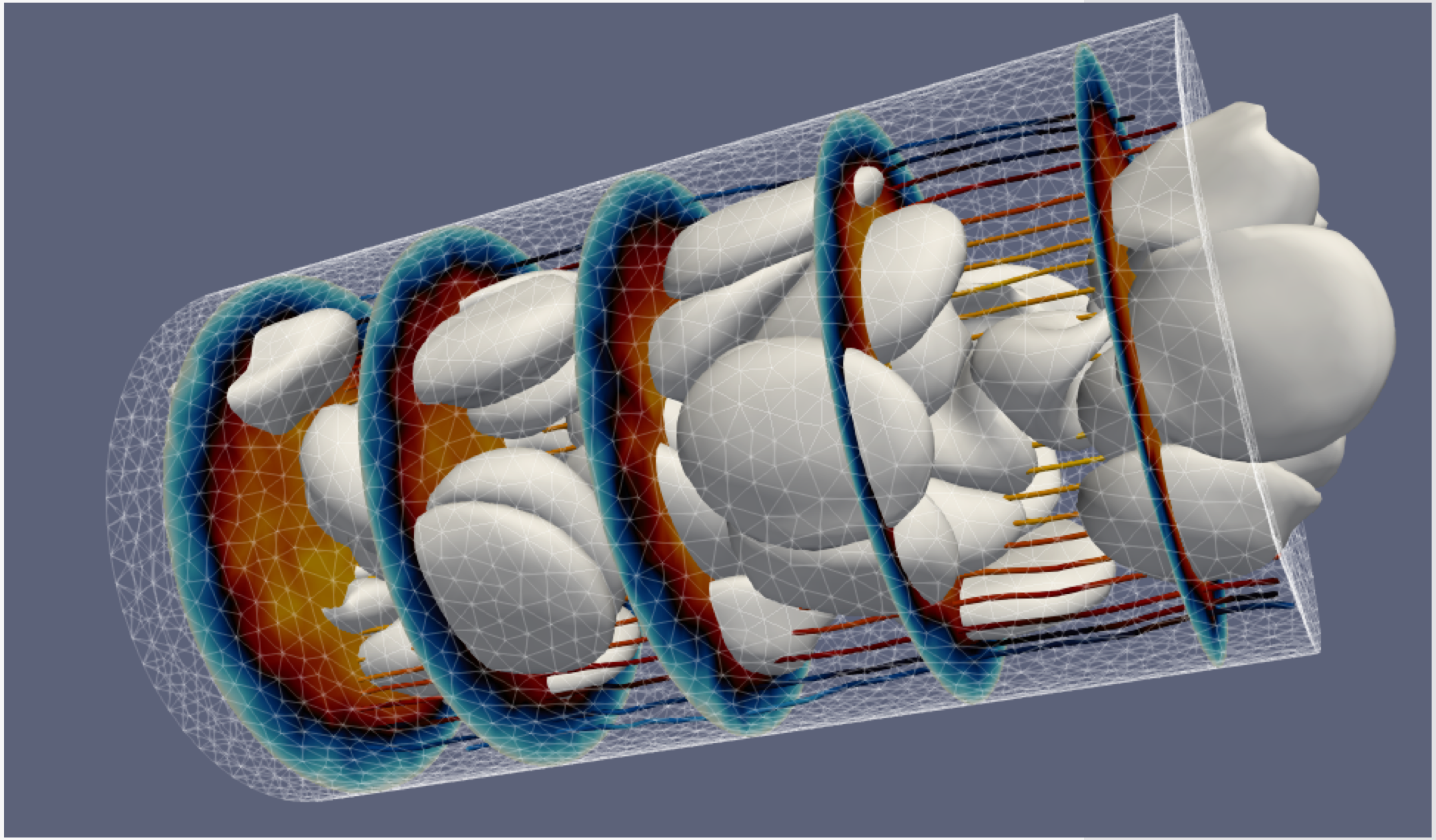
1. Choose File →Save State…
   - .pvsm (for restoring state in interactive mode)
   - saved on the client side
2. Edit → Reset Session 
3. Choose File →Load State…
4. Load State Options
   1. Use File Names From State
   2. Search files under specified directory
   3. Choose File Names



