Scientific Visualization
An Introduction

Featuring ParaView

Vetria L. Byrd, PhD
Assistant Professor

International High Performance Computing Summer School
Ljubljana, Slovenia
Wednesday, June 29, 2016
About Me

Vetria L. Byrd, PhD

Academic Preparation
• Computer Science (PhD, MS)
• Biomedical Engineering (MSMBE)

Where I Am Now
Assistant Professor
Purdue University
Computer Graphics Technology
Research Focus: Data Visualization

What I’ve Done
Visualization Initiatives
• Research Experience for Undergraduates in Collaborative Data Visualization Applications (2014/2015)
• BPViz: Broaden Participation in Visualization (2014/2016)
  • August 3 – 4, 2016
  • Purdue/NCSA

Agent for “Insight”
AGENDA

INTRODUCTION TO SCIENTIFIC VISUALIZATION

- High Level Overview
- Purpose Of visualization
- Visualization Applications
- Data Visualization Process
- Scientific Visualization Process
- Introduction to Scientific Visualization Using ParaView
  - Hands-on Workshop
- Additional Resources
Slides Available on the XSEDE Wiki
What is the purpose of Visualization?
“The purpose of visualization is “insight”, not pictures.”

~Ben Shneiderman
What does Insight lead to?
“Insight” Leads to . . .

Discovery

Source: greenbookblog.org
“Insight” Leads to . . .

Discovery

Visualizing Patterns over Time

Source: greenbookblog.org
“Insight” Leads to . . .

Discovery

Spotting Differences
"Insight" Leads to . . .

Discovery

Spotting Differences

How many 7’s do you see?
“Insight” Leads to . . .

Decision Making

Allows users to answer questions they didn’t know they had

Human Genome Project
https://pradipjntu.files.wordpress.com/2011/05/molecularmachine.jpg
“Insight” Leads to . . .

Analysis Of Data

The Challenger Disaster

http://en.wikipedia.org/wiki/33
File: Challenger_explosion.jpg
"Insight" Leads to . . .

**Explanation** Visualizing Spatial Relationships

http://datafl.ws/197

http://datafl.ws/198

Can anyone think of another reason why “insight” is important?
“Insight” Tells a Story.

The use of visualization in the analysis of pollution and air flows in Manhattan, New York City.
“Insight” Tells a Story

Napoleon’s Invasion of Russia in 1812 By Jacque Minard
“Insight” Tells a Story

Napoleon’s Invasion of Russia in 1812 By Jacque Minard

Army Size: 422,000

Path of retreat
“Insight” Tells a Story

Napoleon’s Invasion of Russia in 1812 By Jacque Minard

Army Size: 422,000

Path of retreat


The number of men present was represented by the width of the colored zone at a rate of one million for every ten thousand men; they are further written across the zones. They designate the men who fell in Russia, the Black, those who have died. The information which has been used to draw up the map has been extracted from various reports of official sources of Chavagne and the unpublished diary of M. Planard, personal secretary of the army from October 21st. Along with the other members of the army, I have assumed that the army of France was made up of three parts: the vanguard, the main body, and rear guard. The band indicates the size of the army at each position.

Moscow

Path of retreat

GRAPHIC TABLE of the temperature in degrees of the Rössner thermometer below zero.

-26 -30 -11 -21 -9
“Insight” Tells a Story

Napoleon’s Invasion of Russia in 1812 By Jacque Minard

Armies Size: 422,000

Path of retreat

Army Size: 100,000

Figure 2: Graphic Table of the Temperature in Degrees of the Reaumur Thermometer below Zero

-26, -30, -11, -21, -9
“Insight” Tells a Story

Napoleon’s Invasion of Russia in 1812 By Jacque Minard

Army Size: 422,000

Path of retreat

Army Size: 100,000

Moscow

Width of band indicates the size of the army at each position

Temperature

-26 -30 -11 -21 -9
"Insight" Tells a Story

Napoleon's Invasion of Russia in 1812 By Jacque Minard


The number of men present are represented by the width of the colored zone at a rate of one million for every 10,000 men. They are further written across the zones. The red designates the men who entered Russia, the black those who died. The information which serves to draw up the map has been extracted from the "Mémoires du Sénat," and "Nouvelles de France," and the unpublished papers of D'Ansembourg.

