FourFlow Documentation

Christoffer Green

July 2012

Contents

Frequently Asked Questions 29

Introduction

FourFlow is an application for visualization and quantification of data from time-resolved three-directional phase contrast magnetic resonance velocity mapping (4D PC-MR). It is based on the Paraview software and has many similarities with it. The documentation for ParaView is complementary to the documentation for FourFlow and it is also distributed with FourFlow.

Recommended Workflow

When we created FourFlow we envisioned a typical workflow as the user first having access to time resolved image data with velocity mapping in the Ensight Gold format and having an intention of visualizing this data in various ways. The recommended way of converting other types of data to the Ensight Gold format is with the FourFlow export plugin in the Segment software but the Ensight Gold format is well documented so it is possible to implement custom data exporters in other software. The next step after having the data in the right format is to import it into FourFlow and apply various visualization filters to it. It is at this point also possible to do some simple quantification.

Importing Data

Data to be imported into FourFlow must be in the Ensight Gold format. The .case files that are to be imported should all be in the same directory. The naming rules for the different files are as follows (the * character signifies that any string of characters are allowed here):

- *Velocity Volume*: should be named velocity*.case and should contain a 3-value vector variable with the name V_3D_01 that contains the velocity data.

- *SAX Stack*: A stack of SAX images should be named sax*.case

- *Endo Surface*: A file with endo surface geometry should be named *Endo.case

- *Epi Surface*: A file with epi surface geometry should be named *Epi.case

- *RVEndo Surface*: A file with reverse endo surface geometry should be named *RVEndo.case

- *RVEpi Surface*: A file with reverse epi surface geometry should be named *RVEpi.case

- *Two Chamber Image Plane*: A two Chamber image plane should be named 2CH*.case

- *Three Chamber Image Plane*: A three Chamber image plane should be named 3CH*.case

- *Four Chamber Image Plane*: A four Chamber image plane should be named 4CH*.case

When importing data via the Import Data button on the main toolbar in the FourFlow GUI the user is presented with a folder browser dialog where the directory containing the data files is to be selected. When such a directory has been selected another dialog is shown where the user can select what files to import from that directory.

The FourFlow Graphical Interface

Main Window
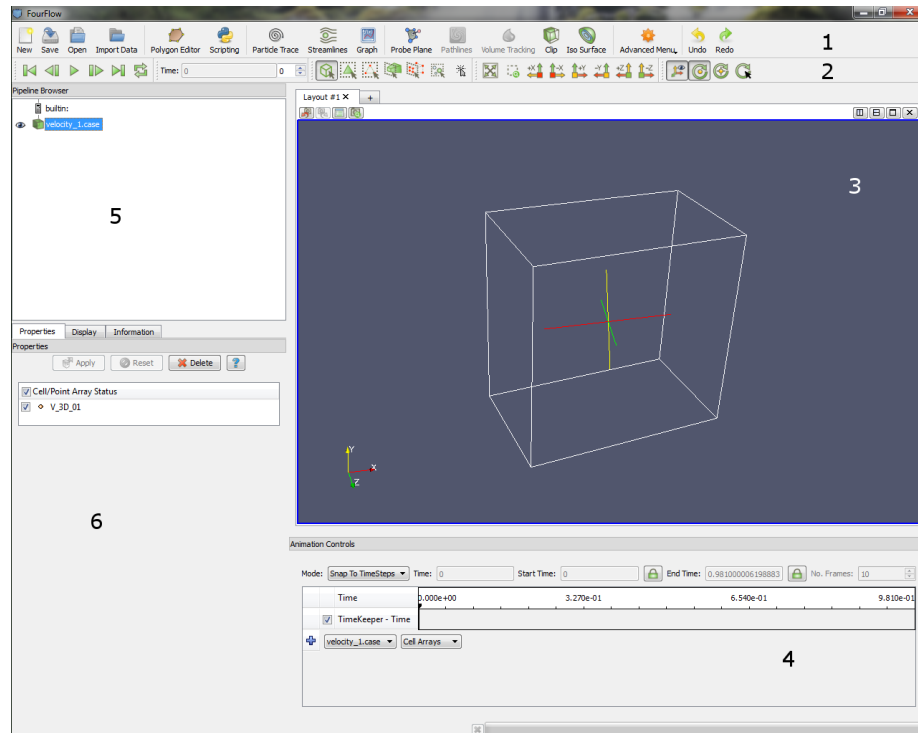
The above image shows the FourFlow GUI as it appears when one volume data set has been loaded, the volume can be seen as a white outline in the 3d view. The default state of the graphical interface of FourFlow consists of 6 sections:

1. Main Toolbar

2. Secondary Toolbar

3. The 3D View

4. Pipeline Browser

5. Animation Controls

6. Properties Editor

Main Toolbar
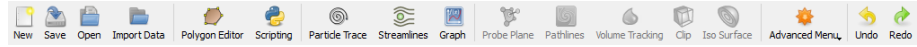
The main toolbar has a number of different functions grouped together by type. Some of these functions are explained in more detail in other parts of this document.

- New: Resets the application to the default state, removing all data currently loaded.

- Save: Saves a copy of the current scene with all the applied filters.

- Open: Open a previously saved scene.

- Import Data: Imports data in Ensight format from a directory.

- Polygon Editor: Is used on the currently selected Probe Plane to display a new window that can be used to create a 2d segmentation of data from a 3d volume.

- Scripting: Shows a window from where python scripts can be written, run and loaded.

- Particle Trace: Applies a particle trace visualization to the currently selected polygon segment in a velocity volume.

- Particle Collector: Enables quantification of how many particles go through polygons (collectors) created with the polygon editor. Since it supports multiple polygons and shows the meassurements in a graph the results the difference in polygon throughput can be seen.

- Streamlines: Applies a streamline visualization to the currently selected polygon segment in a velocity volume.

- Graph: Applies a filter to the currently selected polygon segment that opens a window with quantification information from the velocity data on the volume segment.

- Probe Plane: Adds a widget that can be moved around, rotated and scaled to show a slice of the velocity data in a volume. This can then be used with the Polygon Editor for more advanced visualizations.

- Pathlines: Similar to streamlines but adds lines between the different timesteps that a particle goes through instead of in a single timestep.

- Volume Tracking: Is applied to volume tracking data to visualize the flow of a user defined volume over time.

- Clip: Can be used to clip away unwanted parts of a data set.

- Iso Surface: Displays a polygon shell of a specific magnitude value in a velocity volume.

- Advanced Menu: Contains extra functionality from ParaView, consult the ParaView manual for further information.

Secondary Toolbar

The secondary toolbar has buttons for controlling animation playback, selection, view direction and orientation display. There is more detail on what these controls do in the ParaView documentation.

- The first group of green buttons act like controls on media players, they can be used to rewind, step through and play animations.

- To the right of the green buttons there is a control that is used to set the current time.

- After that there are buttons that control what selecting items in the view does, here you can switch between selecting objects, faces on polygons, cells in volumes among others.

- The next group of buttons controls the position and orientation of the camera and can be used to for example align the camera with the x-axis of the scene.

- The last group pf buttons control how the 3d orientation indicators in the scene are displayed.

