

ParaView

Table of Contents

1 Introduction and Setup
1.1 The SLAC Toolbar
1.2 Macros
1.2.1 Macro Installation via the ParaView GUI
1.2.2 Macro Installation by copying files
2 Common Tasks
2.1 Save a Movie/Animation
2.1.1 Quick and dirty approach:
2.1.1.1 High quality approach:
2.2 Uniform Grid of Cones
2.3 Boundary Selection and Identification
2.4 Finding a Mesh Node/Vertex Id and XYZ Position/Field Values
3 Improving Appearance
3.1 Improved Particle Display and Particle Paths
3.2 Improved Display of Curved Elements
4 Client-Server remote Visualization
4.1 Remote Parallel Visualization on Franklin at NERSC
4.1.1 Setting up on Franklin (one time only):
4.1.2 Running Paraview on Franklin (general case):
4.2 Remote Visualization from Lens.
4.2.1 Initial Setup (one time only)
5 Tips and Tricks
5.1 Exporting Slice Data (Re-sampled to a Uniform Rectilinear Grid)
6 Getting Help

Accelerator Visualization Recipes for ParaView

Introduction and Setup

The SLAC Toolbar

The SLAC Toolbox is an extra toolbar added into ParaView. Its purpose is to streamline work flow by minimizing the amount of GUI interaction needed for commonly performed tasks. Everything done via the SLAC Toolbar can also be done via the normal ParaView GUI. In the toolbar, the steps needed for various accelerator visualization recipes are reduced to a single mouse click.

The first time you run ParaView, you will likely not see the SLAC Toolbar. It's included with the ParaView distribution, but is disabled by default.

To enable the SLAC Toolbar, go to "Tools --> Manage Plugins". Select "SLACTools" and press "Load Selected" which loads the plugin for now. Then, click on the SLACTools "+" and check the box by "Auto Load" so this plugin will default to always load.

When ParaView first comes up, click the SLAC Folder in the toolbar. This has places to enter the name of the mesh, and optionally, the name(s) of field files and/or particle files. On hitting "OK", the other icons in the toolbar become active, if their corresponding data has been loaded (e.g., the "e" and "b" field icons will become active if fields are loaded, but will remain greyed out if no fields were specified).

For mesh viewing (independent of whether fields and/or particles are loaded), these three buttons change how the mesh is displayed. The "Show Mesh as Solid Surface" button shows the mesh surface as an opaque surface. The "Show Mesh as Solid with Wireframe" button shows the surface along with element edges rendered in black. The "Show Wireframe Front and Solid Back" button is a view dependent drawing style; the outward oriented faces are shown as wireframe, while the inward oriented faces are shown as a solid surface. This way, no matter how the mesh is oriented on the screen, you can look through the outer surface to see details inside.

For looking at fields (once they have been loaded), the "e" and "b" buttons in the SLAC Toolbox display pseudocolored electric and magnetic field magnitudes (respectively) on the mesh surface.

To see particles once they have been loaded, pressing the "Toggle Show Particles" button will toggle the display of particles. Particles should always be inside the mesh, so they may not be visible unless the mesh is displayed using the "Show Wireframe Front and Solid Back" style. Particles are rendered as white dots.

To create a line plot of the current field along the z axis, press the "Make a plot of the field along the Z axis" button. The field type for the plot will be the same as the displayed field type.

Macros

When new toolbar features are being developed, prototyped, and tested, we use ParaView's python macros. For CW11, several macros were in place on the euclid cluster. These are not quite ready to run on a standalone installation of ParaView. They will be made available here when ready for release. The current macros are:

- nextmode (cycles through multiple frequency domain mode files for a common mesh) (this one may be ready by roughly October 25, 2011.)
- wakeplot (waiting on good way to ask user for input filename)
- timeFFT (waiting on good way to ask user for input filename)
- resonant (waiting on good way to ask user for input filename)
- sparam (this one requires more work and will likely take longest to appear here)

There are 2 ways to install a macro:

Macro Installation via the ParaView GUI

- download the macro of interest to a known directory. It is a python file, so it should be called yourmacroname.py. In ParaView, click "Macros" -> "Add New Macro", then browse to yourmacroname.py in your known directory and hit "Ok".
- on running (or reloading) ParaView, the you should see yourmacroname (without the ".py" suffix) in one of the macro toolbars. Clicking the macro name will run the macro.

