13th Summer School on SCIENTIFIC VISUALIZATION

Paraview for large data visualization

Raffaele Ponzini – r.ponzini@cineca.it
SuperComputing Applications and Innovation Department

CINECA

OUTLINE

- Paraview architecture details
- Large data visualization
- Filters and data explosion
- Rendering of large data

Paraview architecture details

ParaView is designed as a three-tier client-server architecture.

The three logical units of ParaView are as follows:

Data Server: is responsible for data reading/filtering/writing;

All of the pipeline objects seen in the pipeline browser are contained in the data server.

The data server can be parallel.

Render Server: is the unit responsible for rendering.

The render server can also be parallel, in which case built in parallel rendering is also enabled.

Client: The unit responsible for establishing visualization.

The client controls the object creation, execution, and destruction in the servers, but does not contain any of the data.

If there is a GUI, that is also in the client.

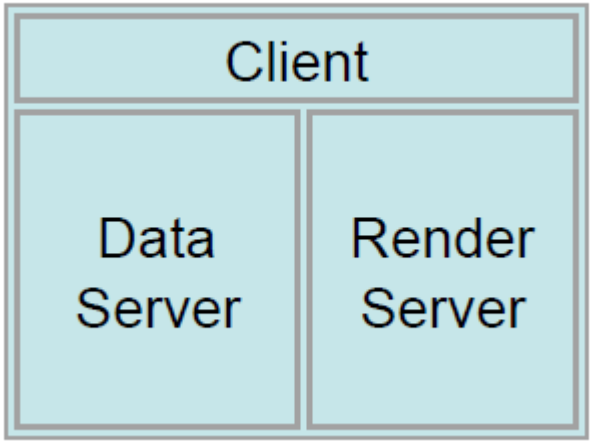
The client is always a serial application.

Paraview architecture details

Standalone mode

In standalone mode, the client, data server, and render server are all combined into a single serial application.

When you run the ParaView application, you are automatically connected to a built-in server so that you are ready to use the full features of ParaView.



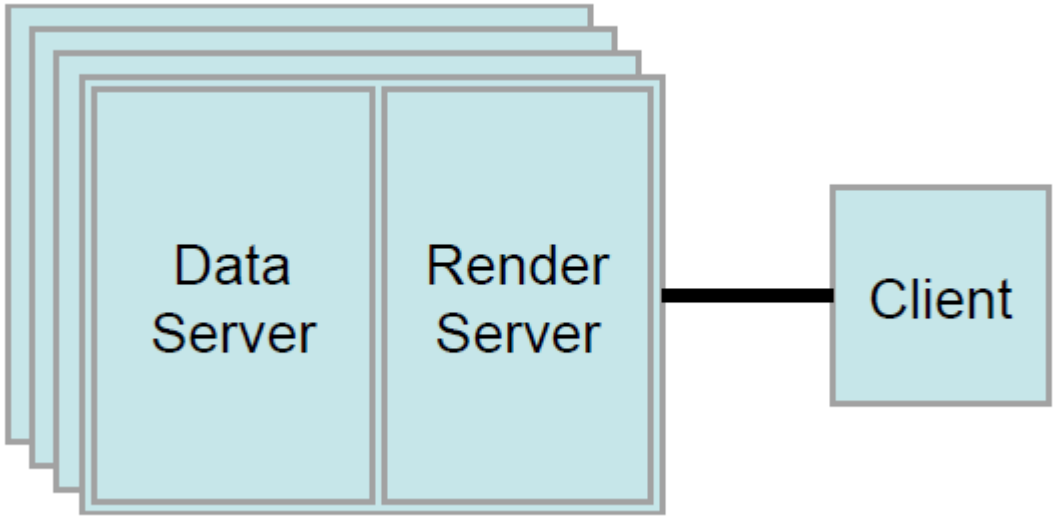
OUR CASE UP TO NOW

Paraview architecture details

Client-server mode. In client-server mode, you execute the *pvserver* program on a parallel machine and connect to it with the ParaView client application.

The *pvserver* program has both the data server and render server embedded in it, so both data processing and rendering take place there.