Pipeline Browser
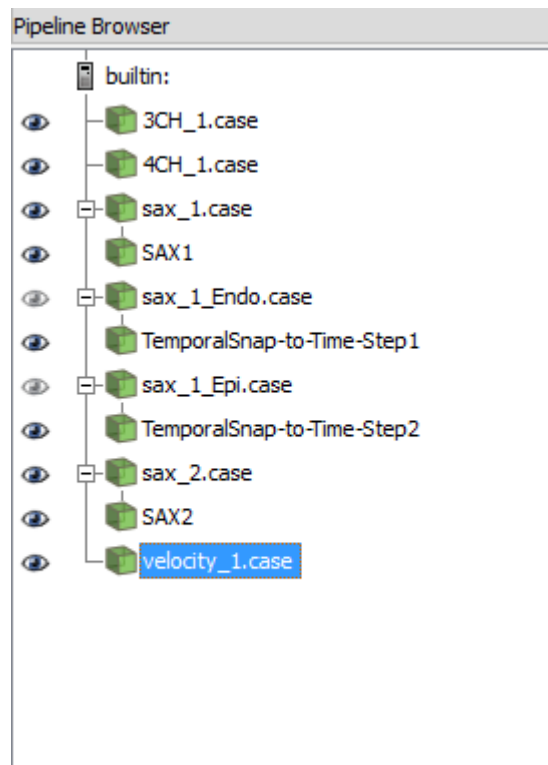
The pipeline browser shows the currently loaded data and the filters that are applied to that data. The eye icon to the left of the nodes control the visibility of that node in the currently active window. For example in the image of the pipeline browser that can be seen above we have a stack of images loaded in the 'sax_1.case' node and that node is connected to a node called 'SAX1', this node extracts and shows one slice of this image stack. A relationship between nodes is indicated by the existence of a '-' button next to the parent node that can be used to hide the child chain of nodes. Nodes can be renamed by selecting them and pressing the keyboard key F2. Nodes can be deleted by selecting them and pressing the 'delete' keyboard key.

Properties Editor

The properties editor is split up into two different tabs.

Properties Tab   The properties tab is used to change node properties and for changing how the currently selected node is shown in the 3d view. In the following image it is displaying the settings for a volume node which has controls for changing what data to display and how to display it.
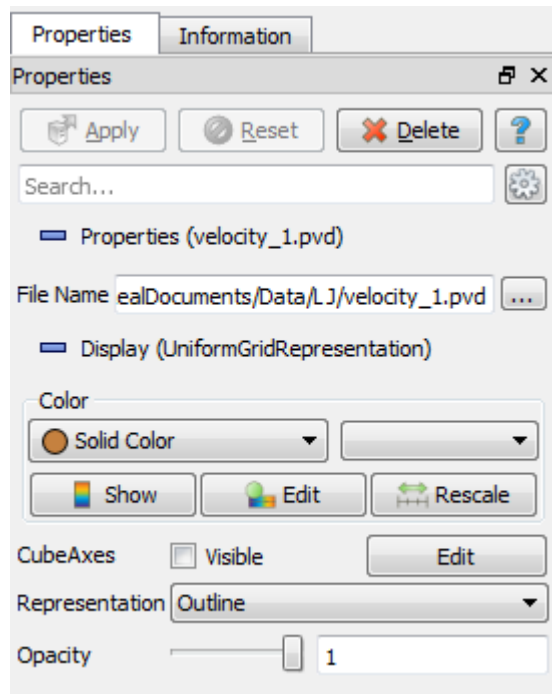
Information Tab The information tab shows information about the data set in the currently selected node. It can show the extent of the data, what type of data it is, how many points or how much ram it takes up.

Animation Controls

The animation controls are used to give a graphical indication of what the current timestep is. It can also be used to insert keyframes on properties for animation.
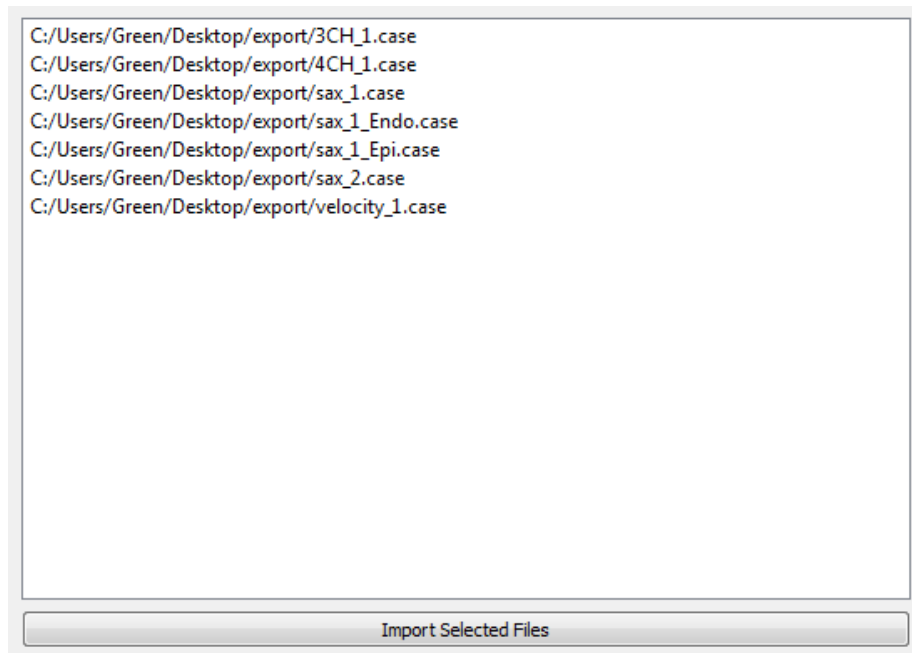
Other Windows
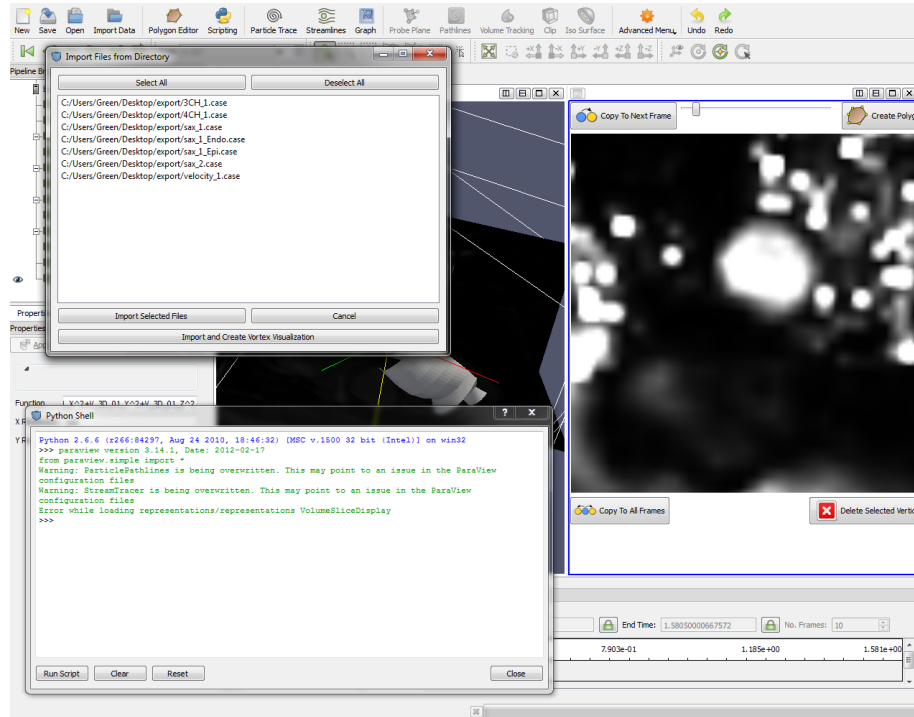
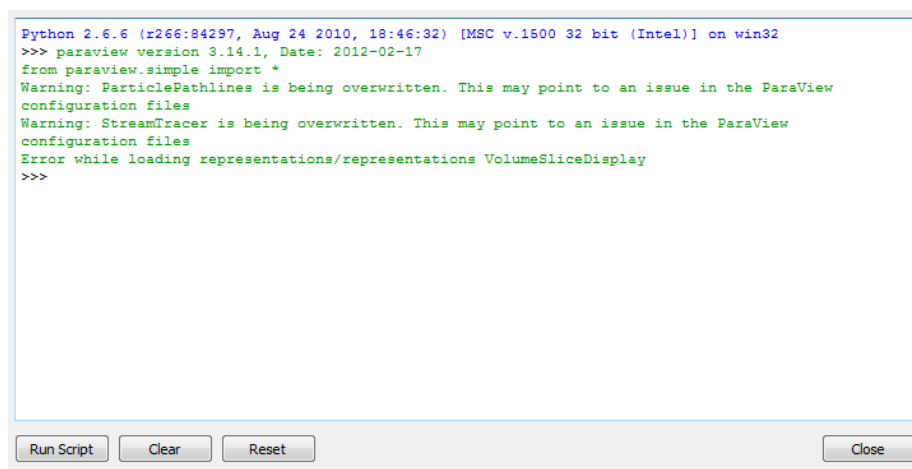There are a couple of other windows as well in FourFlow, they are:

1. Import Files Window: Used when importing data from a directory.

2. Scripting Window: Used for python scripting

3. Polygon Editor Window: Used to create polygon segmentations of volume data.
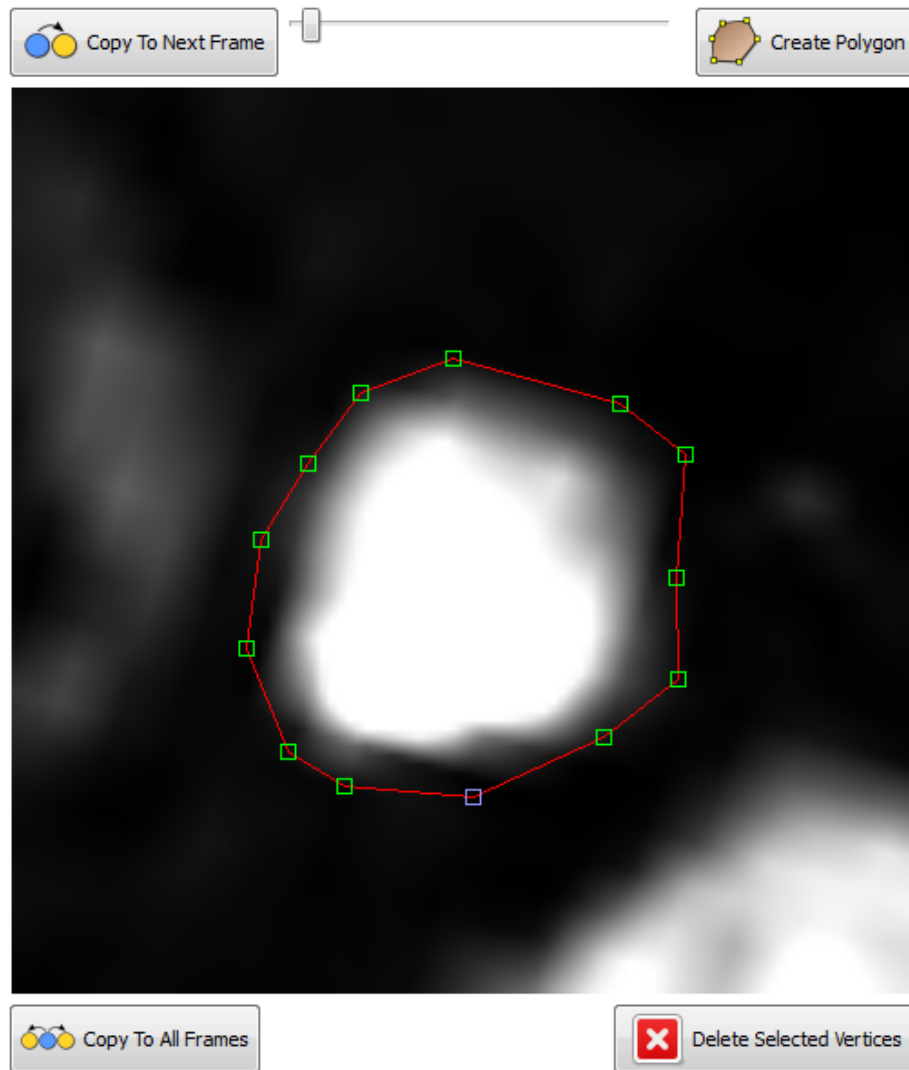
Import Files Window

```
C:/Users/Green/Desktop/export/3CH_1.case
C:/Users/Green/Desktop/export/4CH_1.case
C:/Users/Green/Desktop/export/sax_1.case
C:/Users/Green/Desktop/export/sax_1_Endo.case
C:/Users/Green/Desktop/export/sax_1_Epi.case
C:/Users/Green/Desktop/export/sax_2.case
C:/Users/Green/Desktop/export/velocity_1.case
```

Import Selected Files

This window is used to select Ensight .case files from a specified directory.

Scripting Window

```
Python 2.6.6 (r266:84297, Aug 24 2010, 18:46:32) [MSC v.1500 32 bit (Intel)] on win32
>>> paraview version 3.14.1, Date: 2012-02-17
from paraview.simple import *
Warning: ParticlePathlines is being overwritten. This may point to an issue in the ParaView
configuration files
Warning: StreamTracer is being overwritten. This may point to an issue in the ParaView
configuration files
Error while loading representations/representations VolumeSliceDisplay
>>>
```

Run Script   Clear   Reset   Close

The python script window is used to run and write python scripts.
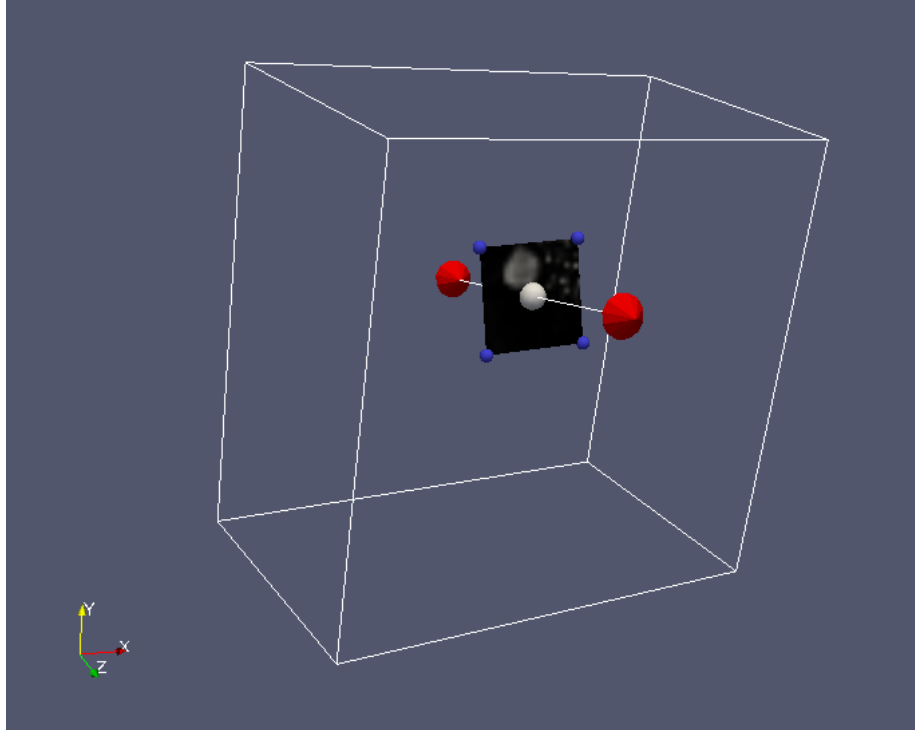
Polygon Editor Window

The polygon editor window is used to create a 2d segmentation of a velocity volume.