Path of retreat

Army Size: 100,000

Army Size: 422,000

Army Size: 10,000

Width of band indicates the size of the army at each position

Temperature

-26
-30
-11
-21
-9

Moscow
“Insight” Tells a Story

Napoleon’s Invasion of Russia in 1812 By Jacque Minard

Best Statistical Graphic, Ever!


The number of men present are represented by the widths of the shaded zones at a rate of one million for every ten thousand men; they are further divided across the zones. The red designates the men who entered Russia, the black those who left it. The information which serves to draw up the map has been extracted from M. Taine’s Chapters of Napoleon, the unpublished papers of General Charette, and the unpublished papers of Napoleon, 1812. The information which serves to draw up the map has been extracted from M. Taine’s Chapters of Napoleon, the unpublished papers of General Charette, and the unpublished papers of Napoleon, 1812.

Width of band indicates the size of the army at each position.

Path of retreat

Temperature

Army Size: 100,000

Army Size: 422,000

Army Size: 10,000
Insight

Analysis of Data

Explanation

Tells a Story

Discovery

Decision Making
Visualization Applications

Why is visualization important?
Biovisualization (BioVis)

The visualization of biological data;
Often grouped with computer animation
Information Visualization (InfoVis)
Interdisciplinary Study of the “visual representation of large-scale collections of non-numerical information

Internet Usage
Source: http://www.cernea.net/wp-content/uploads/2013/03/internet.gif
Geographic Visualization

GeoVis

Communicates geospatial information in ways that, when combined with human understanding, allow for data exploration and decision-making processes.

Scientific Visualization (SciVis)

Primarily concerned with the visualization of three-dimensional phenomena

Emphases on realistic renderings of volumes, surfaces, illumination sources, etc.

http://www.tinkering.net/sciviz/
Scientific Visualization (SciVis)

*The focus of this workshop*

Primarily concerned with the visualization of three-dimensional phenomena

Emphases on realistic renderings of volumes, surfaces, illumination sources, etc.

http://www.tinkering.net/sciviz/
Data Visualization Process
High Level Overview
When do you think about visualizing your data?
Data Visualization Process

acquire → parse → filter → mine → represent → refine → interact

Adopted from Visualizing Data: Exploring and Explaining Data with the Processing Environment by Ben Fry, O’Reilly (p 15)
Data Visualization Process

obtain the data

acquire → parse → filter → mine → represent → refine → interact

Adopted from Visualizing Data: Exploring and Explaining Data with the Processing Environment by Ben Fry, O’Reilly (p 15)
Data Visualization Process

obtain the data

provide structure

acquire → parse → filter → mine → represent → refine → interact

Adopted from Visualizing Data: Exploring and Explaining Data with the Processing Environment by Ben Fry, O'Reilly (p 15)
Data Visualization Process

1. **obtain the data**
2. **provide structure**
3. **remove all but the data of interest**

Steps:
- *acquire*
- *parse*
- *filter*
- *mine*
- *represent*
- *refine*
- *interact*
Data Visualization Process

1. Obtain the data
2. Provide structure
3. Remove all but the data of interest
4. Apply methods from statistics or data mining to discern patterns or place the data in mathematical context

Adopted from Visualizing Data: Exploring and Explaining Data with the Processing Environment by Ben Fry, O’Reilly (p 15)
Data Visualization Process

- Acquire
- Parse
- Filter
- Mine
- Represent
- Refine
- Interact

1. Obtain the data
2. Provide structure
3. Remove all but the data of interest
4. Choose a basic visual model, such as a bar graph, list or tree
5. Apply methods from statistics or data mining to discern patterns or place the data in mathematical context

Adopted from Visualizing Data: Exploring and Explaining Data with the Processing Environment by Ben Fry, O’Reilly (p 15)
Data Visualization Process