The client and server are connected via a socket, which is assumed to be a relatively slow mode of communication, so data transfer over this socket is minimized.



Paraview architecture details

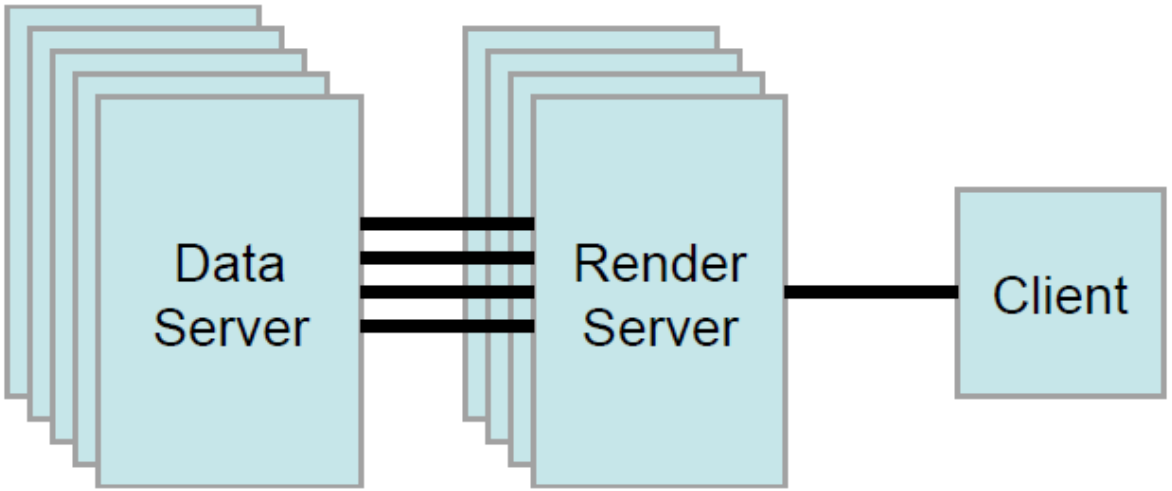
Client-render server-data server mode. In this mode, all three logical units are running in separate programs.

The client is connected to the render server via a single socket connection.