Features

Probe Plane

A probe Plane is used to select an area in a volume for the Polygon Editor tool to place a polygon into. It shows a preview of the values surrounding the Probe plane on its surface.


To use a Probe Plane a volume containing velocity data must be selected in the Pipeline Browser. Clicking the Probe Plane icon in the main toolbar creates a Probe Plane in the middle of the volume. The Probe Plane is a square surface with coloration indicating the intensity or magnitude of the velocity data from the volume it is in.
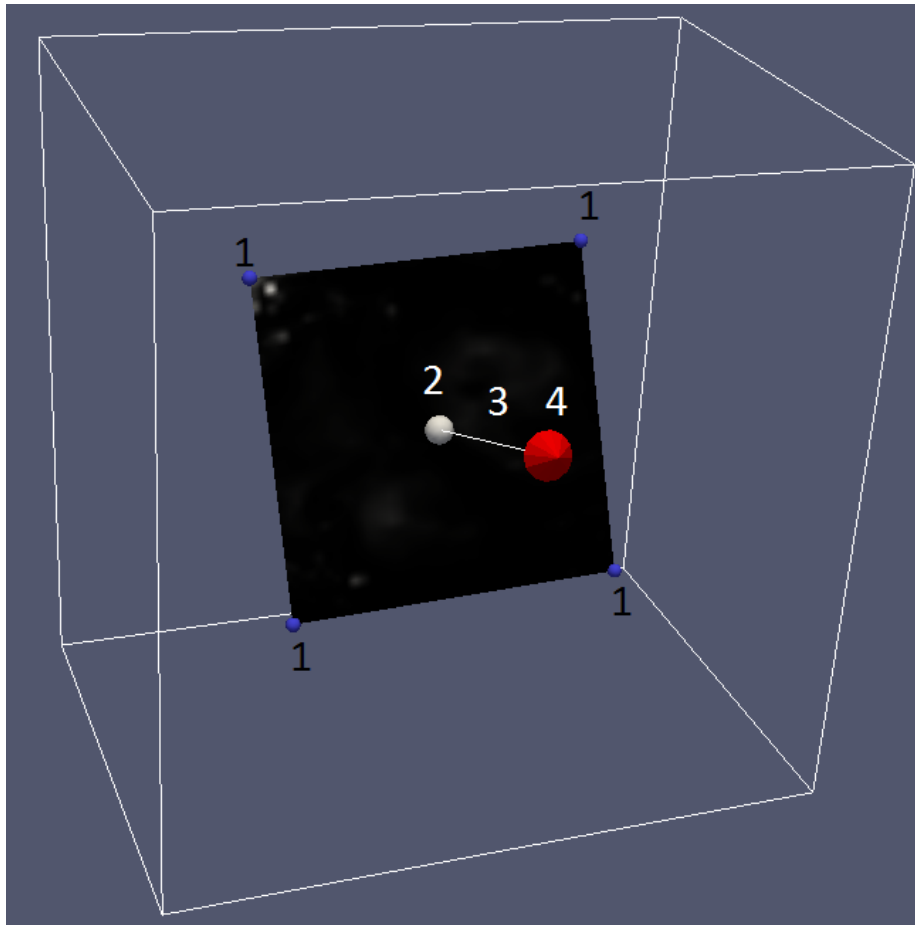
A Probe Plane can be manipulated by clicking and dragging on the 4 different widgets on its surface that are visible when the Probe Plane is selected in the Pipeline Browser. These 4 widgets that can change the position, size and rotation of the Probe Plane are as follows:

1. Resizers: These 4 widgets can be moved to resize the Probe Plane.

2. Center Mover: This widget moves the Probe Plane around within the plane of the current camera.

3. Normal Mover: This widget moves the Probe Plane within it's normal direction.

4. Rotator: This widget rotates the Probe Plane around it's center.

Polygon Editor
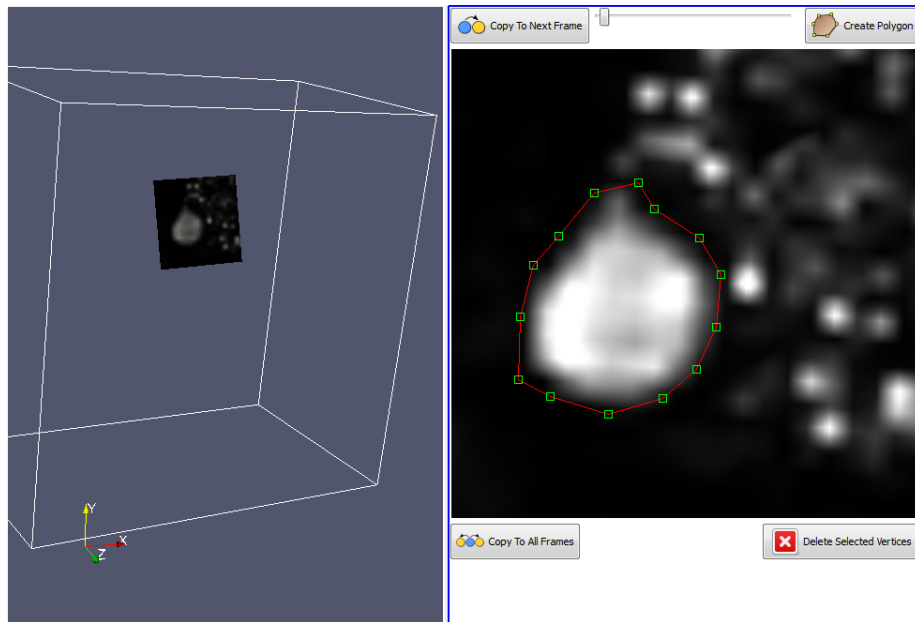
When quantifying blood flow velocities, creating particle trace animations or pathlines a surface region is needed that can act as en emitter or as a measurement region. To create such a region the polygon editor tool in FourFlow is used.
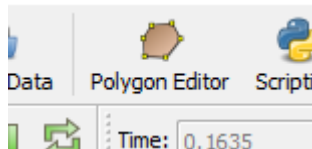

Before launching the polygon editor it is recommended that a Probe Plane has already been created and positioned. The polygon editor is launched by clicking the 'Polygon Editor' button on the main toolbar. This splits the 3d view in two and shows the Polygon Editor as a window with a representation of the probe plane surrounded by several buttons. The following list describes what the buttons do.

- **Copy To Next Frame**: Copies the polygon in the current time frame to the next time frame and overwrites the previous data there.

- **Copy To All Frames**: Copies the polygon in the current time frame to all the other time frames and overwrites any data that was already there.

- **Delete Selected Vertices**: Removes the currently selected vertices in the current time frame.

18

- **Create Polygon**: Creates a polygon from the data in the polygon editor within the current probe plane.

- **Intensity Slider**: This slider changes the intensity of the image on screen, it does not change any data.

To start creating a polygon simply left click in the probe plane visualization view and a vertex will be added, left click again and another vertex connected to the previous one will be added and so on. It is possible to select already added vertices by left clicking them and by holding down the left mouse button and dragging a selection rectangle over several vertices at once. A selected vertex or group of vertices can be moved around by left click-dragging them when the mouse cursor is over them. The polygon is always created in the current time frame, to get to the next time frame make sure the polygon editor window is the currently active window and use the keyboard arrow keys to go to the next or previous frames.