Argonne NATIONAL LABORATORY

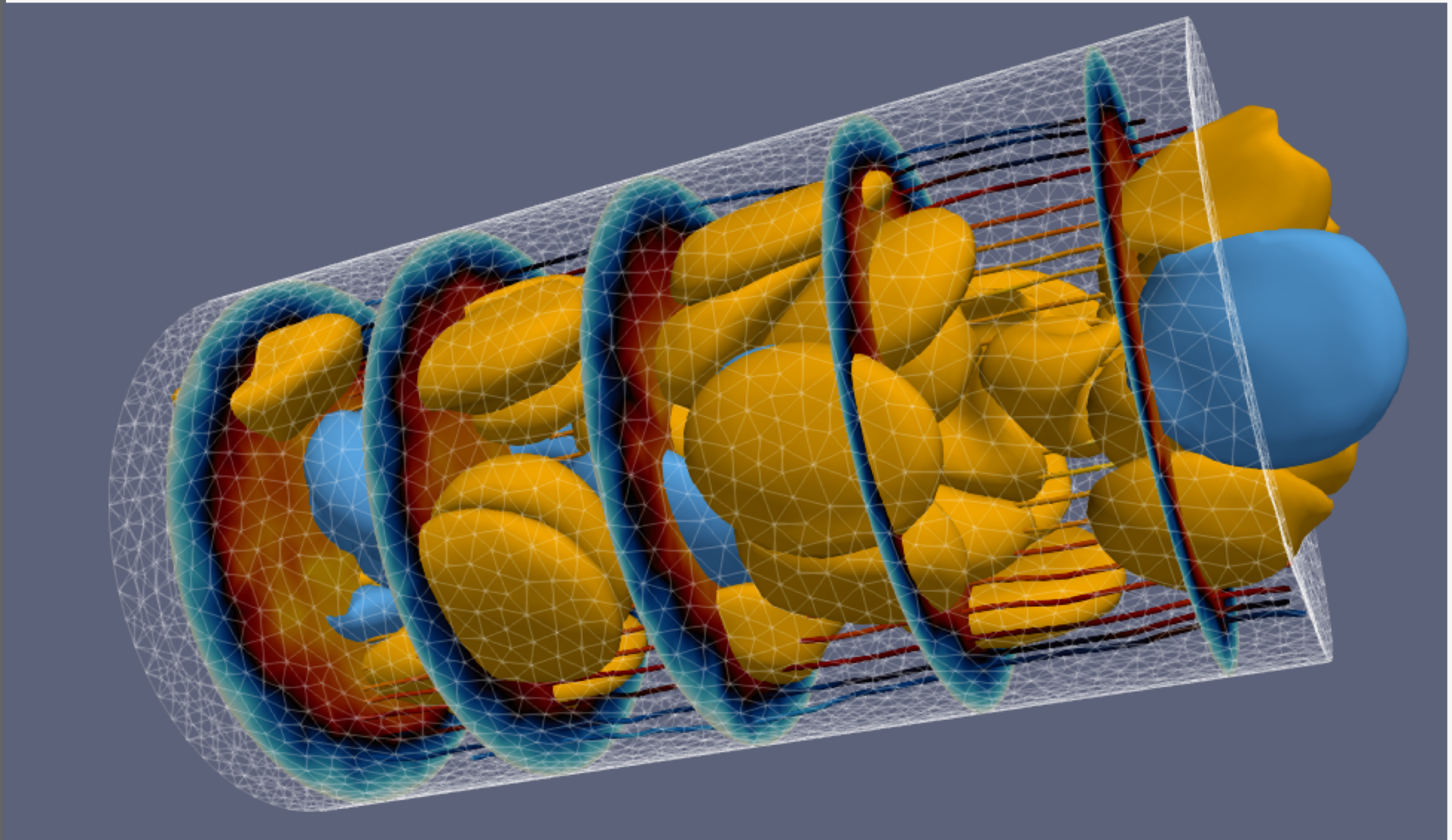
Enter: Red Blood Cells

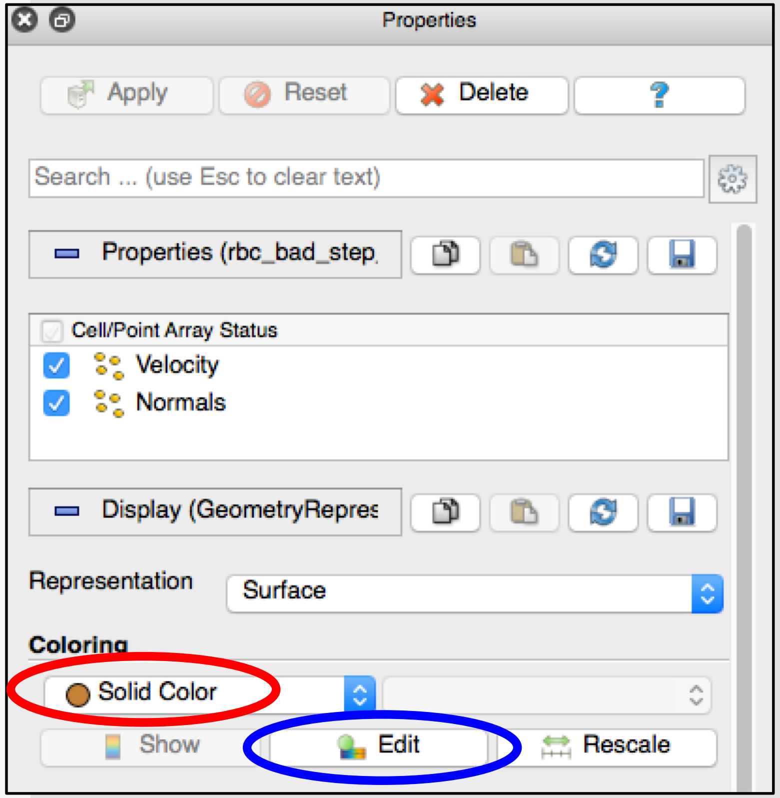
- Unhide good_rbc_step_000* and bad_rbc_step_000*