1. **Obtain the Data**
2. **Provide Structure**
3. **Remove all but the data of interest**
4. **Choose a basic visual model, such as a bar graph, list or tree**
5. **Apply methods from statistics or data mining to discern patterns or place the data in mathematical context**
6. **Improve the basic representation to make it clearer and more visually engaging**

Adopted from Visualizing Data: Exploring and Explaining Data with the Processing Environment by Ben Fry, O’Reilly (p 15)
Data Visualization Process

1. **Obtain the data**
2. **Provide structure**
3. **Remove all but the data of interest**
4. **Choose a basic visual model, such as a bar graph, list or tree**
5. **Apply methods from statistics or data mining to discern patterns or place the data in mathematical context**
6. **Refine the basic representation to make it clearer and more visually engaging**
7. **Interact**

Adopted from Visualizing Data: Exploring and Explaining Data with the Processing Environment by Ben Fry, O’Reilly (p 15)
An iterative process

1. Obtain the data
2. Provide structure
3. Remove all but the data of interest
4. Choose a basic visual model, such as a bar graph, list or tree
5. Apply methods from statistics or data mining to discern patterns or place the data in mathematical context
6. Improve the basic representation to make it clearer and more visually engaging
7. Add methods for manipulating the data or controlling what features are visible
Taking raw data and converting it to a form that is viewable and understandable to humans.
There are several steps between raw data and a finished visualization.
Why do we care?

Why should you care?
Data visualization is becoming an increasingly important component of analytics in the age of big data (SAS: Five big data challenges and how to overcome them with visual analytics)


Between now and 2020, the information in the Digital Universe will grow by a factor of 44; the number of “files” in it to be managed will grow by a factor of 67

When should you think about visualizing your data?

As early as possible
Research Experience for Undergraduates in Collaborative Data Visualization Applications

Purdue Polytechnic Institute Computer Graphics Technology Department invites undergraduates with an interest in visualization to participate in cutting-edge undergraduate research at Purdue University.

The VisREU Experience: 8-Weeks, Paid Research, Professional Development, Network Building, Developing Lifetime Transferrable Skills

What is Visualization
Visualization is the process of transforming raw data into a visual representation of relationships that exist within the data. Visualization leads to insight, better understanding of relationships between variables that exist in the data. Regardless of your academic major, at some point during your academic life you will have a need to visualize something. The VisREU Site is the perfect opportunity to learn about the visualization pipeline, applications and tools to help you create impactful visualizations.

About the Program
The program identifies research collaborators with visualization needs and assigns each student to a...
Visualization is the tool that will take us forward from the traditional output of high performance computing (HPC) that we are used to into a visual medium that allows researchers to collaborate and elaborate on the finding's they’ve got.

Tim Carroll
Director and Global Lead,
Dell Research Computing Solutions
HPC Source (Spring 2011)
• Primarily concerned with the visualization of three-dimensional phenomena (architectural, meteorological, medical, biological, etc.),
Scientific Visualization

- Primarily concerned with the visualization of three-dimensional phenomena (architectural, meteorological, medical, biological, etc.),
- Where the emphasis is on realistic renderings of volumes, surfaces, illumination sources, and so forth, perhaps with a dynamic (time) component.
What's Missing?
Scientific Visualization Pipeline

- Produce Input Data
- Analyze, Filter, Reformat
- Apply Sci Vis Techniques
- Map to Geometry
- Render, Postprocess
- View Results

http://www.bu.edu/tech/research/training/tutorials/introduction-to-scientific-visualization-tutorial/the-scientific-visualization-pipeline/
Scientific Visualization Pipeline: Step 1 . . .

Produce Data

- Simulated Data
- Images
- Numerical
- Some measured value
- Observed Phenomena

Adopted from
http://www.bu.edu/tech/research/training/tutorials/introduction-to-scientific-visualization-tutorial/the-scientific-visualization-pipeline/
Scientific Visualization Pipeline:
Step 2 . . .

