CSC

# **Elmer** Post-processing utilities

Peter Råback ElmerTeam CSC – IT Center for Science

Tampere Univ. Of Tech. 24.4.2014

# **Alternative postprocessors for Elmer**

## **Open source**

- ElmerPost
  - Postprocessor of Elmer suite
- ParaView, Visit
  - Use ResultOutputSolve to write .vtu or .vtk
  - Visualization of parallel data
- OpenDX
  - Supports some basic elementtypes
- Gmsh
  - Use ResultOutputSolve to write data
- Gnuplot, R, Octave, ...
  - Use SaveData to save results in ascii matrix format
  - Line plotting

## Commercial

- Matlab, Excel, …
  - Use SaveData to save results in ascii matrix format
  - Line plotting



## Visualization tools – Poll (3/2014)

#### What visualization software do you use?

You may select up to 10 options ElmerPost 12 18% 11% ElmerGUI VTK postprocessor 1 7 Paraview 41% 27 ViSit 5% - 3 0 No votes Mayavi Gmsh 2 3% GiD 1 2% 8% Matlab 5 gnuplot 4 6% 8% Something else (please specify) 5 Total votes : 66 Submit vote

## **Exporting 2D/3D data: ResultOutputSolve**



- Apart from saving the results in .ep format it is possible to use other postprocessing tools
- ResultOutputSolve offers several formats
  - vtk: Visualization tookit legacy format
  - vtu: Visualization tookit XML format
  - Gid: GiD software from CIMNE: http://gid.cimne.upc.es
  - Gmsh: Gmsh software: http://www.geuz.org/gmsh
  - Dx: OpenDx software
- Vtu is the recommended format!
  - offers parallel data handling capabilities
  - Has binary and single precision formats for saving disk space
  - Suffix .vtu in Post File does this automatically



## **Exporting 2D/3D data: ResultOutputSolve**

An example shows how to save data in unstructured XML VTK (.vtu) files to directory "results" in single precision binary format.

```
Solver n
Exec Solver = after timestep
Equation = "result output"
Procedure = "ResultOutputSolve" "ResultOutputSolver"
Output File Name = "case"
Output Format = String "vtu"
Binary Output = True
Single Precision = True
End
```

## **Overcoming bottle-necks in postprocessing**



Excellent tools available: Paraview & Visit

## Reducing data

- Saving only boundaries
- Uniform point clouds
- A priori defined isosurfaces
- Using coarser meshes for output when hierarchy of meshes exist

## Extracting data

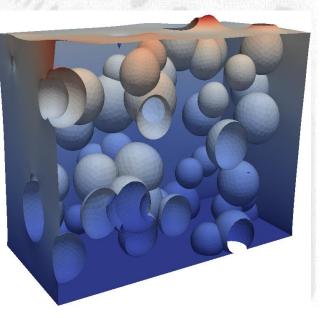
- Dimensional reduction (3D -> 2D)
- Averaging over time
- Integrals over BCs & bodies
- More robust I/O
  - Not all cores should write to disk in massively parallel simulations
  - HDF5+XDML output available for Elmer, mixed experiences

## **Example, File size in Swiss Cheese**

- Memory consumption of vtu-files (for Paraview) was studied in the "swiss cheese" case
- Saving just boundaries in single precision binary format may save over 90% in files size compared to full data in ascii
- With larger problem sizes the benefits are amplified

Binary output	Single Prec.	Only bound.	Bytes/ node
-	Х	-	376.0
Х	-	-	236.5
Х	Х	-	184.5
Х	-	Х	67.2
Х	Х	Х	38.5

Simulation Peter Råback, CSC, 2012.





## **Derived fields**



- Many solvers have internal options for computing derived fields (fluxes, heating powers,...)
- Elmer offers several auxiliary solvers
  - SaveMaterials: makes a material parameter into field variable
  - Streamlines: computes the streamlines of 2D flow
  - FluxComputation: given potential, computes the flux  $q = -c \nabla \phi$
  - VorticitySolver: computes the vorticity of flow,  $w = \nabla \times \phi$
  - PotentialSolver: given flux, compute the potential  $c \nabla \phi = q$
  - Filtered Data: compute filtered data from time series (mean, fourier coefficients,...)

- ..

- Usually auxiliary data need to be computed only after the iterative solution is ready
  - Exec Solver = after timestep
  - Exec Solver = after all
  - Exec Solver = before saving

## **Derived lower dimensional data**

- Derived boundary data
  - SaveLine: Computes fluxes on-the-fly
- Derived lumped (or 0D) data
  - SaveScalars: Computes a large number of different quantities on-the-fly
  - FluidicForce: compute the fluidic force acting on a surface
  - ElectricForce: compute the electrostatic froce using the Maxwell stress tensor
  - Many solvers compute lumped quantities internally for later use

(Capacitance, Lumped spring,...)



## Saving 1D data: SaveLine

- Lines of interest may be defined on-the-fly
- Flux computation using integration points on the boundary not the most accurate
- By default saves all existing field variables



## Saving 1D data: SaveLine...

```
Solver n
Equation = "SaveLine"
Procedure = File "SaveData" "SaveLine"
Filename = "g.dat"
File Append = Logical True
Polyline Coordinates(2,2) = Real 0.0 1.0 0.0 2.0
End
```

```
Boundary Condition m
Save Line = Logical True
End
```



## Saving OD data: SaveScalars

**Operators on bodies** 

- Statistical operators
  - Min, max, min abs, max abs, mean, variance, deviation
- Integral operators (quadratures on bodies)
  - volume, int mean, int variance
  - Diffusive energy, convective energy, potential energy

Operators on boundaries

- Statistical operators
  - Boundary min, boundary max, boundary min abs, max abs, mean, boundary variance, boundary deviation, boundary sum
  - Min, max, minabs, maxabs, mean
- Integral operators (quadratures on boundary)
  - area
  - Diffusive flux, convective flux

Other operators

nonlinear change, steady state change, time, timestep size,...



## Saving OD data: SaveScalars...

```
Solver n
  Exec Solver = after timestep
  Equation = String SaveScalars
  Procedure = File "SaveData" "SaveScalars"
  Filename = File "f.dat"
  Variable 1 = String Temperature
  Operator 1 = String max
 Variable 2 = String Temperature
  Operator 2 = String min
  Variable 3 = String Temperature
  Operator 3 = String mean
End
```

Boundary Condition m Save Scalars = Logical True End





## **Case: TwelveSolvers**

## Natural convection with ten auxialiary solvers

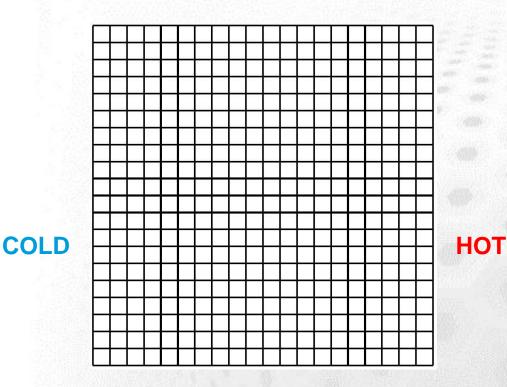
## **Case: Motivation**



- The purpose of the example is to show the flexibility of the modular structure
- The users should not be afraid to add new atomistic solvers to perform specific tasks
- A case of 12 solvers is rather rare, yet not totally unrealitistic

## **Case: preliminaries**

- Square with hot wall on right and cold wall on left
- Filled with viscous fluid
- Bouyancy modeled with Boussinesq approximation
- Temperature difference initiates a convection roll



CSC

## **Case: 12 solvers**

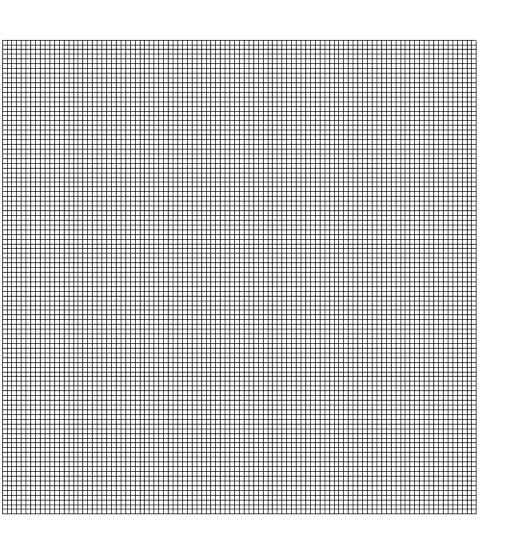
- 1. Heat Equation
- 2. Navier-Stokes

\_ \_ \_ \_ \_ \_ \_ \_ \_ \_

- 3. FluxSolver: solve the heat flux
- 4. StreamSolver
- 5. VorticitySolver
- 6. DivergenceSolver
- 7. ShearrateSolver
- 8. IsosurfaceSolver
- 9. ResultOutputSolver
- 10. SaveGridData
- 11. SaveLine
- 12. SaveScalars



