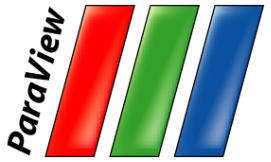


Large Scale Visualization with ParaView

ATPESC 2019

Dave DeMarle
Kitware, Inc.

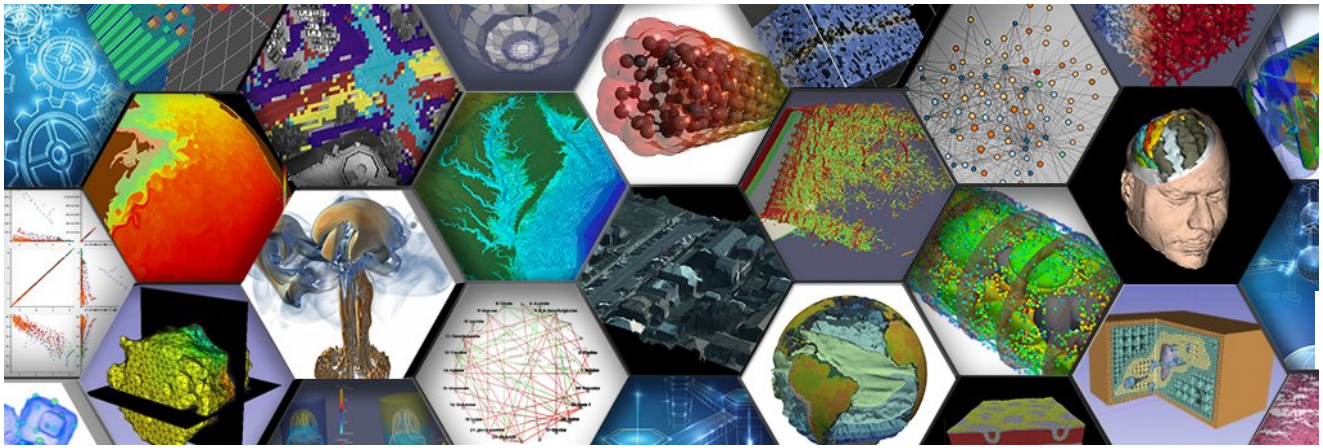
Kitware

Outline

- Kitware
- Introduction
- Basic Usage
- Visualizing Large Models

Kitware

- Computer Vision
- Data and Analytics
- HPC and Visualization
- Medical Computing
- Software Process

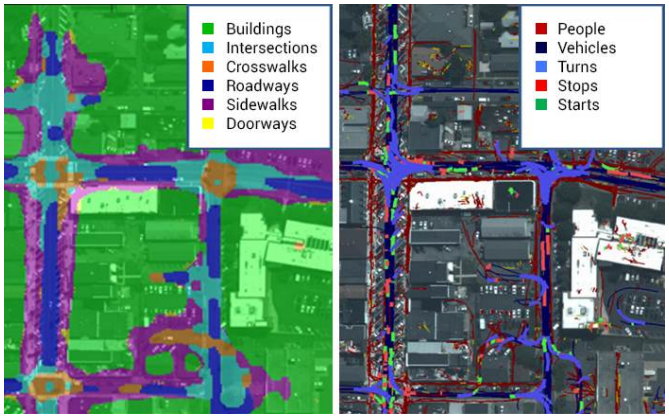


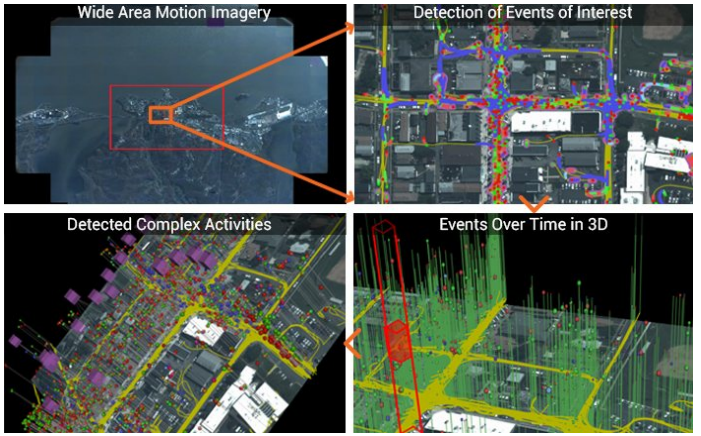
Kitware

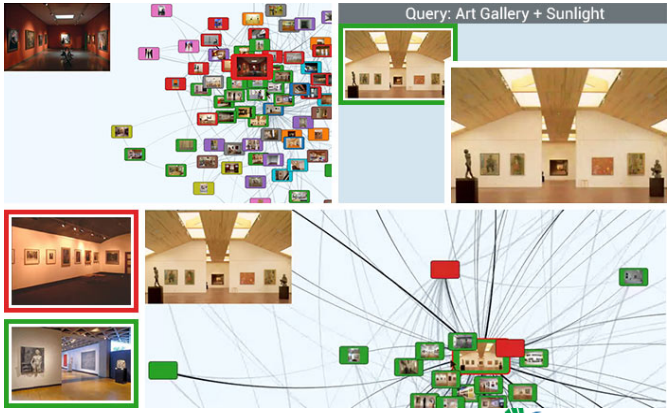
is (always) hiring

Kitware – Computer Vision

KWIVER



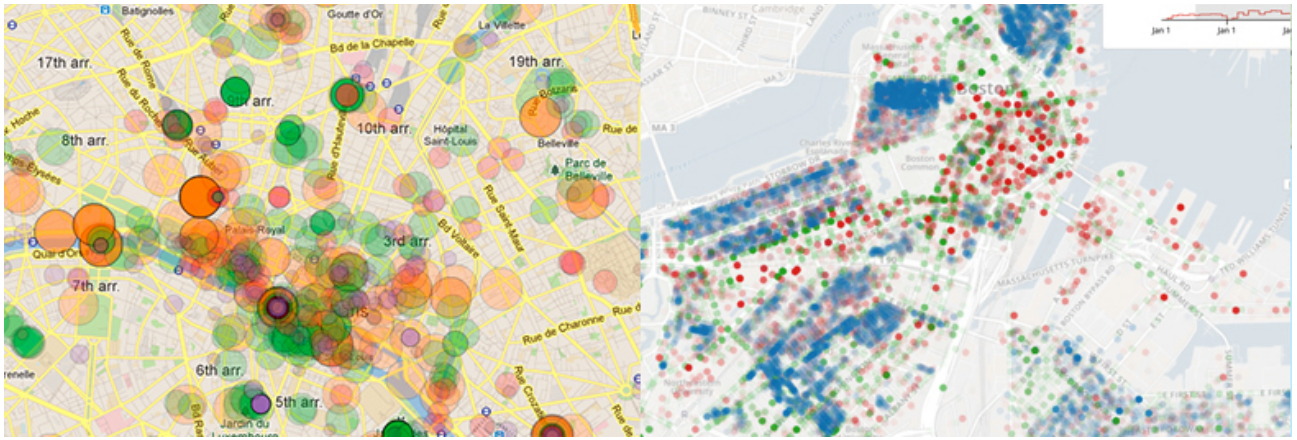


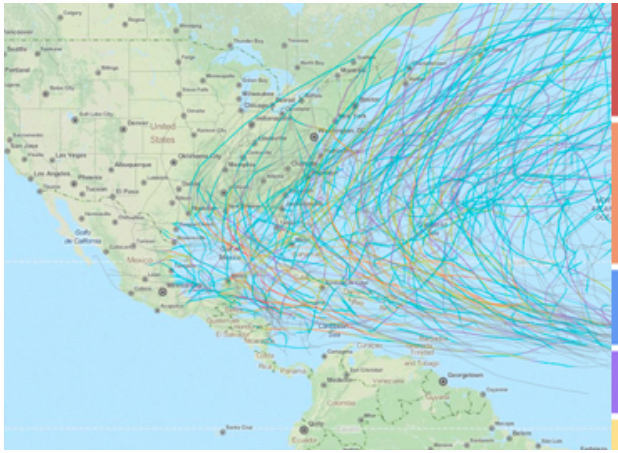


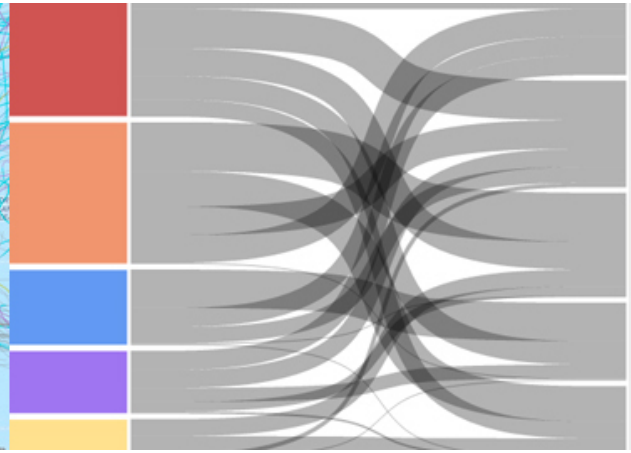
Kitware

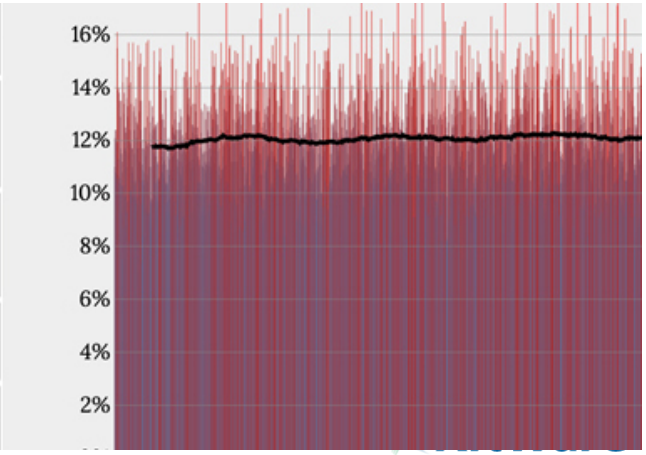
Kitware – Data and Analytics

Resonant



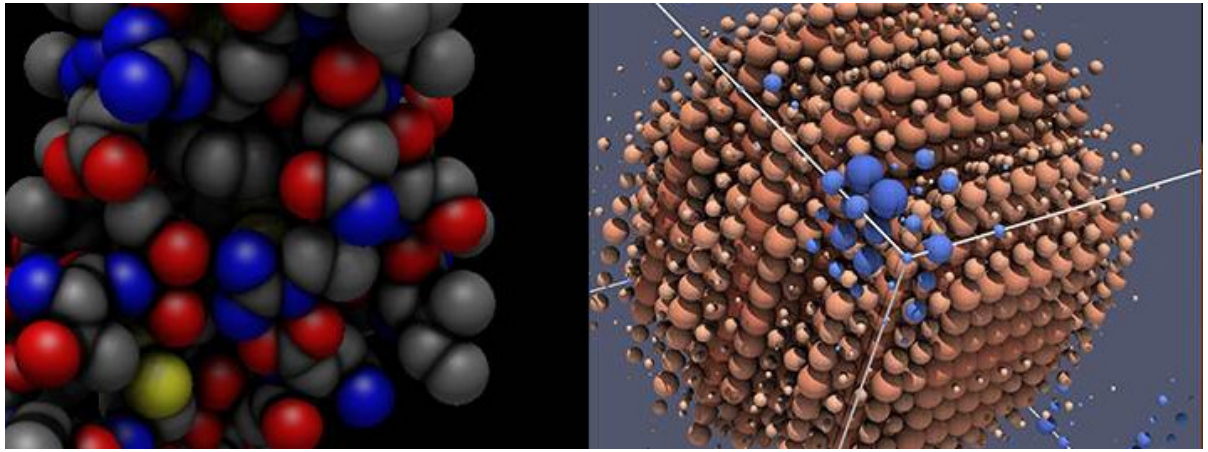


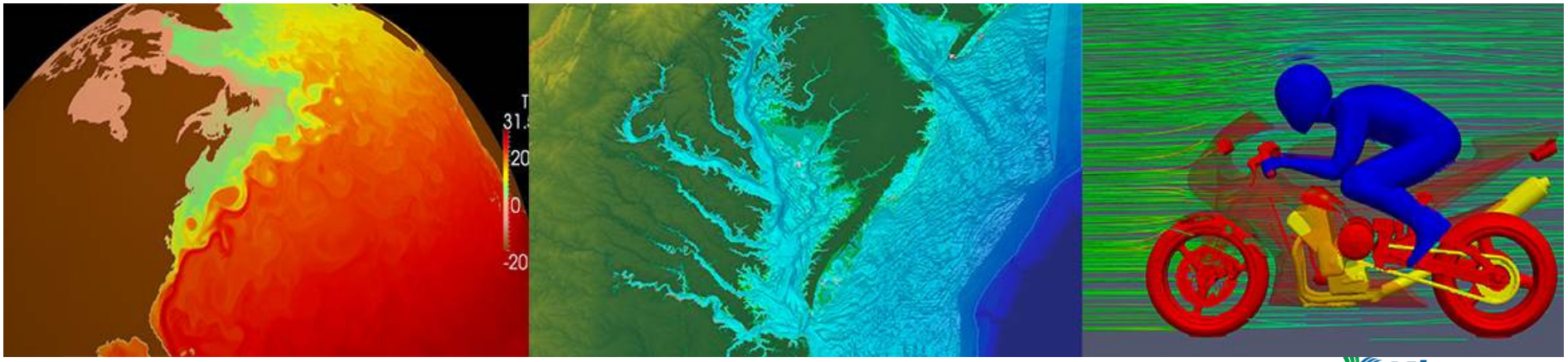




Kitware – HPC and Visualization

VTK ParaView CMB tomviz

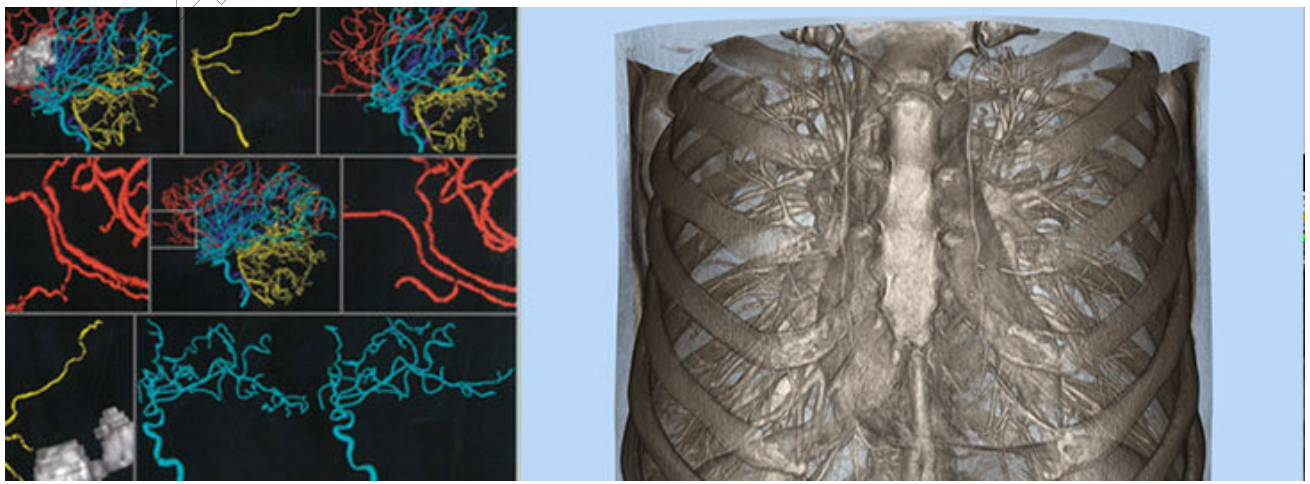


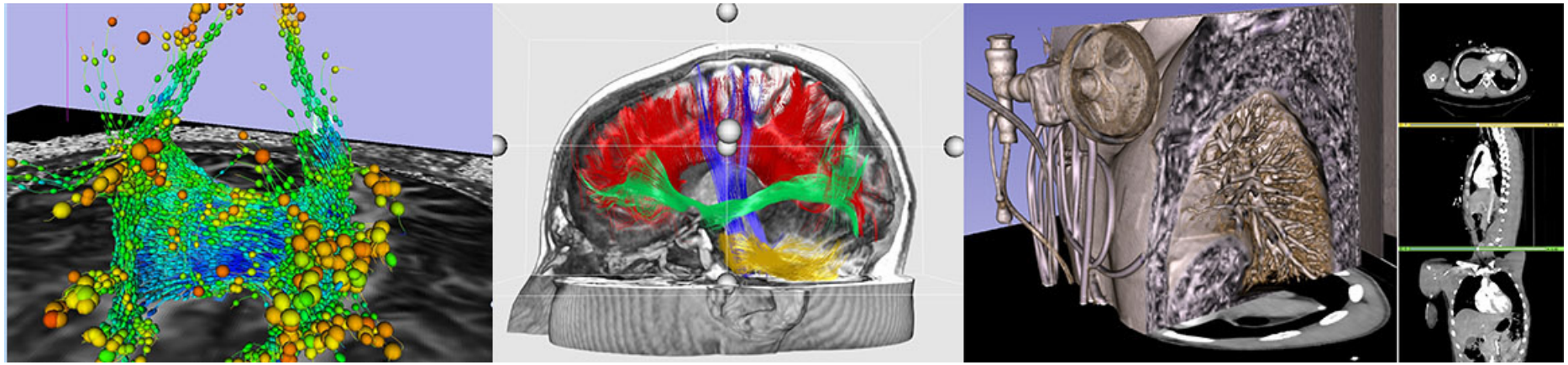


Kitware

Kitware – Medical Computing

ITK 3DSlicer Pulse Physiology Engine ParaView





Kitware – Software Process

- cmake, ctest, cdash, buildbot

Kitware

is (always) hiring

To Follow Along…

- Install ParaView 5.6.1
  - http://www.paraview.org → Download

Introduction

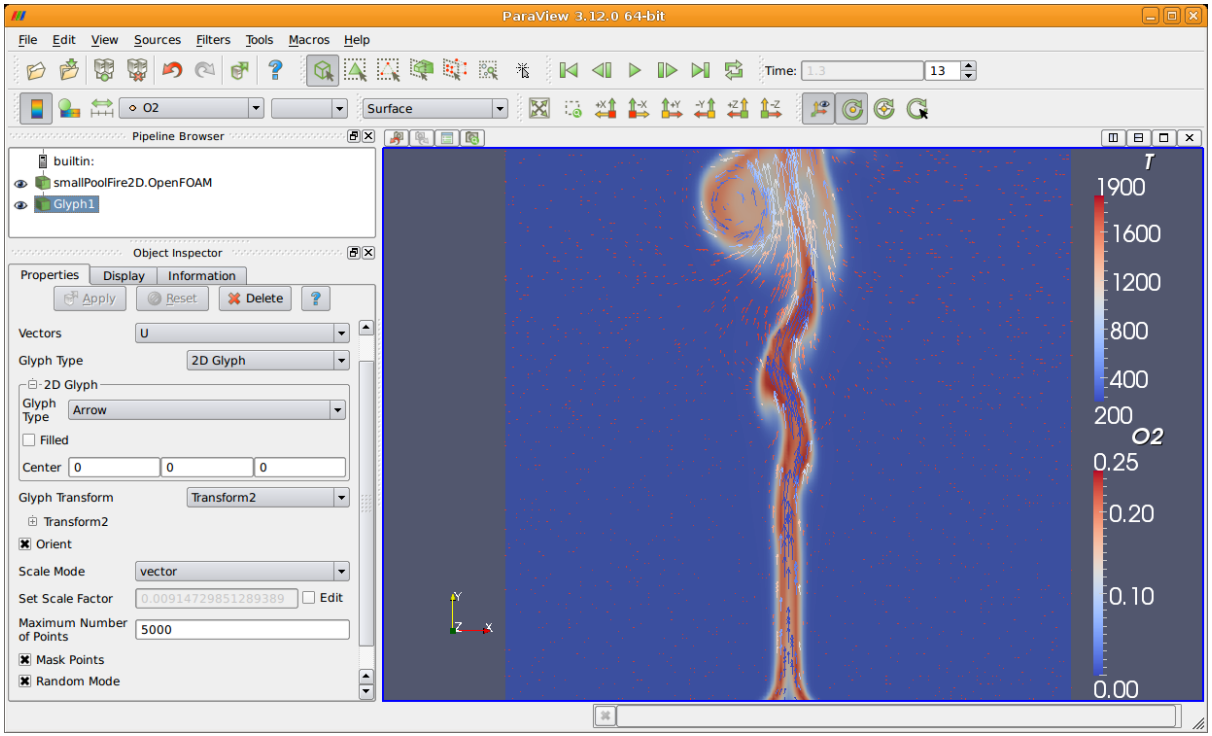
What is ParaView?

- An open-source, scalable, multi-platform visualization application.

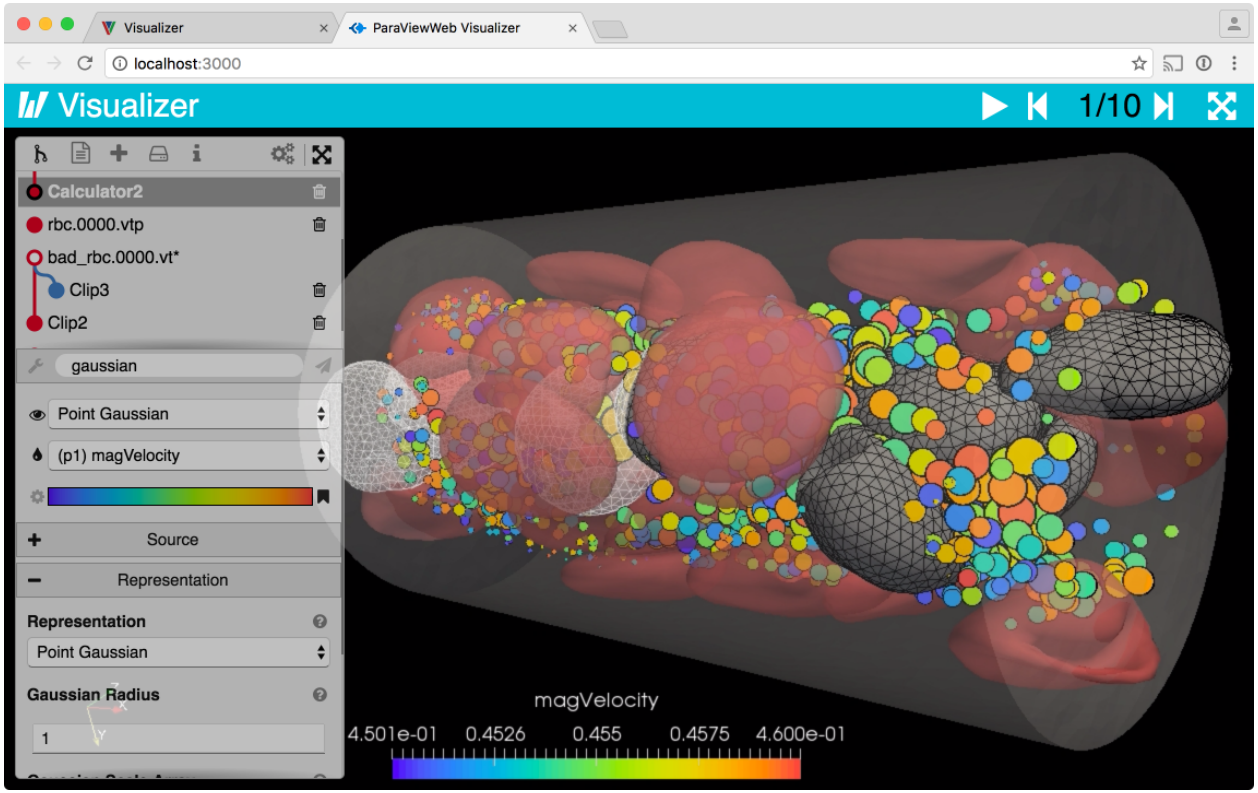
- Support for distributed computation models to process large data sets.
- An open, flexible, and intuitive user interface.
- An extensible, modular architecture based on open standards.
- A flexible BSD 3 Clause license

- Commercial maintenance and support.