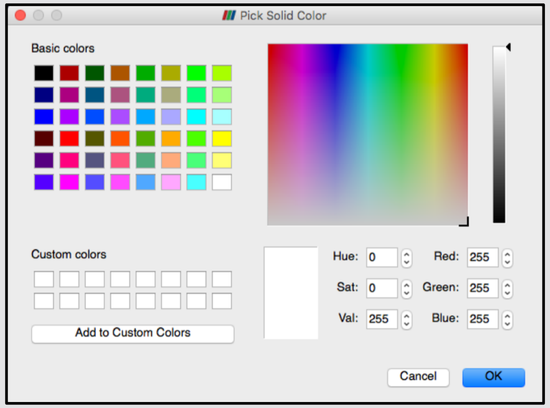


Using color to differentiate data

- Select one of the rbc data sets in the Pipeline Browser
- Coloring: Solid Color
- Edit
- Pick different colors for the two data sets

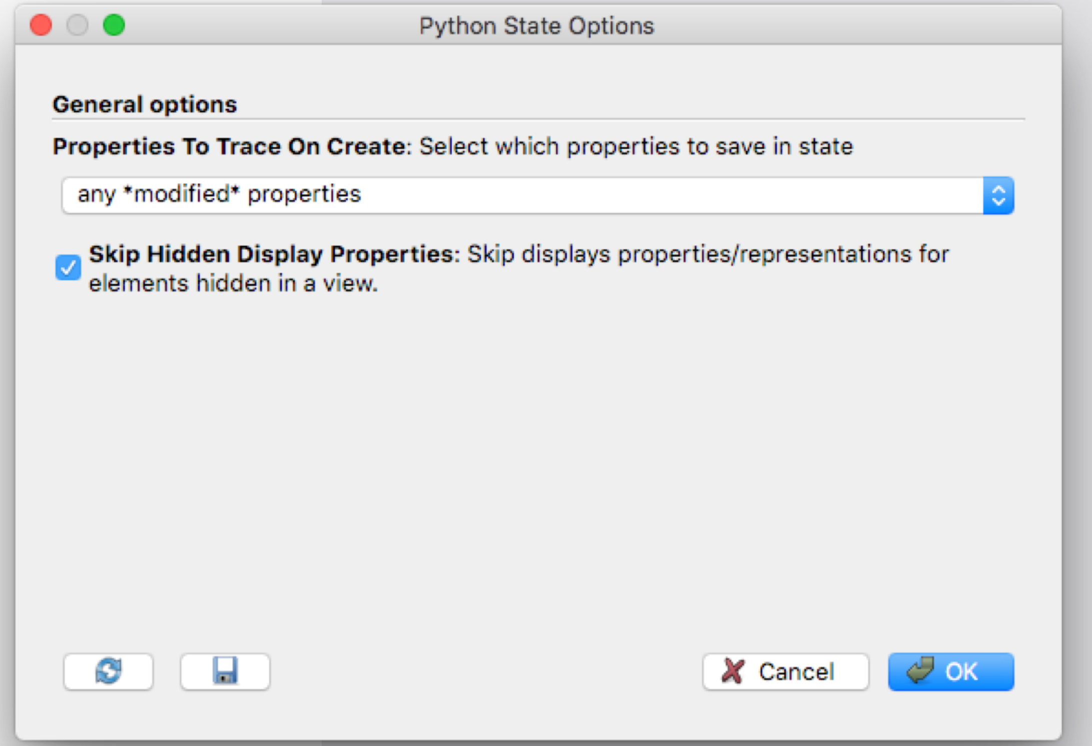






Saving ParaView State and Scripting

1. Choose File →Save State…
   - . py (for use with pvbatch)
     - any *modified* properties
     - all properties
   - saved on the client side
2. Edit .py script
   - short example, loop over time steps, saving images in batch mode



Saving ParaView State and Scripting

At the top of the script:

```
import sys
start_frame=int(sys.argv[1])
num_frames=int(sys.argv[2])
IMAGE_DIR = “/path/to/where/we/should/save/images/
```

At the bottom of the script:

```
RenderView1.ViewSize = [1920, 1080]                   # set resolution appropriately
time_vals = artery_step_0000vtu.TimestepValues       # find data reader object

for i in range(start_frame, start_frame+num_frames):
    RenderView1.ViewTime = time_vals[i]
    RenderView1.StillRender()
    IMAGE_FILE="%s/frame_%04d.png" % (IMAGE_DIR, i)
    print ("saving: " + IMAGE_FILE)
    WriteImage(IMAGE_FILE)
```