## **Case: Computational mesh**

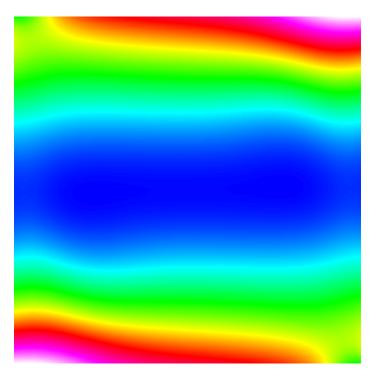


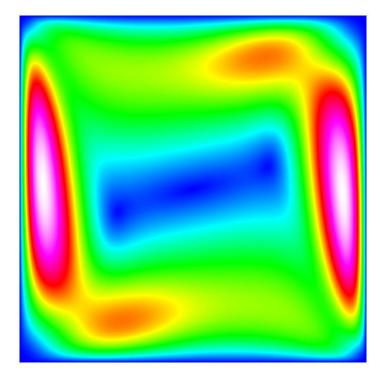
CSC

10000 bilinear elements

## **Case: Navier-Stokes, primary fields**





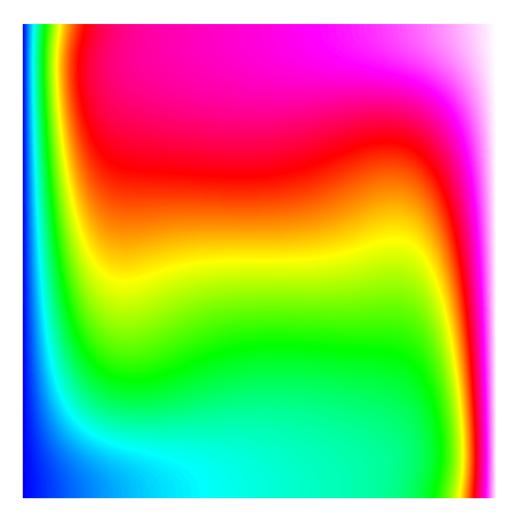


Pressure

Velocity

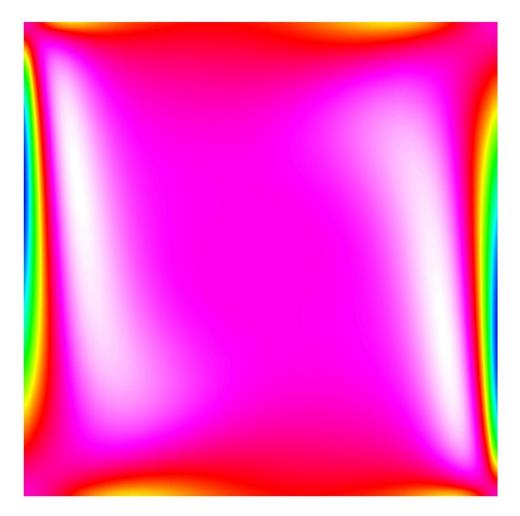
## **Case: Heat equation, primary field**





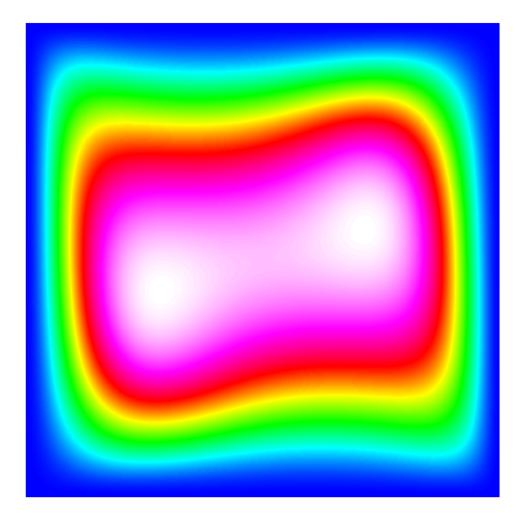
## **Case: Derived field, vorticity**





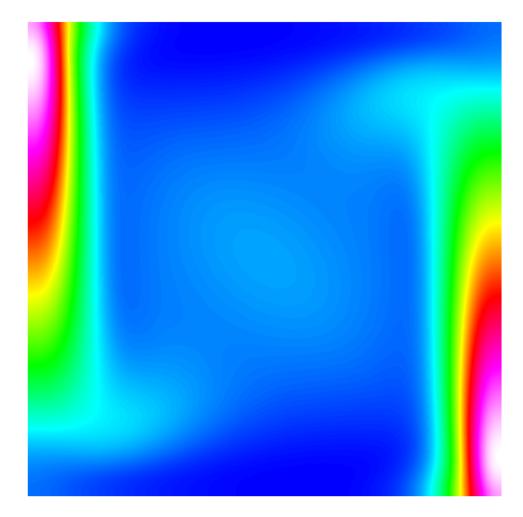
## **Case: Derived field, Streamlines**





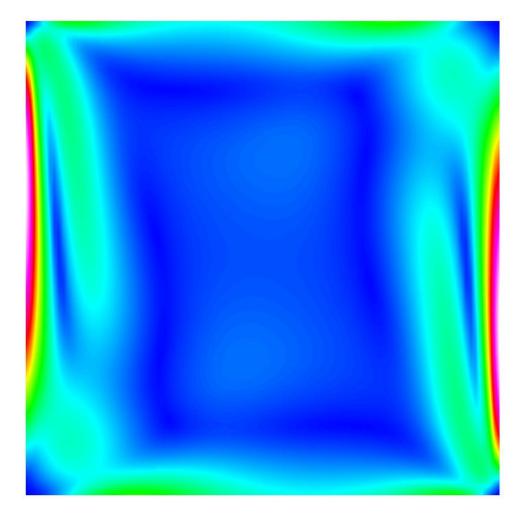
## **Case: Derived field, diffusive flux**





## **Case: Derived field, Shearrate**

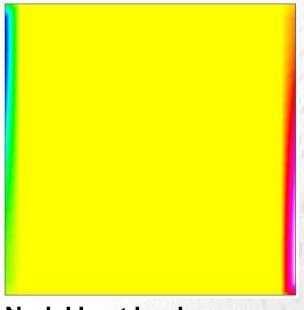




## **Example: nodal loads**

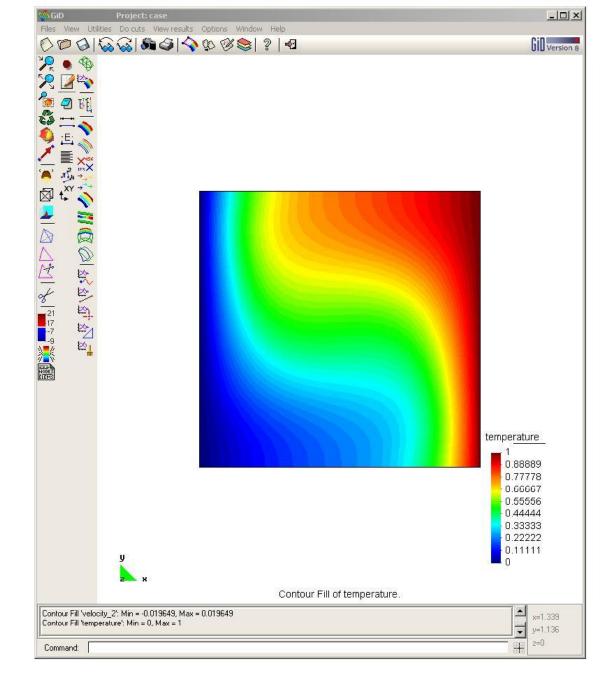


- If equation is solved until convergence nodal loads should only occur at boundaries
- Element size h=1/20 ~weight for flux