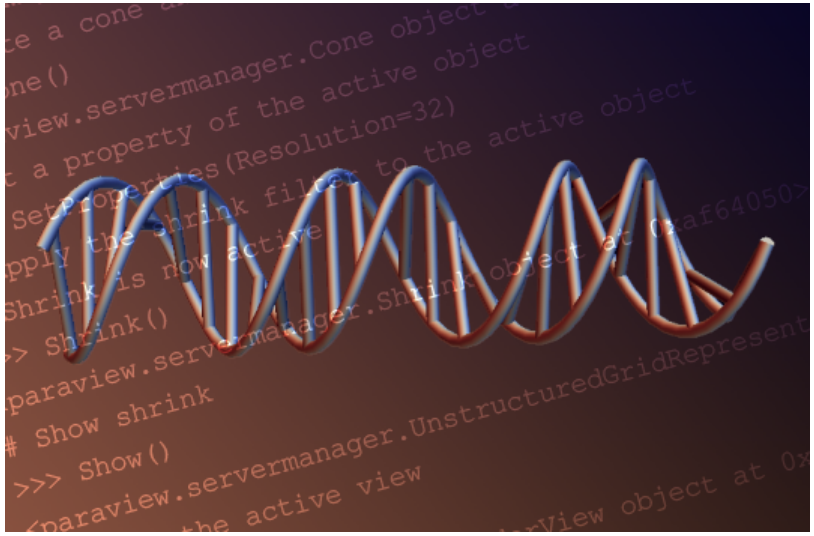
ParaView on the Desktop



ParaView on the Web

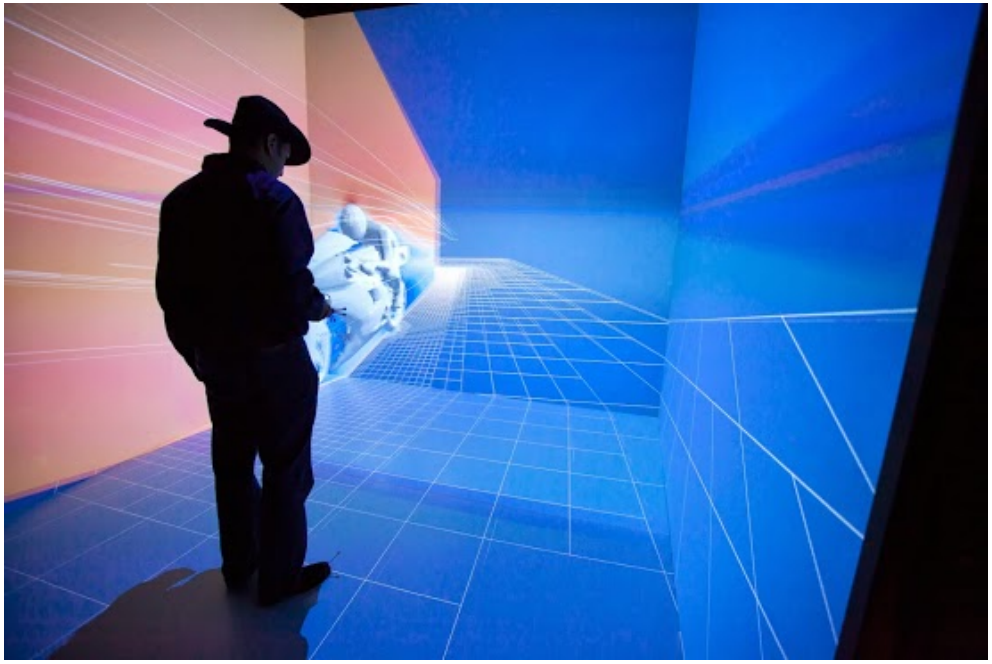


ParaView Scripting - Python



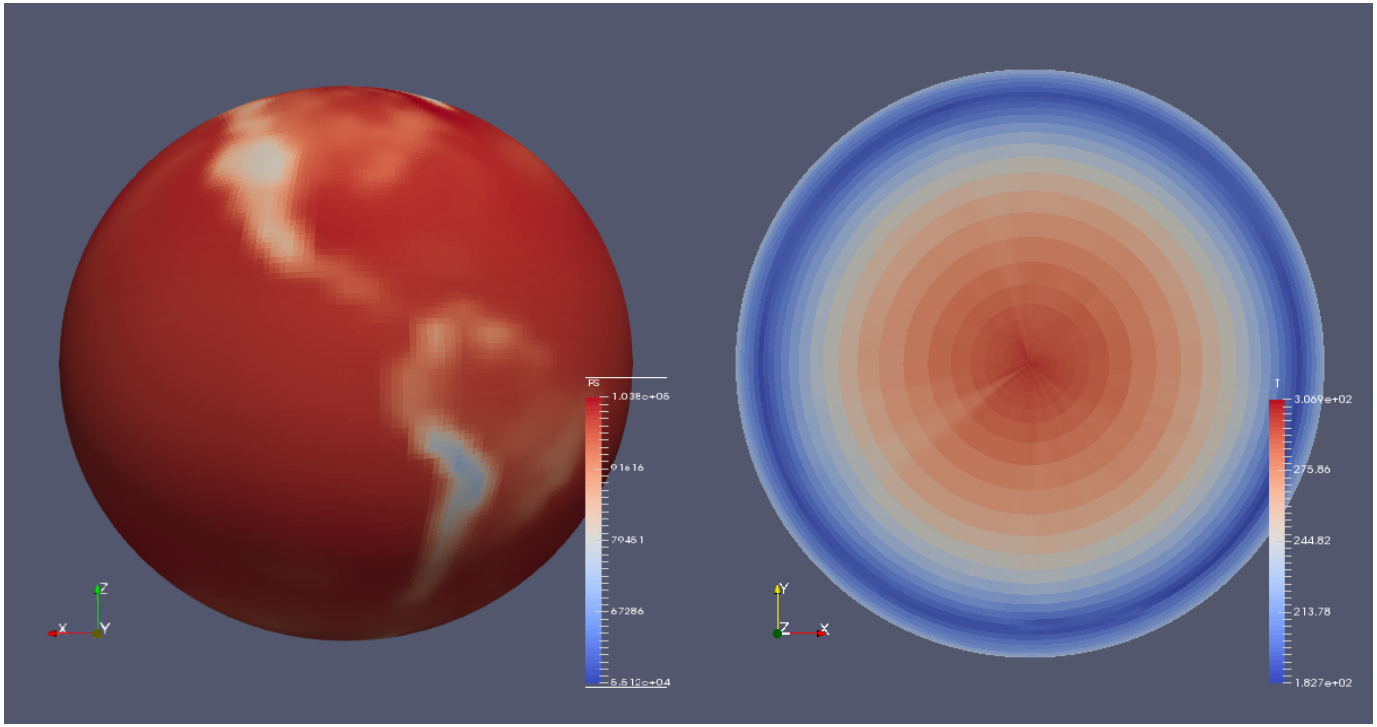
Python scripts can control ParaView with or without the GUI in order to create reproducible and customizable visualizations.

ParaView Immersive



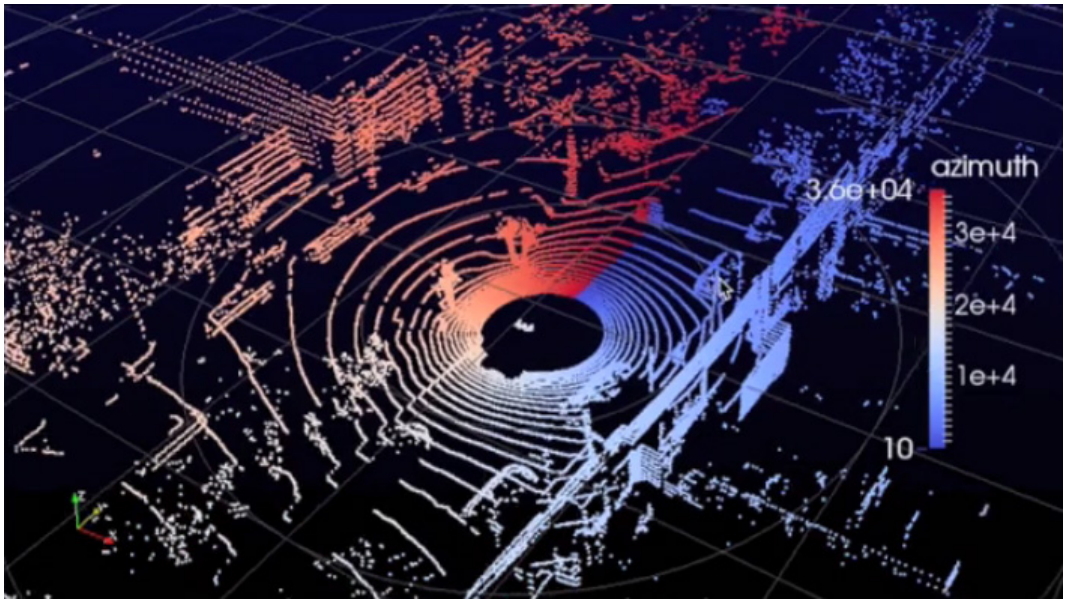
ParaView for HPC

ParaView Catalyst



Community Atmosphere Model (CAM5) 2D (PS) 3D data (T), Spectral Element dynamic module.

ParaView Custom Application VeloView



Visualization of 3D LIDAR data.

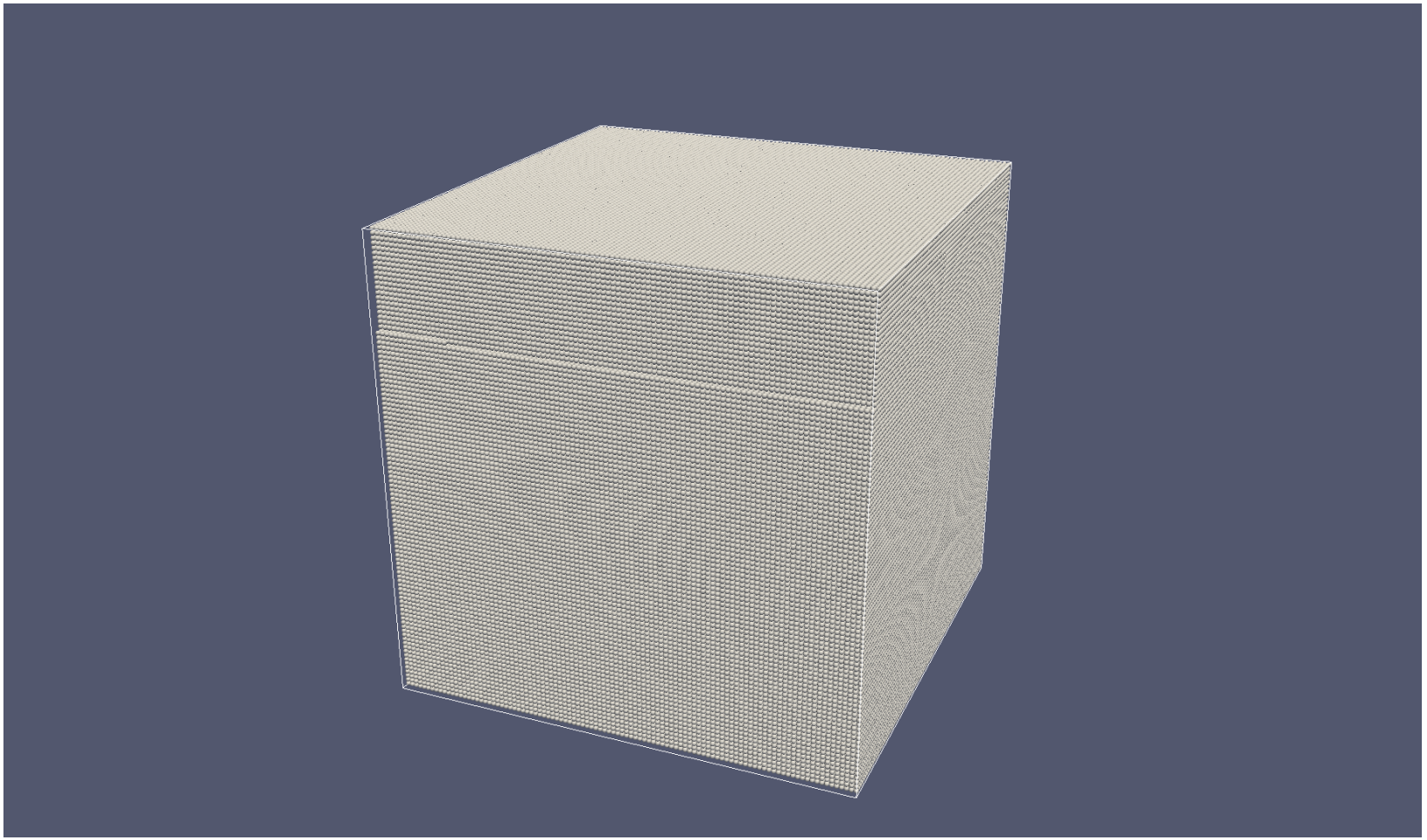
Current ParaView Usage

- Used by academic, government, and commercial institutions worldwide.
- Downloaded ~135K times per year.
- HPCwire Editors' Choice 2010/2016 and HPCwire Readers' Choice 2010/2012/2015 Awards for Best Visualization Product or Technology.

Data Ranges

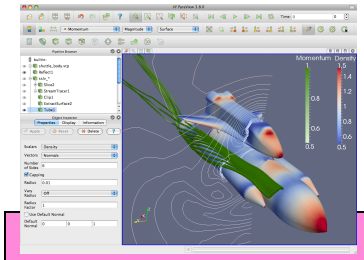
- Used for all ranges of data size.
- Landmarks of usage:
  - 6 billion structured cells (2005).
  - 250 million unstructured cells (2005).
  - Billions of AMR cells (2008).
  - Scaling test over 1 Trillion cells (2010).
  - 6.33 billion unstructured cells in Catalyst (2016).
  - 1.1 trillion unstructured cells scaling test (2016).

Trinity Run each sphere = 1E6 Tris



LA_UR_16_23941

ParaView Application Architecture



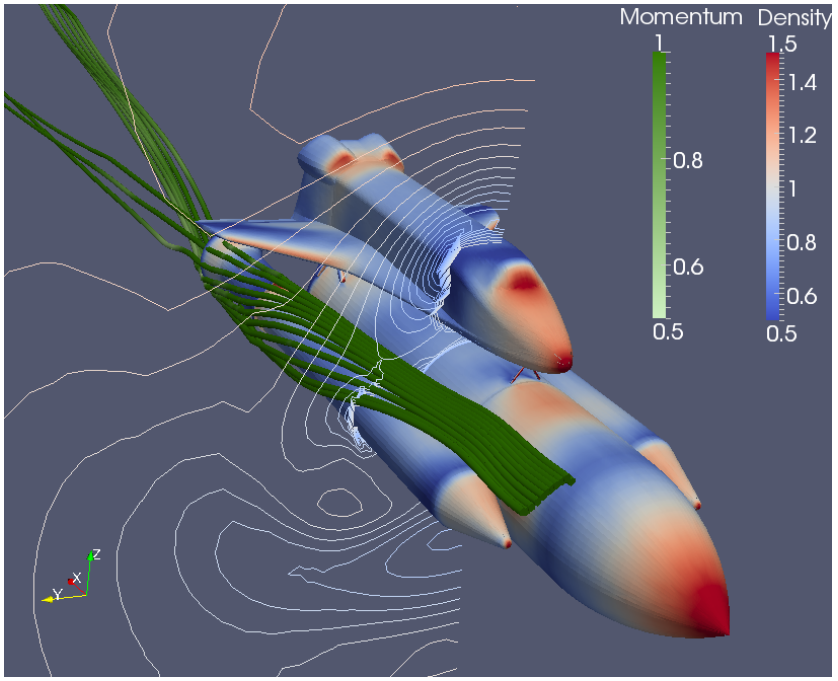
ParaView Client	pvpython	ParaWeb	Catalyst	Custom App
UI (Qt Widgets, Python Wrappings)				
ParaView Server				
VTK				
OpenGL	MPI	IceT	Etc.	

ParaView Development

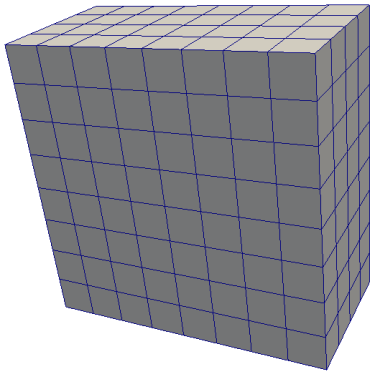
- Started in 2000 as collaborative effort between Los Alamos National Laboratories and Kitware Inc. Sandia has been a major contributor since 2005.
  - ParaView 0.6 released October 2002.
- Paraview 3.0 release in May 2007.
  - GUI rewritten to be more user friendly and powerful.
- ParaView 4.0 released in June 2013.
  - Properties panel redesign for smoother interaction.
- ParaView 5.0 released in January 2016.
  - Updated to OpenGL 3.2 features. Huge performance improvements.

Basics of Visualization

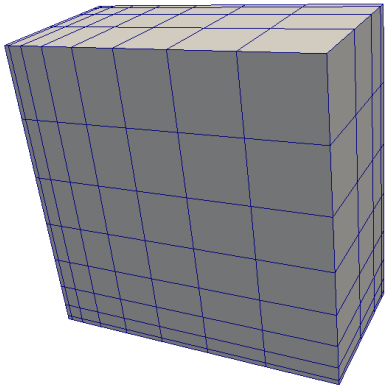
```
0265640 132304 133732 032051 037334 024721 015013 052226 001662
0265660 025537 064663 054606 043244 074076 124153 135216 126614
0265700 144210 056426 044700 042650 165230 137037 003655 006254
0265720 134453 124327 176005 027034 107614 170774 073702 067274
0265740 072451 007735 147620 061064 157435 113057 155356 114603
0265760 107204 102316 171451 046040 120223 001774 030477 046673
0266000 171317 116055 155117 134444 167210 041405 147127 050505
0266020 004137 046472 124015 134360 173550 053517 044635 021135
0266040 070176 047705 113754 175477 105532 076515 177366 056333
0266060 041023 074017 127113 003214 037026 037640 066171 123424
0266100 067701 037406 140000 165341 072410 100032 125455 056646
0266120 006716 071402 055672 132571 105645 170073 050376 072117
0266140 024451 007424 114200 077733 024434 012546 172404 102345
0266160 040223 050170 055164 164634 047154 126525 112514 032315
0266200 016041 176055 042766 025015 176314 017234 110060 014515
0266220 117156 030746 154234 125001 151144 163706 136237 164376
0266240 137055 062276 161755 115466 005322 132567 073216 002655
0266260 171466 126161 117155 065763 016177 014460 112765 055527
0266300 003767 175367 104754 036436 172172 150750 043643 145410
0266320 072074 000007 040627 070652 173011 002151 125132 140214
0266340 060115 014356 015164 067027 120206 070242 033065 131334
0266360 170601 170106 040437 127277 124446 136631 041462 116321
0266400 020243 005602 004146 121574 124651 006634 071331 102070
0266420 157504 160307 166330 074251 024520 114433 167273 030635
0266440 133614 106171 144160 010652 007365 026416 160716 100413
0266460 026630 007210 000630 121224 076033 140764 000737 003276
0266500 114060 042647 104475 110537 066716 104754 075447 112254
0266520 030374 144251 077734 015157 002513 173526 035531 150003
0266540 146207 015135 024446 130101 072457 040764 165513 156412
0266560 166410 067251 156160 106406 136770 030516 064740 022032
0266600 142166 123707 175121 071170 076357 037233 031136 015232
0266620 075074 016744 044055 102230 110063 033350 052765 172463
```



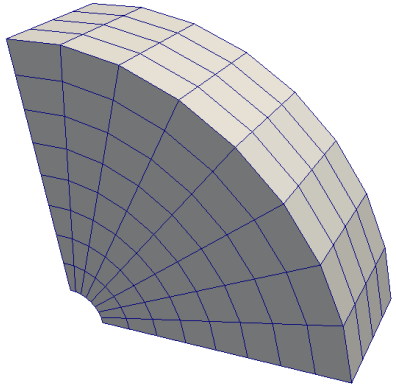
Data Types



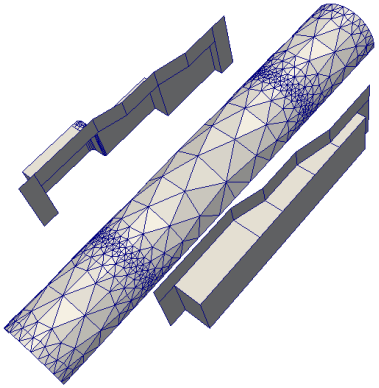
Uniform Rectilinear
(vtkImageData)



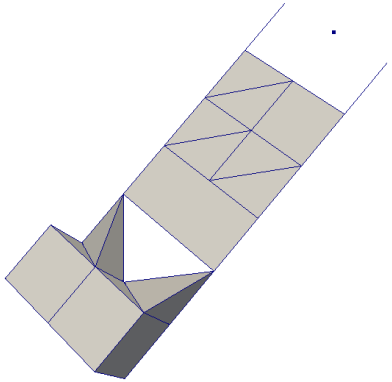
Non-Uniform Rectilinear
(vtkRectilinearData)



Curvilinear
(vtkStructuredData)



Polygonal
(vtkPolyData)



Unstructured Grid
(vtkUnstructuredGrid)

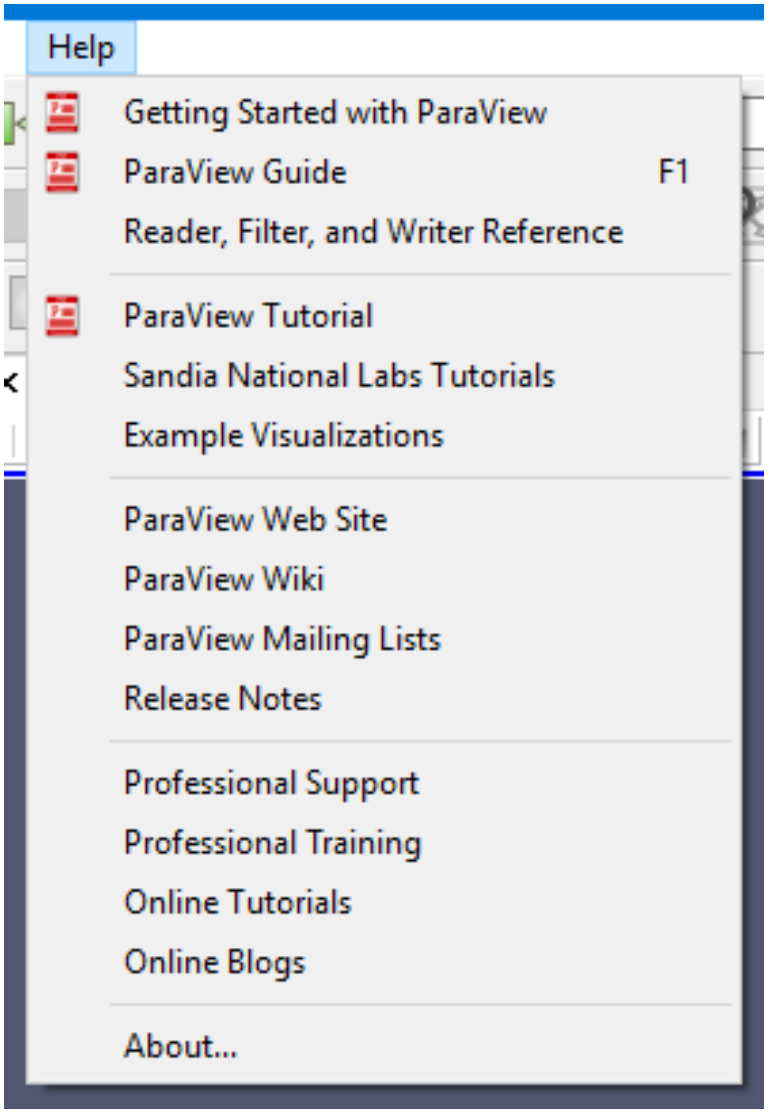
Multi-block

Hierarchical Adaptive Mesh Refinement (AMR)

Hierarchical Uniform AMR

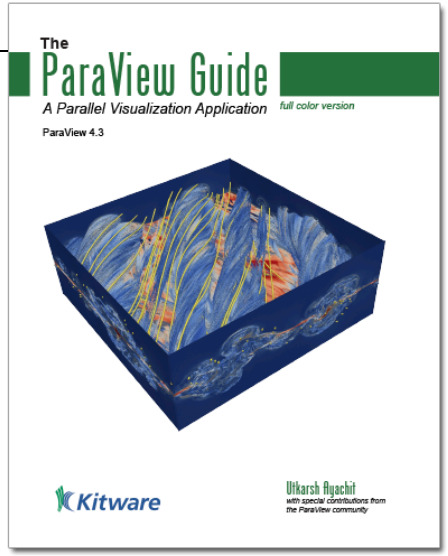
Octree

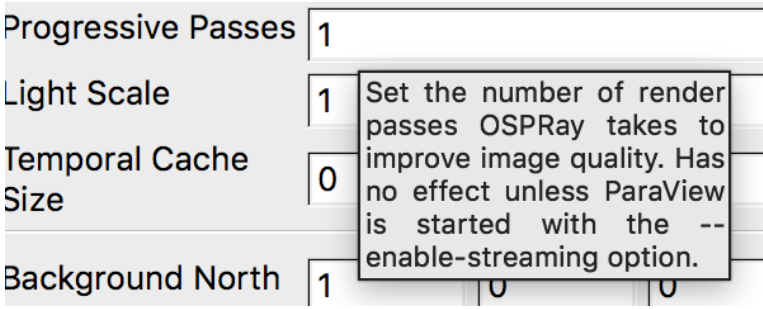
Help Menu


Help
Getting Started with ParaView
ParaView Guide F1
Reader, Filter, and Writer Reference
ParaView Tutorial
Sandia National Labs Tutorials
Example Visualizations
ParaView Web Site
ParaView Wiki
ParaView Mailing Lists
Release Notes
Professional Support
Professional Training
Online Tutorials
Online Blogs
About...