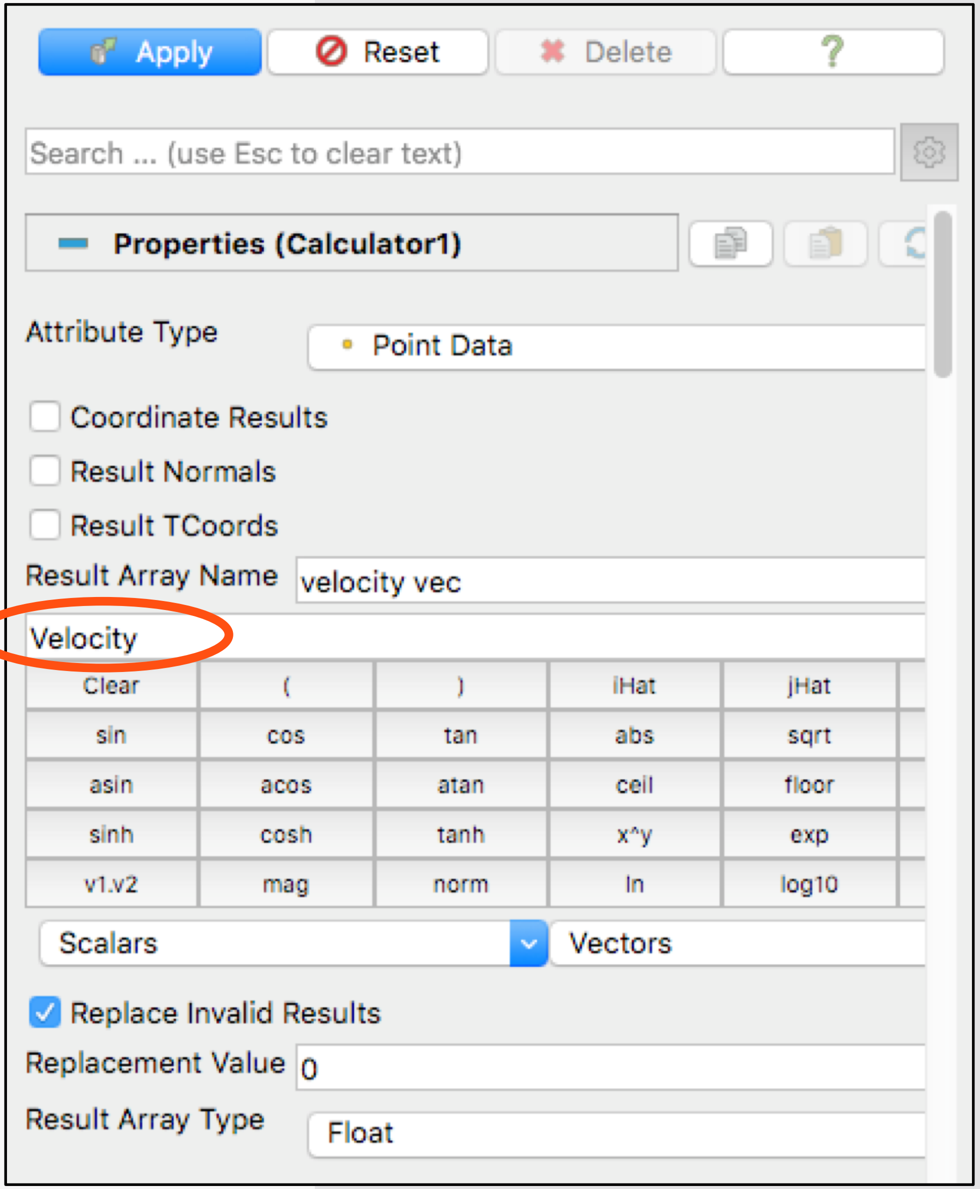
Saving ParaView State and Scripting

- Copy python script to Cooley
- Submit batch job to render animation
  - qsub –n 1 –t 60 –A Comp_Perf_Workshop -q training2 /soft/visualization/paraview/v5.8.0/bin/pvbatch python_script.py 0 50
  - qsub –n 1 –t 60 –A Comp_Perf_Workshop -q training2 /soft/visualization/paraview/v5.8.0/bin/pvbatch python_script.py 50 50

- Encode frames using ffmpeg
  - soft add +ffmpeg
  - ffmpeg -r 25 -i /path/frame_%04d.png -r 25 -pix_fmt yuv420p movie_01.mp4