Macro Installation by copying files

- ParaView stores macros in one of two places, depending on your operating system. In windows, it's "%APPDATA%/ParaView/Macros". For all other operating systems, it's "~/.config/ParaView/Macros". For windows users, one way to find out where "%APPDATA%" really is, is to run paraview, then "Tools" -> "Python Shell". Within the python shell, type "import os; print os.environ['APPDATA']".
- You can then install a macro by copying its file into the appropriate ParaView/Macros directory for your operating system.
- on running (or reloading) ParaView, the you should see yourmacro name (without the ".py" suffix) in one of the macro toolbars. Clicking the macro name will run the macro.

Common Tasks

Save a Movie/Animation

Quick and dirty approach:

1. get things to look the way you want onscreen
2. "File -> Save Animation", accept the defaults, and hit "Save Animation"
3. for File Name, enter the base portion of a frame name "frame", for example.
4. set the type to .png (<-- This is important; other formats will burn disk space or butcher your image quality.)
5. Hit OK.
6. Encode your frames into a movie (using the encode.py script for a plays-on-linux-only x264 movie, or quicktime H.264 for a portable movie) [High Quality Movie Encoding](#)

High quality approach:

1. Please see the full writeup here: [How to Make Movies with ParaView](#)
1. Setup
 - a. Load your data (mesh, fields and/or particles), and get them to look the way you want
 - b. preview your animation using the VCR style controls in the VCR Toolbar ("View -> Toolbars -> VCR Controls" if not already visible)
 - c. For your images, figure out your desired resolution and aspect ratio (aspect = pixel_width/pixel_height).
 - i. Here is some information on [Choosing Frame Resolution](#). Please read and understand that before proceeding.
 - ii. It may help to lock your view to that ratio ("Tools -> Lock View Size Custom...")
 - d. It helps to see the Animation View ("View -> Animation View" if it's not already visible)

Uniform Grid of Cones

1. Create a Plane Glyph as described in "Re-sampling Volume Data to a Planar Uniform Rectilinear Grid", but skip the Save Data step
2. select the Plane Glyph in the Pipeline Browser
3. apply the Glyph filter
 - a. "Filters -> Alphabetical -> Glyph" (or skip the menus and just hit the glyph icon)
 - b. Set Vectors to the field you want to show (defaults to efield)
 - c. Set "Glyph Type" to "Cone"
 - d. Deselect "Masked Points"
 - e. Accept other defaults. Hit the green "Apply" button. Note: cones will probably be the wrong scale at this point.
 - f. Scale the cones by checking "Edit", and adjust the scale factor (usually down by an order of magnitude or so).
 - g. Remember to hit "Apply" each time after adjusting the scale value.
 - h. If, in the Pipeline Browser, the PlaneGlyphs was not automatically hidden when Glyphs was created, hide it.
4. Finishing touches
 - a. The cones tend to look better when they are skinnier. Try setting radius to roughly 0.2
 - b. If the visualization results are to be published or shown to others, the cones look much better at resolution 12 to 18 or so.

Boundary Selection and Identification

1. Select your mesh in the Pipeline Browser
 - a. If the Selection Inspector isn't active, turn it on "View -> Selection Inspector". This usually shows up in the lower left part of the ParaView window, under the Object Inspector.
2. if you **don't know the surface ID**, but want to find it out by mousing on the mesh,
 - a. use "Edit -> Select Block", (or just hit the "b" key), and then use the mouse to drag out a rectangular region on the mesh which surrounds a portion of the surface(s) of interest.
 - b. Selected boundaries will highlight as pink wireframe triangles, and the selected blocks will be checked in the Selection Inspector, telling you the ID(s) of interest. (For instance, a check next to "DataSet 6 (7)" corresponds to the mesh surface ID of 6.)
 - c. if you got too many blocks, you can check/uncheck blocks in the selection inspector.
3. If you **know the surface ID** ("composite ID"),
 - a. Set "Selection Type" to "Blocks", and "Field Type" to "CELL".
 - b. In the Selection Inspector (scroll down if needed), check "Blocks", and then check/uncheck whichever blocks you care about. You will see those corresponding surfaces in the 3D view highlight as a pink wireframe collection of triangles. (For instance, if you want surface ID 6, then check the box next to "DataSet 6 (7)").

Finding a Mesh Node/Vertex Id and XYZ Position/Field Values

1. Orient the mesh and zoom in so that the region of the mesh containing the nodes of interest is visible.
2. Set surface type for the displayed mesh to "Surface" (i.e., not "wireframe").
3. Click the "Select Points On" icon (or "Edit -> Select Points On" via menus)
4. Drag out a rectangular region surrounding the nodes/vertices of interest.
 - a. The Selection Inspector will list the index corresponding to these points.