More Information

- Help Menu
  - Getting Started
  - *The ParaView Tutorial*
  - *The ParaView Guide*
    - *aka The Book*
  - The ParaView web page
    - www.paraview.org
  - ParaView discussion forum
    - https://discourse.paraview.org/
- In app ? and pop ups





Basic Usage

User Interface

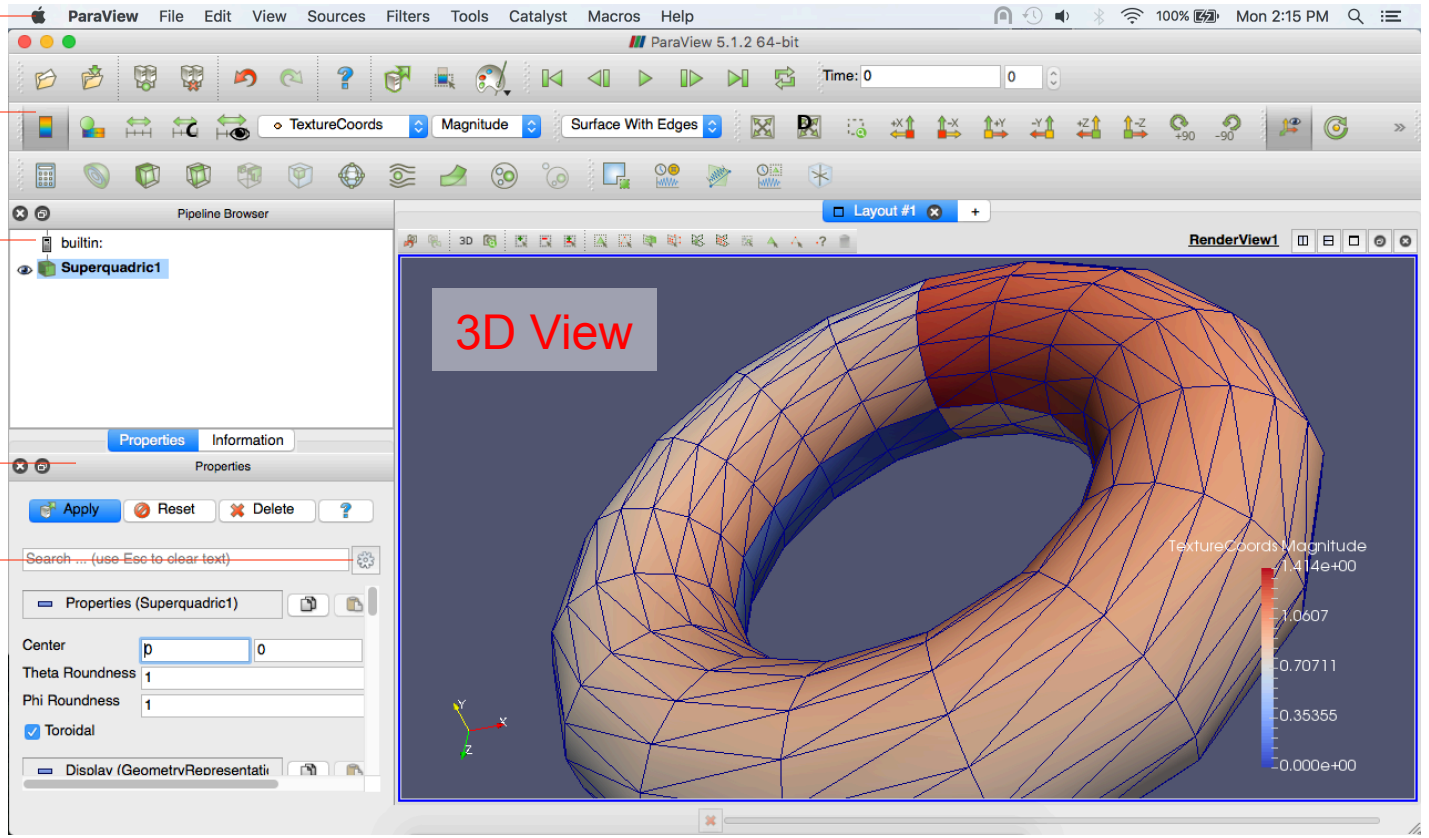
Menu Bar

Toolbars

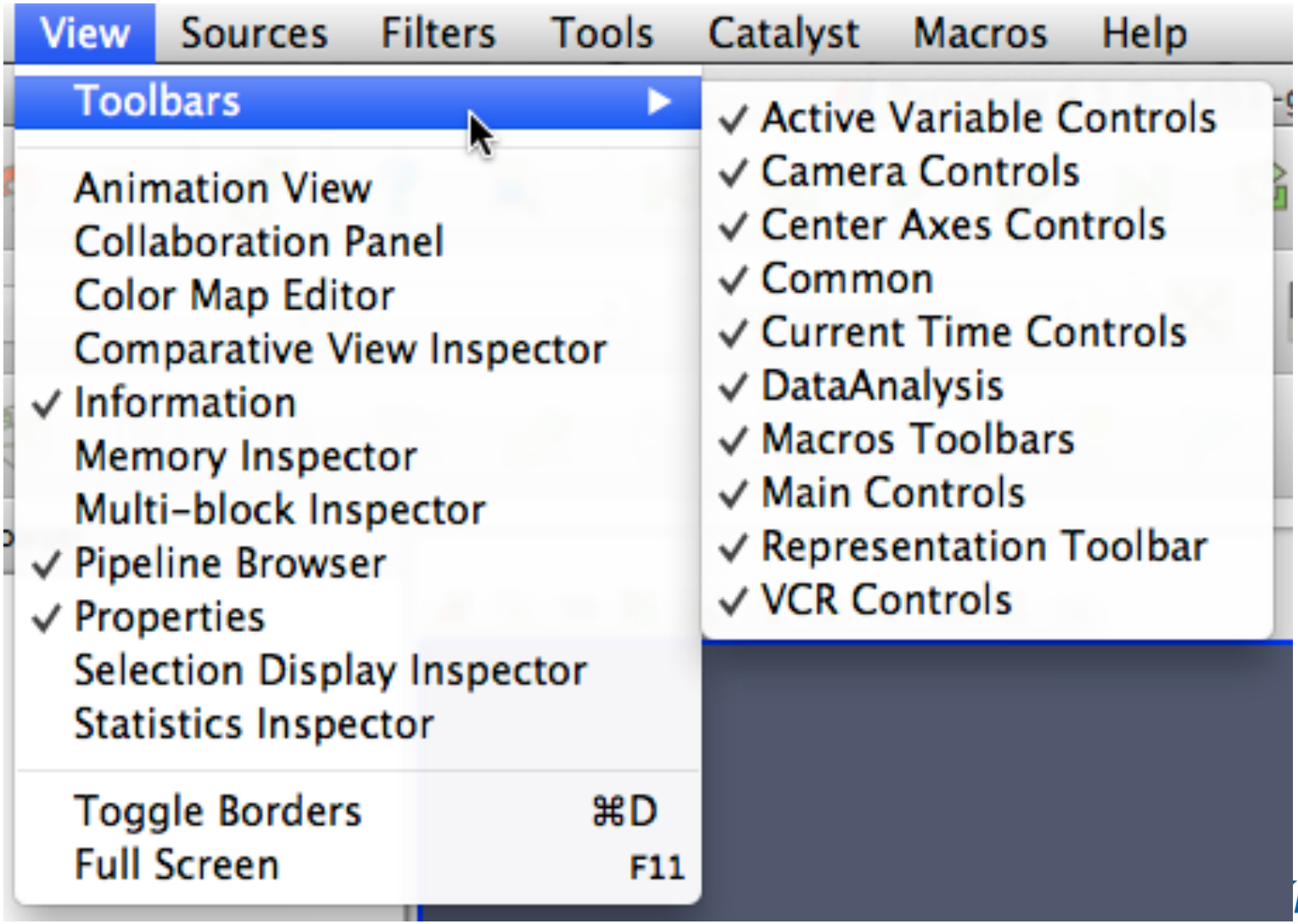
Pipeline Browser

Properties Panel

Advanced Toggle



Getting Back GUI Components

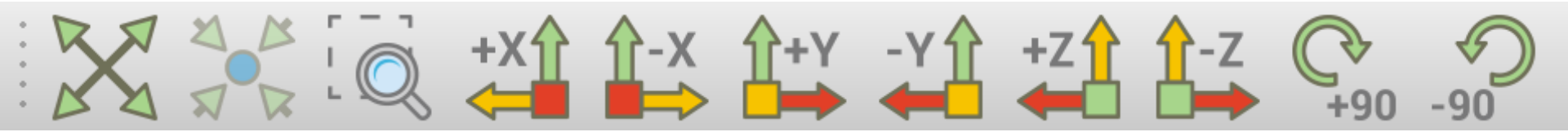


Creating a Cylinder Source

1. Go to the Sources menu and select Cylinder.
2. Click the Apply button to accept the default parameters. 

Simple Camera Manipulation

- Drag left, middle, right buttons for rotate, pan, zoom.
  - Also use Shift, Ctrl, Alt modifiers.
  - Also try holding down x, y, or z.



Creating a Cylinder Source

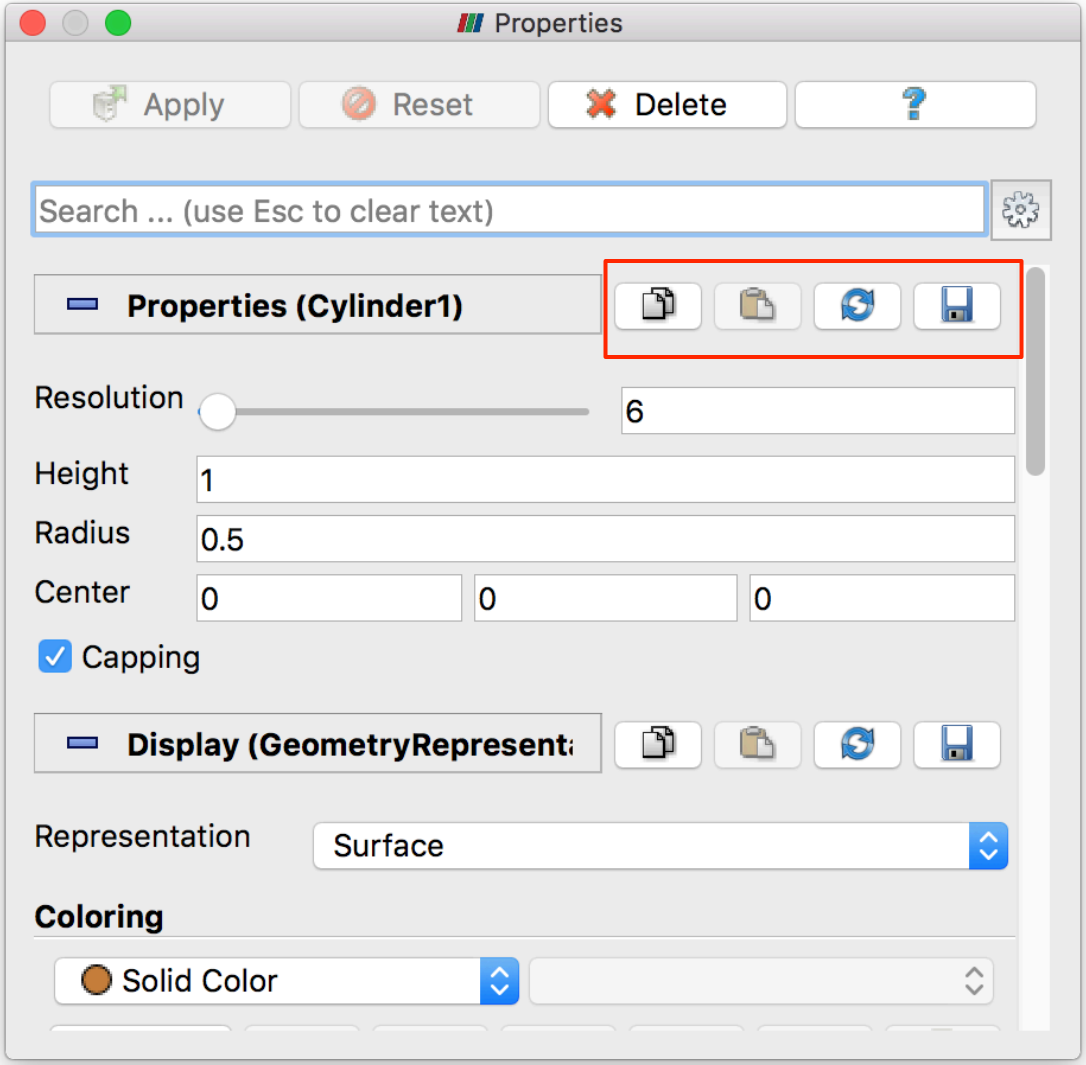
1. Go to the Source menu and select Cylinder.
2. Click the Apply button to accept the default parameters. 

3. Increase the Resolution parameter.

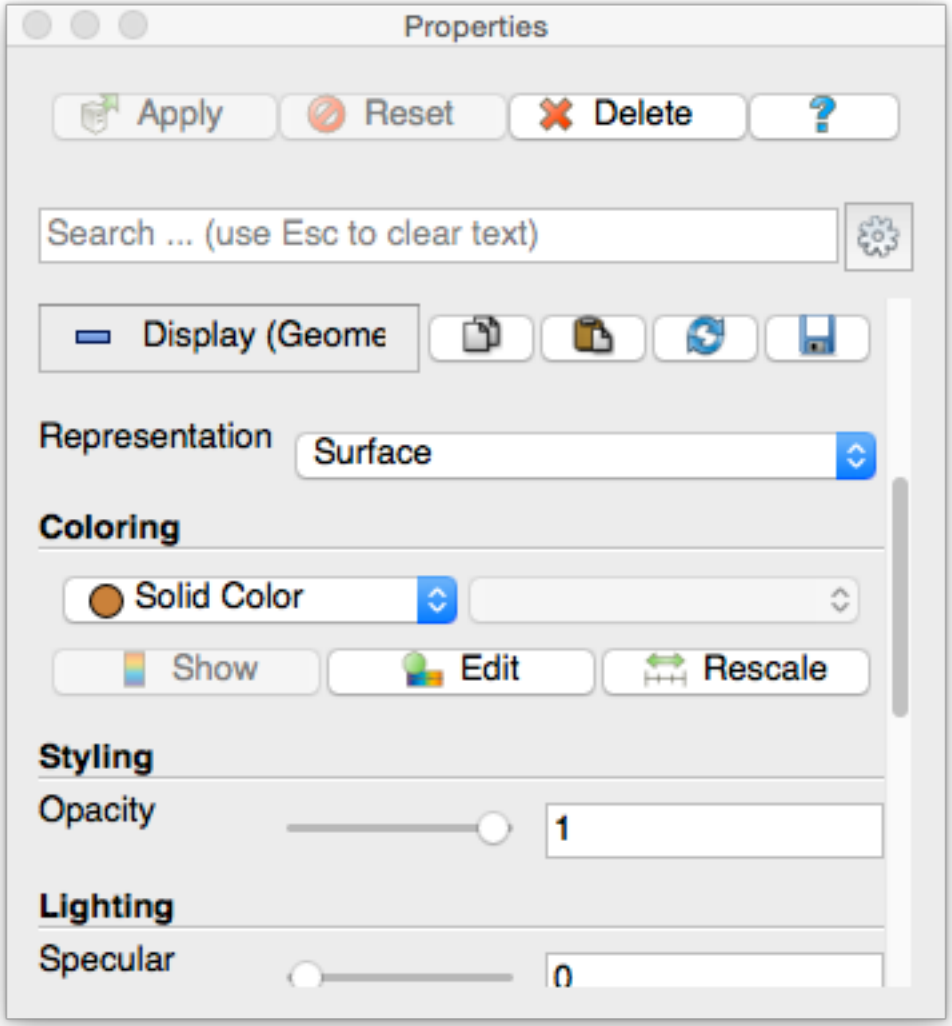
4. Click the  button again.

Pipeline Object Controls

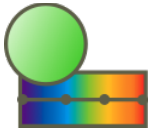
Copy/Paste/Reset/Save Parameters



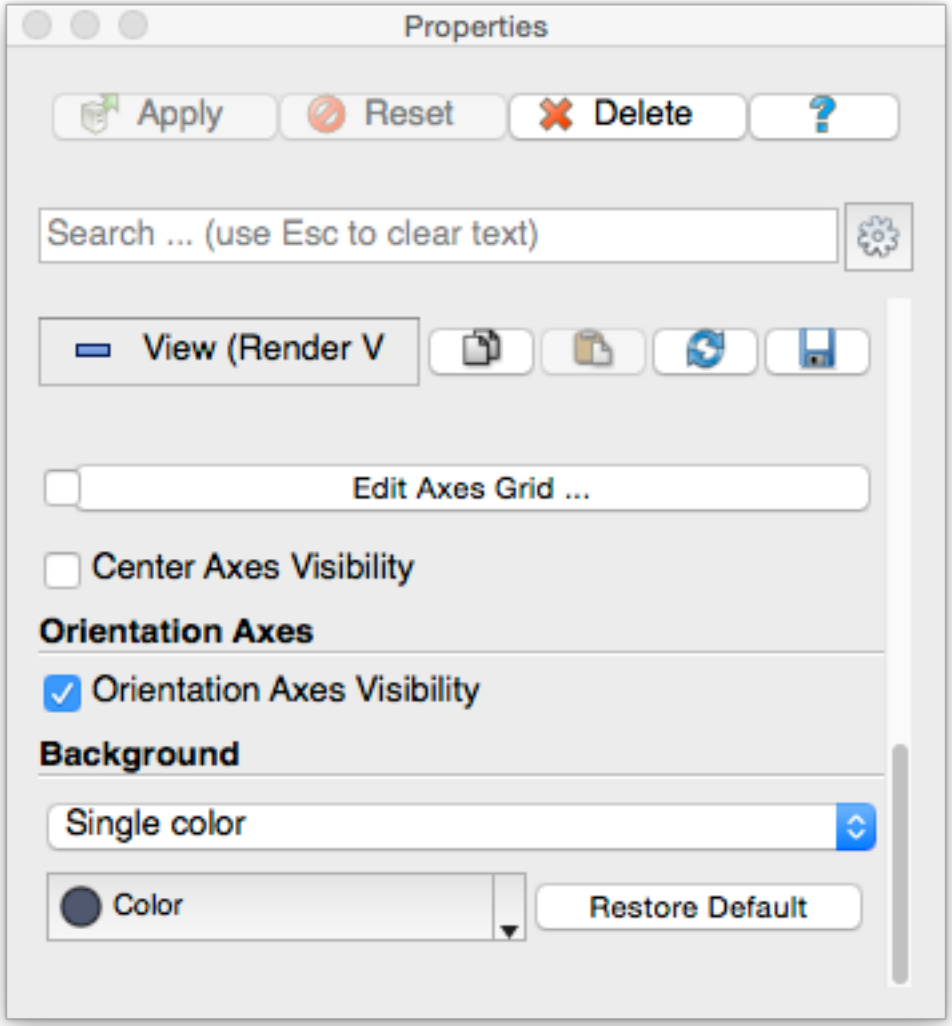
Display Properties



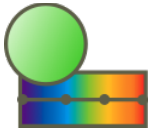
Change Render Properties

1. Scroll down to the Display group.
2. Click the  Edit button. (This button is replicated in the toolbar.)
3. Select a new color for the cylinder.

Render View Options

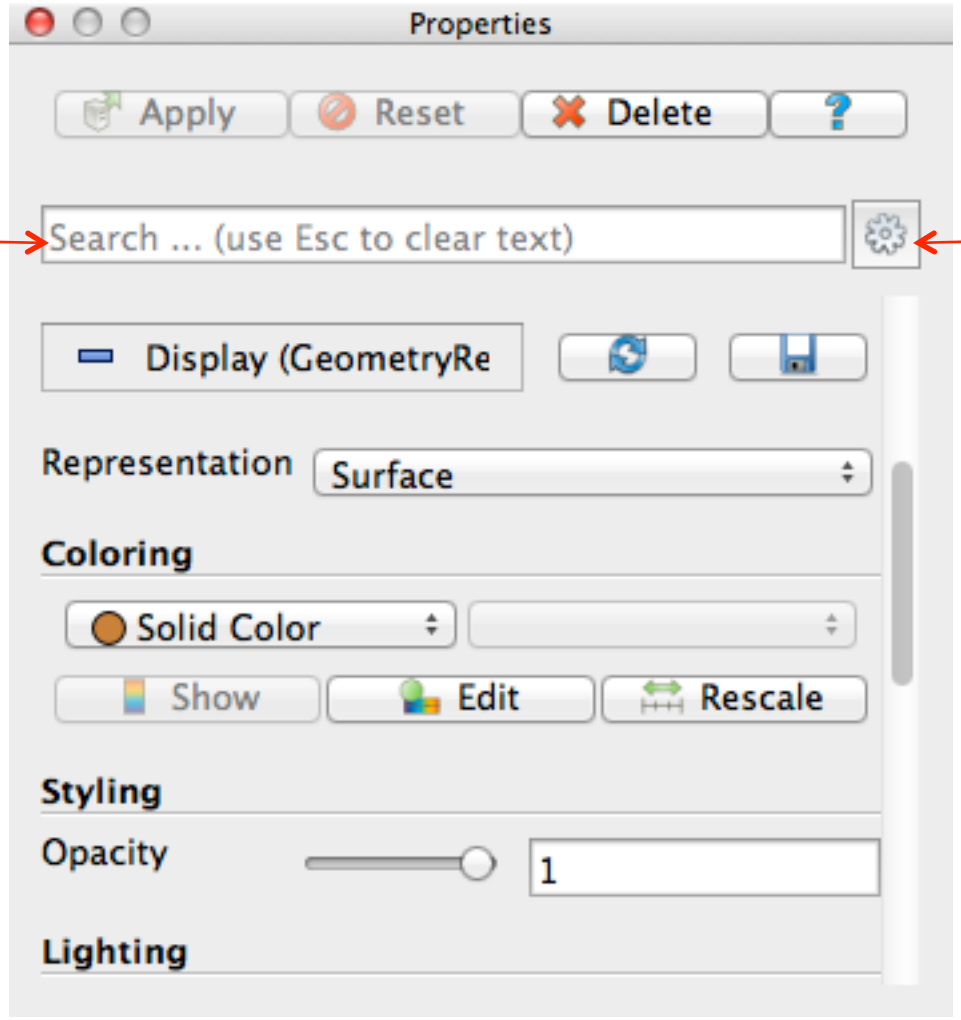


Change Render Properties

1. Scroll down to the Display group.
2. Click the  Edit button. (This button is replicated in the toolbar.)
3. Select a new color for the cylinder.
4. Scroll down to the View group.
5. Turn on the Axis Grid.

Advanced Properties

Search Properties



Toggle Advanced Properties

Searching Properties

1. Type “specular” in the properties search box
2. Change Specular value to 1 (makes the cylinder shiny)

Searching Properties

1. Type “specular” in the properties search box
2. Change Specular value to 1 (makes the cylinder shiny)

Other interesting properties:

- Axes Grid
- Opacity
- Lights

Undo Redo

Undo

Redo

Camera Undo

Camera Redo

Creating a Cylinder Source

1. Go to the Source menu and select Cylinder.
2. Click the Apply button to accept the default parameters. 

3. Increase the Resolution parameter.

4. Click the 

button again.
5. Delete the Cylinder. 

Supported Data Types

- ParaView Data (.pvd)
- VTK (.vtp, .vtu, .vti, .vts, .vtr)
- VTK Legacy (.vtk)
- VTK Multi Block (.vtm,.vtmb,.vtmg,.vthd,.vthb)
- Partitioned VTK (.pvtu, .pvti, .pvts, .pvtr)
- ADAPT (.nc, .cdf, .elev, .ncd)
- ANALYZE (.img, .hdr)
- ANSYS (.inp)
- AVS UCD (.inp)
- BOV (.bov)
- BYU (.g)
- CAM NetCDF (.nc, .ncdf)
- CCSM MTSD (.nc, .cdf, .elev, .ncd)
- CCSM STSD (.nc, .cdf, .elev, .ncd)
- CEAucd (.ucd, .inp)
- CGNS (.cgns)
- CMAT (.cmat)
- CML (.cml)
- CTRL (.ctrl)
- Chombo (.hdf5, .h5)
- Claw (.claw)
- Comma Separated Values (.csv)
- Cosmology Files (.cosmo, .gadget2)
- Curve2D (.curve, .ultra, .ult, .u)
- DDCMD (.ddcmd)
- Digital Elevation Map (.dem)
- Dyna3D(.dyn)
- EnSight (.case, .sos)
- Enzo boundary and hierarchy
- ExodusII (.g, .e, .exe, .ex2, .ex2v.., etc)
- ExtrudedVol (.exvol)
- FVCOM (MTMD, MTSD, Particle, STSD)
- Facet Polygonal Data
- Flash multiblock files
- Fluent Case Files (.cas)
- GGCM (.3df, .mer)
- GTC (.h5)
- GULP (.trg)
- Gadget (.gadget)
- Gaussian Cube File (.cube)
- JPEG Image (.jpg, .jpeg)
- LAMPPS Dump (.dump)
- LAMPPS Structure Files
- LODI (.nc, .cdf, .elev, .ncd)
- LODI Particle (.nc, .cdf, .elev, .ncd)
- LS-DYNA (.k, .lsdyna, .d3plot, d3plot)
- M3DCI (.h5)
- MFIX Unstructred Grid (.RES)
- MM5 (.mm5)
- MPAS NetCDF (.nc, .ncdf)
- Meta Image (.mhd, .mha)
- Miranda (.mir, .raw)
- Multilevel 3d Plasma (.m3d, .h5)
- NASTRAN (.nas, .f06)
- Nek5000 Files
- Nrrd Raw Image (.nrrd, .nhdr)
- OpenFOAM Files (.foam)
- PATRAN (.neu)
- PFLOTRAN (.h5)
- PLOT2D (.p2d)
- PLOT3D (.xyz, .q, .x, .vp3d)
- PLY Polygonal File Format
- PNG Image Files
- POP Ocean Files
- ParaDIS Files
- Phasta Files (.pht)
- Pixie Files (.h5)
- ProSTAR (.cel, .vrt)
- Protein Data Bank (.pdb, .ent, .pdb)
- Raw Image Files
- Raw NRRD image files (.nrrd)
- SAMRAI (.samrai)
- SAR (.SAR, .sar)
- SAS (.sasgeom, .sas, .sasdata)
- SESAME Tables
- SLAC netCDF mesh and mode data
- SLAC netCDF particle data
- Silo (.silo, .pdb)
- Spheral (.spheral, .sv)
- SpyPlot CTH
- SpyPlot (.case)
- SpyPlot History (.hscth)
- Stereo Lithography (.stl)
- TFT Files
- TIFF Image Files
- TSurf Files
- Tecplot ASCII (.tec, .tp)
- Tecplot Binary (.plt)
- Tetrad (.hdf5, .h5)
- UNIC (.h5)
- VASP CHGCA (.CHG)
- VASP OUT (.OUT)
- VASP POSTCAR (.POS)
- VPIC (.vpc)
- VRML (.wrl)
- Velodyne (.vld, .rst)
- VizSchema (.h5, .vsh5)
- Wavefront Polygonal Data (.obj)
- WindBlade (.wind)
- XDMF and hdf5 (.xmf, .xdmf)
- XMol Molecule

Custom Data Import: Plugin Containing a Reader

- Plugins: shared object libraries that can be dynamically loaded into ParaView.
- C++ code and XML description of the interface.
- Any VTK reader object can be added.