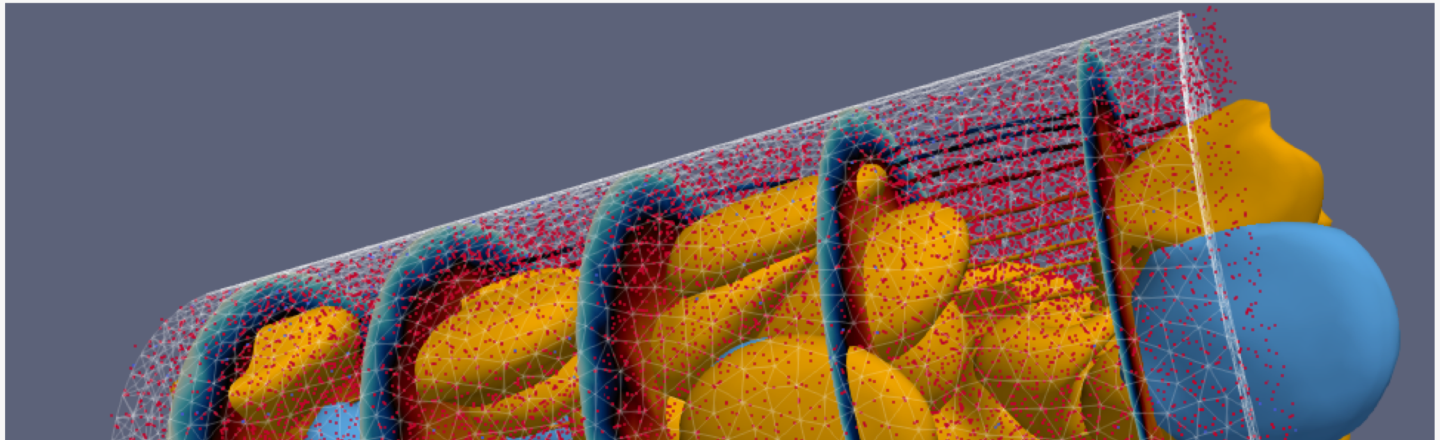
Particles as Glyphs

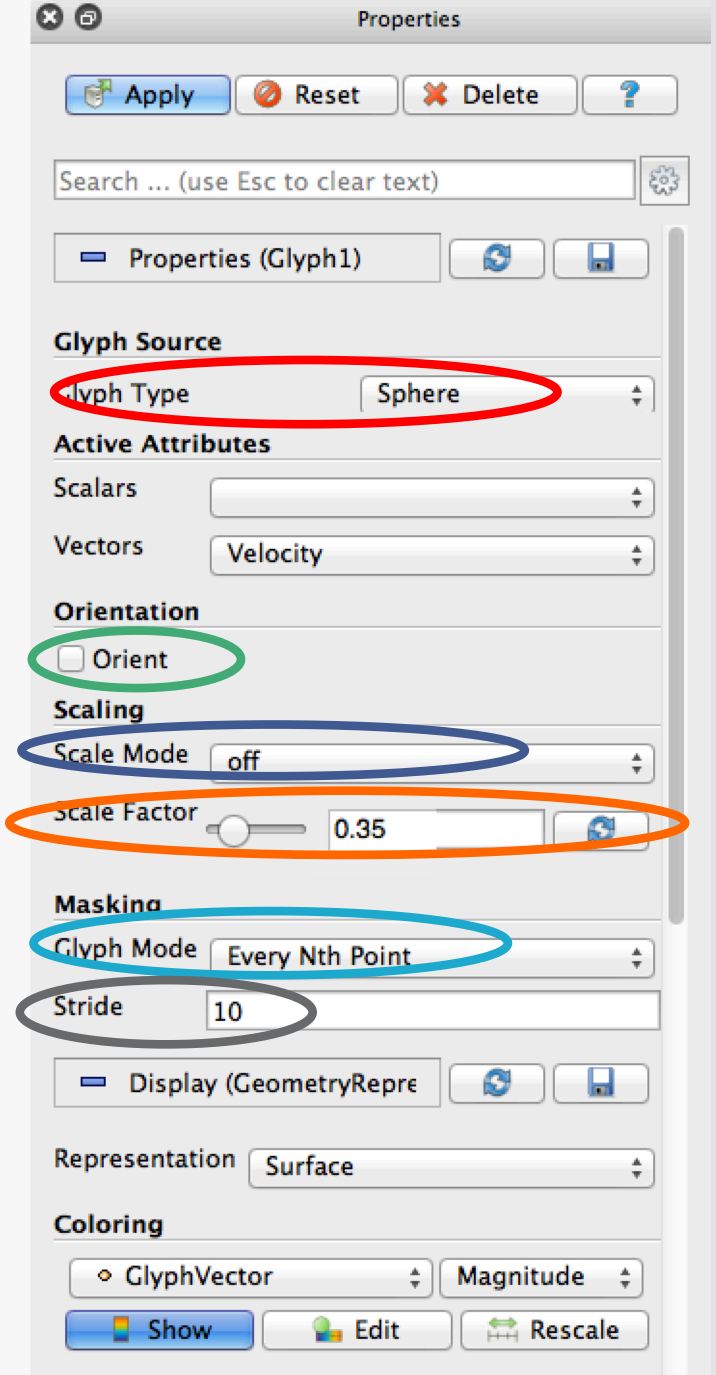
- First need to add a Calculator Filter 
- (this seems to be a bug with glyphs and vector data arrays)
- Select particles.000*
- Apply Calculator filter
- Set equation to Velocity