- b. The index for these points can be shown on screen by scrolling down in the Selection Inspector "Display Style" section, clicking on "Point Label", and checking "Visible".
5. To see XYZ locations, field values, do the following
 - a. In the upper right corner of the 3D display, click the "Split Vertical" tab, and press "Spreadsheet View" in the new area.
 - b. Now go to the Object Inspector, Display tab, and check "Show only selected Elements". The spreadsheet view will now only show data for the selected vertices. Note that vector values in the spreadsheet have their column width default to only wide enough to show the x component, so grab the column title dividers and stretch the vector columns to be wide enough to show all 3 components.

Improving Appearance

Improved Particle Display and Particle Paths

1. Load the mesh and particles via the folder icon in the SLAC Toolbar.
2. Zoom to where the particles are, and make sure you can see them.
 - a. It may be necessary to turn off the mesh surface or switch to wireframe, because the mesh rendered as a surface will usually hide the particles inside.
 - b. It may be necessary to advance beyond the first timestep to see particles because in some simulations, particles are not created until after the first timestep.
 - c.
3. Apply the particle pathline filter
 - a. Make sure you have selected the particle data in the Pipeline Browser, and have them set to visible (small eye icon toggles this).
 - b. Create the filter ("Filters -> Alphabetical -> Particle Pathlines")
 - c. Adjust parameters
 - i. "Mask Points" will only use a subset of the total particles to increase performance. A value of 1 means show every particle, 2 means every other particle, 10 means every 10th particle, and so on. Use 1 unless you need otherwise.
 - ii. Max Track Length and Max Step Distance need to be adjusted to taste. Accept the default for now.
 - iii. Hit "Apply". This will create two new pipeline objects "Pathlines" and "Particles". If your particles disappear here, possibly your Mask Points is too large (so try a value of 1).
4. Handle particle heads and tails.
 - a. For **particle heads**, It often helps to show them **as glyphs** (spheres or cones, for example)
 - i. Select the newly created "Particles" source in the Pipeline Browser.
 - ii. "Filters -> Alphabetical -> Glyph"
 1. "Glyph Type": Sphere
 2. "Scale Mode": Off (to keep all particles the same size)
 3. "Mask points" is to reduce the number of particles that are shown to a subset. Turn this off.
 4. "Random Mode" doesn't matter if "Mask Points" is unchecked. But if you are masking, random mode chooses a random subset to hopefully chose a distributed set of particles rather than just the first n, which may all be bunched together.
 5. "Orient" is not necessary for sphere glyphs, but would make sense for cone glyphs.
 6. Finer Theta/Phi resolution will keep the spheres looking more spherical, but takes longer to render, and may not be needed if the camera never gets very close to the particles.
 7. Hit "Apply".
 8. Adjust the "Radius" according to your taste.
 9. Select the coloring you prefer. "EmissionType" is default, but "Momentum" or "Solid Color" can be useful as well. You may need to adjust the field scaling and color map via the "Edit Color Map" button.
 10. If "Particles" is still visible you may see small dots together with the sphere glyphs. Remember to turn off "Particles" visibility so that only the sphere glyphs are seen. Though often it is helpful to leave them on temporarily while setting up glyphs just so you have something to see and provide context.
 - iii. Head Troubleshooting:
 1. I don't see any particles.
 - a. Does your dataset have any particles in the first timestep? It may be necessary to go to a later timestep.
 2. My particles have crazy different sizes.
 - a. Make sure "Scale Mode" is set to "off".
 3. My particles have crazy (but consistent sizes).
 - a. Adjust "Radius" accordingly. Often the default radius is much too large.
 4. My particles have crazy colors.
 - a. Check that you're mapping the right thing onto them: Momentum or Emission Type, or Solid Color
 5. My particles have reasonable colors, but too much (or not enough) red.
 - a. When you hit the "Rescale to Data Range" button by the "Edit Colormap" button, it's rescaling based on the *current* timestep. So, for multipacting, for example, momentum is usually highest just before impact. So move forward/backward in time to just before the impact and "Rescale to Data Range" for that timestep. For more red, "Rescale to Data Range" even earlier.
 6. I see (probably) white dots on most of my sphere glyphs.
 - a. Turn off visibility for you particle data, and for "Particles" in your pipeline browser. The corresponding glyph should be left visible, however.
 - b. For **particle trails**, it is often helpful to use apply a **tube filter**. This takes longer to render, but shows depth relationships more clearly, and gives the particles position history.
 - i. Select "Pathlines" in the Pipeline Browser.
 - ii. "Filters -> Alphabetical -> Tube" (or, for older ParaView versions, "Filters -> Alphabetical -> Generate Tubes").
 1. The defaults are not bad.
 2. But "Number of sides" usually needs to be increased to at least 8 for tubes to have better shading
 3. Adjust "Radius" to taste, and to coordinate well with the sphere glyphs for the particles.
 - a. Too thick will result in a lot of occlusion, meaning much of your data will be hidden in the image.
 - b. Too thin doesn't leave room for shading, and prevents good depth discernment.
 - c. A tail that appears to be half as wide as the particle sphere usually works reasonably well.
 4. Map something appropriate to the tail color