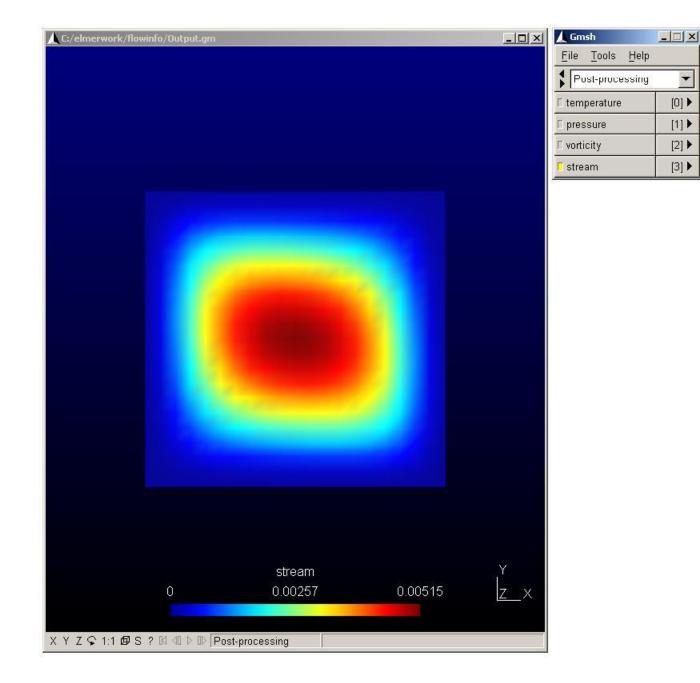


**Nodal heat loads** 

#### Example: view in GiD

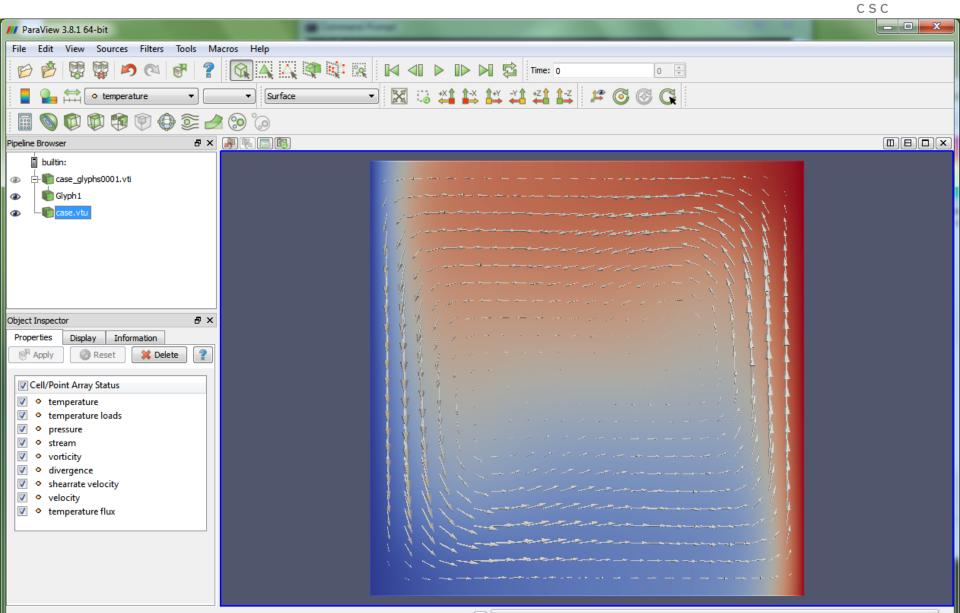


#### **Example:** view in Gmsh



-

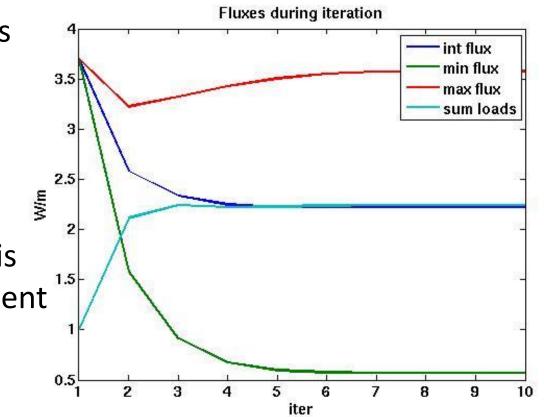
#### **Case: View in Paraview**



## **Example: total flux**



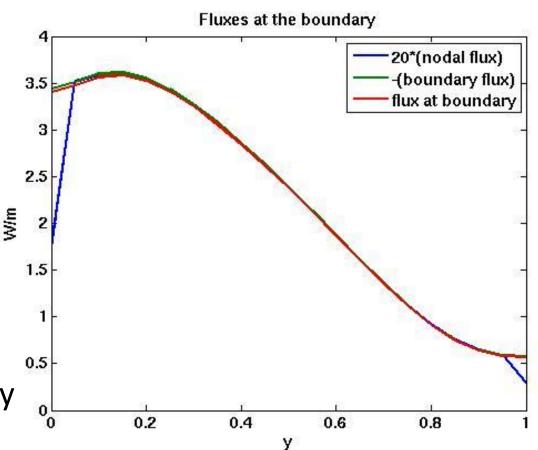
- Two ways of computing the total flux give different approximations
- When convergence is reached the agreement is good



CSC

## **Example: boundary flux**

- Saved by SaveLine
- Three ways of computing the boundary flux give different approximations
- At the corner the nodal flux should be normalized using only h/2



CSC

## Exercise



- Study the command file with 12 solvers
- Copy-paste an appropriate solver from there to some existing case of your own
  - ResultOutputSolver for VTU output
  - StreamSolver, VorticitySolver, FluxSolver,...
- Note: Make sure that the numbering of Solvers is consistant
  - Solvers that involve finite element solution you need to activate by Active Solvers
- Run the modified case
- Visualize results in ElmerPost or Paraview

## **Overcoming bottle-necks in postprocessing**

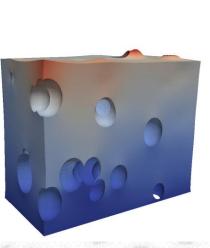


- Despite good visualization tools sometimes the amount of data may become a bottle-neck
- Reducing data
  - Saving only boundaries
  - Uniform point clouds
  - A priori defined isosurfaces
  - Using coarser meshes for output when hierarchy of meshes exist
- Extracting data
  - Dimensional reduction (3D -> 2D)
  - Averaging over time
  - Integrals over BCs & bodies
- More robust I/O
  - Not all cores should write to disk in massively parallel simulations
  - Preliminary HDF5+XDML output available for Elmer, mixed experiences

## Example, File size in Swiss Cheese

- Memory consumption of vtu-files (for Paraview) was studied in the "swiss cheese" case
- The ResultOutputSolver with different flags was used to write output in parallel
- Saving just boundaries in single precision binary format may save over 90% in files size compared to full data in ascii
- With larger problem sizes the benefits are amplified

Binary output	Single Prec.	Only bound.	Bytes/node
-	Х	-	376.0
Х	-	-	236.5
Х	Х	-	184.5
Х	-	Х	67.2
Х	Х	Х	38.5





## Example, saving boundaries in .sif file

```
Solver 2

Exec Solver = Always

Equation = "result output"

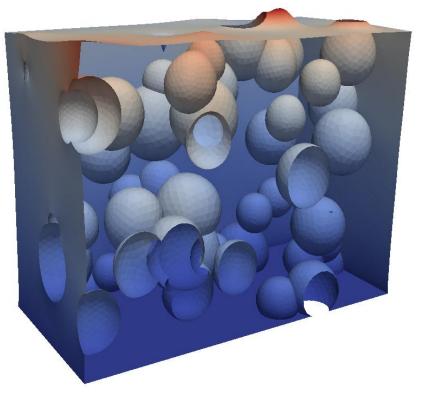
Procedure = "ResultOutputSolve" "ResultOutputSolver"

Output File Name = case

Vtu Format = Logical True

Save Boundaries Only = Logical True

End
```



## Conclusions



- It is good to think in advance what kind of data you need
  - 3D volume and 2D surface data
  - Derived fields
  - 1D line data
  - OD lumped data
- Often the same operations may be done also at later stages but with significantly greater effort



# **Visualization with Paraview**



# **Exporting 2D/3D data: ResultOutputSolve**

By setting suffix for **Post File** to **.vtu** paraview format is saved automatically.

An example shows how to save data in unstructured XML VTK (.vtu) files to directory "results" in single precision binary format.

```
Solver n
Exec Solver = after timestep
Equation = "result output"
Procedure = "ResultOutputSolve" "ResultOutputSolver"
Output File Name = "case"
Output Format = String "vtu"
Binary Output = True
Single Precision = True
Save Geometry Ids = True
End
```

### **Filename conventions**



- Suffix of unstructured XML based VTU file is .vtu
- Timesteps numbered #step
- Partitions numbered with #partpar#step
- Holder for vtu files in parallel is .pvtu