The image below shows the result of pressing the 'Create Polygon' button.
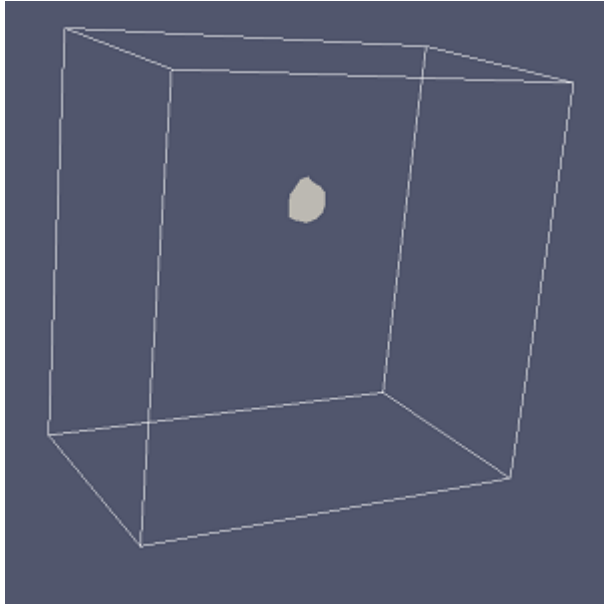
Particle Trace
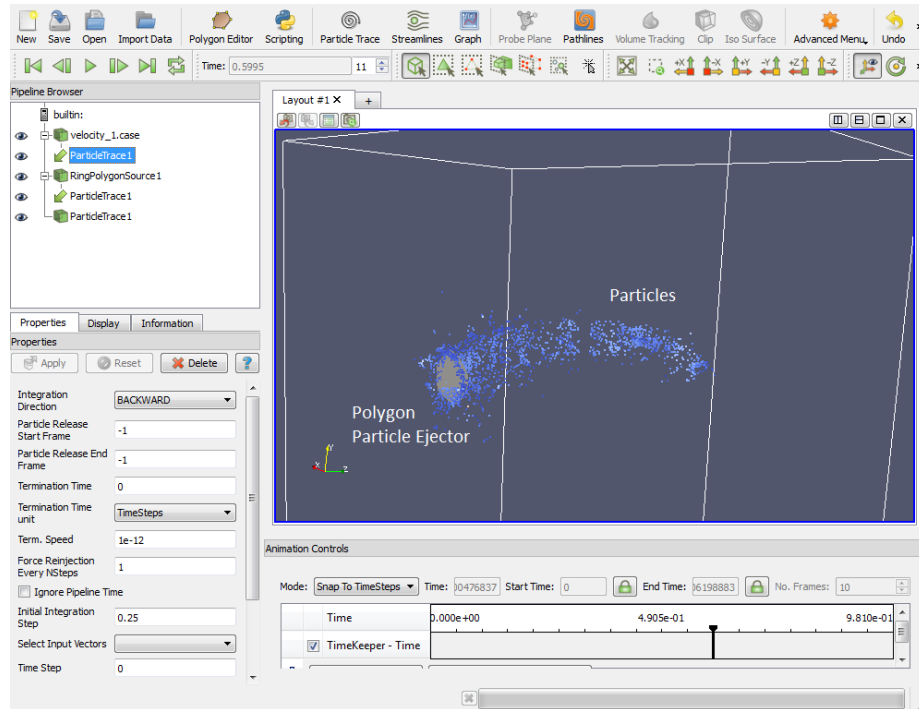
Particle tracing works in FourFlow by ejecting particles into a velocity volume from a polygon ring created via the polygon editor. The position of these particles are changed on every timestep based on the velocity data in the volume. The workflow for creating a particle trace visualization is as follows:

1. Import a volume with velocity data and select its node in the Pipeline Browser.

2. Click the Probe Plane icon and position the probe plane to where you want to eject particles.

3. Click the Polygon Editor button to open the polygon editor window and give an outline of the area of interest, if the area of interest should eject particles from every timestep make sure that every timestep has a region of interest in the polygon editor.

4. Create the polygon from the region of interest by clicking the Create Polygon button in the Polygon Editor.

5. Click the Particle Trace button to set up a particle trace node network.

The attributes that can be changed on a particle trace node are the following:

- Integration Direction: Determines what the direction of the movement of the particles should be through the volume.

- Particle Release Start Frame: Sets the first frame in which particles should be emitted from the polygon particle emitter. If set to -1 then particles are emitted from the first possible frame. (this is currently unimplemented)

- Particle Release End Frame: Sets the first frame in which particles should end being emitted from the polygon particle emitter. If set to -1 then particles are never stopped being emitted. (this is currently unimplemented)

- Termination Time: Sets the lifetime of individual particles.

- Termination Time Unit: Sets how the Termination Time attribute should be interpreted.

- Term. Speed: If particles have a speed below this value then they are removed from the simulation.

- Force Reinjection Every NSteps: Sets how often new particles should be injected from the polygon ejector.

Particle Collector

Enables quantification of how many particles go through polygons (collectors) created with the polygon editor. Since it supports multiple polygons and shows the meassurements in a graph the results the difference in polygon throughput can be seen.

To create a particle collector graph first there must be a particle a particle trace simulation already set up. The particle collectors are created as normal polygons with the polygon editor on probe planes, there is no upper limit to how many are supported.

When clicking the Particle Collector button on the main toolbar there is a window that pops up asking for inputs and sources. The particle trace simulation node is the input and one or more polygons are the sources.