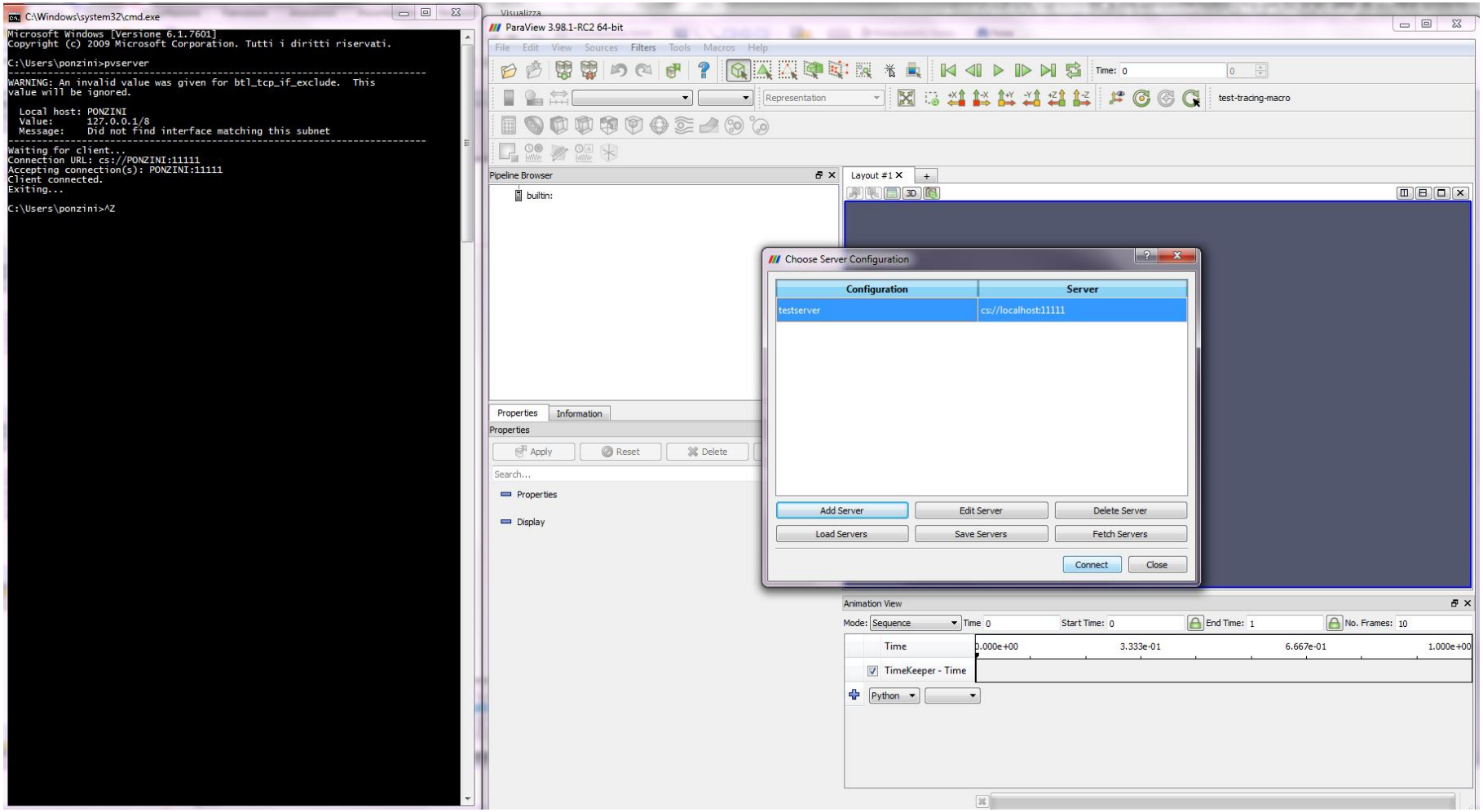
The render server and data server are connected by many socket connections, one for each process in the render server. Data transfer over the sockets is minimized.

Not recommend since it born to take advantage of heterogeneous environments where one might have a large, powerful computational platform and a second smaller parallel machine with graphics hardware in it.

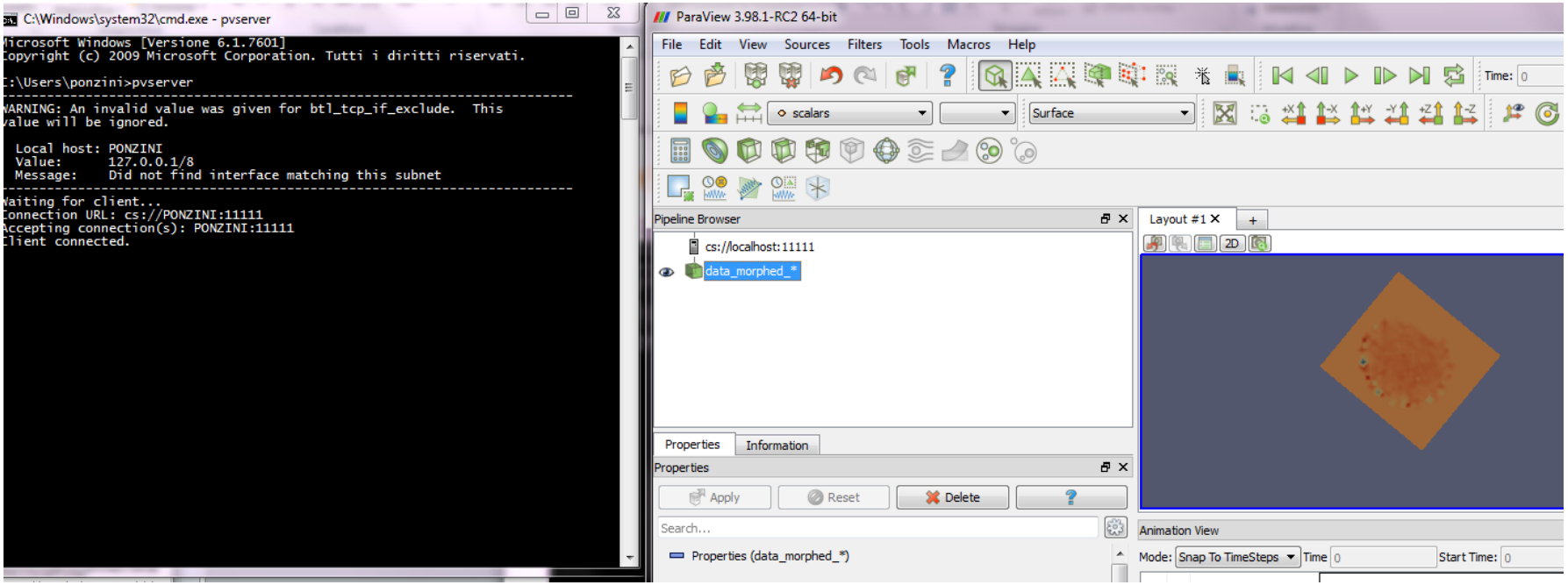
In practice benefits are almost always outstripped by the time it takes to move geometry from the data server to the render server.