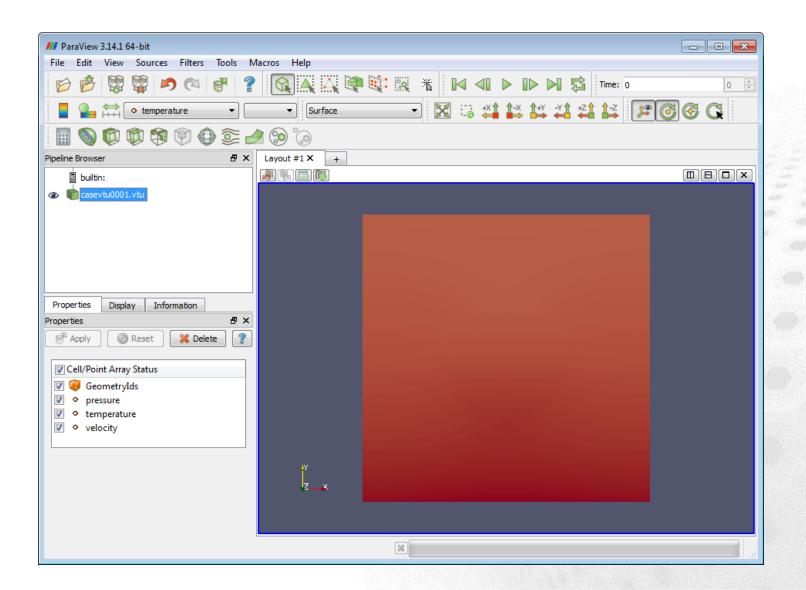
### Loading data

📶 ParaView 3.14.1 64	-bit	
File Edit View	Sources Filters Tools Macros Help	
68	🖉 🔊 🔍 💣 🦹 💽 🔍 🔍 🔍 🔍 🔍 🗮 🗮 🕷 🐁 🕪 🕨 🛸 Time: O	0
	) 🗐 🏵 🈂 🖉 😳	
Pipeline Browser	/// Open File: (open multiple files with < ctrl> key.)	
builtin:	Look in: C:/elmerwrk/Viz/    G O G	
Properties Display Properties Rapply @	My Documents   Desktop   Favorites   A:\   C:\   D:\   F:\   F:\	
	File name:     casevtu0001.vtu       Files of type:     Supported Files (*.xyz *.okc *.h5 *.vsh5 *.vld *.rst *.POS* *.CHG*    Cancel	Ø

CSC

Note: Paraview may have several datasets at the same time!

### Solid color



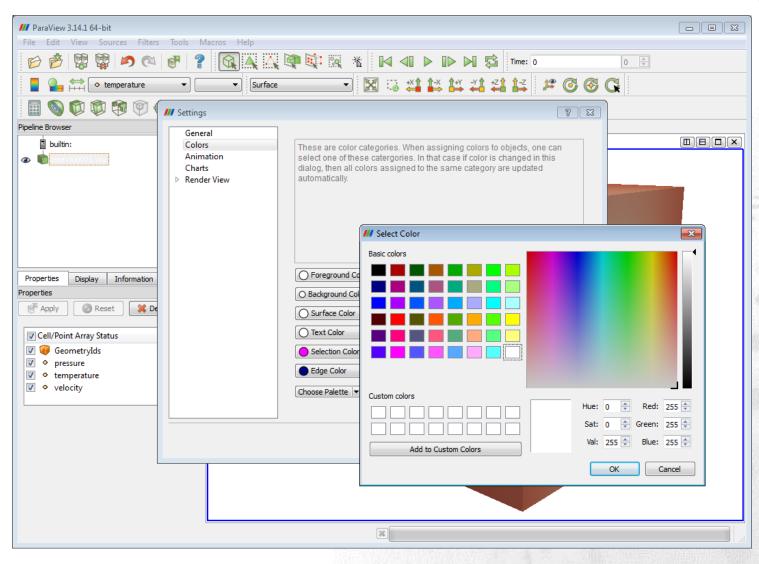


# **Moving object in Paraview**

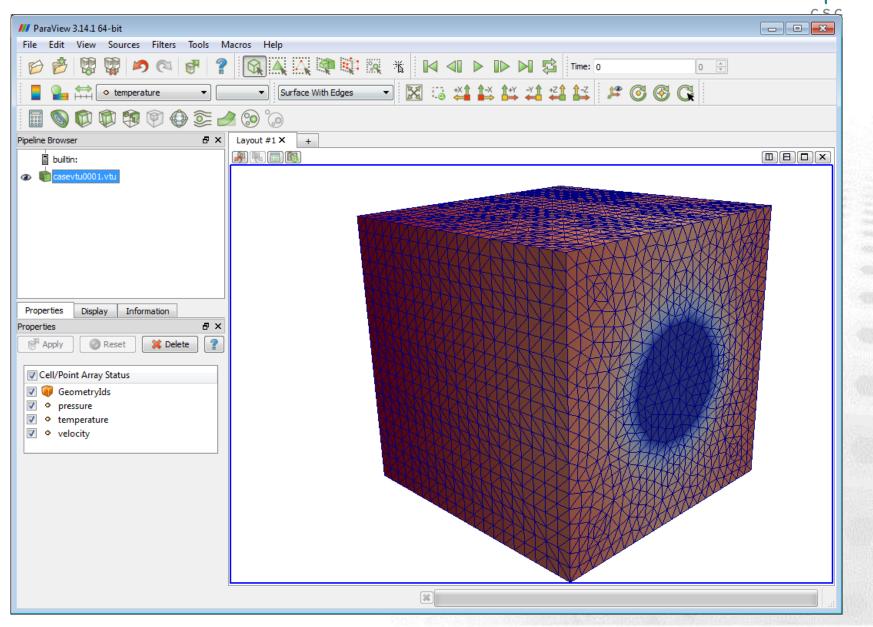


- Rotate
  - Mouse: Left bottom
- Scale
  - Mouse: Right bottom
- Translate
  - Mouse: Center bottom

### Setting background color



### **Color mesh with surface + edges**



AMR Contour

AMR Dual Clip

- Annotate Time Filter
- \_\_\_\_\_

Append Attributes

Append Datasets

Append Geometry

Block Scalars Calculator

Cell Centers Cell Data to Point Data

Clean

Clean Cells to Grid

Clean to Grid

🗊 Clip

Clip Closed Surface Clip Generic Dataset

Compute Derivatives

. Connectivity

Contingency Statistics

Contour

Contour Generic Dataset Curvature

D3

Decimate

Delaunay 2D

Delaunay 3D

**Descriptive Statistics** 

Elevation

Extract AMR Blocks

Extract Block

Extract CTH Parts

Extract Cells By Region Extract Edges

Extract Generic Dataset Surface

Extract Level
 Extract Selection

Extract Subset

Extract Surface

FFT Of Selection Over Time FOF/SOD Halo Finder

Feature Edges

Gaussian Resampling

Generate Ids

Generate Quadrature Points

Generate Quadrature Scheme Dictionary

Generate Surface Normals

Glyph
 Glyph With Custom Source

Gradient Gradient Of Unstructured DataSet Grid Connectivity Group Datasets

Histogram Image Data to Point Set

Integrate Variables Interpolate to Quadrature Points

Intersect Fragments Iso Volume K Means

Level Scalars Linear Extrusion

Loop Subdivision

Mask Points

- Material Interface Filter
- Median

Merge Blocks Mesh Quality

#### Multicorrelative Statistics

Normal Glyphs Octree Depth Limit Octree Depth Scalars

> Outline Outline Corners Outline Curvilinear DataSet

Particle Pathlines

ParticleTracer Plot Data

Plot Global Variables Over Time

Plot On Intersection Curves Plot On Sorted Lines

💉 🛛 Plot Over Line

- Plot Selection Over Time Point Data to Cell Data Principal Component Analysis
  - Probe Location
- Process Id Scalars
- {...} Programmable Filter Python Calculator

Quadric Clustering Random Vectors

Rectilinear Data to Point Set Rectilinear Grid Connectivity

Reflect Resample With Dataset

Ribbon

Rotational Extrusion

Scatter Plot Shrink

Slice Generic Dataset

Stream Tracer
 Stream Tracer For Generic Datasets
 Stream Tracer With Custom Source
 Subdivide
 Surface Flow
 Surface Vectors
 Table To Points
 Table To Structured Grid
 Temporal Cache
 Temporal Interpolator
 Temporal Shift Scale
 Temporal Statistics
 Tessellate

Tetrahedralize Texture Map to Cylinder Texture Map to Plane Texture Map to Sphere

Threshold
Transform
Triangle Strips
Triangulate

Tube

Warp By Scalar

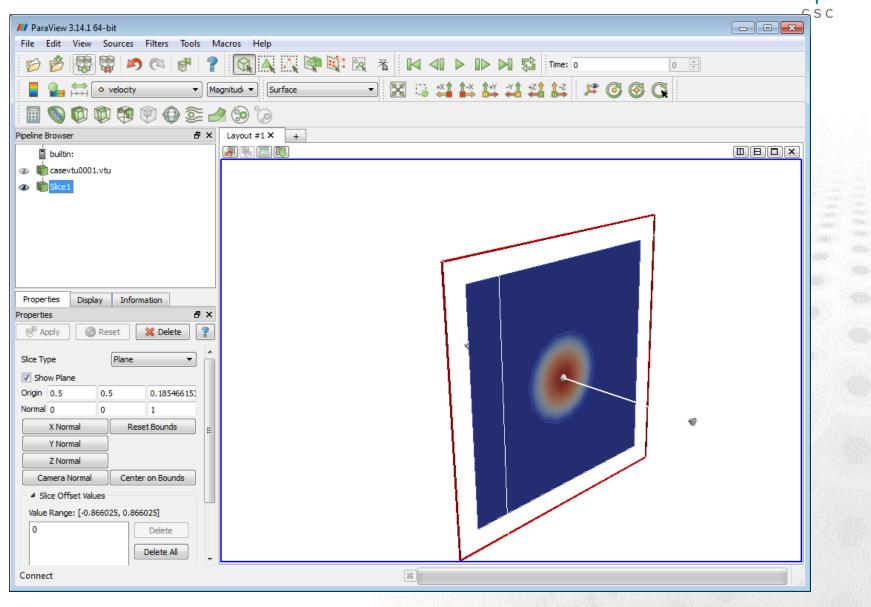
Warp By Vector Youngs Material Interface builtin:
 builtin:
 case0001.pvtu
 Connectivity1
 Slice1
 Glyph1