Custom Data Import: Python Reader/Filter Plugin

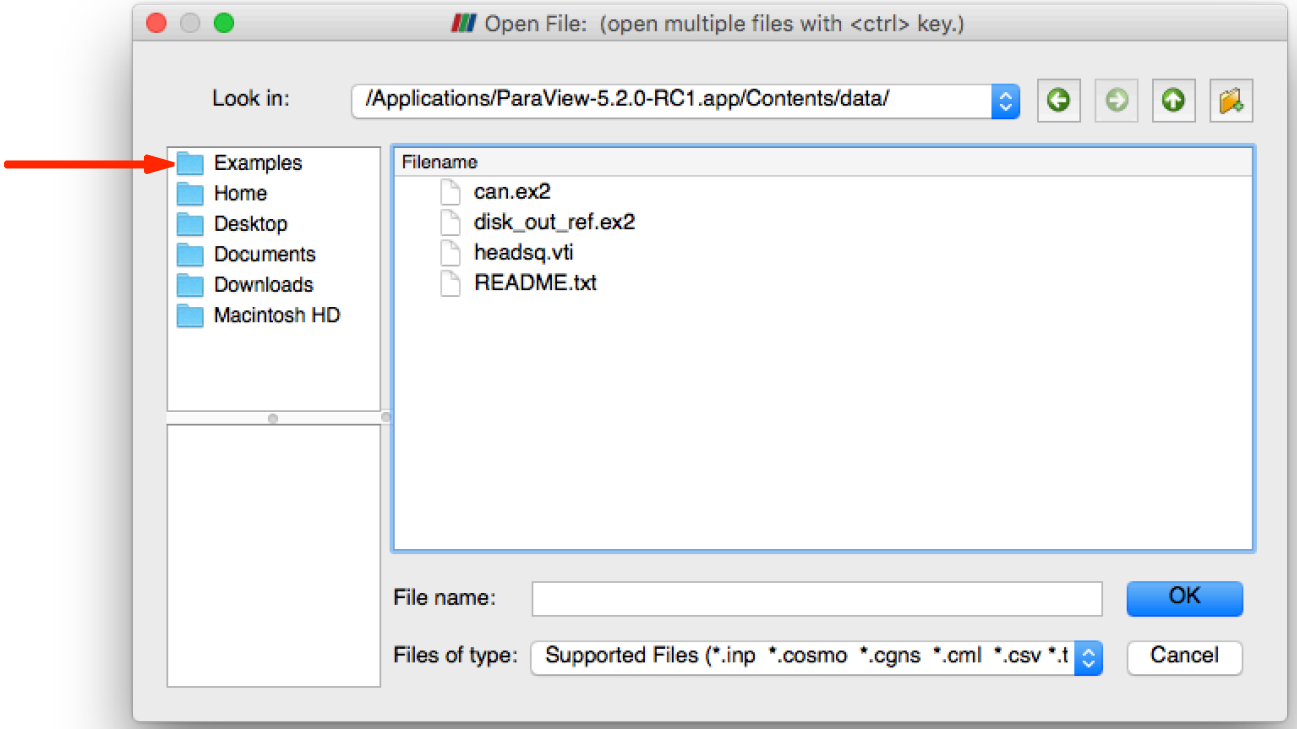
- Available since 5.5.2
- Python code to open file and populate VTK data objects
- Python decorators to tell ParaView GUI what to do with it (similar with the XML description).

```
@smproxy.reader(name="PythonCSVReader”,
                label="Python-based CSV Reader",
                extensions="csv”,
                file_description="CSV files")
class PythonCSVReader(VTKPythonAlgorithmBase):
```

- PythonAlgorithmExamples.py

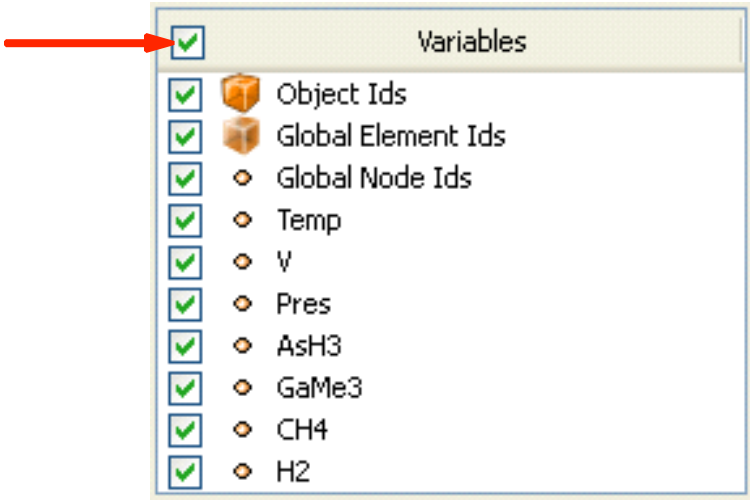
Load disk_out_ref.ex2

1. Open the file disk_out_ref.ex2 from the Examples directory.



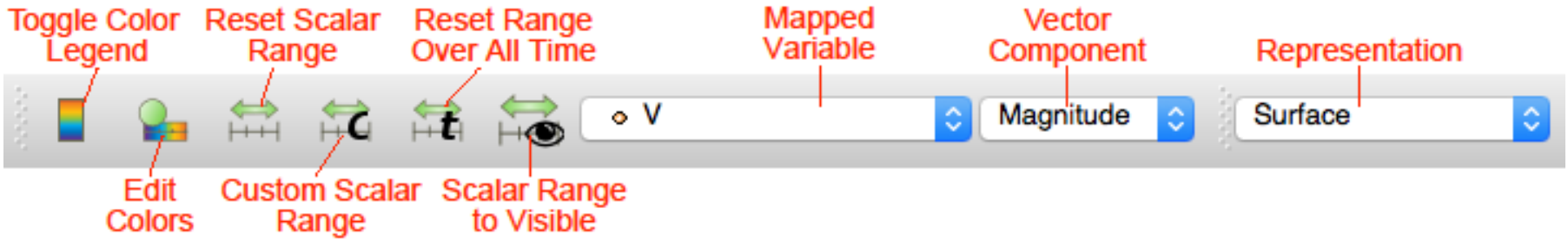
Load disk_out_ref.ex2

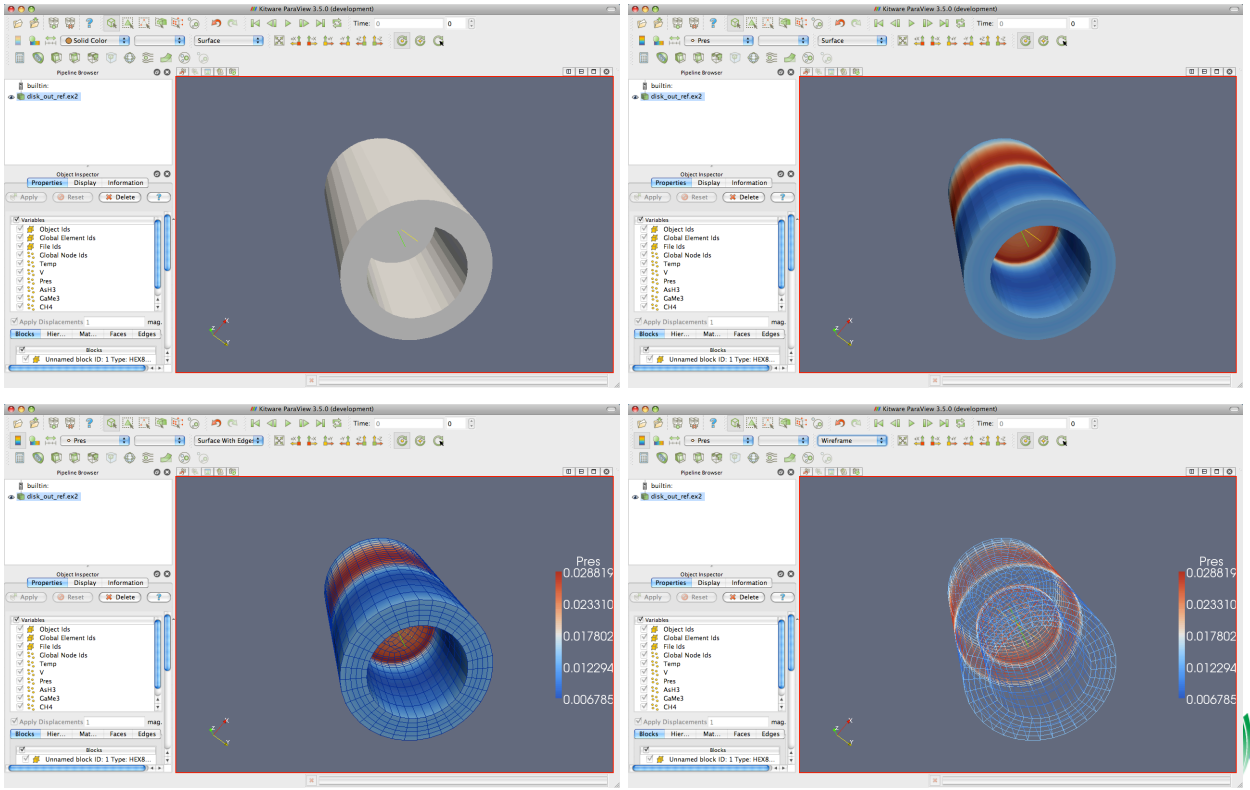
1. Open the file disk_out_ref.ex2 from the Examples directory.
2. Load all data variables.



3. Click 

Data Representation





Common Filters


Calculator


Contour


Clip


Slice


Threshold


Extract Subset


Glyph

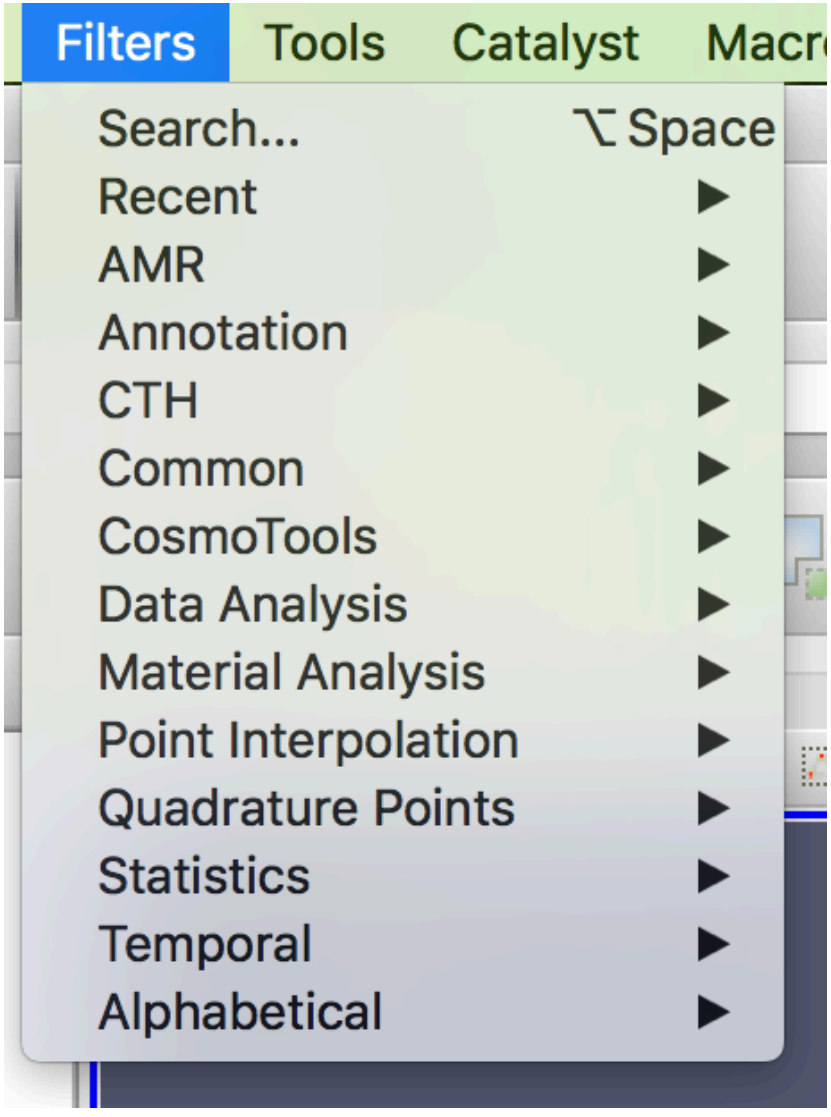

Stream Tracer


Warp (vector)


Group Datasets

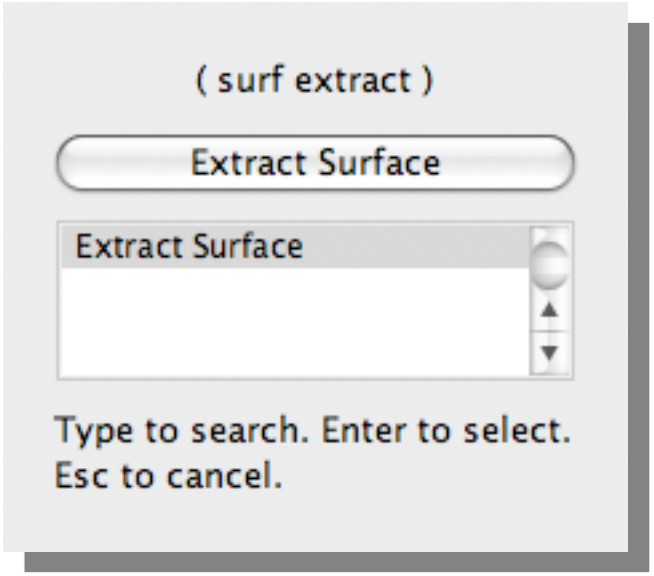

Extract Level

Filters Menu



~ 150 filters
+ C++ plugins
+ python filters

Quick Launch



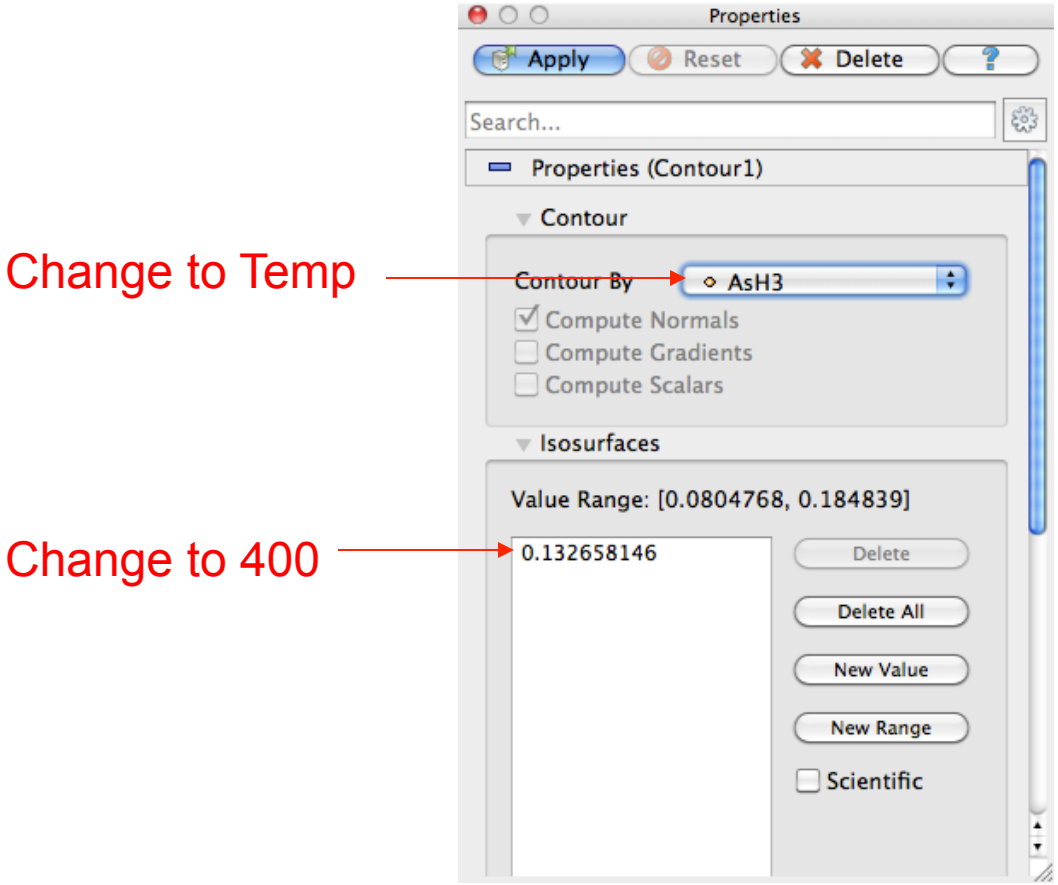
- Used for searching for filters by name
- Keyboard shortcut
  - Ctrl-space for Windows & Linux
  - Alt-space for Mac

Apply a Filter

1. Make sure that disk_out_ref.ex2 is selected in the pipeline browser.
2. Select the contour filter. 

Apply a Filter

3. Change parameters to create an isosurface at Temp = 400K.



Apply a Filter

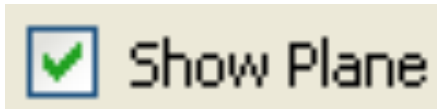
1. Make sure that disk_out_ref.ex2 is selected in the pipeline browser.
2. Select the contour filter. 
3. Change parameters to create an isosurface at Temp = 400K.
4. 

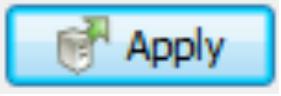
Create a Cutaway Surface

1. Select disk_out_ref.ex2 in the pipeline browser.
2. From the quick launch, select Extract Surface.
3. 

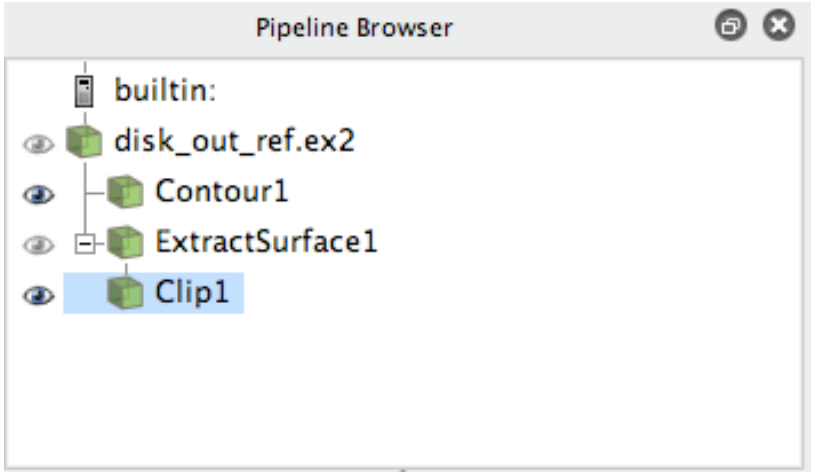
Create a Cutaway Surface

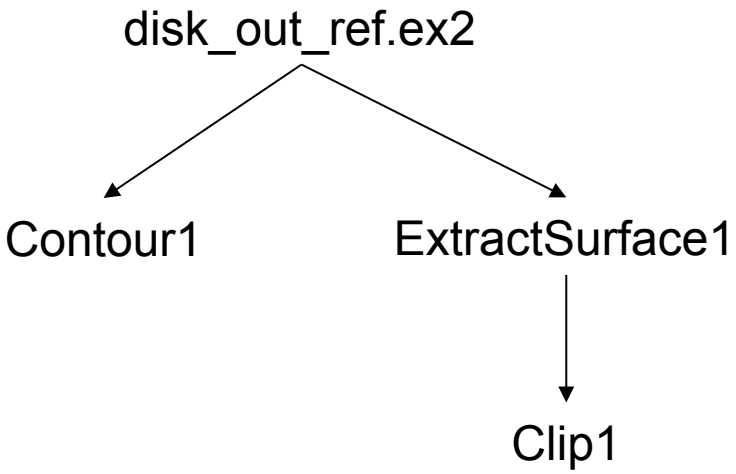
1. Select disk_out_ref.ex2 in the pipeline browser.
2. From the quick launch, select Extract Surface.
3. 

4. Create a clip filter. 
5. Uncheck Show Plane. 

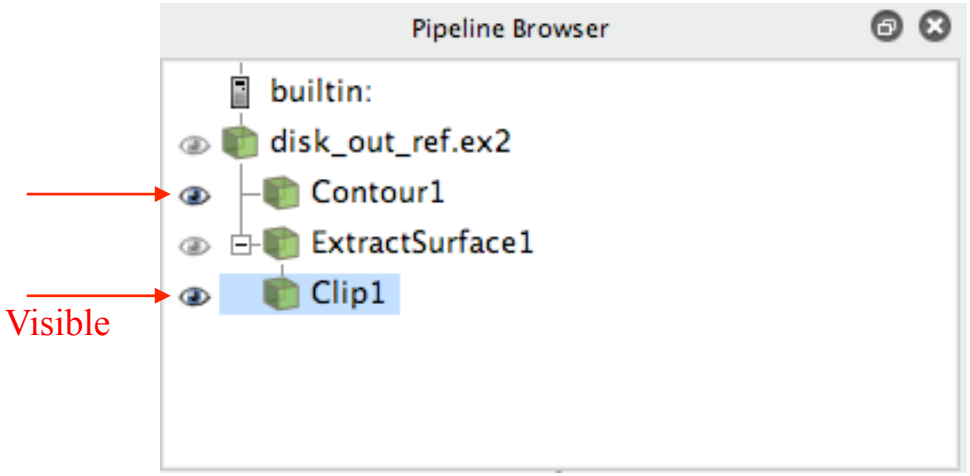
6. 