Particles as Glyphs

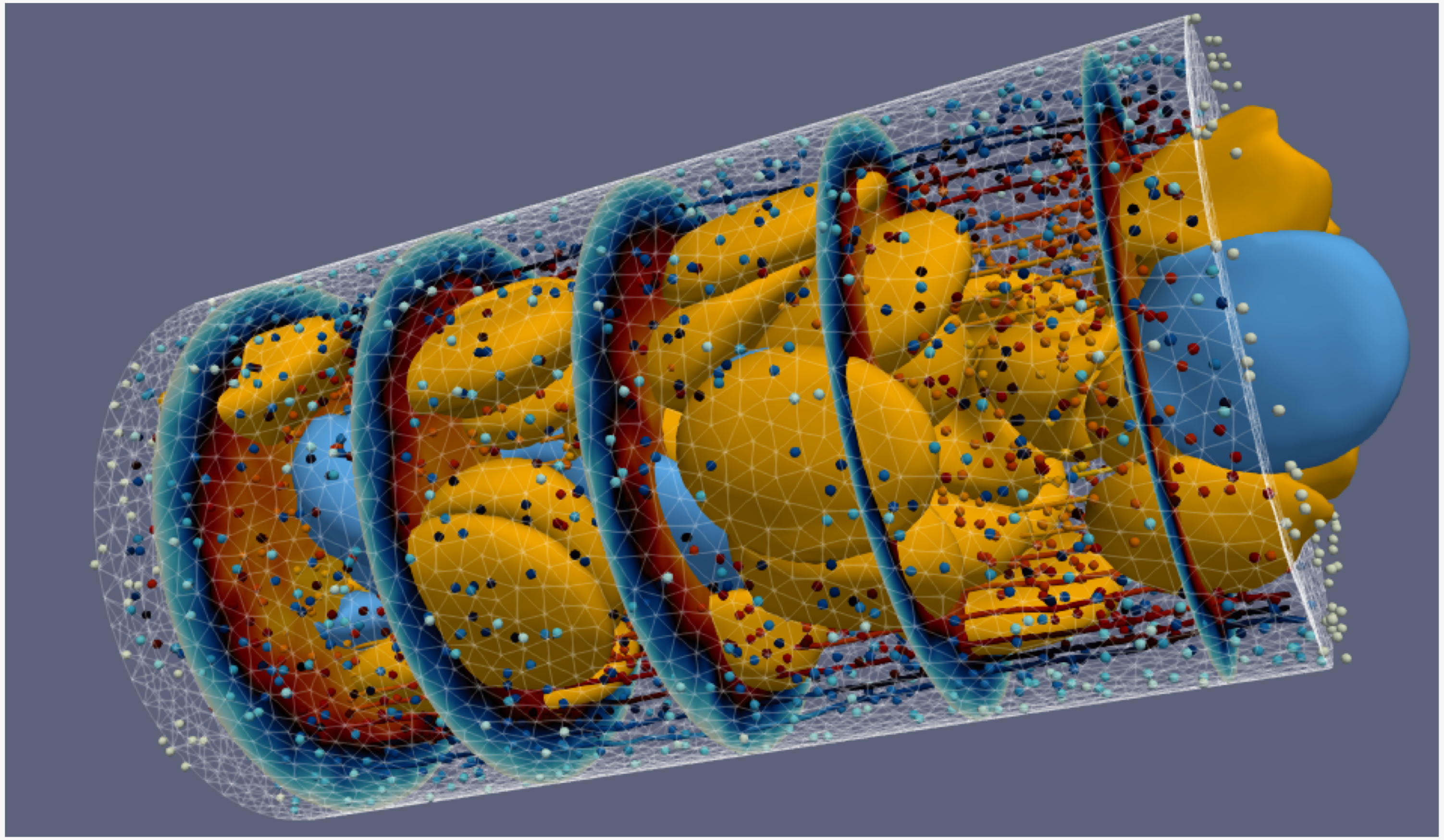
- With Calculator1 selected in Pipeline Browser
- Filters->Alphabetical->Glyph
- Glyph Type: Sphere
- Orient: Unchecked (optional)
- Scale Mode: off
- Set Scale Factor: 0.35
- Glyph Mode: Every Nth Point
- Stride: 10