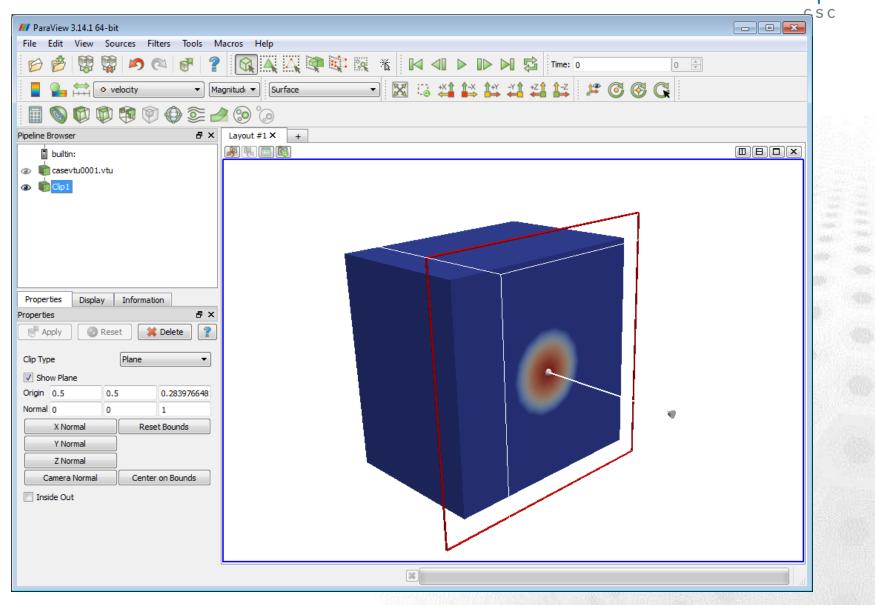
 Paraview uses extensively *filters* to create new datasets

- Filters and datasets may be set active or passive by clicking the eye
- Several datasets may be visualized at the same time

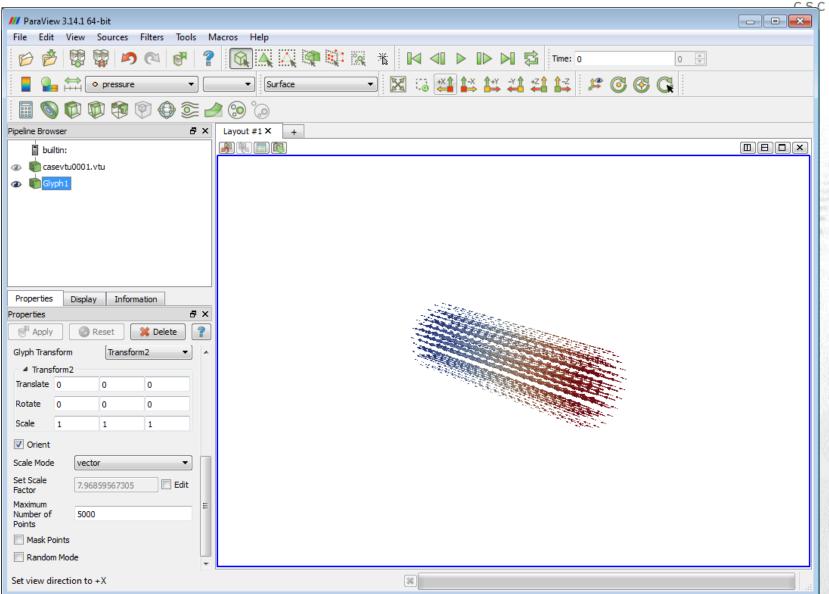
### **Plotting a slice**



### **Plotting a clip**



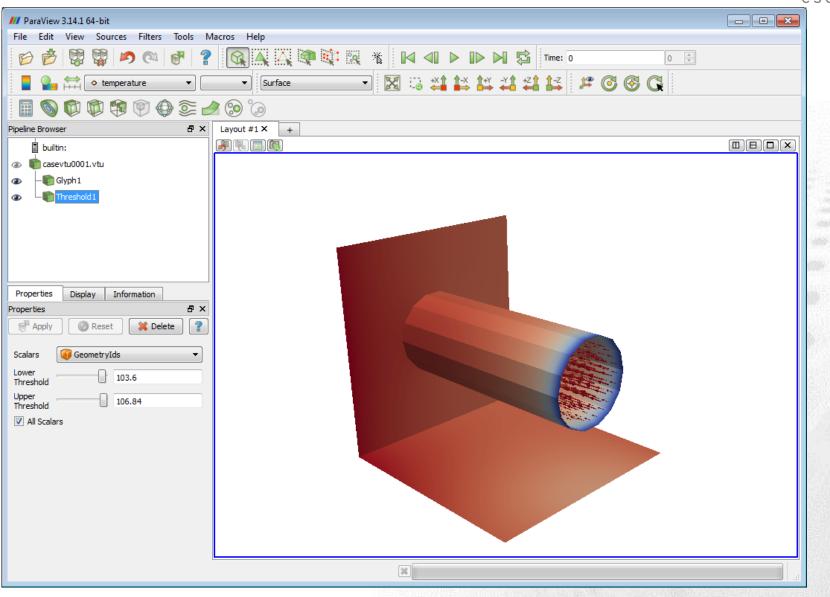
### **Vector plot**



### **Vector plot + opaque solid surface**

/// ParaView 3.14.1 64-bit			
File Edit View Sources Filters Tools Macros Help			
🛿 😥 🖄 🞇 🔊 🔍 🕐 🧣 🧌 🦗 🎇 👯 🕅 🐴 🕨 🖉 🔝 Time: O 🕞 🛛			
Surface			
Pipeline Browser B × Layout #1 × +			
Properties   Display     Style   Representation   Surface   Interpolation   Gouraud   Point size			
Line width 1,00 + Opacity 0,10 + Subdivision 1 + Edge Style Set Edge Color + Volume Volume Projected tetra +			

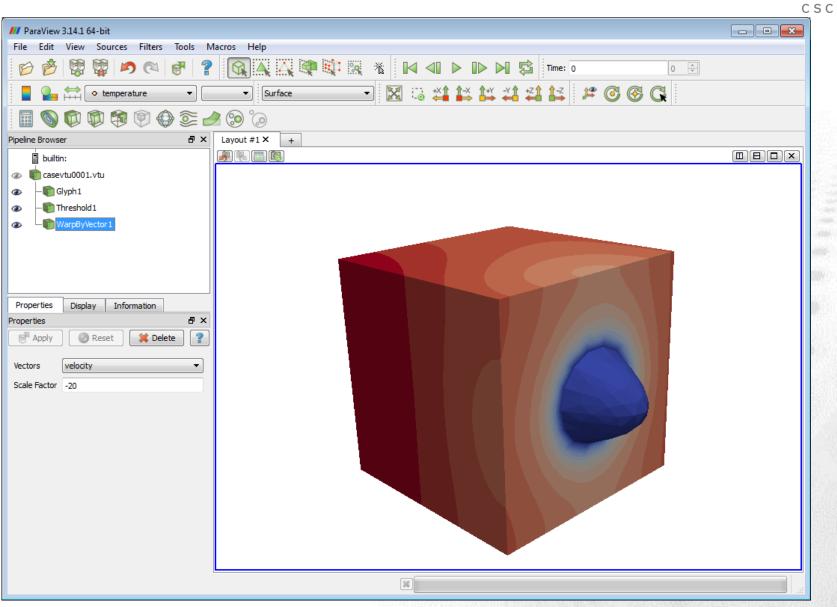
### Vector plot + solid surface with Id treshold