The resulting graphs show the number of vertices that the polygons have as an identification of what polygon matches to what graph line. To tell how many vertices a polygon has select it's node and switch to the Information tab, at the top there is a "Number of points:" field that tells how many there are.
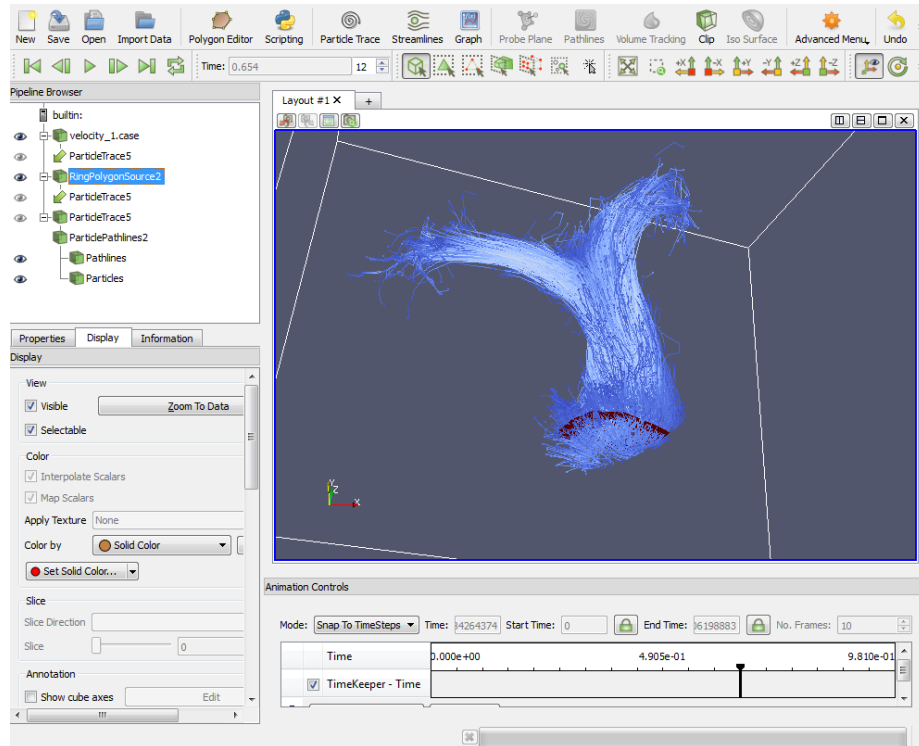
Streamlines
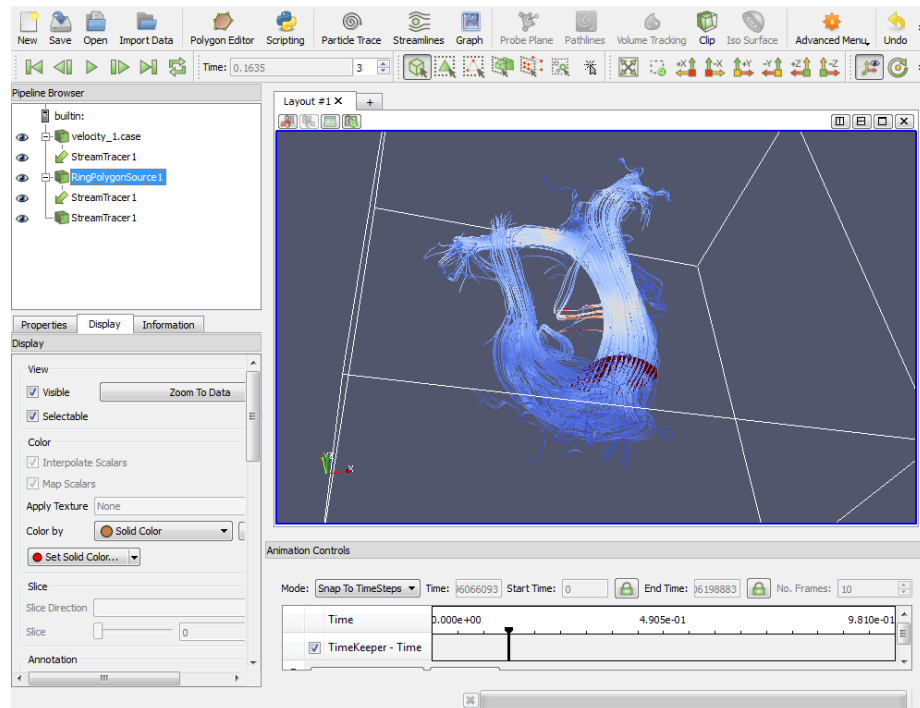
Streamlines are very similar to particle tracing and works in FourFlow by ejecting

particles into a velocity volume from a polygon ring created via the polygon editor. The lines in a streamline are created with the values in the velocity data volume at the current timeframe. The workflow for creating a streamline follows:

1. First import a volume with velocity data and select its node in the Pipeline Browser.

2. Click the Probe Plane icon and position the probe plane to where you want to eject streamlines.

3. Click the Polygon Editor button to open the polygon editor window and give an outline of the area of interest, if the area of interest should eject streamlines from every timestep make sure that every timestep has a region of interest in the polygon editor.

4. Create the polygon from the region of interest by clicking the Create polygon button in the Polygon Editor.

5. Click the Streamline button to set up a streamline node network.

Pathlines

Pathlines are lines that connect the individual particles in a particle trace

simulation, creating a visualization that looks similar to streamlines but unlike streamlines are calculated over the entire timestep range. To create pathlines first create a particle trace simulation and then click on the pathlines button.

Flow Velocity Quantification
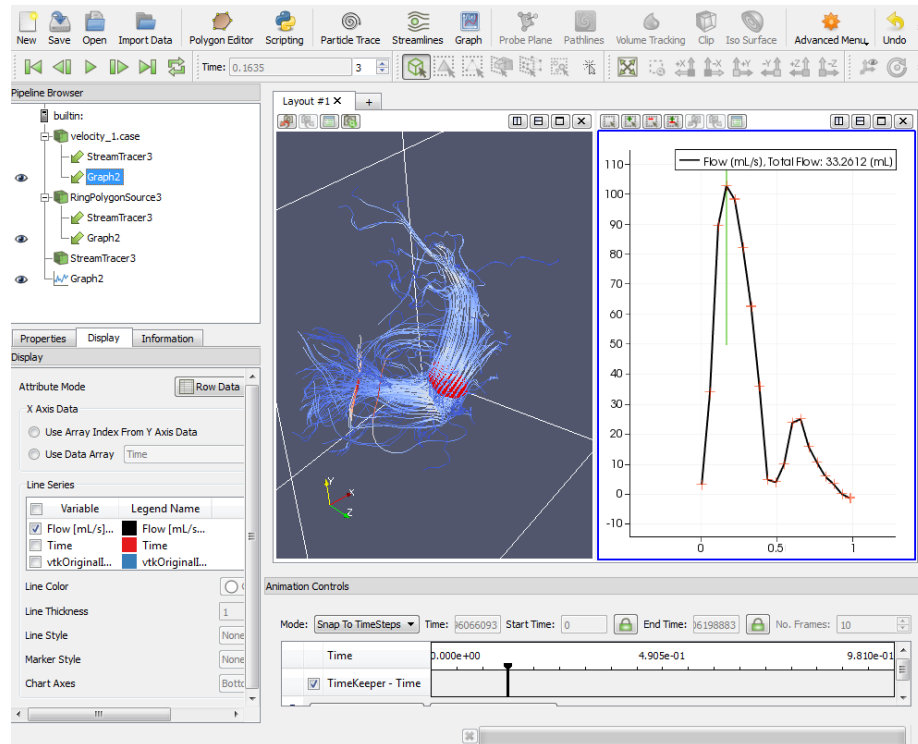
Its possible to quantify the flow through a polygon created via the polygon editor in FourFlow with the Graph tool. The workflow to do this is as follows:

1. Import a volume with velocity data.

2. Add a probe plane to the volume and position it at the region of interest.

3. Start the polygon Editor and create a polygon around the region of interest.

4. Make sure all the time frames where velocity quantification is desired has a region of interest.

5. Create the polygon with the Create polygon button in the Polygon Editor.

6. Create the velocity quantification and show it in a window by pressing the Graph button.

Volume Tracking
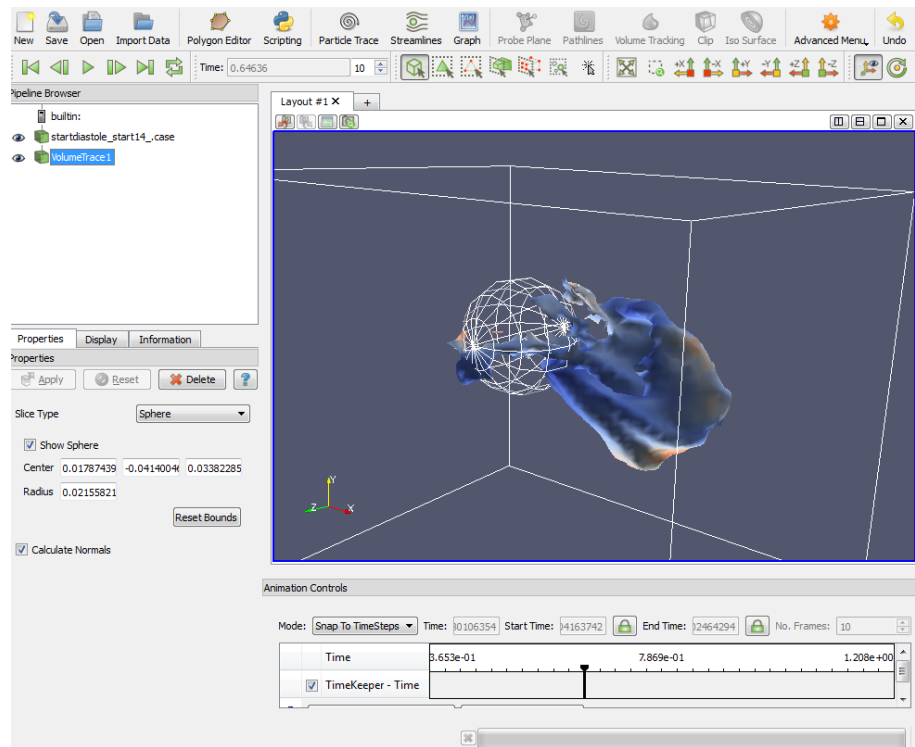
Volume Tracking is a method for tracking the flow of volumes over time, the workflow for using it is described below:

1. Import a volume tracking volume and select its node in the Pipeline Browser.

2. Press the Volume Tracking button.

The volume tracking node includes widgets for changing the origin that the volume starts to get tracked from and also includes the ability to track volumes from both a plane and a sphere.
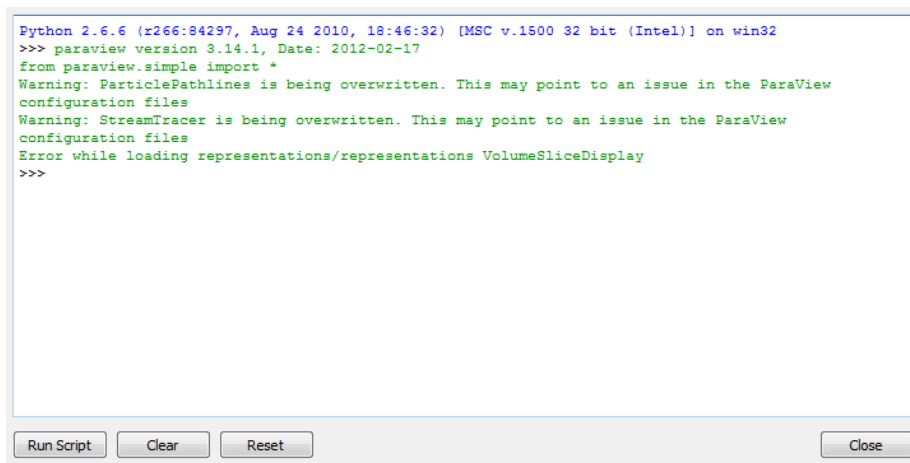
Isosurfacing
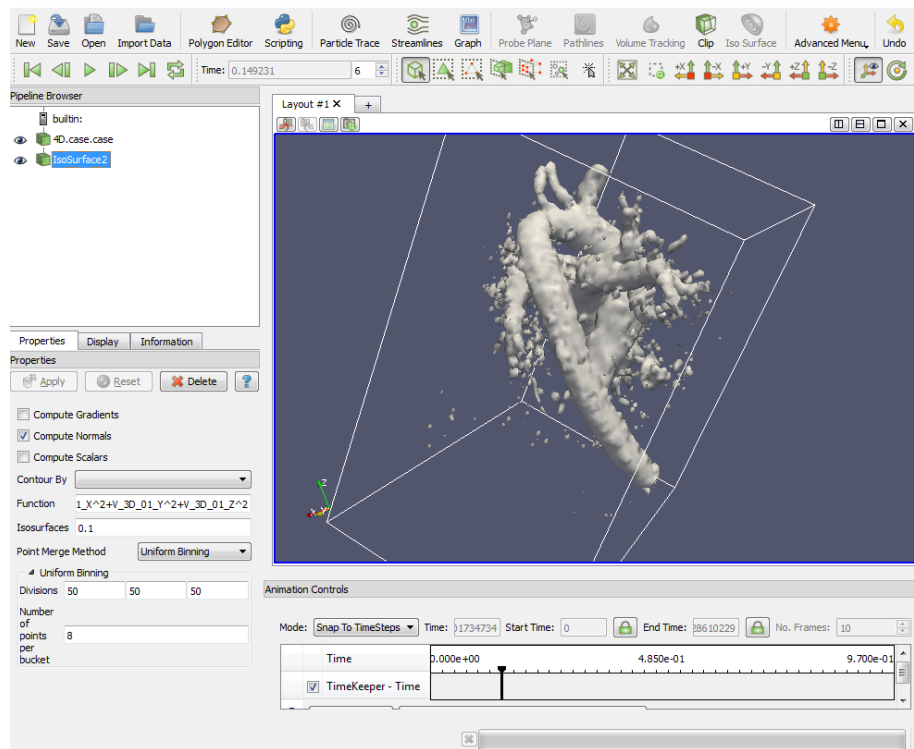
An isosurface is a surface constructed from a volume where the volume data is of the same value. For a velocity volume this means where the velocity has the same magnitude. To create such a surface select a velocity volume node and click the Isosurface button. The isosurface node has settings for changing what value in the velocity volume the surface gets created on.

Vortex Cores
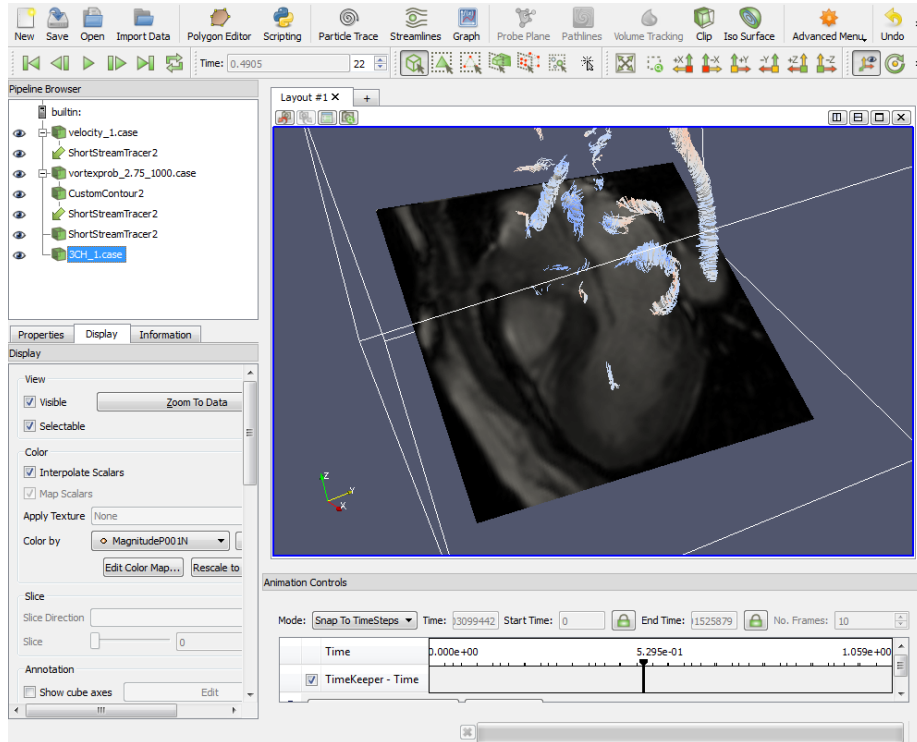
A Vortex Core visualization shows the existence of vortex cores in a velocity volume. Since the actual calculation of the vortex cores cannot be done in FourFlow they must be done in another application (such as Segment). To show such a visualization import both a velocity volume (named velocity*.case) at the same time as importing a vortex core volume (named vortexprob*.case).