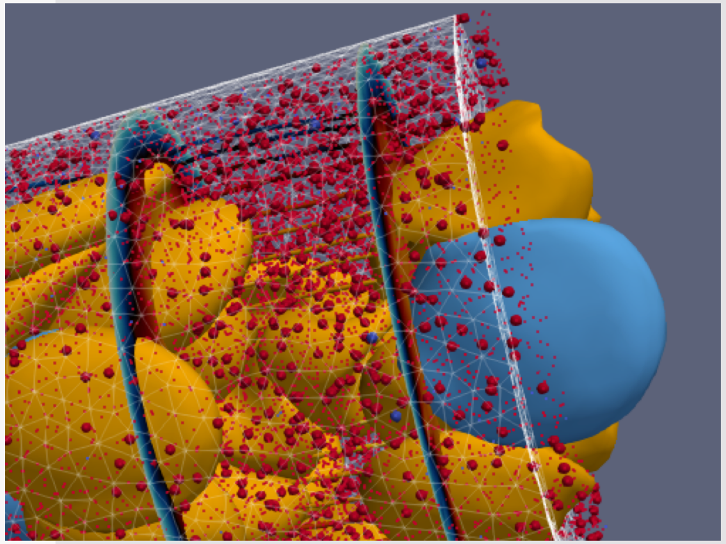




Finalizing Glyphs

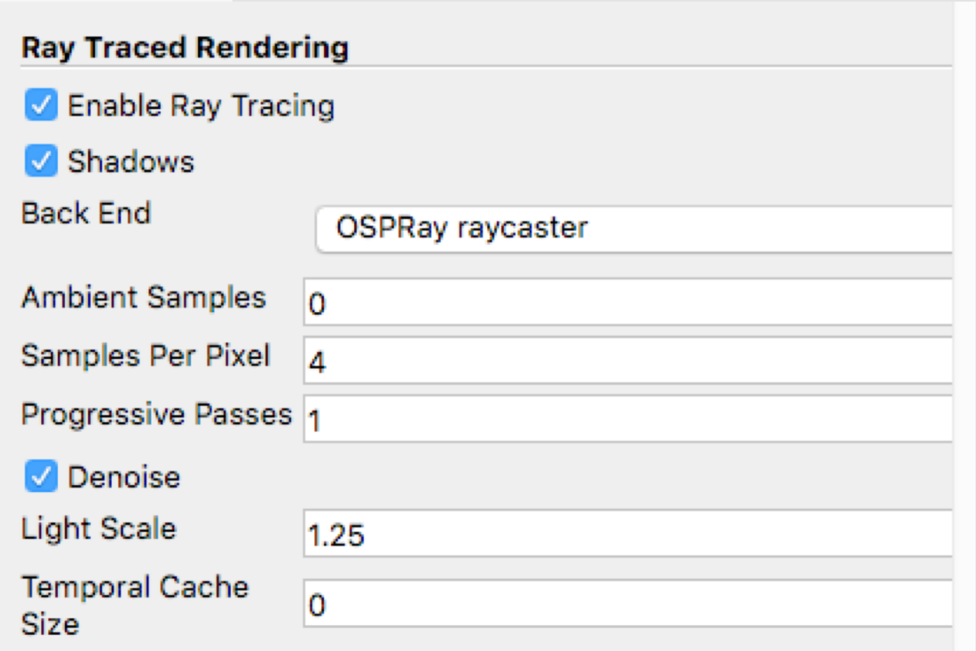
- Hide the Calculator1: Eye icon in Pipeline Browser
- Color Glyph1 by: Velocity