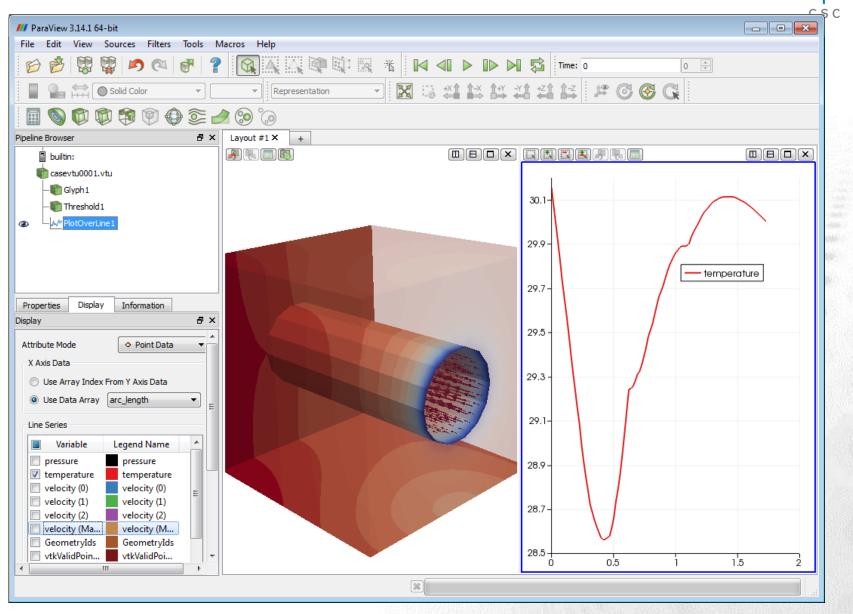
# **Change of colormap**

		<u> </u>
M ParaView	3.14.1 64-bit	
File Edit	View Sources Filters Tools Macros Help	
ÐB	11 Color Scale Editor Time: 0	
	Color Scale Color Legend	
	Render View Immediately     Save     Choose Preset	
Pipeline Brow		
<ul> <li>② </li> <li>② </li> <li>■ </li> <li></li></ul>	Color Scalar Value NaN Color Color Space Diverging	
o -	Use Logarithmic Scale	
	V Automatically Rescale to Fit Data Range	
	Minimum: 10 Maximum: 34.3139	
	Rescale Range         Rescale to Data Range         Rescale to Temporal Range	
	✓ Use Discrete Colors	
	Resolution 16	
Properties		
Display		
View		
Visible	Apply Make Default Close	
V Select		
Color		
🔽 Interpola	ate Scalars	
√ Map Sca	lars	
Apply Textu	re None	
Color by		
	Edit Color Map Rescale to	
Slice		
Slice Directio		
Slice		
•		

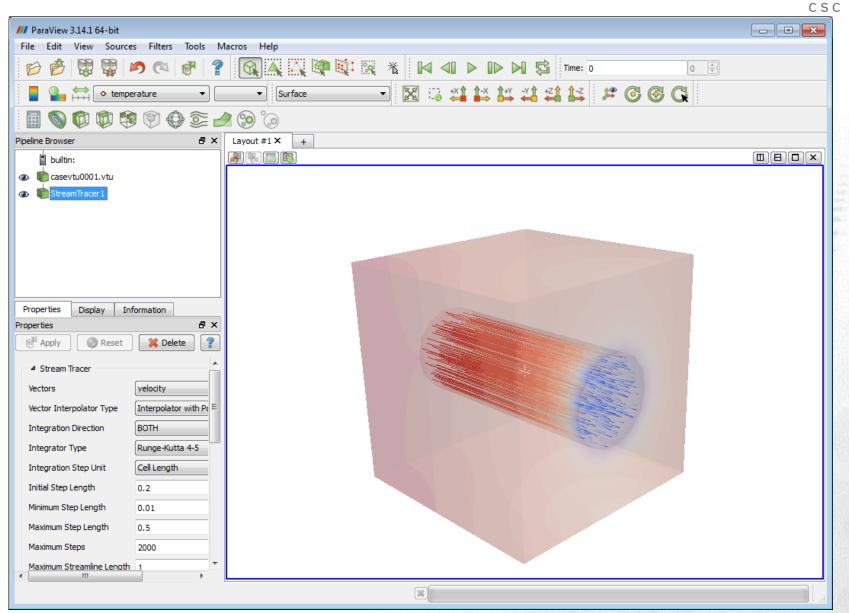
## **Deformation – WarpByVector filter**



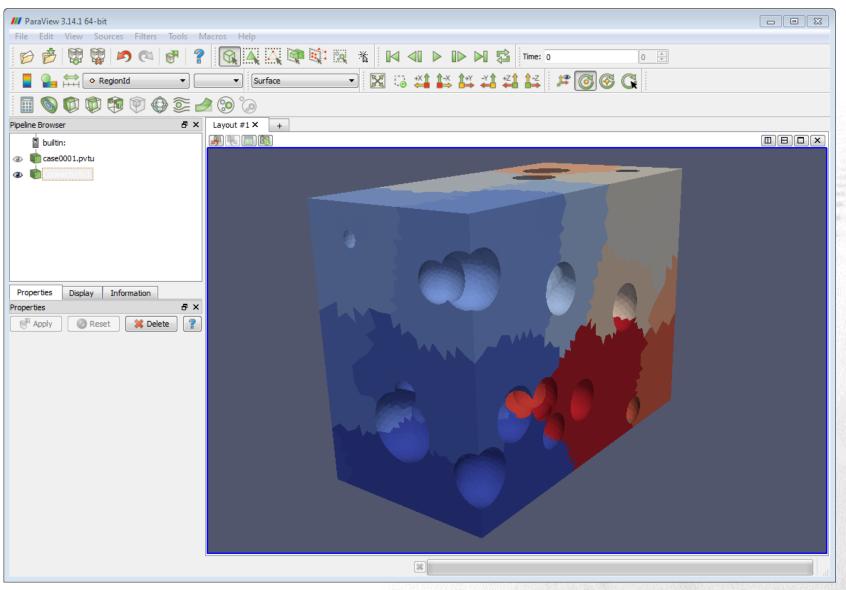
### **Plot line – PlotOverLine filter**



### **Streamlines – Filter StreamTracer**



### **Partitioning – Connectivity filter**



# **Saving figures**

Image: Application of the state of the	
😥 🥙 🕎 💋 🔍 🕐 🧣 🦳 🍇 🔍 🎕 💱 🔣 🐁 🕪 💵 ы 🖓 🌠 Time: O 💿 🖯	
Pipeline Browser B × Layout #1 × +	
III Save Screenshot:	
Look in: C:/elmerwrk/Viz/	
My Docun 🔺 Filename	
Desktop Favorites	
Dia 🚺 Viz	
File name: case.png OK	
Files of type: PNG image (*.png)	
✓ Map Scalars	
Apply Texture None	
Color by <a>temperature</a>	
Edit Color Map Rescale to	
Sice	
Slice Direction	
Slice 0	
	the later

### **Saving animations with Paraview**

- CSC
- The only packing method that comes with Paraview by default is motion AVI
- It is advicable to save the animation as separate files
- You may use ElmerClips to make mpg animations of the separate png figures



# **Visualization with ElmerPost**

### How to write files for ElmerPost

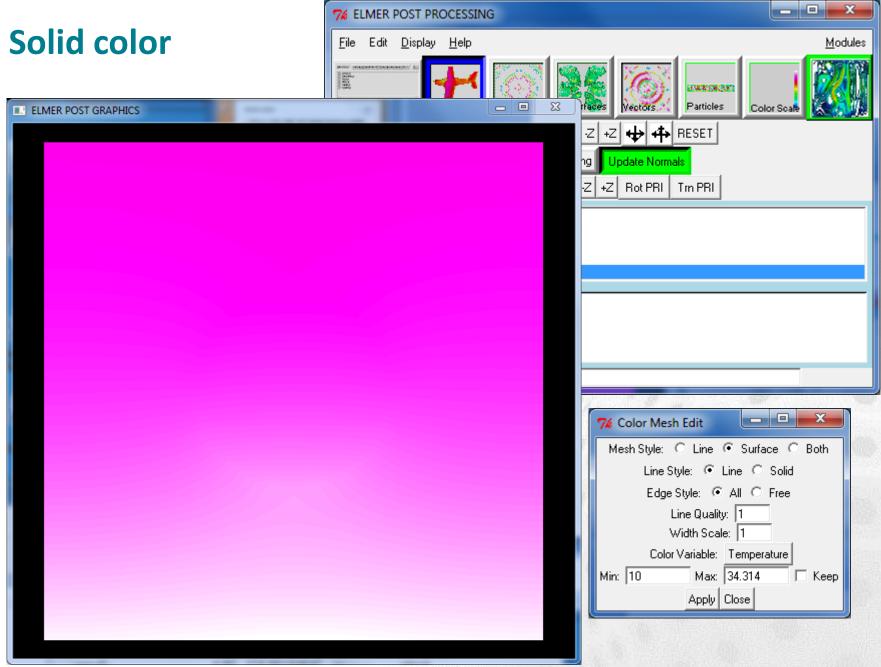
- Default suffix is .ep
- May be requested in Simulation section Post File = case.ep
- Or using ResultOutputSolver with
  Output format = ElmerPost



### **Loading data**

- Assume data in case.ep
- File -> Open ->
  case.ep
- Here the timesteps are chosen
- If element edges or sides are not defined for BCs they may have to be created here