Connecting to a server



Connecting to a server



Large data visualization

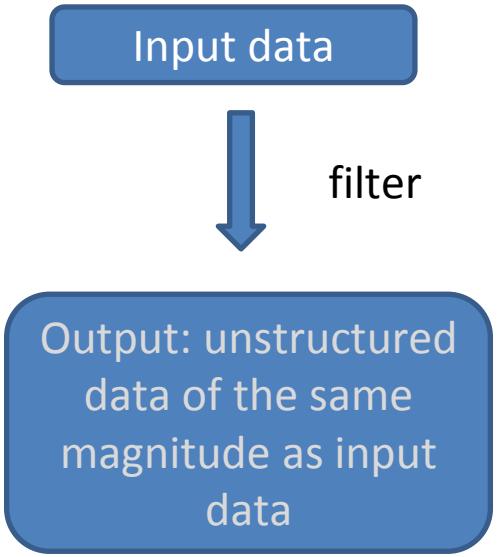
- Loose coupling between components → flexible framework
- Drawback → larger memory footprint

- Why: Each stage of this pipeline maintains its own copy of the data. Whenever possible, ParaView performs **shallow copies** of the data so that different stages of the pipeline point to the same block of data in memory. However, any filter that creates new data or changes the values or topology of the data must allocate new memory for the result. If ParaView is filtering a very large mesh, inappropriate use of filters can quickly deplete all available memory.

- **When visualizing large data sets**, it is important to **understand the memory requirements of filters**.

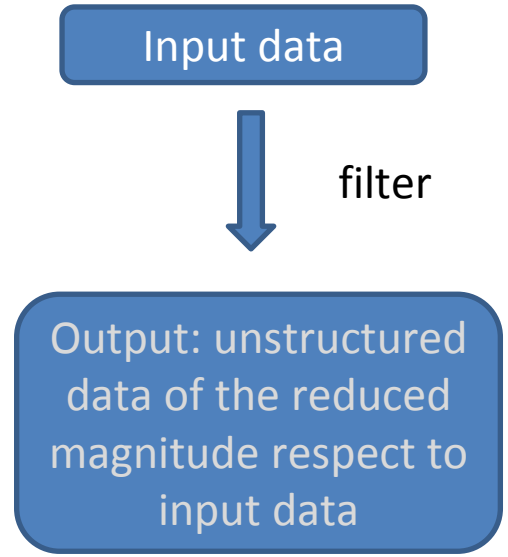
Filters and data explosion

- Append Datasets
- Append Geometry
- Clean
- Clean to Grid
- Connectivity
- D3
- Delaunay 2D/3D
- Extract Edges
- Linear Extrusion
- Loop Subdivision
- Reflect
- Rotational Extrusion
- Shrink
- Smooth
- Subdivide
- Tessellate
- Tetrahedralize
- Triangle Strips
- Triangulate