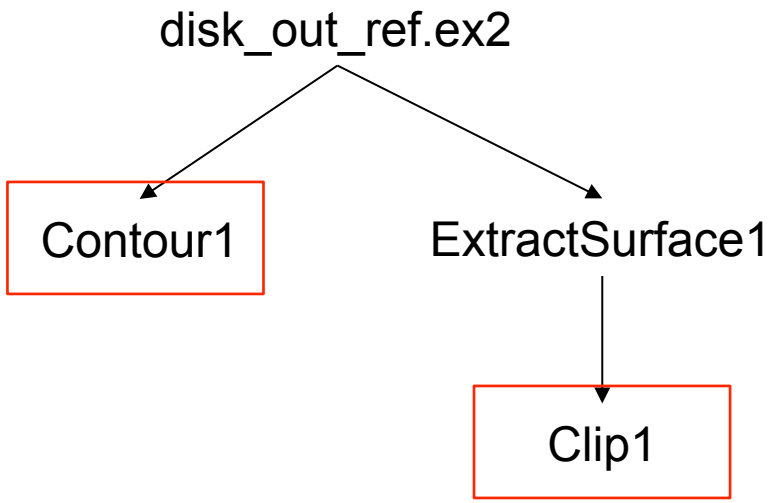
Pipeline Browser Structure



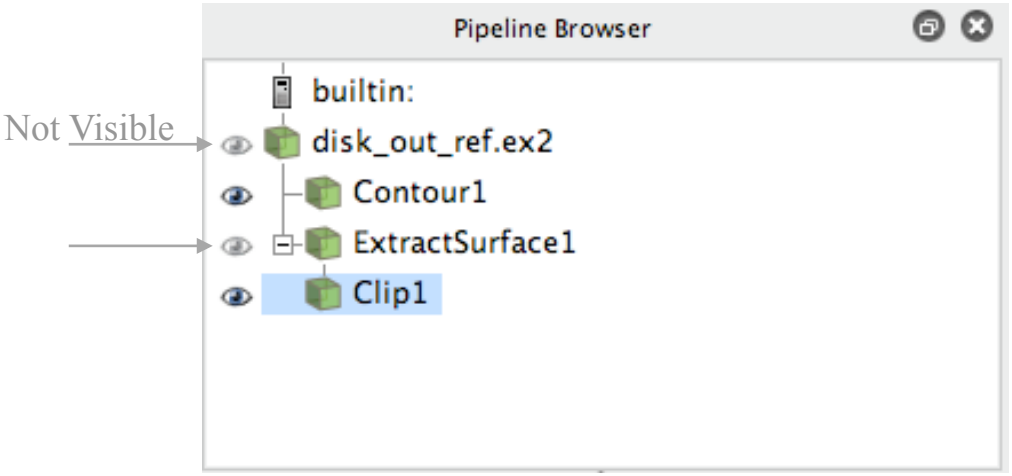


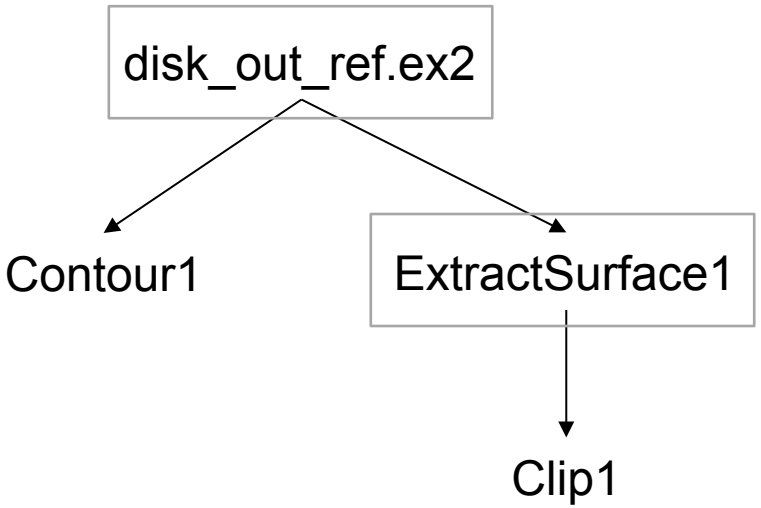
Pipeline Browser Structure





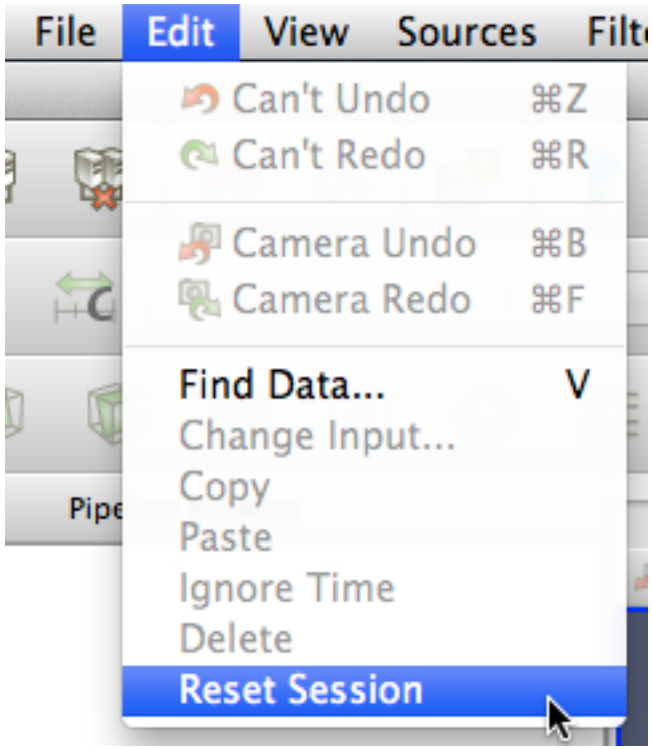
Pipeline Browser Structure





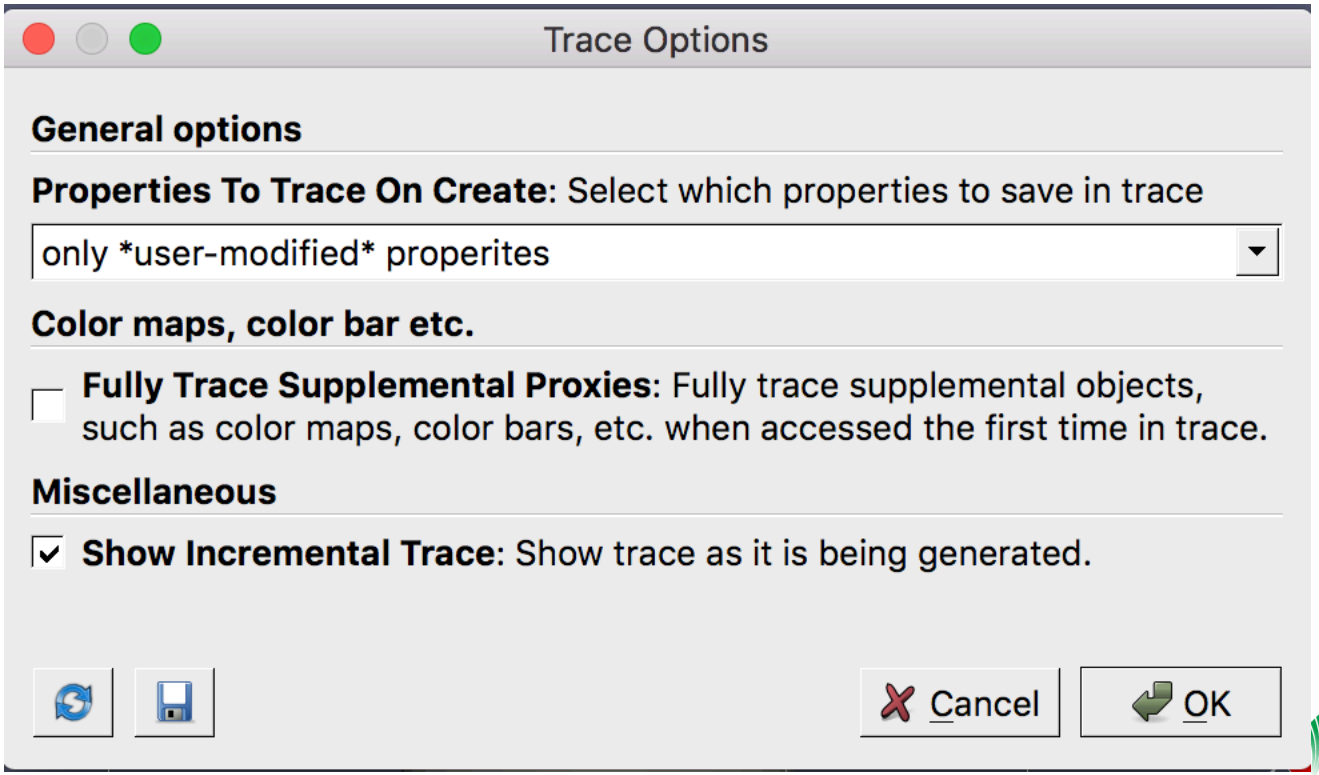
Reset ParaView

Edit → Reset Session



Python Scripting

- Everything you can do with GUI you can do programmatically
- Tools->Start Trace



Python Scripting

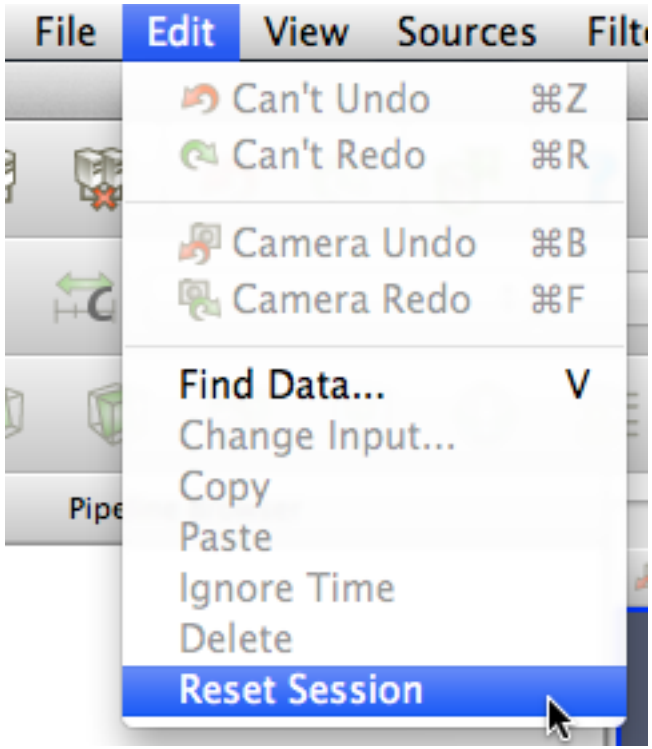
- Open the can data set
- Apply a filter
- Change view parameters
- Tools->Stop Trace
- Save File
- Reset Session
- View->Python Shell -> Run Script

Better yet

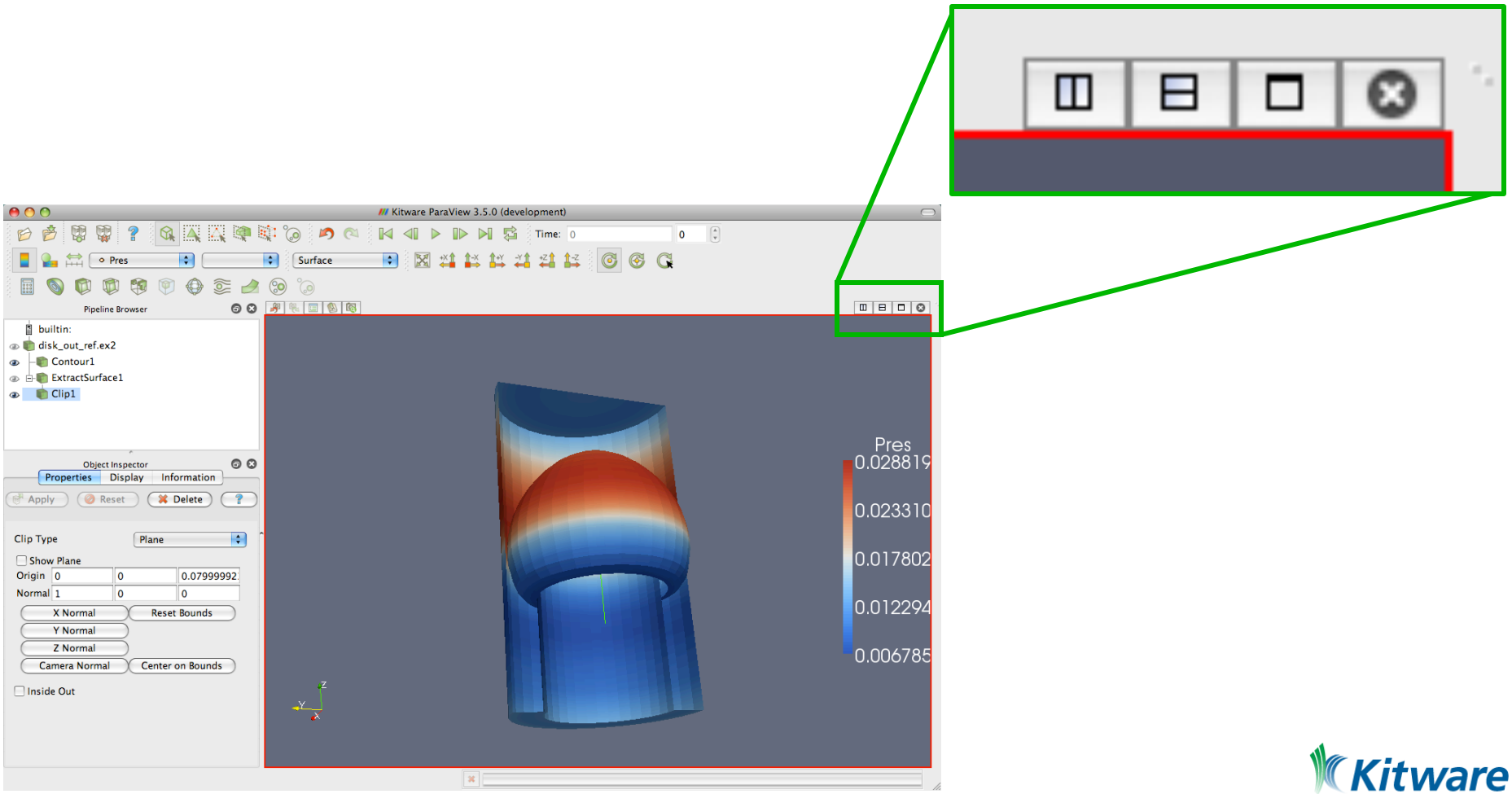
>> qsub mpiexec −n N pvbatch myscript.py

Reset ParaView

Edit → Reset Session



Multiview


Kitware ParaView 3.5.0 (development)
Time:
Pres
Surface
Pipeline Browser
builtin:
disk_out_ref.ex2
Contour1
ExtractSurface1
Clip1
Object Inspector
Properties
Display
Information
Apply
Reset
Delete
Clip Type
Plane
Show Plane
Origin
Normal
X Normal
Reset Bounds
Y Normal
Z Normal
Camera Normal
Center on Bounds
Inside Out
Pres
0.028819
0.023310
0.017802
0.012294
0.006785

Multiview

1. Open disk_out_ref.ex2. Load all variables. 

2. Add Clip filter. 
3. Uncheck Show Plane. 

4. 

5. Color surface by Pres.

Multiview

6. Split the view horizontally. 
7. Make Clip1 visible. 
8. Color surface by Temp.

Multiview

6. Split the view horizontally. 
7. Make Clip1 visible. 
8. Color surface by Temp.
9. Right-click view, Link Camera...
10. Click other view.

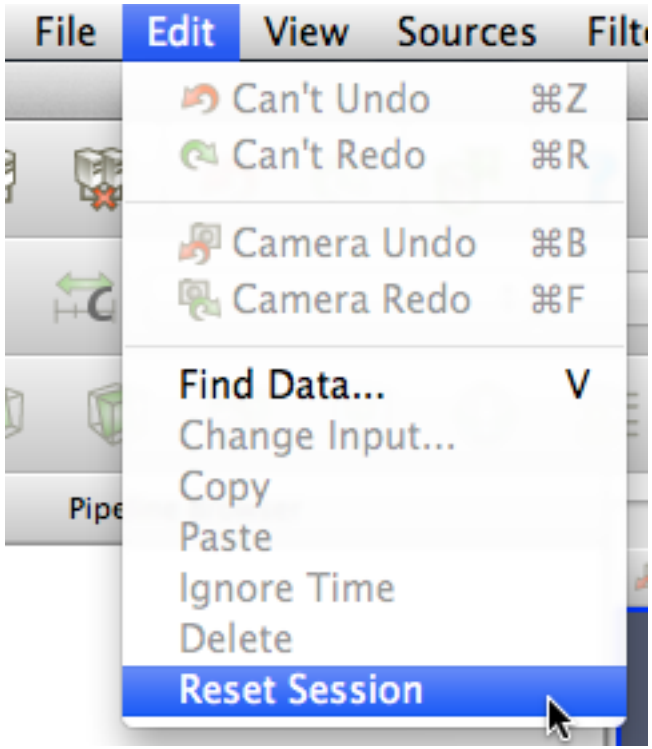
Multiview

6. Split the view horizontally. 
7. Make Clip1 visible. 
8. Color surface by Temp.
9. **Right-click view, Link Camera…**
10. Click other view.
11. Click 

and zoom in a bit.

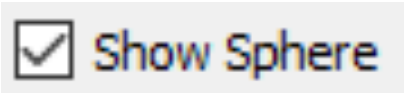
Reset ParaView

Edit → Reset Session



Streamlines

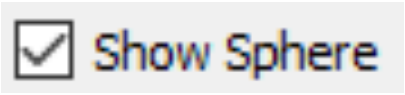
1. Open disk_out_ref.ex2. Load all variables. 

2. Add Stream Tracer. 
3. Change Seed Type to Point Source.
4. Uncheck Show Sphere. 

5. 

Streamlines

1. Open disk_out_ref.ex2. Load all variables. 

2. Add Stream Tracer. 
3. Change Seed Type to Point Source.
4. Uncheck Show Sphere. 

5. 

6. From the quick launch, select Tube
7. 

Getting Fancy

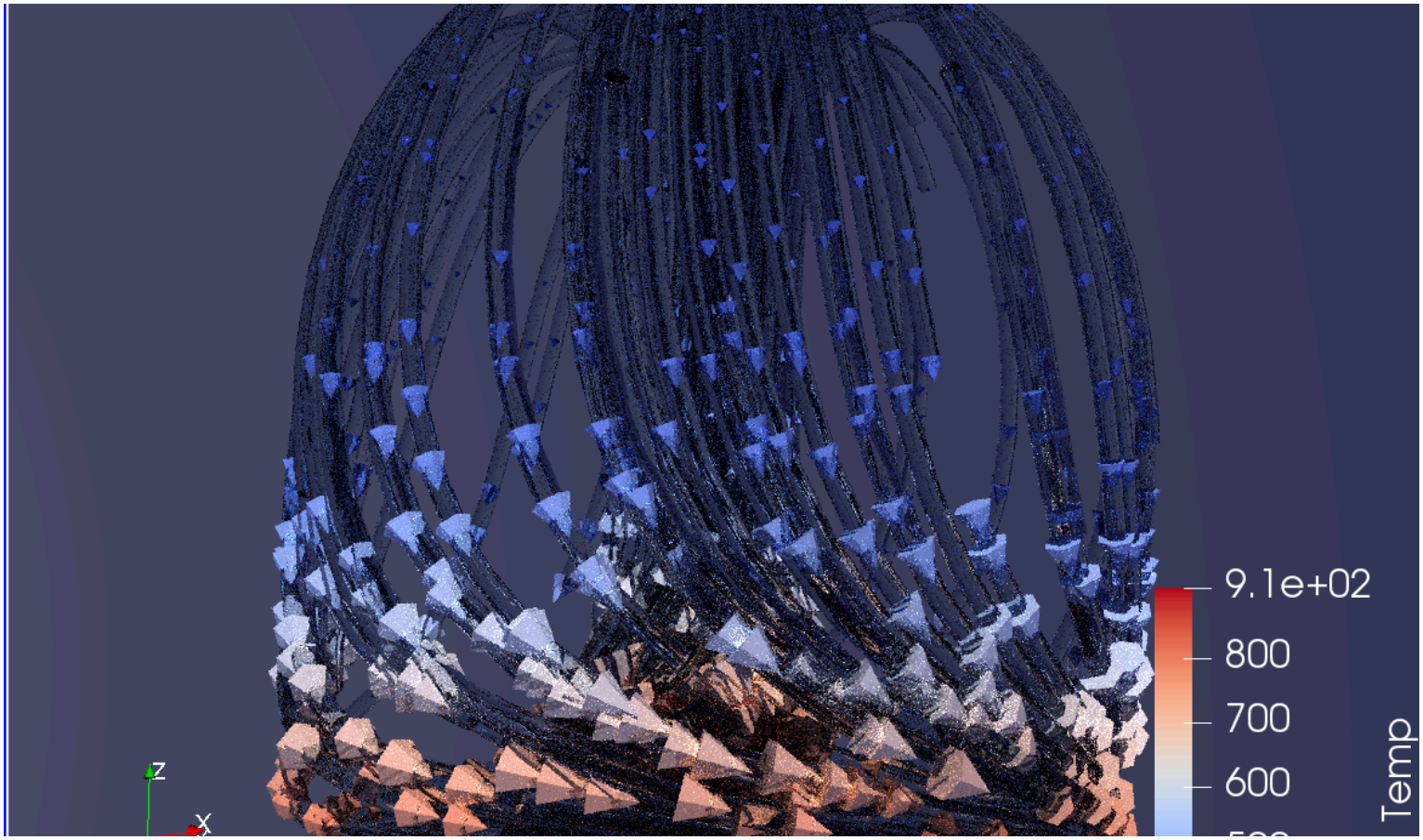
8. Select StreamTracer1.
9. Add Glyph filter. 
10. Change Glyph Type to Cone.
11. Change Vectors to V.
12. Change Scale Mode to vector.
13. Click reset  next to Scale Factor.
14. 

15. Color by Temp.

Getting Really Fancy

16. Enable OSPRay
17. Enable Shadows
18. OSPRay Renderer -> pathtracer
19. Select Tube1
20. OSPRay Material -> “water”
21. Background to Gradient

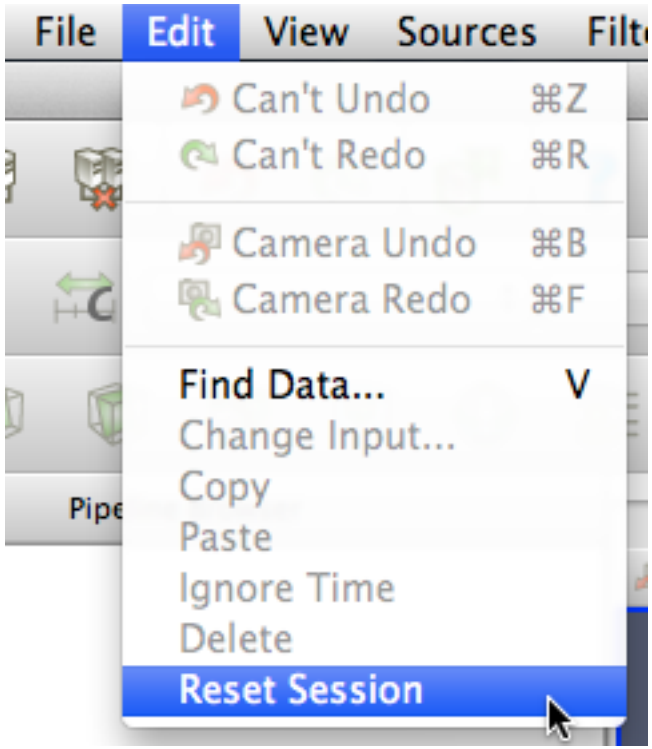
Getting Really Fancy



See also: https://www.paraview.org/gallery/#videos

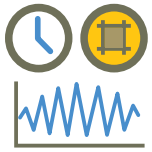
Reset ParaView

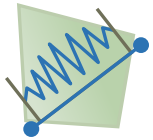
Edit → Reset Session



Common Data Analysis Filters


Extract Selection


Plot Global Variables Over Time


Plot Over Line


Plot Selection Over Time

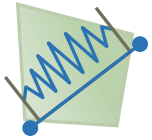

Probe Location

Plotting

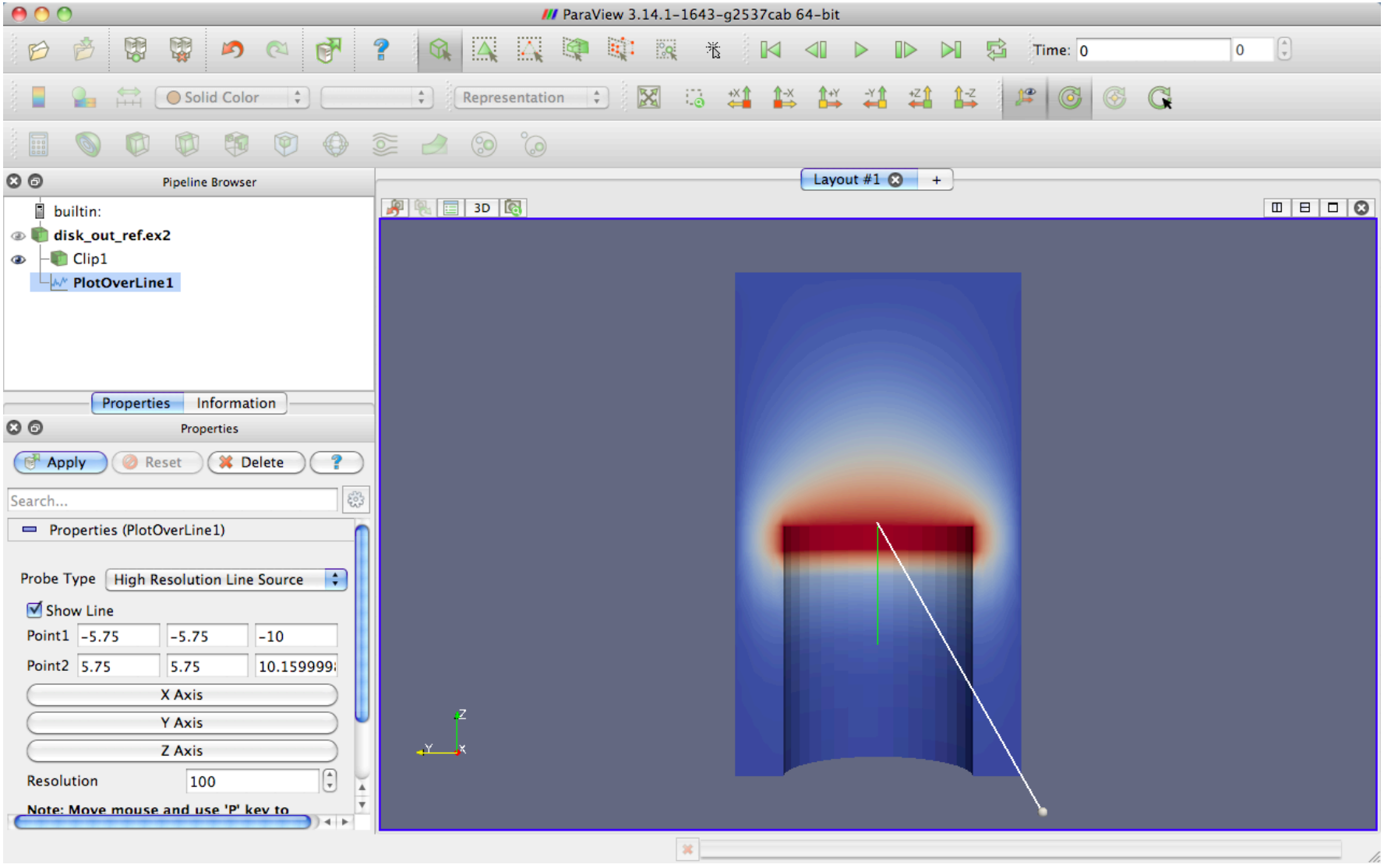
1. Open disk_out_ref.ex2. Load all variables. 

2. Clip,  uncheck, 

, 

3. Select disk_out_ref.ex2.
4. Add Plot Over Line filter. 

3D Widgets


ParaView 3.14.1-1643-g2537cab 64-bit
Time: 0
Solid Color
Representation
Pipeline Browser
builtin:
disk_out_ref.ex2
Clip1
PlotOverLine1
Properties
Information
Properties
Apply
Reset
Delete
Search...
Properties (PlotOverLine1)
Probe Type
High Resolution Line Source
Show Line
Point1 -5.75 -5.75 -10
Point2 5.75 5.75 10.159999
X Axis
Y Axis
Z Axis
Resolution 100
Note: Move mouse and use 'P' key to
Layout #1
3D

Placing 3D Line Widget Endpoints

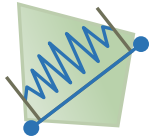
- Use the p key to place alternating points.
  - Ctrl+p places at nearest mesh point.
- Use the 1 or 2 key to place the start or end point.
  - Ctrl+1 or Ctrl+2 places at mesh point.
- Drag the endpoints.
  - Use x, y, or z key to constrain to axis.
- Use widgets in Properties panel
  - E.g. Use Z Axis button and then edit points to place from (0,0,0) to (0, 0, 10).