CSC		
🎋 Read Model File 📃 🖂		
Status: Header Read		
Options:		
Generate Surface Element Sides		
🧮 Generate Volume Element Sides		
Generate Volume Element Edges		
File Information:		
Nodes: 11949 Elements: 69792 Timestps: 2 DOFS: 5 Vector: Velocity Scalar: Pressure Scalar: Temperature		
Select timesteps:		
First: 1 Last: 1 Increment 1 All		
Select file:		
Model file: C:/elmerwrk/Viz/case.ep Browse		
Read header Read file OK Close		



# **Moving object in ElmerPost**



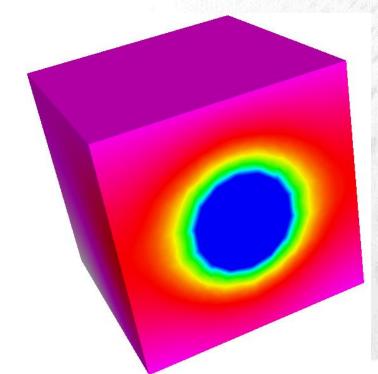
- Rotate
  - Mouse: Right bottom
  - Click:
  - Command line, e.g.: rotate 30 45 60
- Scale
  - Mouse: Both bottoms
  - Click: 🕁 🕁
  - Command line: scale 1 10 1
- Translate
  - Mouse: Left bottom
  - Click: ⇐⇒♥♠
  - Command line: translate 1 2 3

# Setting background color

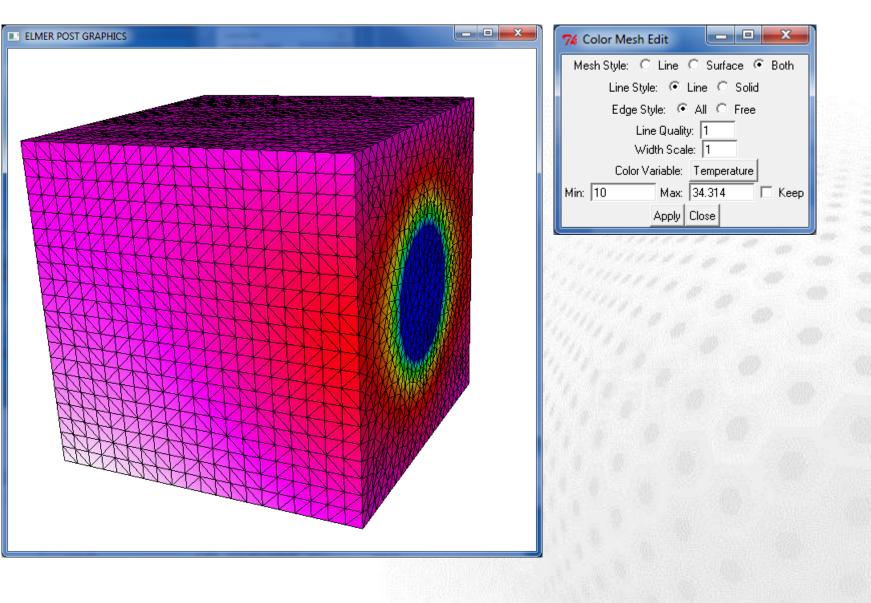
CSC

### Olick:

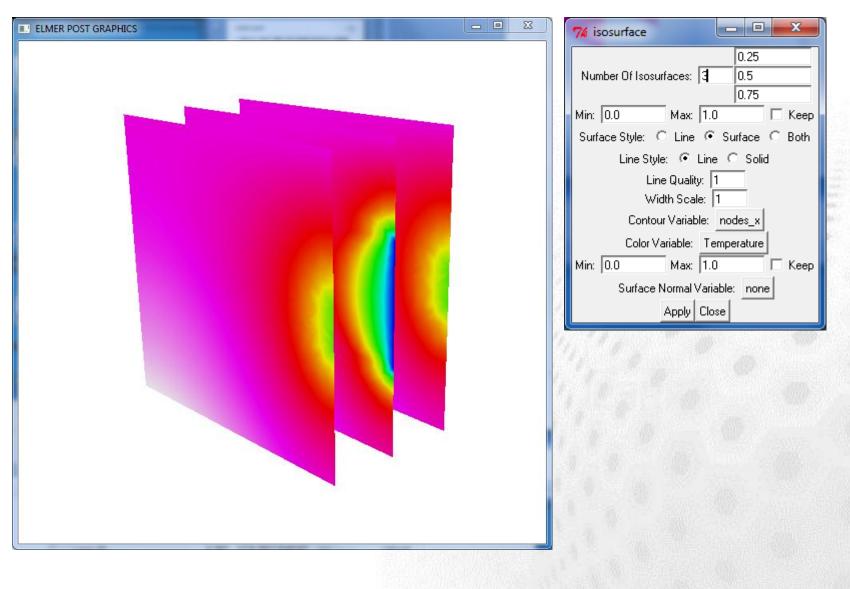
- Edit -> Background
- Set 100.0 100.0 100.0 for white
- Command line
  - background 100 100 100



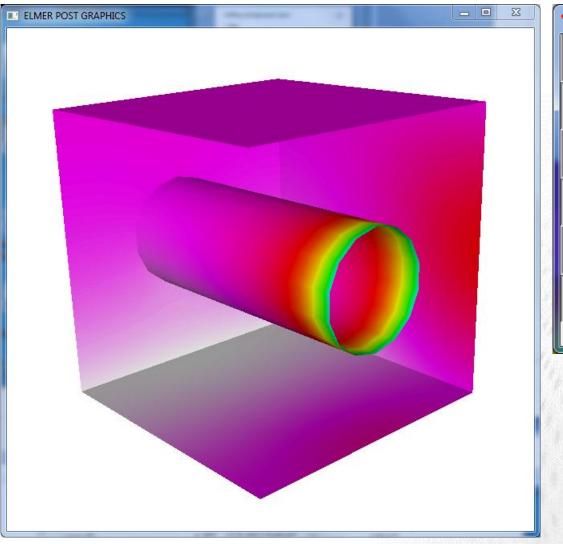
### **Color mesh with surface + edges**



### **Plotting isosurfaces**

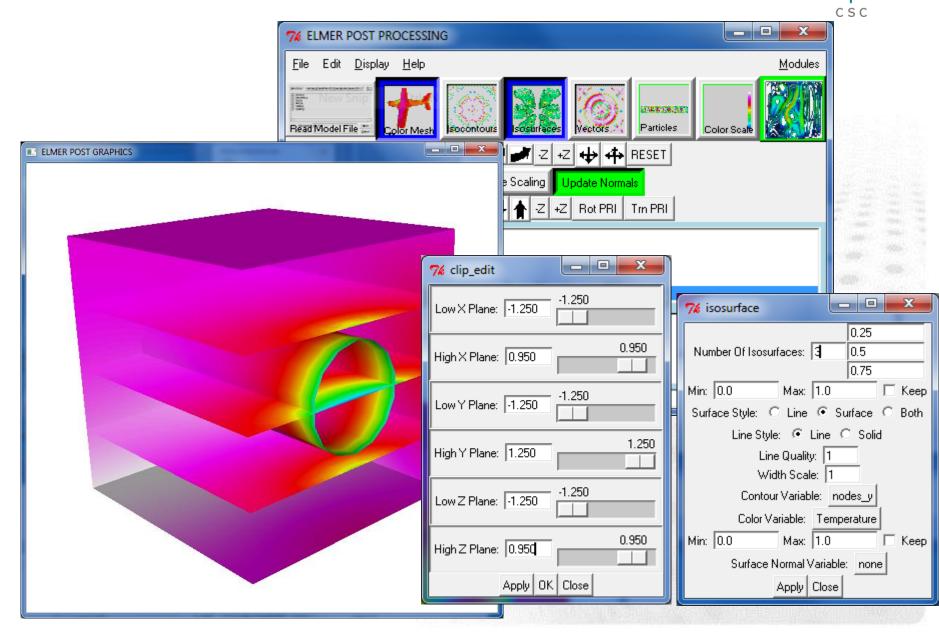


# Using clip planes



% clip_edit	_ <b>D</b> X
Low X Plane: -1.250	-1.250
High X Plane: 0.950	0.950
Low Y Plane: 1.250	-1.250
High Y Plane: 1.250	1.250
Low Z Plane: 1.250	-1.250
High Z Plane: 0.950	0.950
Apply Of	Close