Scripting

```
Python 2.6.6 (r266:84297, Aug 24 2010, 18:46:32) [MSC v.1500 32 bit (Intel)] on win32
>>> paraview version 3.14.1, Date: 2012-02-17
from paraview.simple import *
Warning: ParticlePathlines is being overwritten. This may point to an issue in the ParaView
configuration files
Warning: StreamTracer is being overwritten. This may point to an issue in the ParaView
configuration files
Error while loading representations/representations VolumeSliceDisplay
>>>
```

[Run Script] [Clear] [Reset]   [Close]

The scripting functionality in FourFlow uses the Python language. Please consult the ParaView documentation that ships with FourFlow for information about python scripting.

Frequently Asked Questions

Changing the light intensity of objects
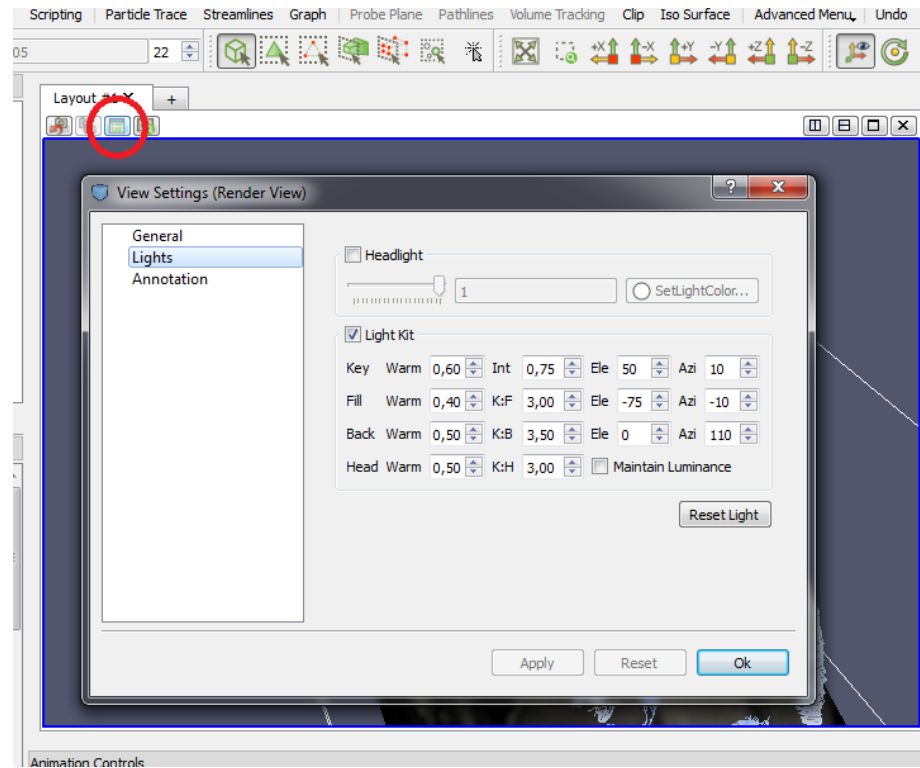
Sometimes it is desirable to change the light intensity of the objects that are displayed in the 3d view, for example the volume tracking visualizations tend to be a bit on the dark side by default. To do this, the scene light settings can be accessed via a small button on the active view toolbar. The int value of the key light tends to give a very bright appearance if set to about 1,7.

Changing the colorization of objects

To access the colour editor in FourFlow first select the node that you want to change the color of in the node editor and then change the property editor to the display tab. Clicking the Edit Colour Map... button brings up the colour gradient editor.
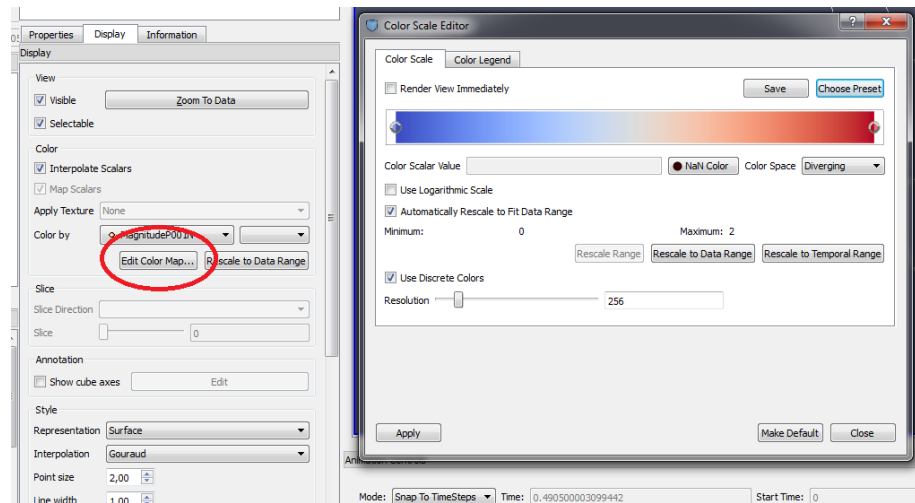
Dealing with slow animations

Some visualizations can use up more ram then FourFlow allows itself to use causing animation calculations to become slow. To increase the ram available for use open up the Application Settings window from the Advanced menu and increase the Cache limit from the Animation menu item.
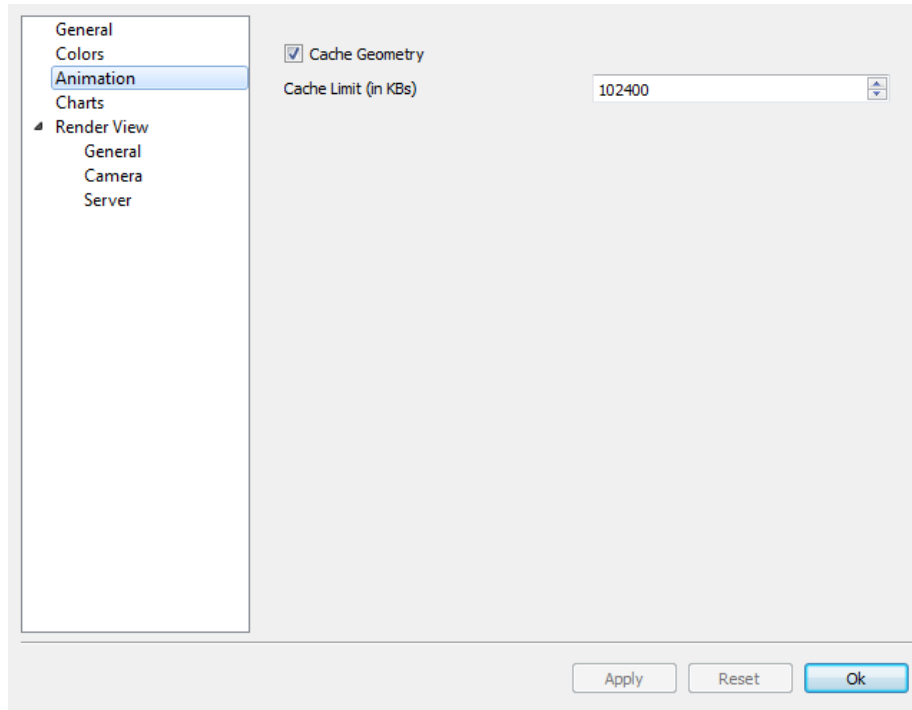
File Path Best Practises

It is always best to store all the ensight (*.case) files in a search path that does not use strange characters. Doing so can cause FourFlow to throw up an error dialog stating something like: "Error parsing XML in stream at line 3622, column 52, byte index 186700: not well-formed (invalid token)" when loading a file that references *.case files in such a file path. If such an error has happened move the files to a better path and edit the FourFlow save file in a simple text editor (like notepad on windows) and change the file paths from the old one to the new one.