Plotting

1. Open disk_out_ref.ex2. Load all variables. 

2. Clip,  uncheck, 

, 

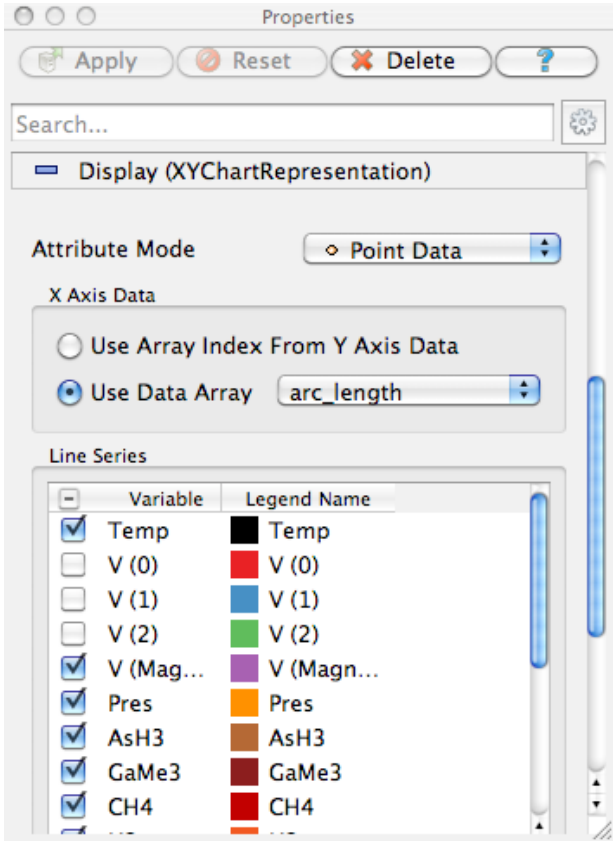
3. Select disk_out_ref.ex2.
4. Add Plot Over Line filter. 
5. Once line is satisfactorily located, 

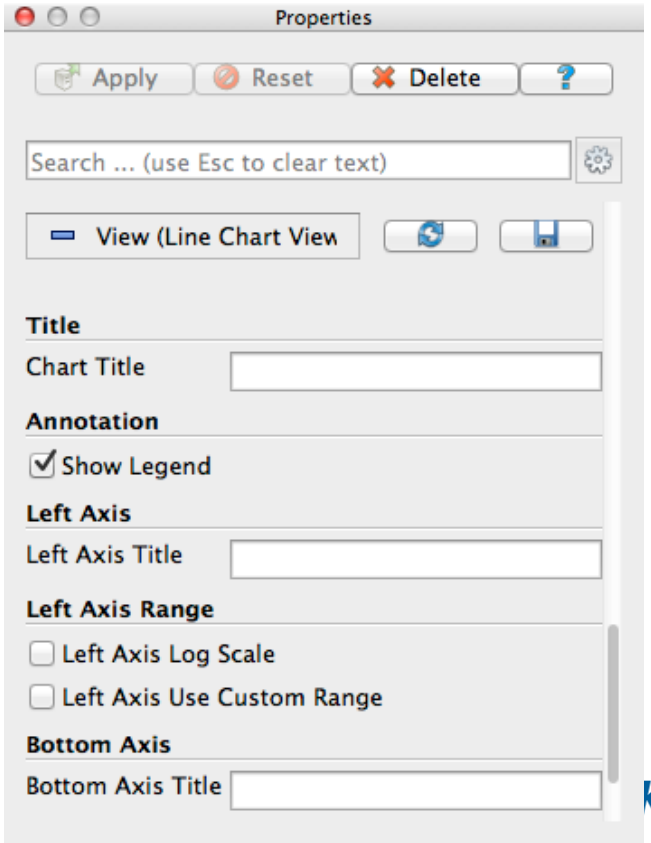
Interacting with Plots

- Left, middle, right buttons to pan, zoom.
- Mouse wheel to zoom.
- Reset view to plot ranges. 

Plots are Views

- Move them like Views.
- Save screenshots.





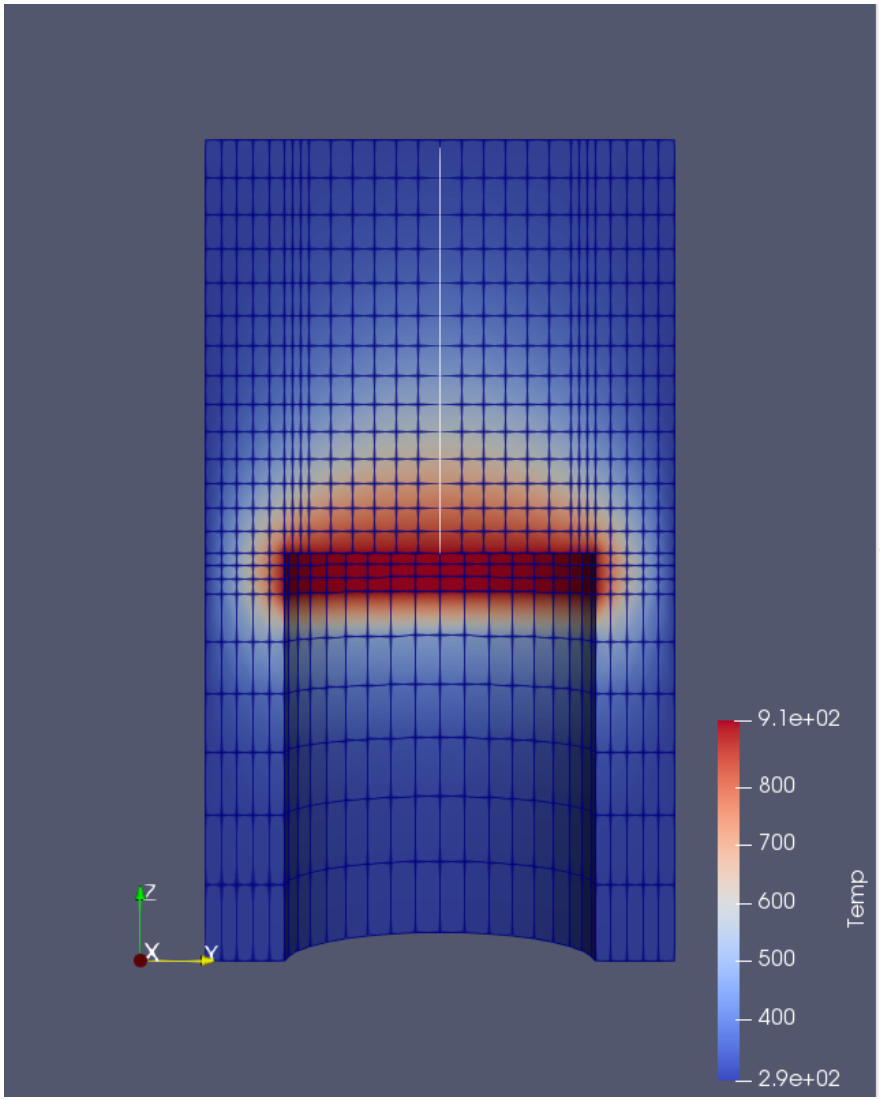
Adjusting Plots

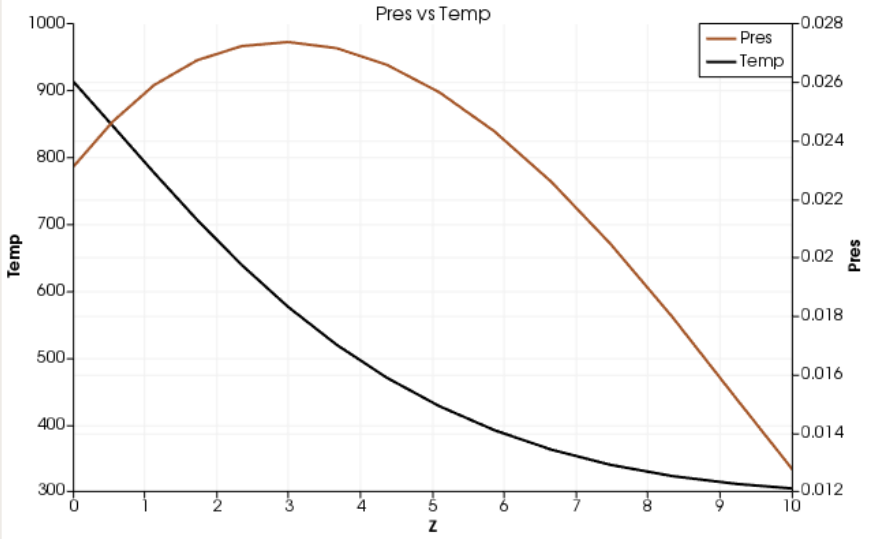
1. In Display section of properties panel, turn off all variables except Temp and Pres.
2. Select Pres in the Display options.
3. Change Chart Axis to Bottom – Right.
4. Verify the relationship between temperature and pressure.

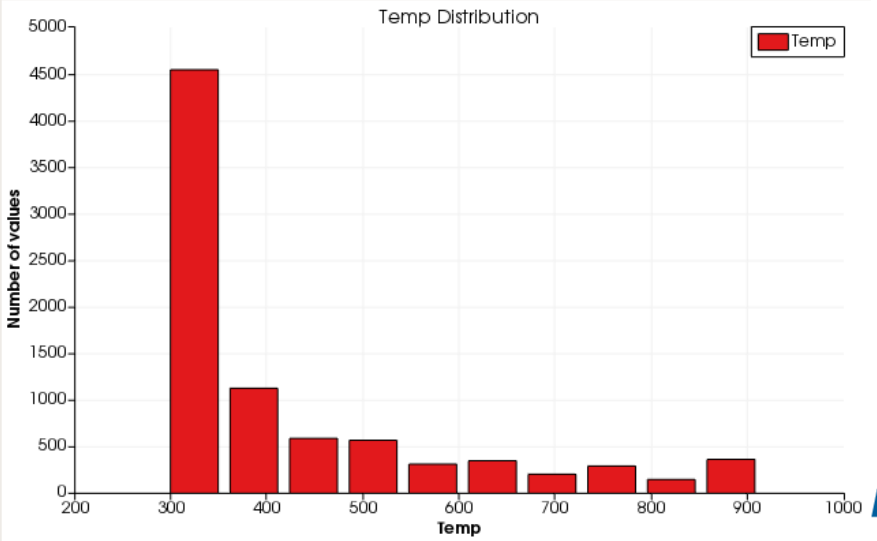
Histogram / Bar Chart

1. Select disk_out_ref.ex2.
2. Filters → Data Analysis → Histogram 
3. Change Input Array to Temp.
4. 

Histogram / Bar Chart

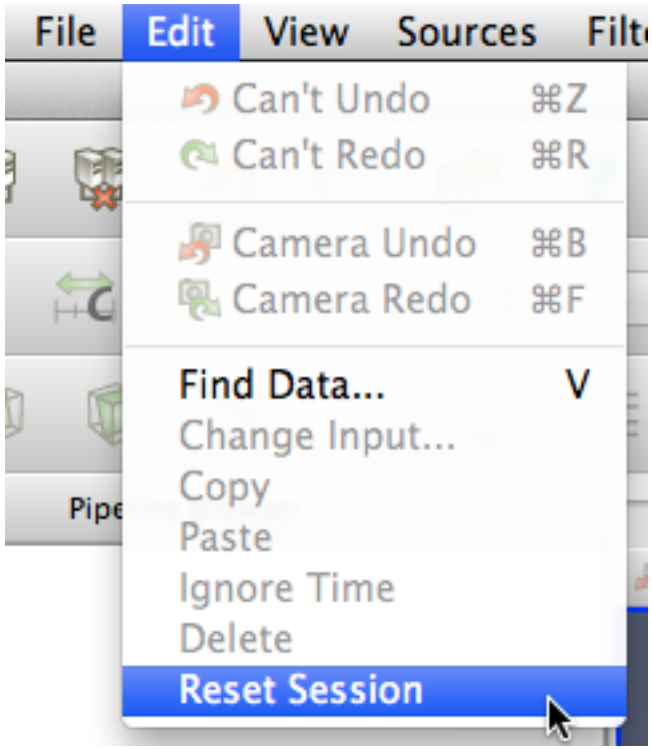






Reset ParaView

Edit → Reset Session



Volume Rendering

1. Open disk_out_ref.ex2. Load all variables. 

2. Change variable viewed to Temp.
3. Change representation to Volume.
4. In the Are you Sure dialog box, click Yes.

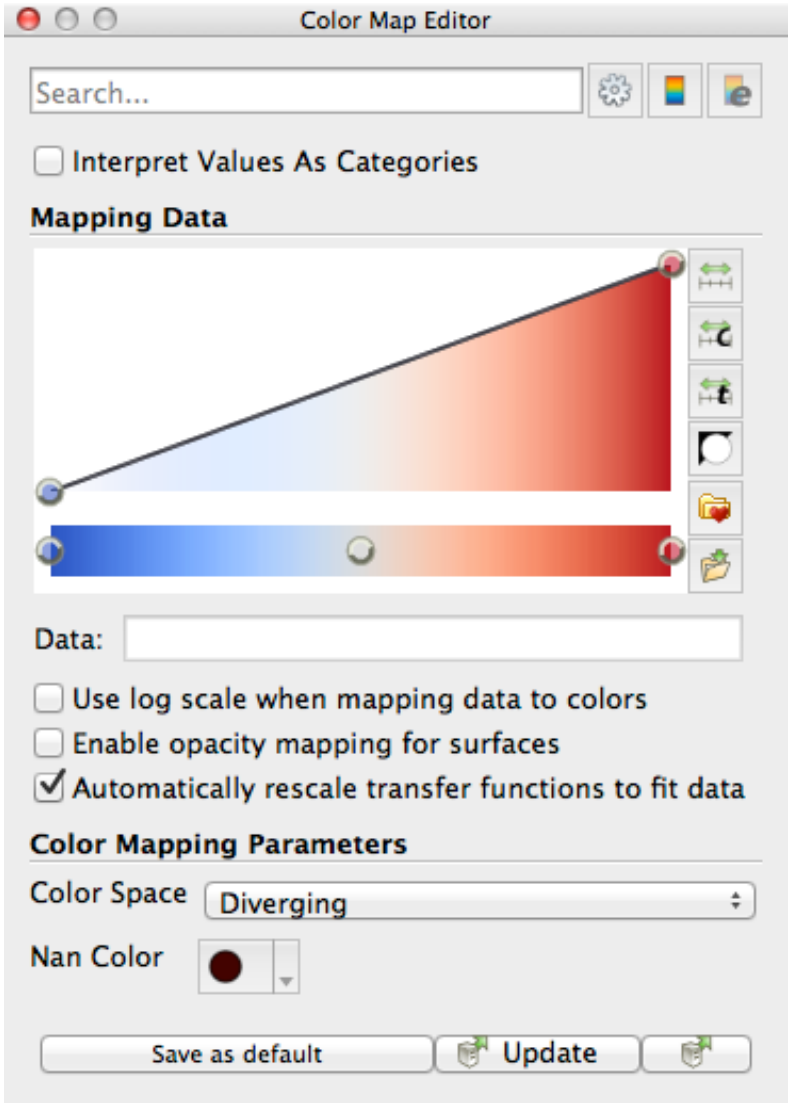
Volume Rendering + Surface Geometry

1. Open disk_out_ref.ex2. Load all variables. 

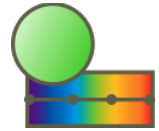
2. Change variable viewed to Temp.
3. Change representation to Volume.
4. Add Stream Tracer.  

5. Optional: Add Tubes and Glyphs.

Transfer Function Editor

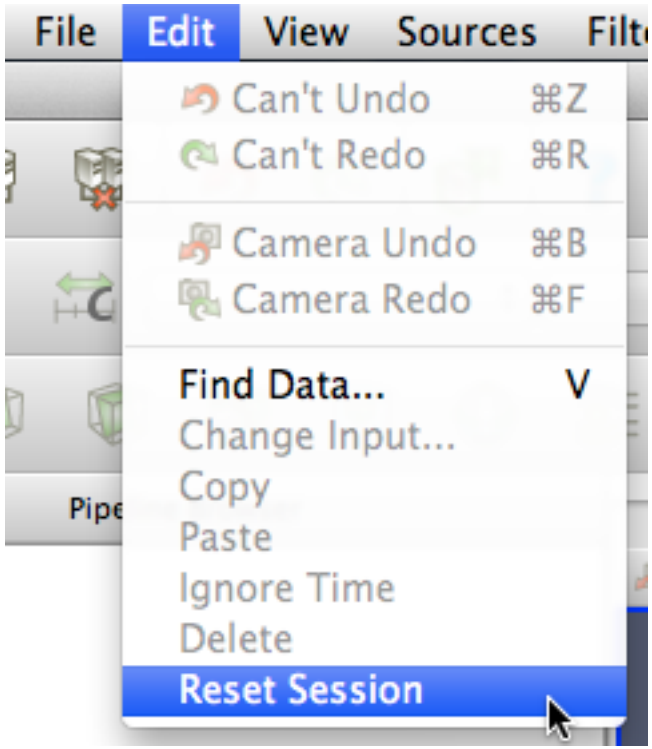


Modify Transfer Function

1. Select disk_out_ref.ex2.
2. Click Edit Color Map  .
3. Click Choose preset  .
4. Select Black-Body Radiation. Apply. Close.
5. Try adding and changing control points.

Reset ParaView

Edit → Reset Session

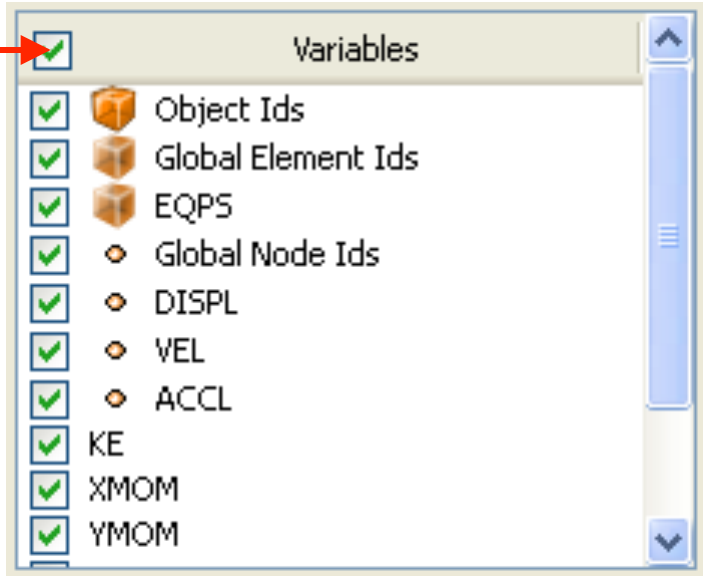


Loading Data with Time

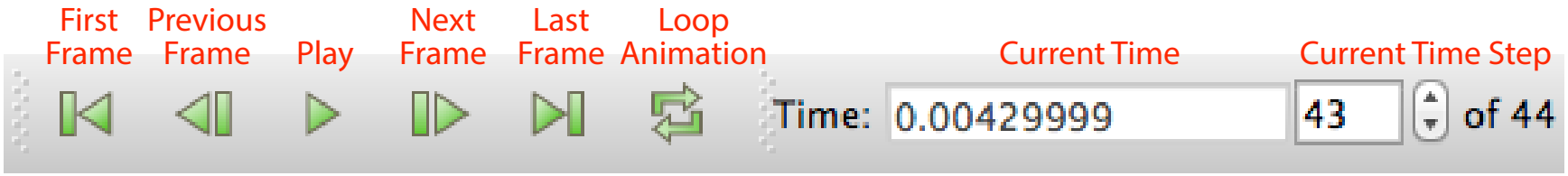
1. Open the file can.ex2.
2. Select all variables.
3. 

4. 

5. 

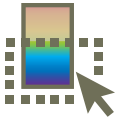


Animation Toolbar

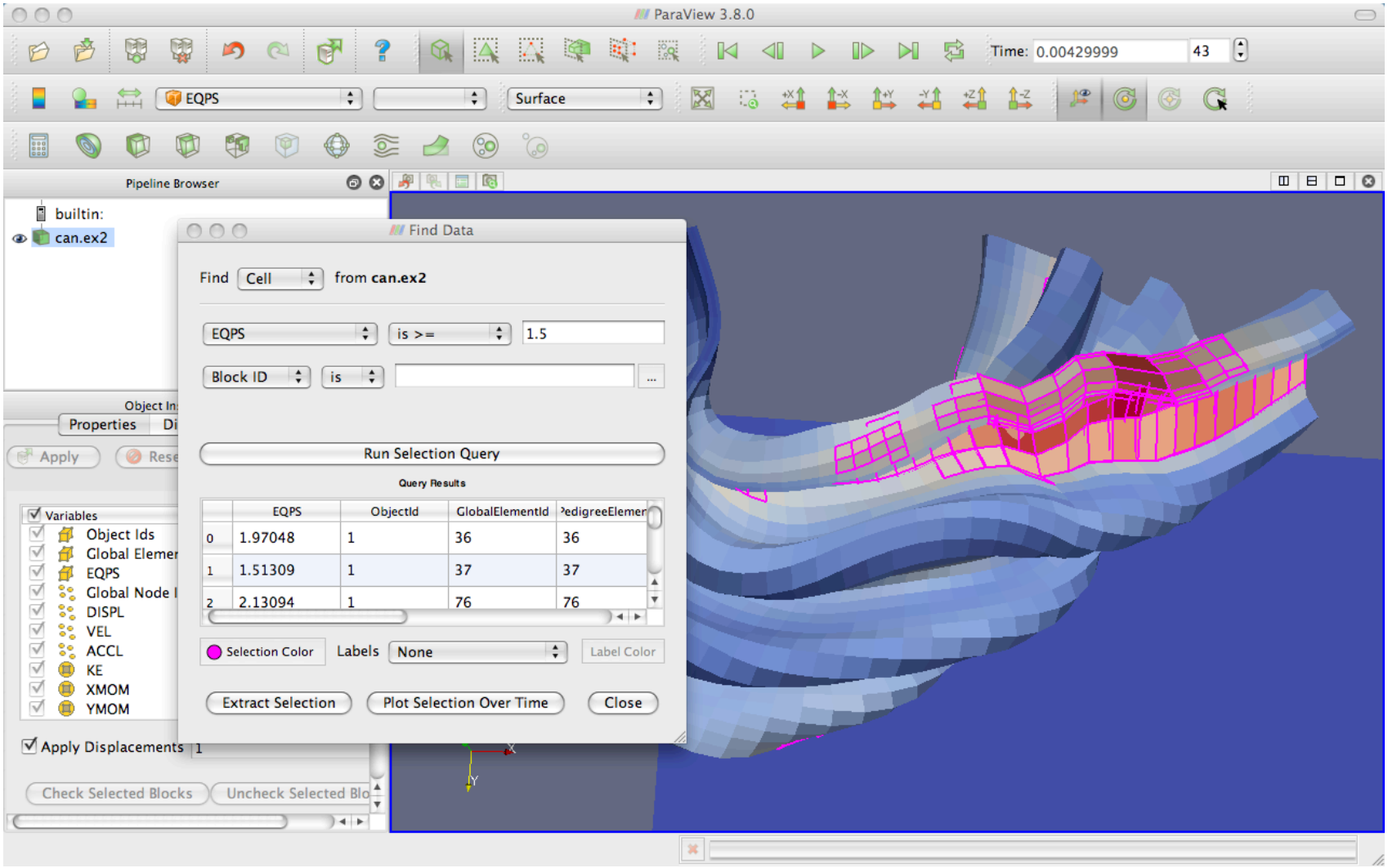


Query-Based Selection

1. Open can.ex2. All variables. 

2. Go to last time step. 
3. Edit → Find Data. 
4. Top combo box: find Cells.
5. Next row: EQPS, is >=, and 1.5.
6. Click Run Selection Query.

Query-Based Selection



Brush Selection


Surface Cell Selection
(shortcut: s)


Surface Point Selection
(shortcut: d)


Through Cell Selection
(shortcut: f)


Through Point Selection
(shortcut: g)


Select Cells (polygon)


Select Points (polygon)


Block Selection
(shortcut: b)

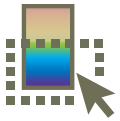

Interactively Select Cells


Interactively Select Points

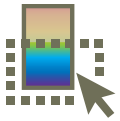

Hover Point Query


Hover Cell Query

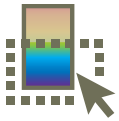
Selections

1. Open Find Data. 
2. Make various brush selections.
3. Observe results in the Find Data dialog box.
4. Play with the Invert Selection and Show Frustum options.

Adding Labels

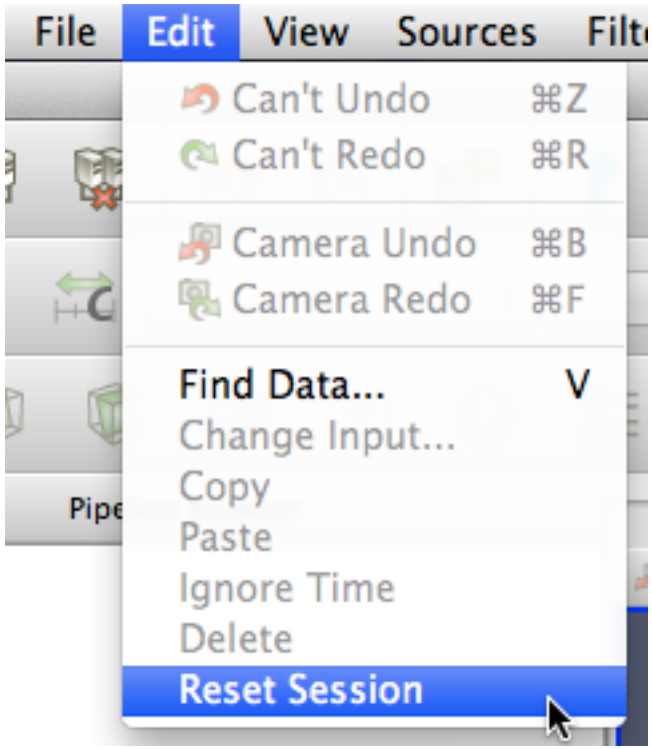
1. Go to the last time step.
2. Open Find Data. 
3. Create query Global ID is min. Click Run Selection Query.
4. In the Cell Labels chooser, select EQPS.

Adding Labels

1. Go to the last time step. 
2. Open Find Data. 
3. Create query Global ID is min. Click Run Selection Query.
4. In the Cell Labels chooser, select EQPS.
5. When you are done, turn off the EQPS labels.