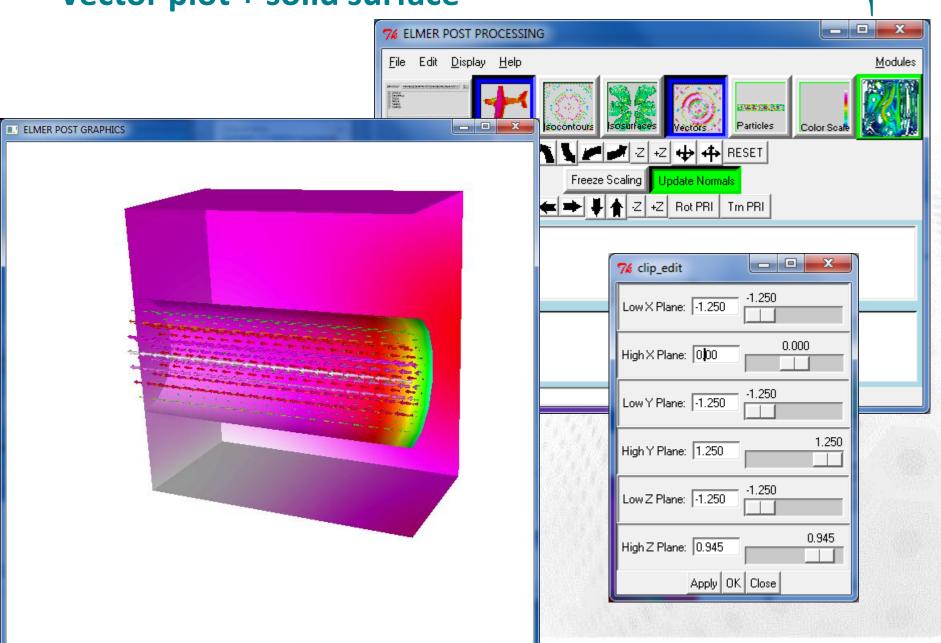
### Isosurface + surface plot + clip planes

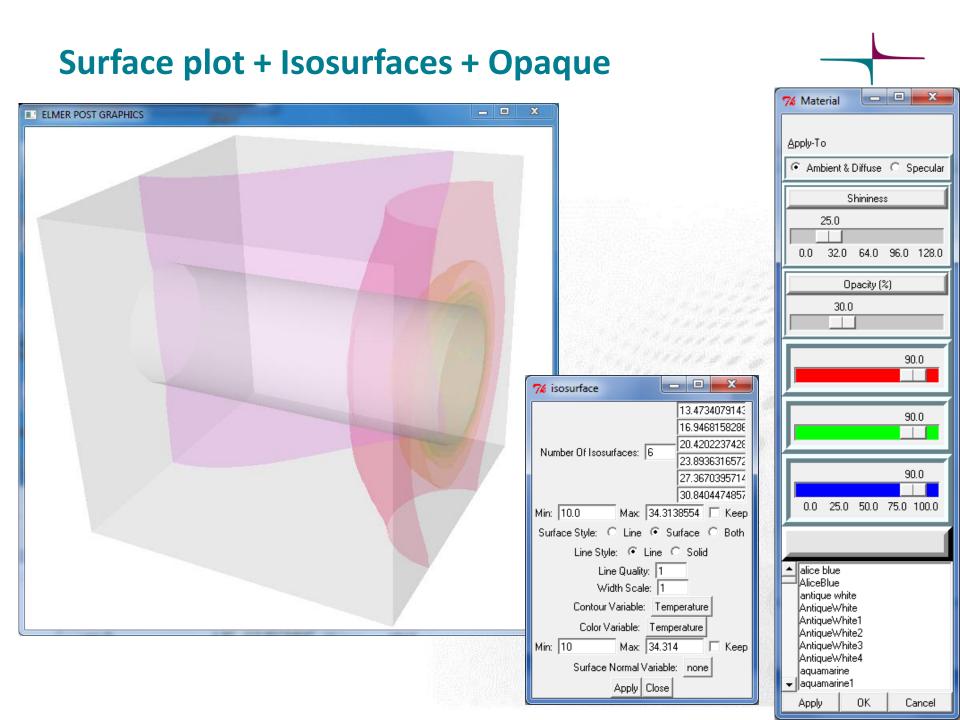


# **Vector plots**

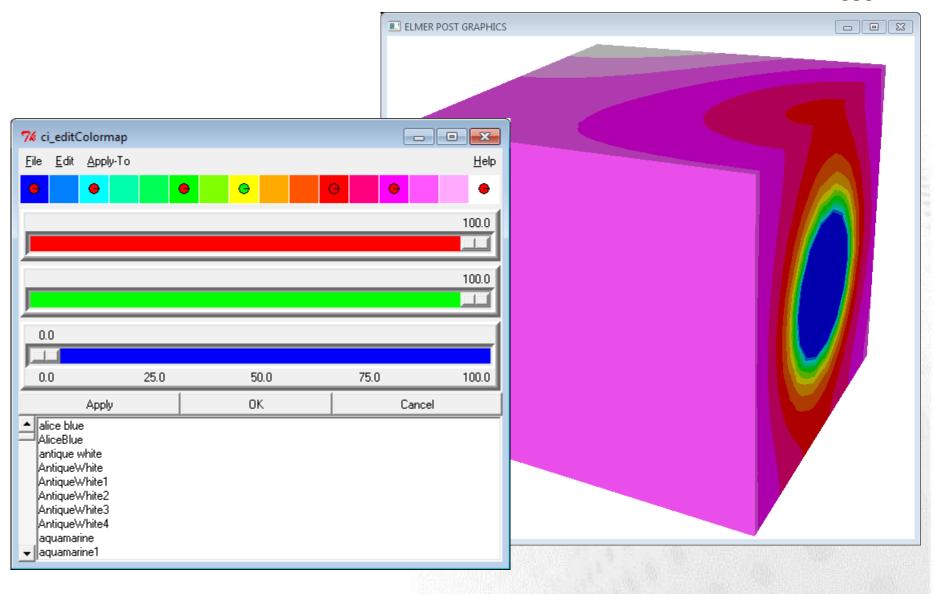
ELMER POST GRAPHICS	Vector         Vector Length Scale:         Line Style:         Line Quality:         Width Scale:         Threshold Variable:         Nin:         0.0         Max:         1.0         Color Variable:         Velocity_abs	
	Length Variable: Velocity_abs Arrow Variable: Velocity Apply Close	

## **Vector plot + solid surface**

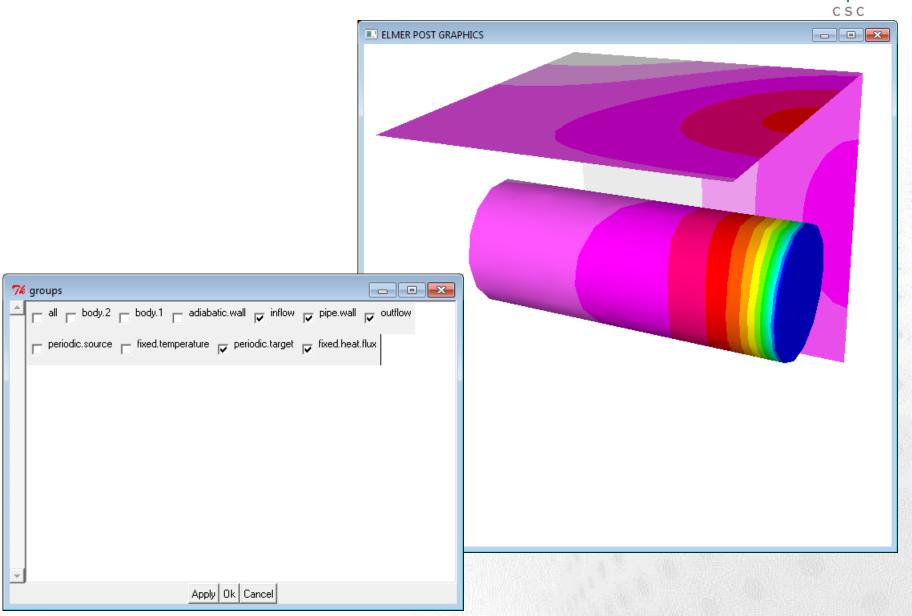




# **Change of colormap**



# Selecting active geometric entities

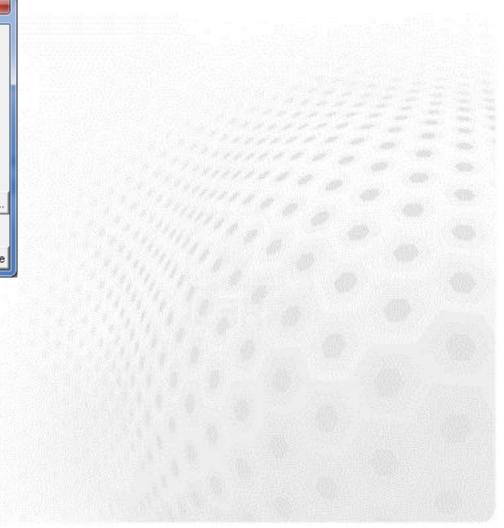


# **Saving figures**



### File -> Save Image -> jpg

76 Save Screen	
Save as:	
O Postscript	
Fit PS to page	
C PPM Image	
JPG Image	
Select file:	
File Name:	Browse
	Save Close
	Save Close



### **Deformation in geometry**



- Assume displacement field in variable "Displacement"
- Set in command windows: math n0=nodes math nodes=n0+Displacement
- Replot