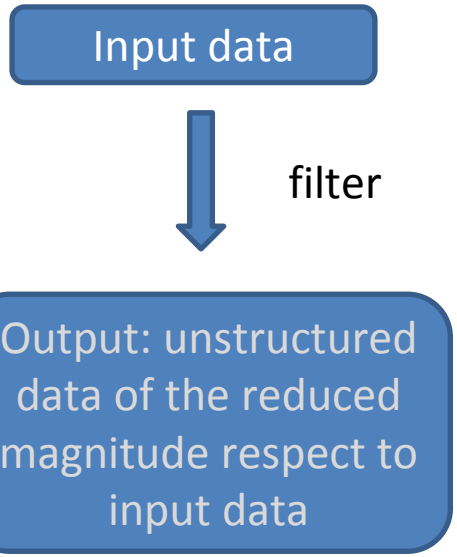
Fifilters and data explosion

- Clip
- Decimate
- Extract Cells by Region
- Extract Selection
- Quadric Clustering
- Threshold



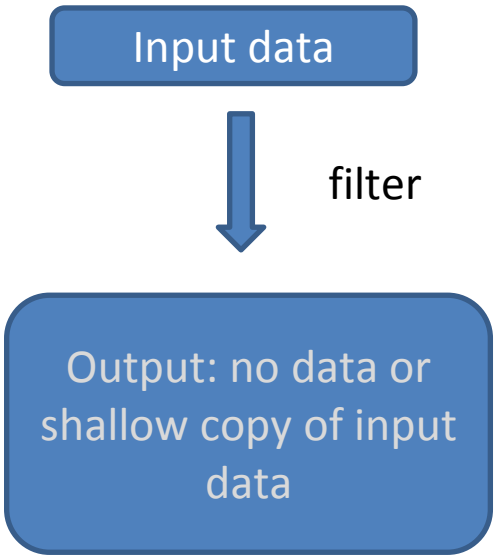
Filters and data explosion

- Cell Centers
- Contour
- Extract CTH Fragments
- Extract CTH Parts
- Extract Surface
- Feature Edges
- Mask Points
- Outline (curvilinear)
- Slice
- Stream Tracer



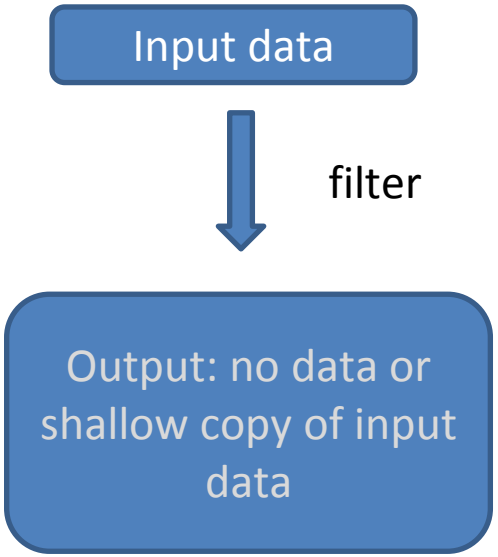
Filters and data explosion

- Block Scalars
- Calculator
- Cell Data to Point Data
- Curvature
- Elevation
- Generate Surface Normals
- Gradient
- Level Scalars
- Median
- Mesh Quality
- Octree Depth Limit
- Octree Depth Scalars
- Point Data to Cell Data
- Process Id Scalars
- Python Calculator
- Random Vectors
- Resample with dataset
- Surface Flow
- Surface Vectors
- Texture Map to...
- Transform
- Warp (scalar)
- Warp (vector)



Fifilters and data explosion

- Annotate Time
- Append Attributes
- Extract Block
- Extract Datasets
- Extract Level
- Glyph
- Group Datasets
- Histogram
- Integrate Variables
- Normal Glyphs
- Outline
- Outline Corners
- Plot Global Variables Over Time
- Plot Over Line
- Plot Selection Over Time
- Probe Location
- Temporal Shift Scale
- Temporal Snap-to-Time-Steps
- Temporal Statistics



Culling out large data

- Extract iso-surfaces from a volume using the Contour filter;
- Perform a Slicing over the data
- Clipping
- Threshold
- Extract Selection
- Extract Subset

NOTE: all of these filters with the exception of Extract Subset will convert structured data types to unstructured grids. Therefore, they should not be used unless the extracted cells are of at least an order of magnitude less than the source data.