- a. "Solid Color" works well for lighter colors (white, yellow, for example) for clearer shading and depth discernment. Solid colors also won't compete with other colors so that visual information is more clearly presented.
 - b. "Momentum" can be meaningful, though can also result in a lot of visual clutter.
 - c. Using a white trail and mapping momentum (or emission type) to the sphere instead, is usually more clear.
 - 5. Play the animation (using the play button in the VCR toolbar) to get a sense of how long your trails should be whether they are too thick/thin, and whether things are too cluttered. Adjust the corresponding parameters accordingly.
- iii. Tail Troubleshooting
 - 1. I don't see any trails.
 - a. The trails are constructed from the results of playing consecutive timesteps. So you'll need to start at a time when you have particles, and play forward a few steps so that the trails can grow. If you step backward in time, or forward by more than one step, the Particle Pathline filter gets confused.
- 5. General Troubleshooting
 - a. Once a particle bounces off a wall, sometimes the portion of the tail from before the bounce disappears.
 - i. This is a known issue. There is presently no workaround.
 - b. I'm working on trial runs of generating a movie. But rendering is very slow.
 - i. keep the resolution (number of sides) for the spheres and tubes as low as still looks good.
 - ii. temporarily disable the spheres/tubes, and mapping the same quantities on the points/lines. Points and lines can have their sizes/widths adjusted (within limits), and are much faster to draw.

Improved Display of Curved Elements

The first method is to load the surface midpoints by clicking "Read Midpoints" (note: a bug prevents this from working in ParaView 3.8.1-RC1).

The other method is to use averaged surface normal.

1. Load internal volume by click "Read Internal Volume" and Apply.
2. Hide the "External Surface" by clicking on the eye icon to it's left in the Pipeline Browser.
3. Select the "Internal Volume" in the Pipeline Browser.
4. Apply "Filter -> Alphabetical -> Extract Surface"
5. Then filter the extracted surface "Filter -> Alphabetical -> General Surface Normal"
6. Adjust the "Feature Angle" property if sharp creases appear in areas which should be smooth, or vice versa.

A third approach combines the first two. Generating surface normals when midpoints are loaded produces better results than either approach alone.

Client-Server remote Visualization

Remote Parallel Visualization on Franklin at NERSC

Setting up on Franklin (one time only):

1. Configure your ParaView Client to know about the server on Franklin at NERSC
 - a. Download http://www.slac.stanford.edu/~schussma/Franklin_community.pvsc server configuration file and save a copy local to your desktop machine.
 - b. Edit your copy of Franklin_community.pvsc to change the single occurrence of the word "zebra" to your Franklin login name.
 - c. On your desktop machine, run paraview 3.8.1 (must be 3.8.1 to match what's on Franklin; other versions will not work)
 - d. Hit the "connect" button (with the green dot).
 - e. Click "Load Servers".
 - i. Browse to where you saved the attached .pvsc file, select it, hit "Ok".
 - ii. Select "Franklin_community" in the "Choose Server" dialog, hit "Connect".
 - iii. The defaults should be ok. Hit "Connect".
 - f. Two dialogs will pop up ("Output Messages" and "Starting Server"). When they appear, ignore them...
 - g. Instead, go to the terminal on your desktop from which you originally ran ParaView, and there type in your Franklin Password.
 - i. If all is well, Franklin should respond to your password with something that looks like "7049275.nid00003".
 - ii. You may need to wait a minute or so more until Franklin schedules your parallel server job.
 - h. When the "Starting Server" dialog disappears, you have connected successfully.