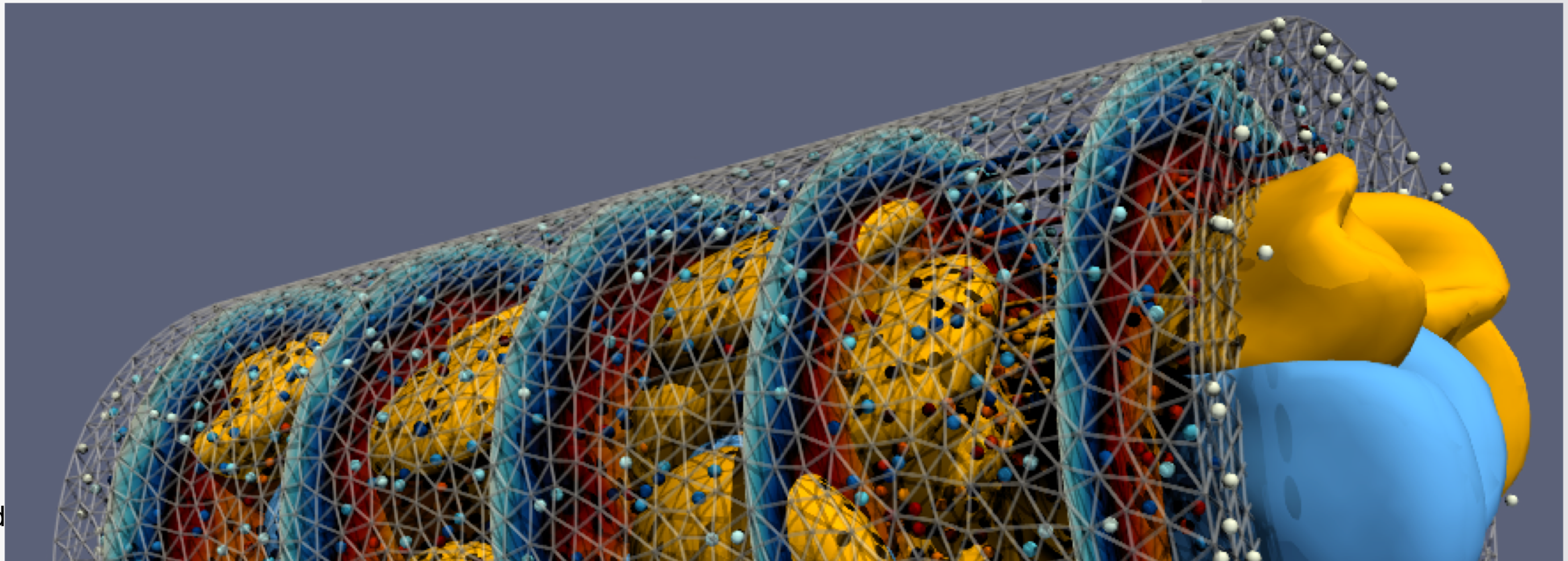




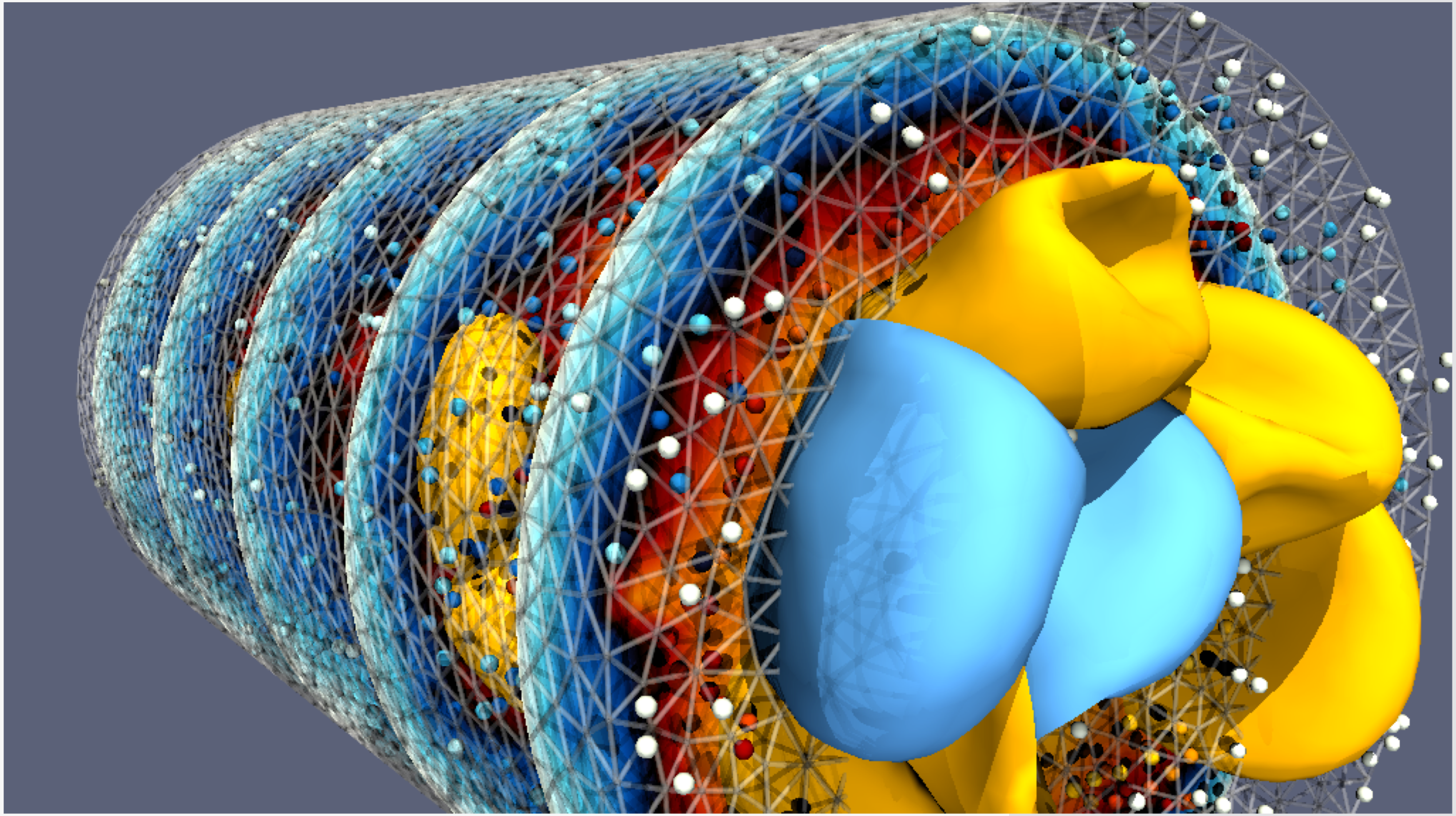
Making it Look Pretty

- Ray Traced Rendering
  - Enable Ray Tracing
  - Shadows
  - Samples Per Pixel
  - Denoise
  - Light Scale





Making it Look Pretty



QUESTIONS?

Joe Insley
insley@anl.gov

Silvio Rizzi
srizzi@anl.gov

Janet Knowles
jknowles@alcf.anl.gov