- When possible, replace the use of a filter that extracts 3D data with one that will extract 2D surfaces. For example, if you are interested in a plane through the data, use the Slice filter rather than the Clip filter.
- If you are interested in knowing the location of a region of cells containing a particular range of values, consider using the Contour filter to generate surfaces at the ends of the range rather than extract all of the cells with the Threshold filter.
- Be aware that substituting filters can have an effect on downstream filters. For example, running the Histogram filter after Threshold will have an entirely different effect then running it after the roughly equivalent Contour filter.

Rendering of large data

Rendering is the process of synthesizing the images that you see based on the dataset.

The ability to effectively interact with your data depends highly on the speed of the rendering. The speed of rendering is proportional to the amount of data being rendered.

The interactive render is a compromise between speed and accuracy.

ParaView supports two modes of rendering that are automatically clipped as necessary.

Mode-1: **still render**, the data is rendered at the highest level of detail. This rendering mode ensures that all of the data is represented accurately. At any time when interaction of the 3D view is not taking place, ParaView uses a still render so that the full detail of the data is available as you study it.

Mode-2: **interactive render**, speed takes precedence over accuracy. This rendering mode endeavors to provide a quick rendering rate regardless of data size.

As you drag your mouse in a 3D view to move the data, you may see an approximate rendering while you are moving the mouse, but the full detail will be presented as soon as you release the mouse button.

Basic parameters settings

Use Immediate Mode Rendering When checked, geometry is sent to the graphics card for immediate rendering. When unchecked, the geometry is first compiled into display lists for more efficient rendering. The display lists usually render faster, but require initial time to compile during the first frame and extra memory to store.

LOD Threshold Controls when to replace the geometry with a decimated version of the geometry during interactive rendering. The checkbox turns the feature on or off. When on, the slider gives a threshold for the feature. If the geometry size is below the threshold, it is considered small enough to render. When the geometry size is above the threshold, the decimated form is used during rendering. When unchecked, the full geometry is always rendered.

LOD Resolution Controls the size of geometry to create for geometric level of detail. Moving the slider to the right results in a more coarse representation.

Lock Interactive Render Specify a pause between an interactive render and a still render. This pause allows you to let go of the mouse and drag again (to perhaps change from rotate to pan) without having to wait for a full still render.

Use outline for LOD rendering Normally ParaView uses a decimated version of the geometry for its interactive render. When this box is checked, ParaView instead uses a simple bounding box outline. This saves the time to make the decimated geometry and any rendering time.

Allow Rendering Interrupts When checked, a still render may be interrupted by a request to perform an interactive render. Not all rendering modes support interrupts.

Basic parameters settings

Outline Threshold When an unstructured dataset exceeds this threshold, the default representation mode is set to outline instead of surface.

Enable Depth Peeling ParaView uses an algorithm called depth peeling to properly render translucent surfaces. With it, the top surface is rendered and then “peeled away" so that the next lower surface can be rendered and so on. Can be expensive: try shutting o depth peeling or adjust the number of depth peels used. Using more peels → more depth complexity → slower (less peels → faster).

Coincident Topology Resolution It is sometimes the case you wish to render lines that are coincident with a surface (for example, rendering a wireframe of cells on top of their surface). Rendering systems generally can have problems resolving hidden surfaces in these circumstances. These options allow you to change the “tricks" ParaView's rendering uses to resolve these issues. Generally you do not need to change these options unless there is a problem with the rendering.

Use Offscreen Rendering for Screenshots ParaView usually uses offscreen rendering when creating screenshots to avoid having other windows on your desktop aspect the images. However, some hardware places restrictions on offscreen rendering that can cause artifacts. If you and that your saved images look different then the ones on screen (for example, more dark), try disabling this feature.