Running Paraview on Franklin (general case):

1. On your desktop machine, run paraview 3.8.1 (must be 3.8.1 to match what's on Franklin; other versions will not work)
2. Hit the "connect" button (with the green dot).
3. Select "Franklin_community" in the "Choose Server" dialog, hit "Connect".
4. The defaults should be ok. Hit "Connect".
 - a. Two dialogs will pop up ("Output Messages" and "Starting Server"). When they appear, ignore them...
 - b. Instead, go to the terminal on your desktop from which you originally ran ParaView, and there type in your Franklin Password.
 - c. If all is well, Franklin should respond to your password with something that looks like "7049275.nid00003".
 - d. You may need to wait a minute or so more until Franklin schedules your parallel server job.
 - e. When the "Starting Server" dialog disappears, you have connected successfully.

Remote Visualization from Lens.

To run a ParaView client on your local machine, and connect that to a parallel ParaView server running on Lens, you need to do some one-time-only setup before you can connect in the usual way.

Initial Setup (one time only)

1. go to <http://www.paraview.org/files/v3.8/> and download the appropriate "ParaView-3.8.0-RC1" for your platform. It must be 3.8.0, and must be RC1, because that is what matches what is already installed on lens. Install this client on your system.
2. save a copy of [lens.pvsc](#) on your machine as well.
 - (If your username on your local machine does not match your username on lens, edit the lens.pvsc file, change
 - <Argument value="lens.ccs.ornl.gov"/> to include your lens username:
 - <Argument value=username@lens.ccs.ornl.gov"/> and change
 - <Server name="Lens" resource to
 - <Server name="Lens-Username" resource ... FIXME (left off here)
3. run ParaView 3.8.0-RC1 on your machine
4. Configure your local plugins.
 - a. Tools -> Manage Plugins
 - i. Under "Local Plugins", select "SLACTools" and hit "Load Selected" if it's not already loaded. Also, click the grep triangle next to "SLACTools" to expand a menu, and make sure "Auto Load" is checked so that you don't have to repeat this every time you run ParaView.
 - ii. You may want to do the same for "CSCS_PointSpritePlugin", which is useful particle visualization (though it doesn't presently work on Mac).
5. connect to lens
 - a. Press the "Connect" button in the main controls toolbar.
 - i. press the "Load Servers" button.
 - ii. Navigate to your "lens.pvsc" file, select it, and hit "OK".
 - b. Under "Choose a Server" you should now see "lens". Select that and hit "Connect".
 - c. A dialogue will pop up. The defaults are probably ok. Don't select less than 16 for "Number of Cores", because that makes the X11 server unhappy for some reason. Hit "Connect".
 - d. Two more windows should open up, one of which is an X terminal. In it, enter your lens passcode.

Tips and Tricks

Exporting Slice Data (Re-sampled to a Uniform Rectilinear Grid)

Occasionally it is useful to re-sample unstructured data to a structured grid. For example, this can be used to export data in a form more usable by other software.

1. Load the mesh and mode as usual, making sure that "Read Internal Volume" is enabled.
2. Set up an appropriate mesh view.
 - a. Hide the External Surface and show the Entire Volume (click the corresponding eye icons).
 - b. Set Entire Volume drawing style to Surface if it isn't already (might default to Outline, which appears as a wireframe box)
 - c. Set the surface to display a field you are interested in (so you can know that you will orient the sampling plane correctly)
 - d. Select "Entire Volume" so that text is highlighted
3. Create the sampling filter
 - a. create the filter: "Filters -> Alphabetical -> Plane Glyphs". This will show a wireframe bounding box and a red outline for the plane orientation.
 - b. Position the plane where you want it. The most consistent and repeatable way to do this is to put values in for "Center" and "Normal" which specify a point in the plane and the plane's normal vector, respectively. Change "Resolution" to 100, and hit "Apply"
4. (Optionally) Save the data
 - a. Presumably, this re-sampled data is to be exported, so make sure that only the re-sample filter is active and visible in the Pipeline Browser, then
 - b. "File -> Save Data", type in a file name, set "Files of Type" to be "CSV" (Comma Separated Values) which is readable by spreadsheets and Matlab. Hit "OK".
 - c. For "Configure Writer", you probably don't want to Write All Time Steps. And "Field Association" should be "Points". Hit "OK". (The file should be written to the same directory from which you originally ran ParaView, unless you navigated elsewhere.)

Getting Help

If you have any questions, or if something in this page is missing or unclear or incorrect, please send email to Greg Schussman schussman@slac.stanford.edu.