Reset ParaView

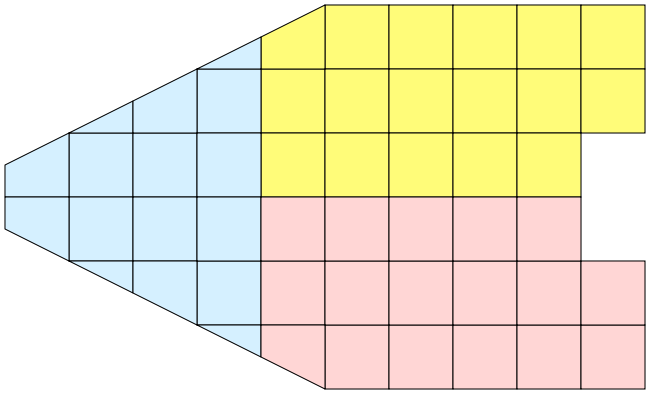
Edit → Reset Session



Visualizing Large Models

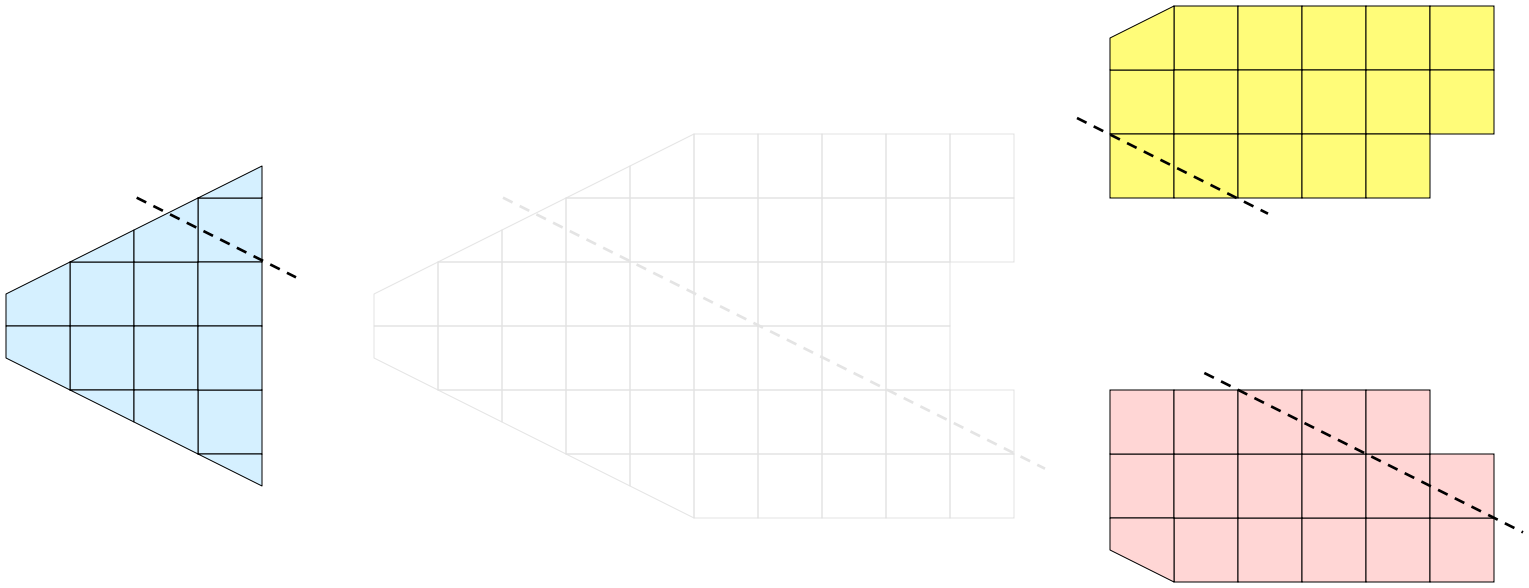
Data Parallel Pipelines

- Duplicate pipelines run independently on different partitions of data.



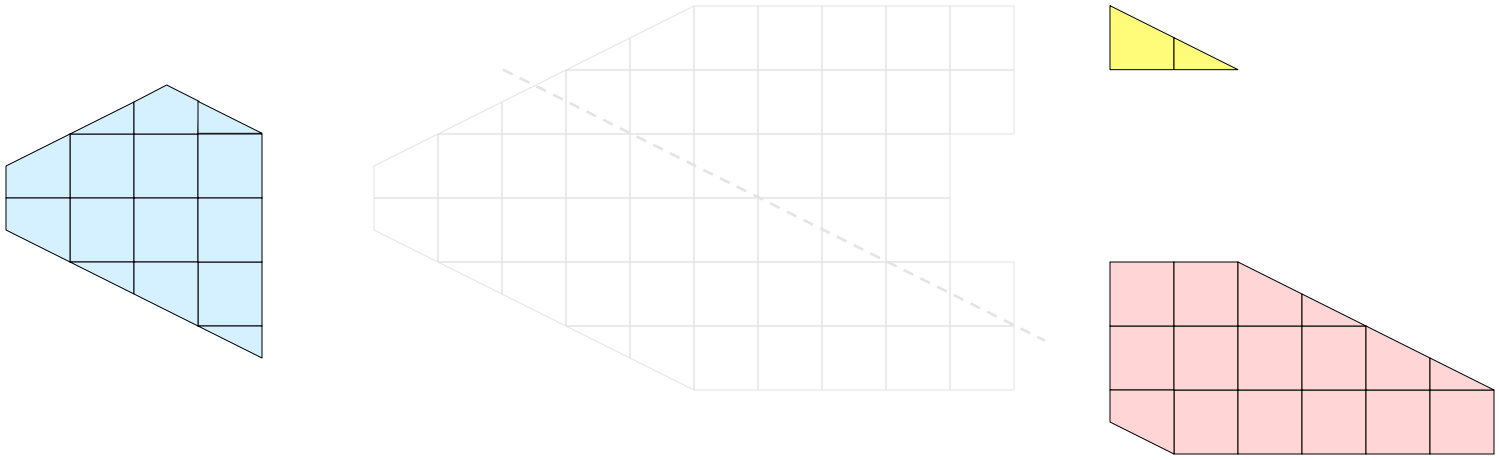
Data Parallel Pipelines

- Many operations will work regardless.
  - Example: Clipping.



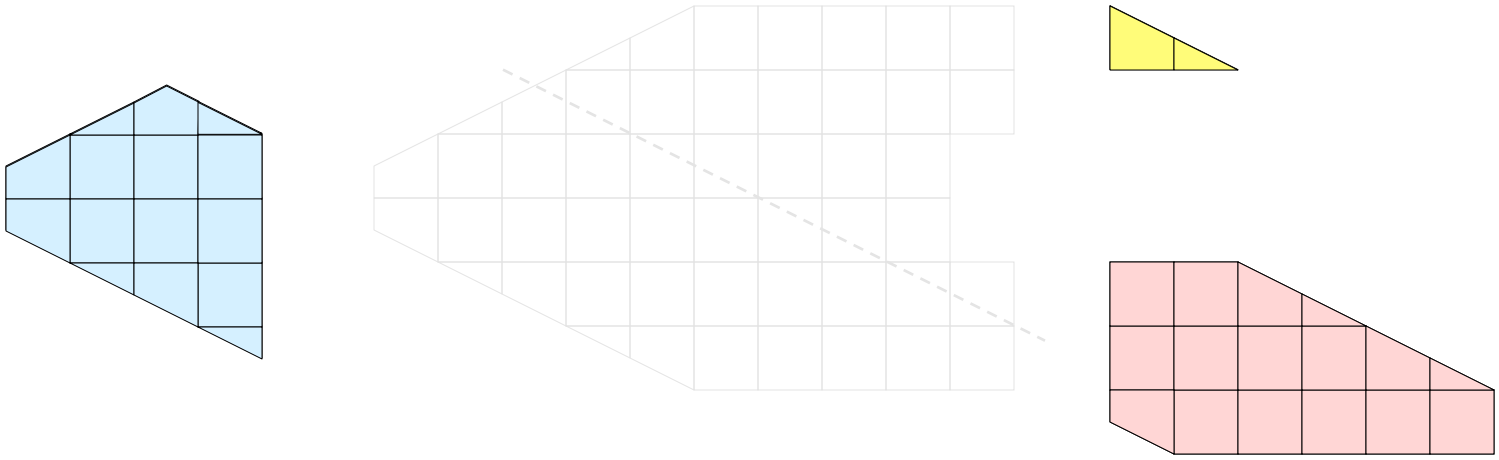
Data Parallel Pipelines

- Many operations will work regardless.
  - Example: Clipping.



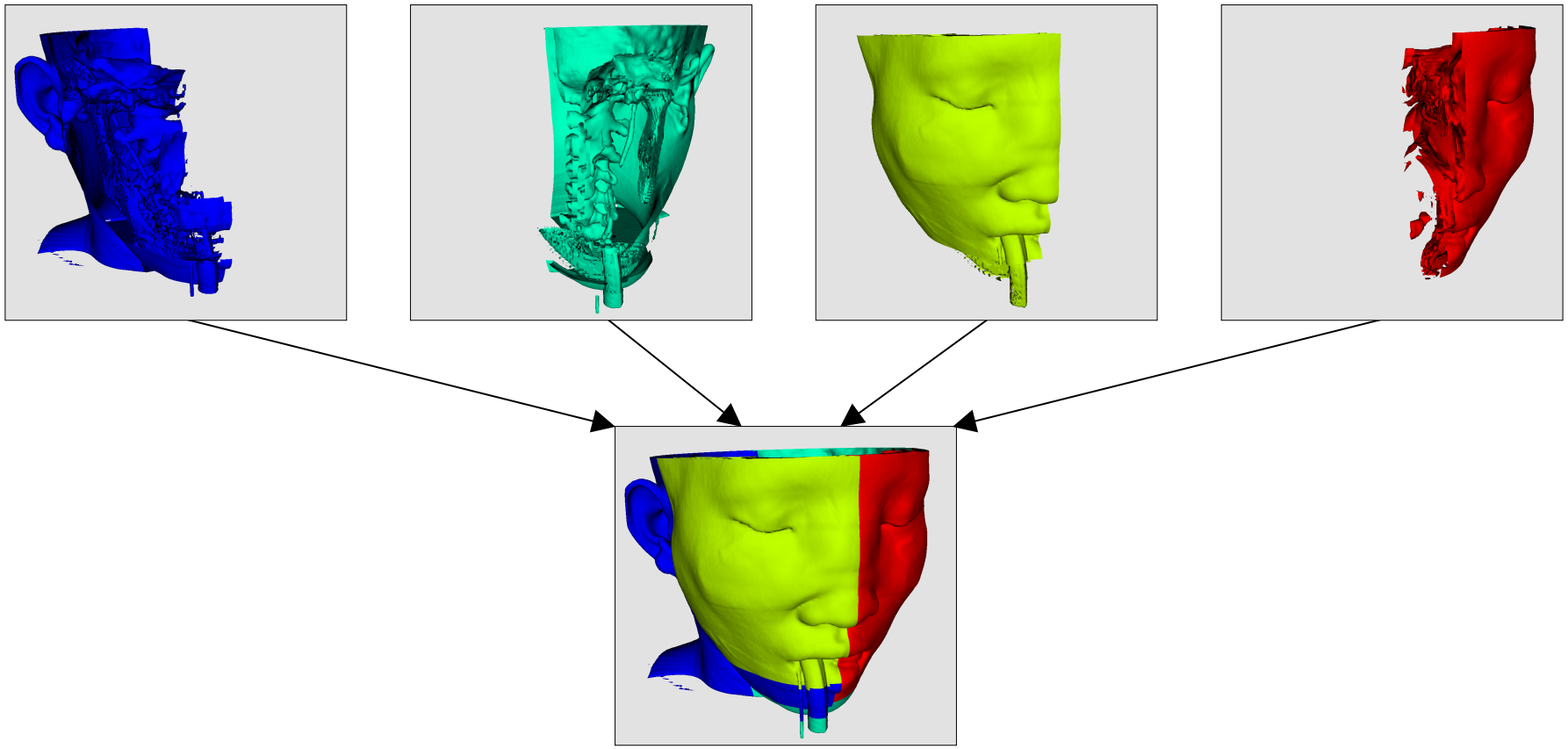
Data Parallel Pipelines

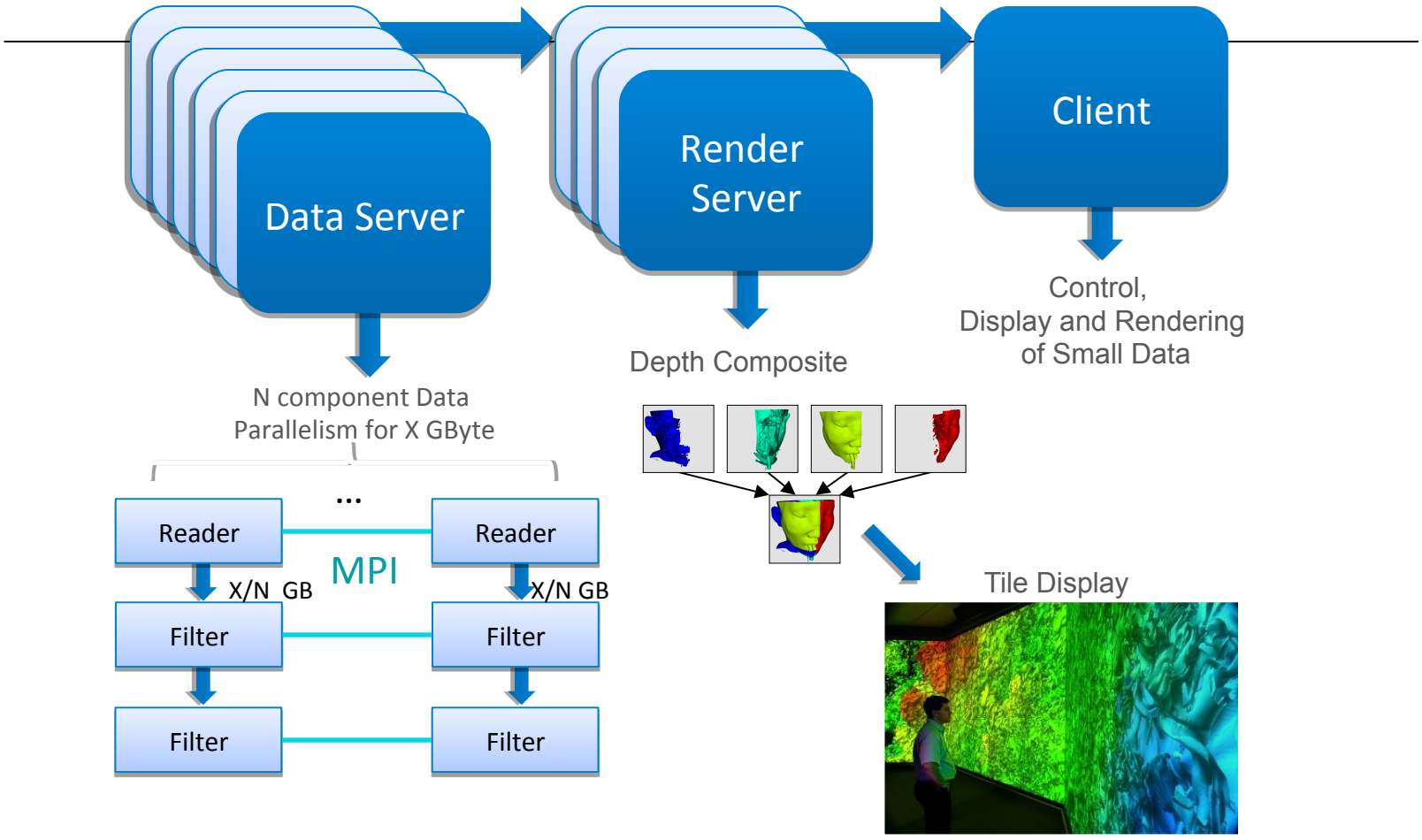
- Many operations will work regardless.
  - Example: Clipping.



- Will discuss those that don't later

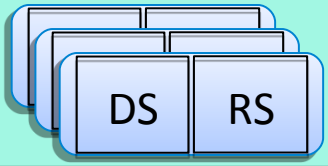
Parallel Rendering




Data Server
Render Server
Client
Control,
Display and Rendering
of Small Data
Depth Composite
N component Data
Parallelism for X GByte
Reader
Reader
MPI
X/N GB
X/N GB
Filter
Filter
Filter
Filter
Tile Display

ParaView's Running Modes

Builtin aka Standalone aka Serial		all components within one process (client may be GUI or pvpython) `paraview || pvpython`
Combined Server		data processing and parallel rendering in MPI job of combined processes. control from TCP connected client. `mpiexec -n x pvserver &; paraview` `#` or `pvpython` #+ Connect

Batch		server is an MPI job which directly runs a python script
		`mpiexec -n x pvbatch \`
		`vis_script.py`

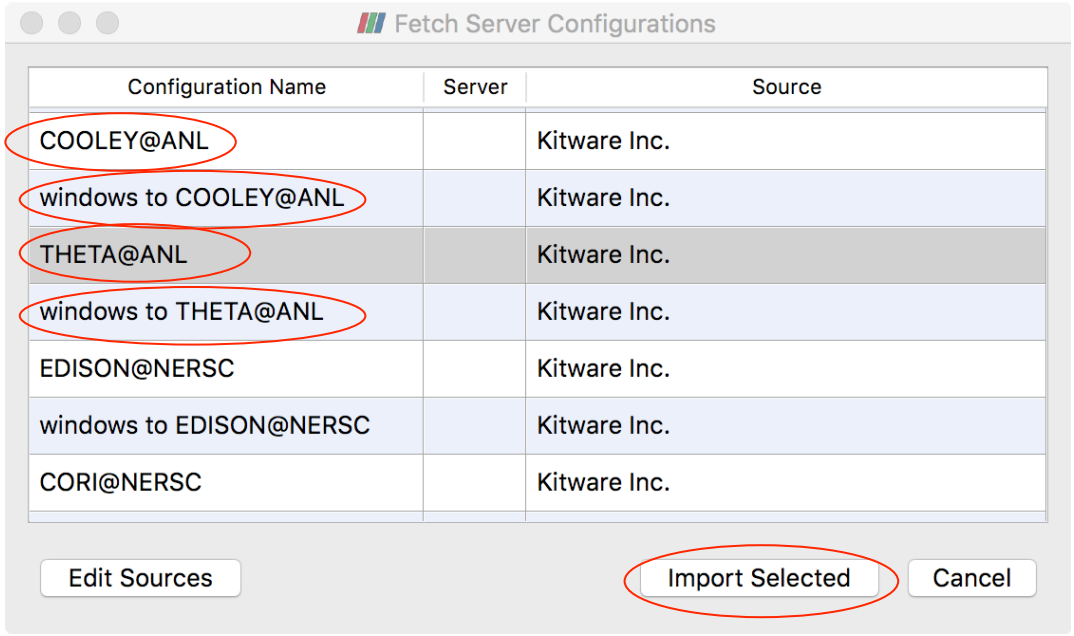
DS = data server

RS = render server

Kitware

Fetch Server Configuration

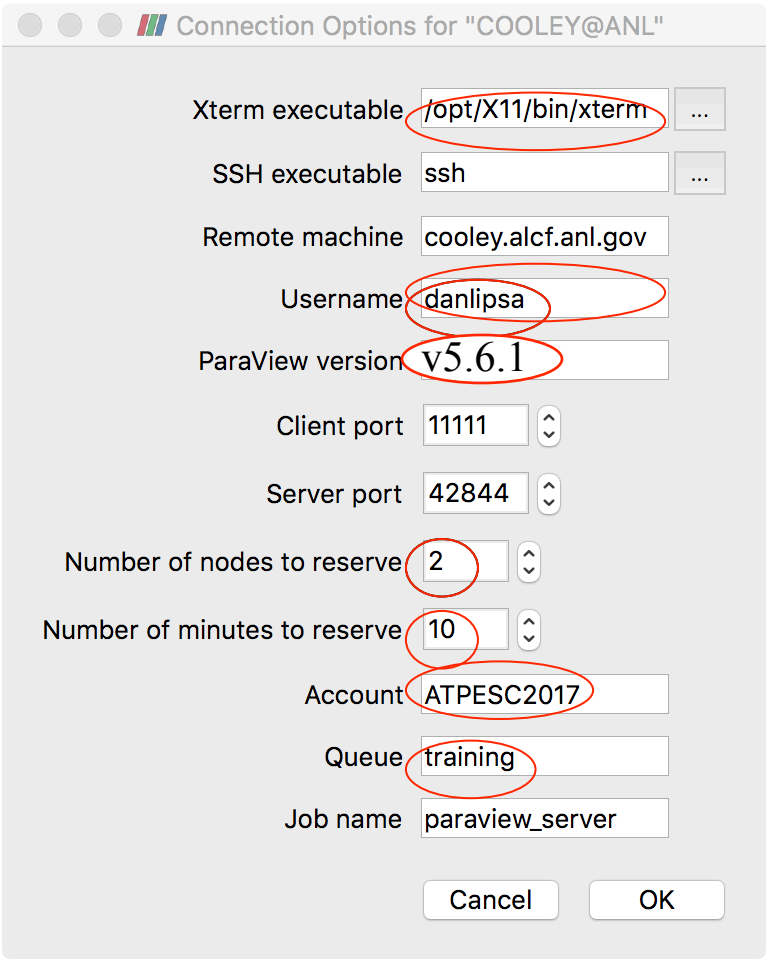
- File > Connect > Fetch Servers



Note: Bug in PV 5.6.0 on Mac. Workaround fetch w/ any other version to fetch then connect as normal. or: https://www.paraview.org/files/pvsc

Connect from Unix/Mac

Mac Os: Install Xquartz

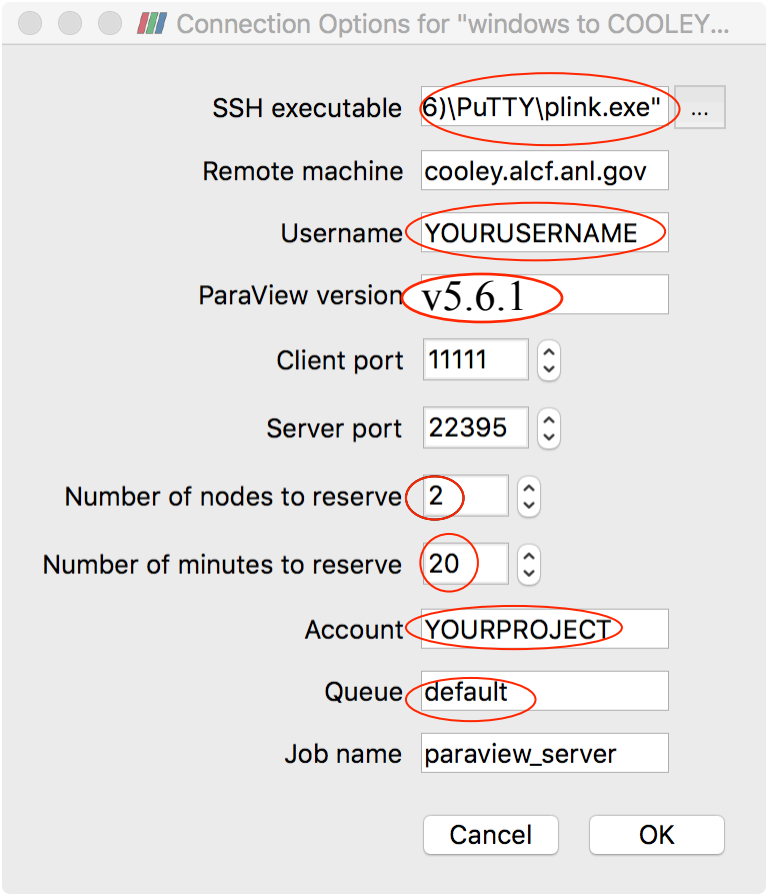


Queue:

- cooley R.ATPESC2019_0805_1
- theta R.ATPESC2019

Connect from Windows

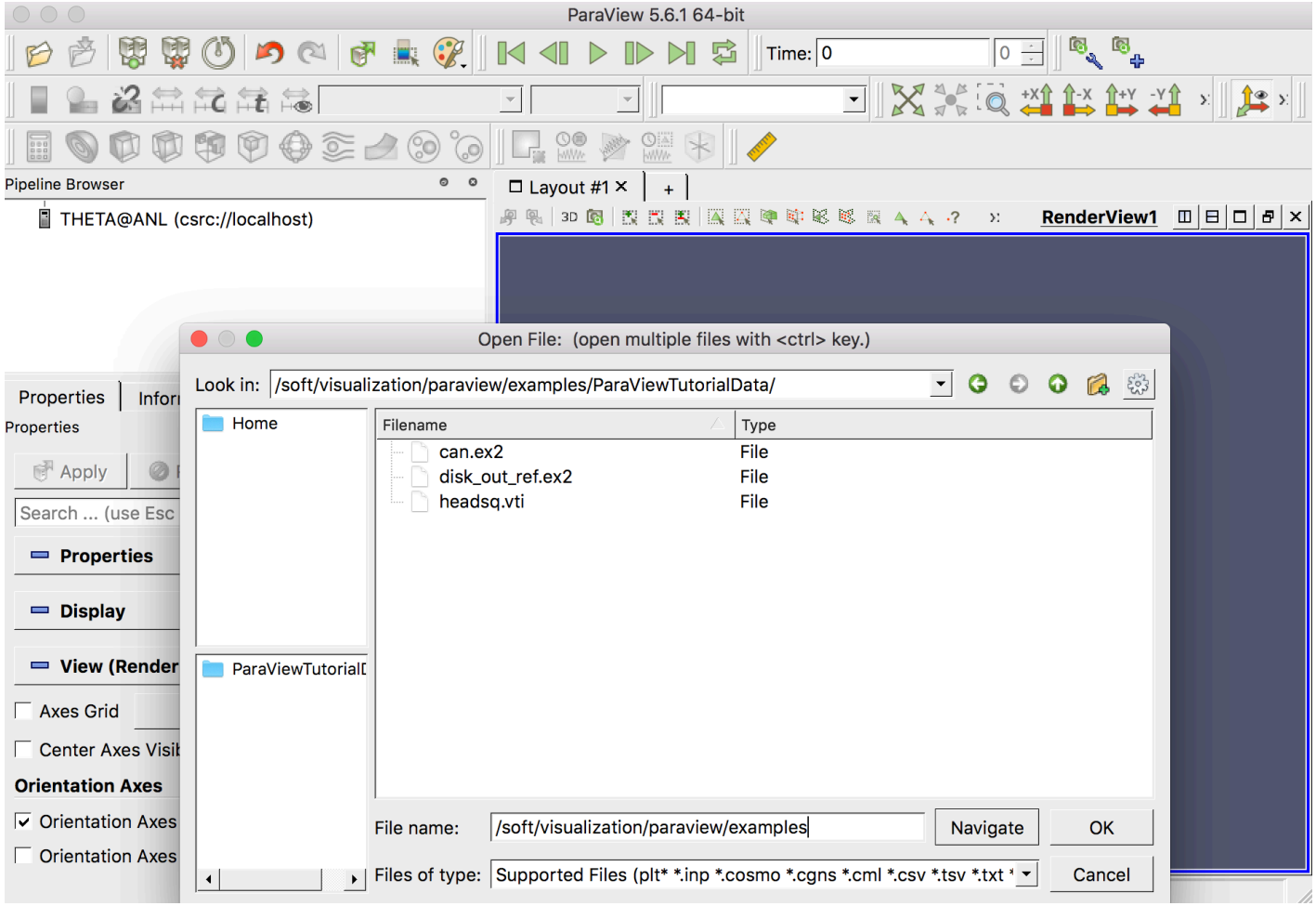
Windows: Install PuTTY



Queue:

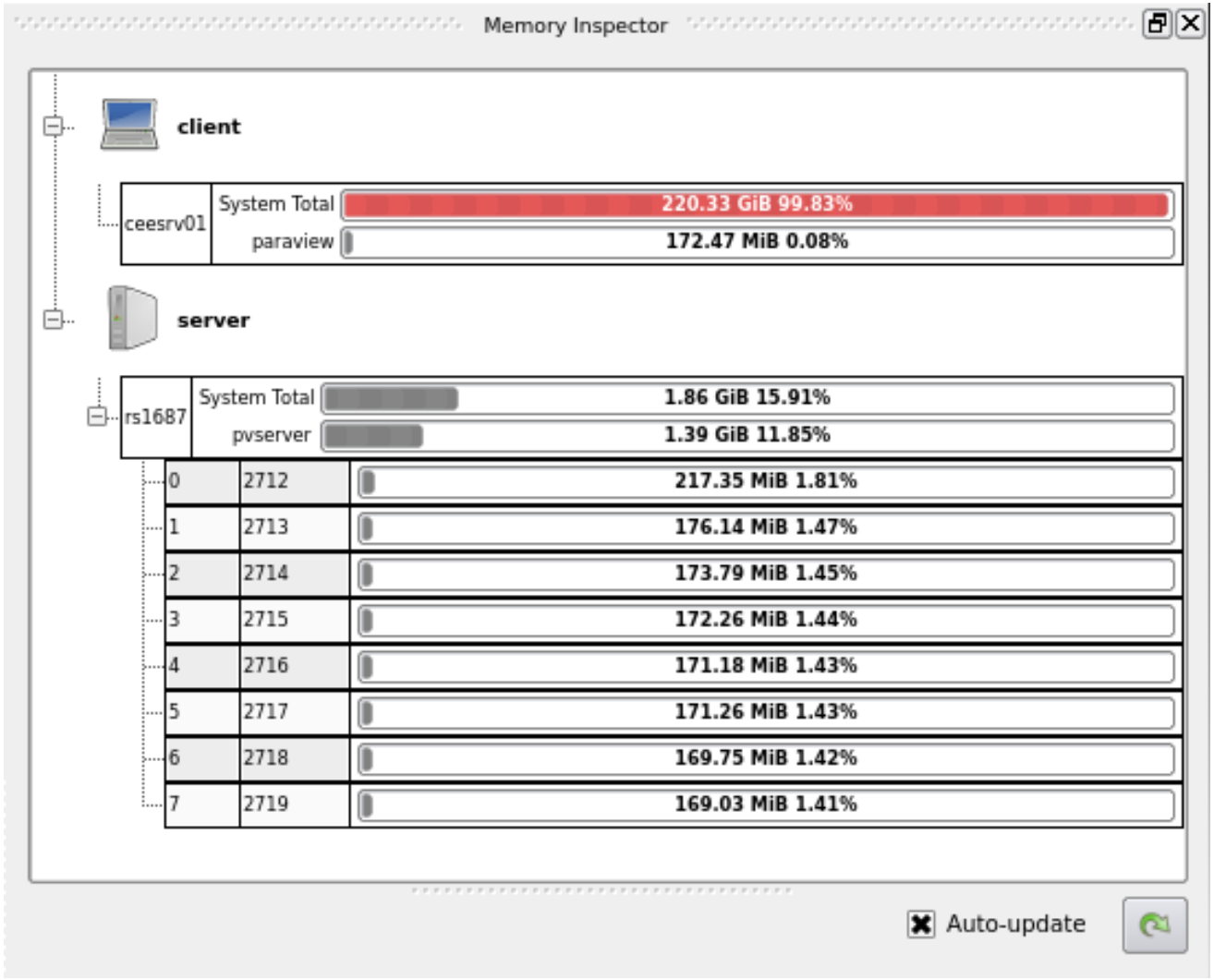
- cooley R.ATPESC2019_0805_1
- theta R.ATPESC2019

File->Open browse remote disks



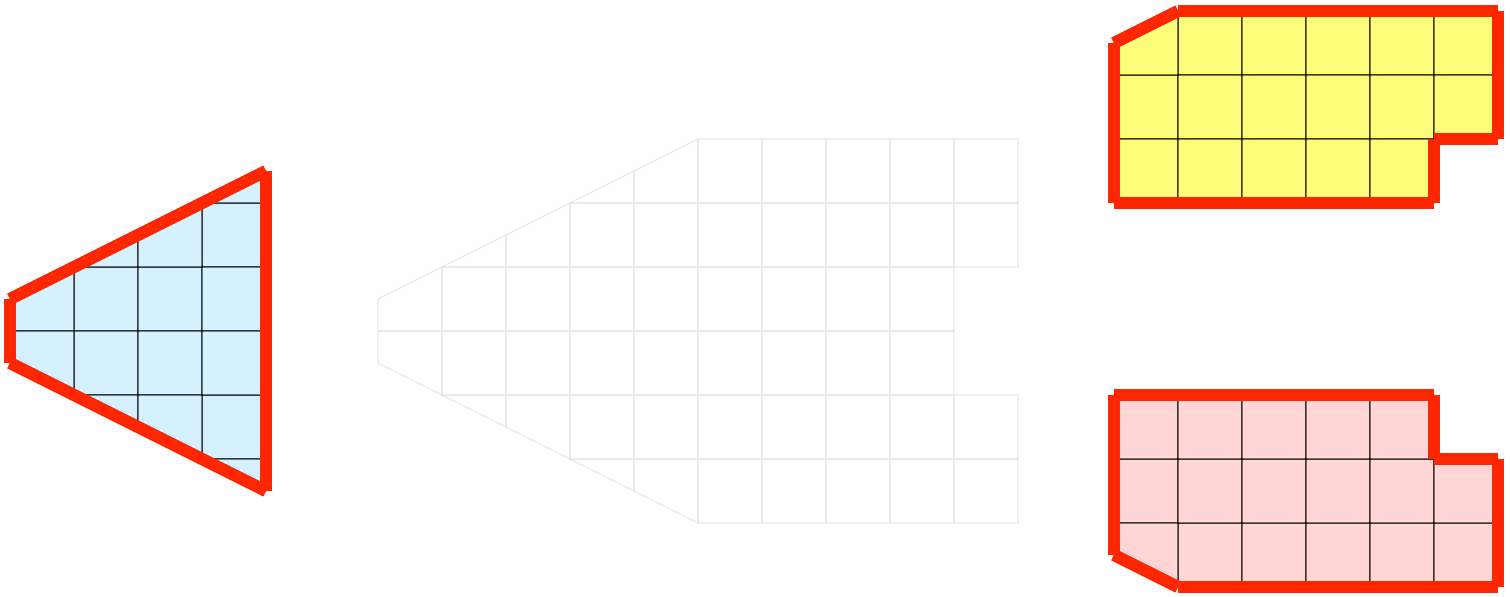
/soft/visualization/paraview/examples/

Memory Inspector



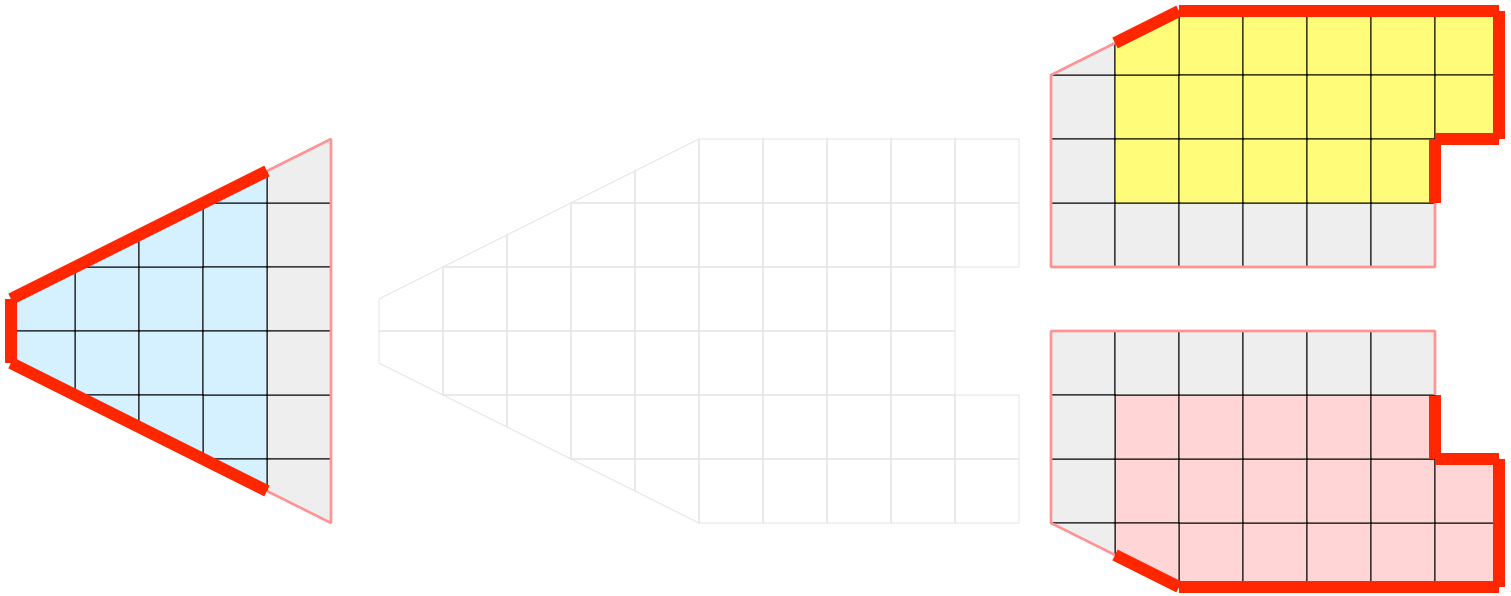
Advanced Data Parallel Pipelines

- Some operations will have problems.
  - Example: External Faces



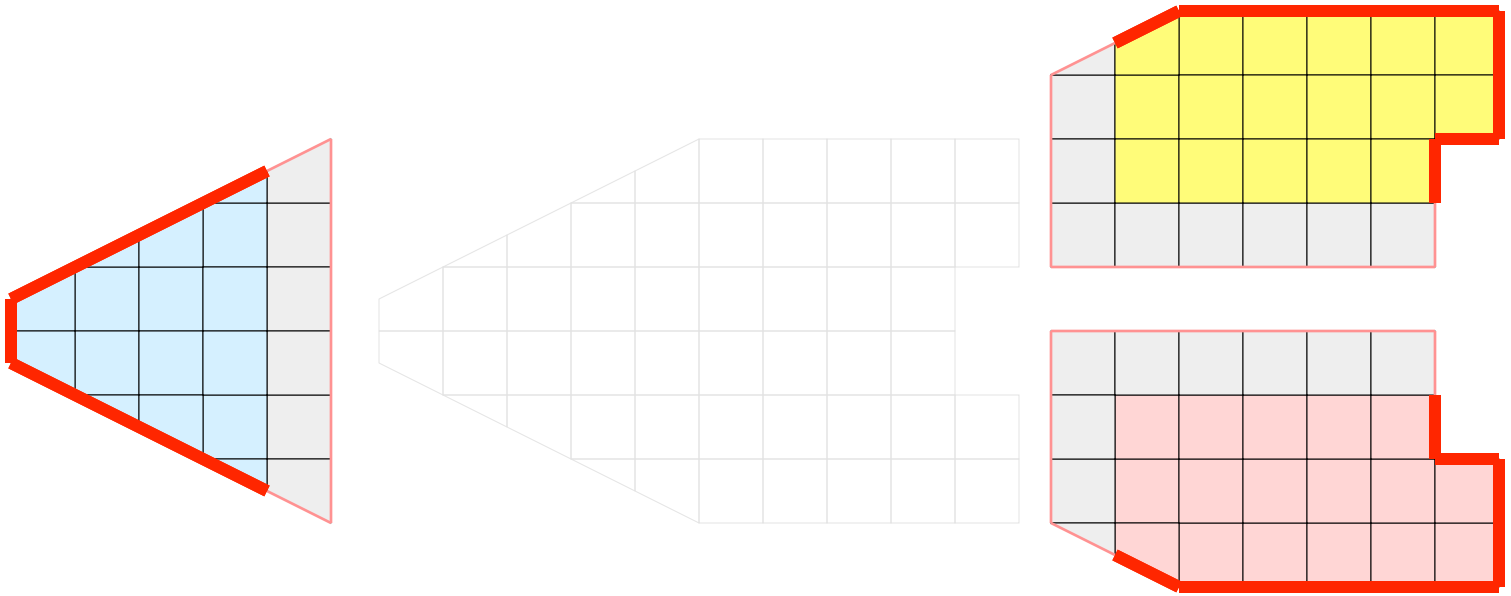
Advanced Data Parallel Pipelines

- Ghost cells can solve most of these problems.



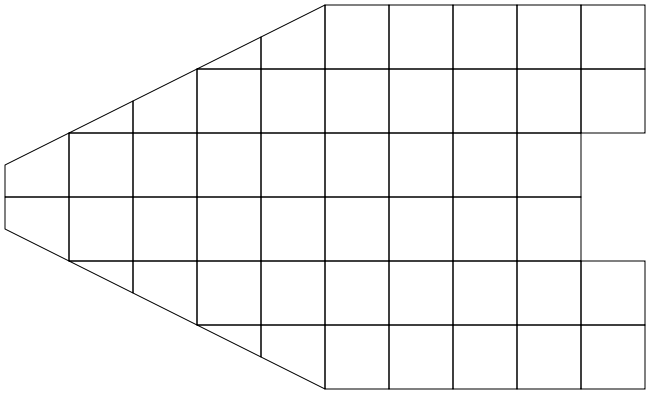
Advanced Data Parallel Pipelines

- Ghost cells can solve most of these problems.



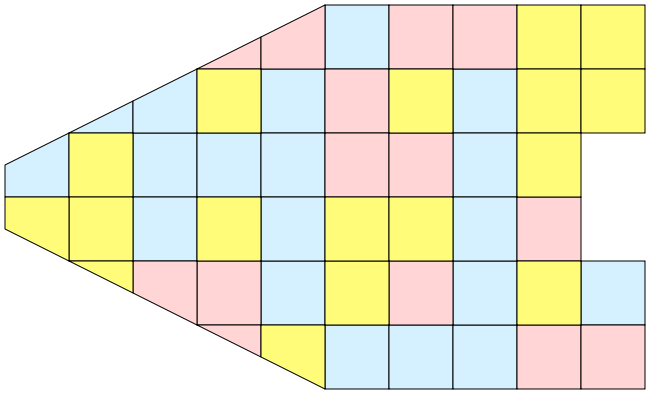
Data Partitioning

- Partitions should be **load balanced** and **spatially coherent**. Why?



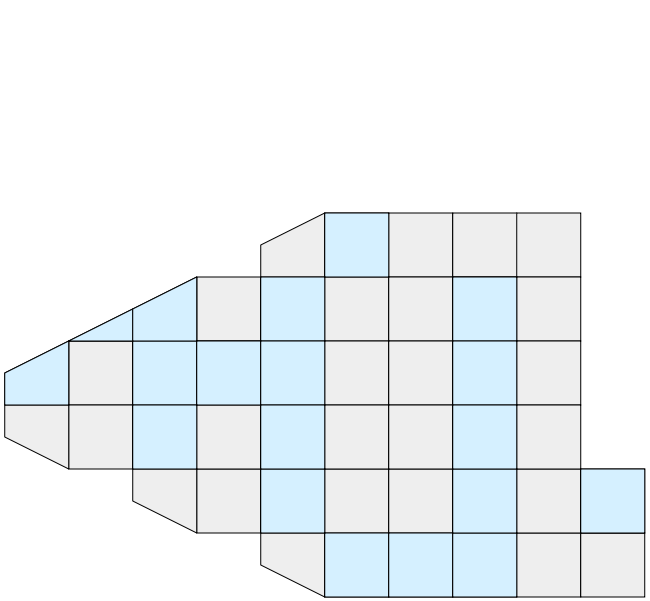
Data Partitioning

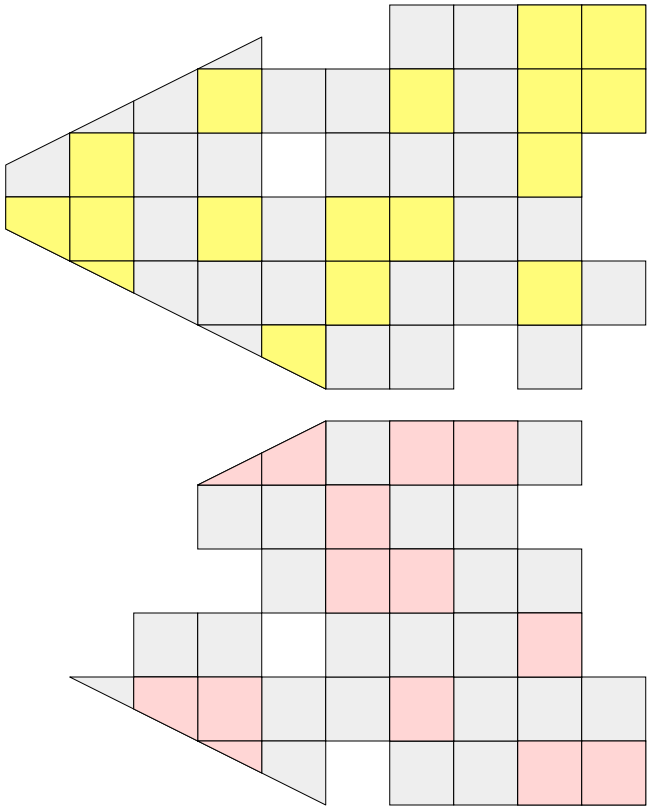
- Partitions should be **load balanced** and **spatially coherent**.



Data Partitioning

- Partitions should be load balanced and spatially coherent. A random partition with ghost cells will replicate the entire dataset on all nodes.

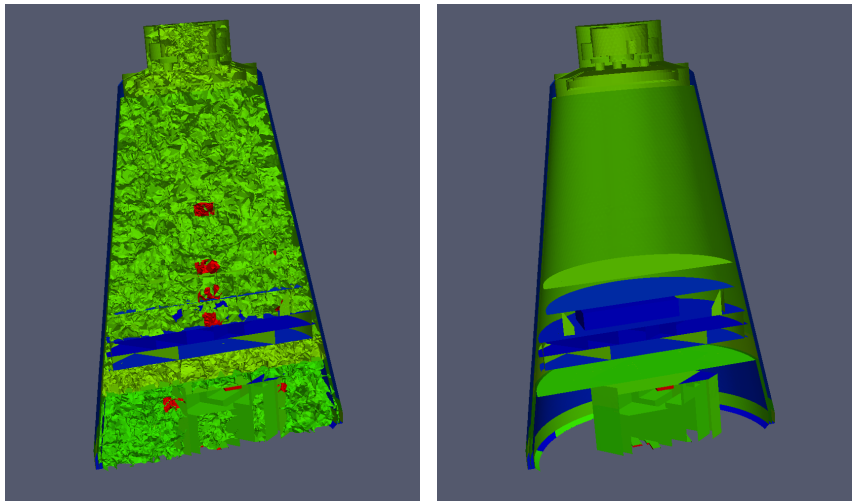




Load Balancing/Ghost Cells

- Automatic for Structured Meshes.
- Partitioning/ghost cells for unstructured is “manual.”
- Use Ghost Level Generator to create
- Legacy option: D3. Also repartitions

Extract Surface without ghost cells



Extract Surface after D3

ParaView Catalyst

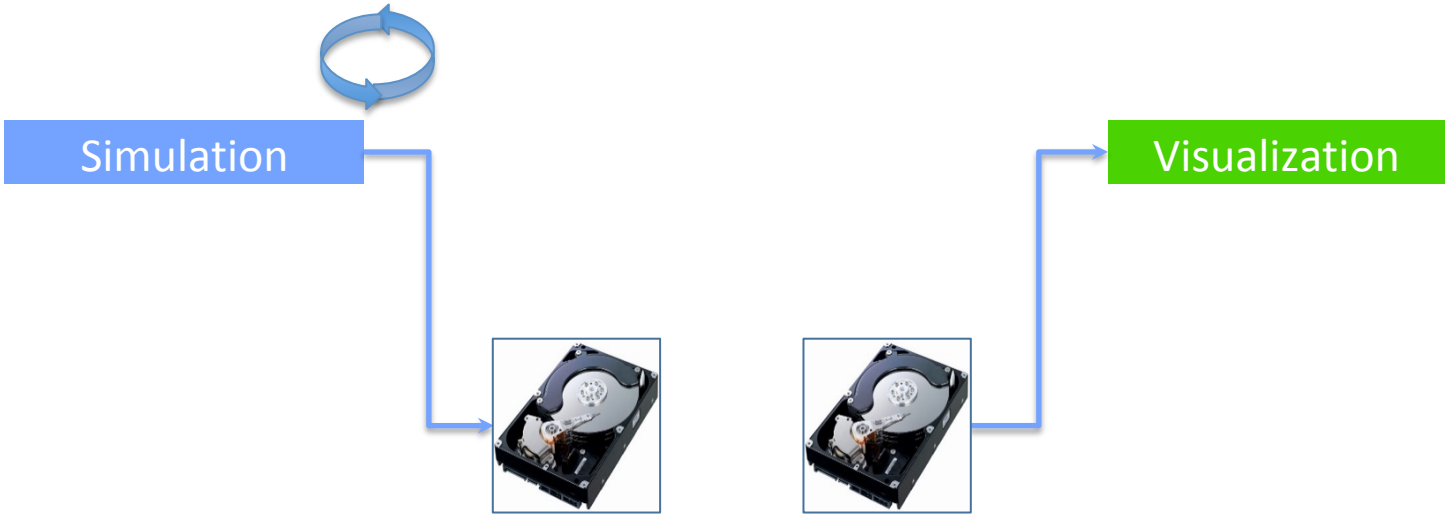
- ParaView as a library

  Anything ParaView can do, simulation can too.

- Efficient, because largely eliminate IO

- UI can be as simple recording a python script

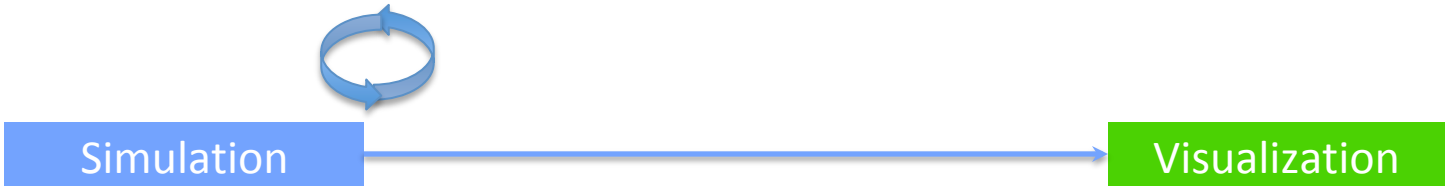
Traditional/Post Hoc Analysis



Simulation runs in a loop, infrequently saving out large files to slow disk.

Sometime later, user manually loads files into an analysis package to inspect the data.

In Situ Analysis



Embed analysis library into the simulation code. Simulation periodically makes calls to generate on-the-fly results.

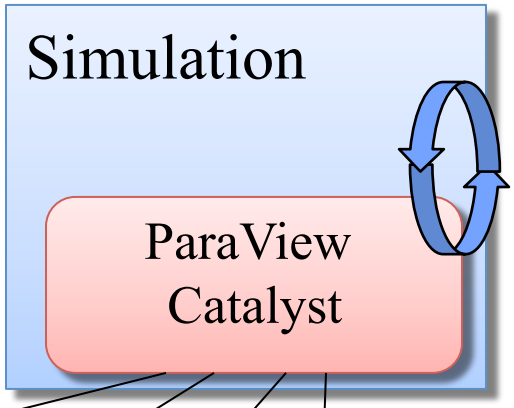
Largely avoids I/O, therefore increases achievable temporal granularity.

Define pipeline and data products ahead of time, then viable path to exascale.

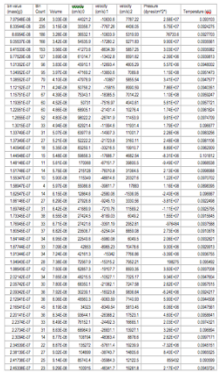
In Situ Analysis and Visualization

See Catalyst Menu

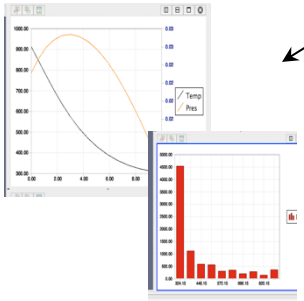
- Define Exports
- Export Catalyst Script



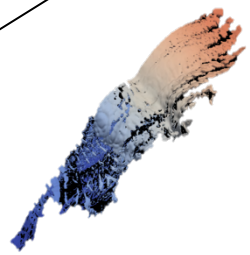
Output
Processed
Data



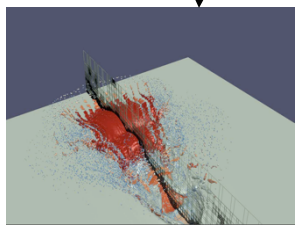
Statistics



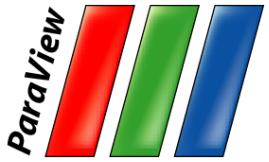
Series Data



Polygonal Output
with Field Data



Rendered Images



Thank you!

Questions?

is (always) hiring