Analyze, Filter, Reformat

- Cleaning up the data
  - Removing noise
  - Replacing missing values
  - Clamping values to be within a specific range of interest

- Performing operations to yield more useful data

Adopted from
http://www.bu.edu/tech/research/training/tutorials/introduction-to-scientific-visualization-tutorial/the-scientific-visualization-pipeline/
Scientific Visualization Pipeline: Step 3

Apply SciVis Techniques

- Converts raw information into something more understandable
- Visually extracting meaning from a scientific data set using various techniques

Adopted from
http://www.bu.edu/tech/research/training/tutorials/introduction-to-scientific-visualization-tutorial/the-scientific-visualization-pipeline/
Scientific Visualization Pipeline: Step 3

**Apply SciVis Techniques**

- Converts raw information into something more understandable
- Visually extracting meaning from a scientific data set using various techniques

- Contour
- Clip
- Threshold
- Glyphs
- Streamlines

Adopted from
http://www.bu.edu/tech/research/training/tutorials/introduction-to-scientific-visualization-tutorial/the-scientific-visualization-pipeline/
Scientific Visualization Pipeline

Step 4 . . .

Map to Geometry

- Scalars, vectors, tensors
- 1D, 2D, 3D
- Mesh

Adopted from
http://www.bu.edu/tech/research/training/tutorials/introduction-to-scientific-visualization-tutorial/the-scientific-visualization-pipeline/
Scientific Visualization Pipeline:
Step 5 . . .

Render, Post Process

Adopted from
http://www.bu.edu/tech/research/training/tutorials/introduction-to-scientific-visualization-tutorial/the-scientific-visualization-pipeline/
Scientific Visualization Pipeline: Step 6 . . .

View Results

Output from ParaView

Pseudocolor Rendering
Vector / Tensor Glyphs
Parallel Coordinates

Streamlines
Volume Rendering

Adopted from http://www.bu.edu/tech/research/training/tutorials/introduction-to-scientific-visualization-tutorial/the-scientific-visualization-pipeline/
Large data produced by large simulations produce large visualization results and require large visualization resources.
Visualization Toolkit (VTK)

- Open source, multiplatform
- Supports distributed computation models
- Extensible modular architecture
- Available for 3D computer graphics, image processing and visualization

- Collection of C++ libraries
- Leveraged by many applications
- Divided into logical areas
  - Filtering
  - Information Visualization
  - Volume Rendering
- Cross platform, using OpenGL
- Wrapped in Python, Tool Command Language (Tcl) and Java
ParaView is an end-user application with support for

- Parallel Data Archiving
- Parallel Reading
- Parallel Processing
- Parallel Rendering
- Single node, Client-Server, MPI Cluster Rendering
Introduction to Scientific Visualization Using ParaView
• Multi-platform parallel data analysis and visualization application
• Mature, feature-rich interface
• Good for general purpose, rapid visualization

Mac
Windows
Linux
• Open Source . . . It’s Free!
• Built upon the Visualization Toolkit (VTK) library
• Primary contributors:
  - Kitware, Inc.
  - Sandia National Laboratory
  - Los Alamos National Laboratory
  - Army Research Laboratory
Uniform Rectilinear (Image Data)
A uniform rectilinear grid is a one- two- or three- dimensional array of data. The points are orthonormal to each other and are spaced regularly along each direction.

**Grid** – regular structure, all voxels (cells) are the same size and shape
Curvilinear (Structured Grid)
Curvilinear grids have the same topology as rectilinear grids. However, each point in a curvilinear grid can be placed at an arbitrary coordinate (provided that it does not result in cells that overlap or self-intersect). Curvilinear grids provide the more compact memory footprint and implicit topology of the rectilinear grids, but also allow for much more variation in the shape of the mesh.

Curvilinear – regularly gridded mesh shaping function applied

Adopted from The ParaView Tutorial, The Basics of Visualization, version 3.98
Unstructured Grid
Unstructured data sets are composed of points, lines, 2D polygons, 3D tetrahedra, and nonlinear cells. They are similar to polygonal data except that they can also represent 3D tetrahedra and nonlinear cells, which cannot be directly rendered.

**Unstructured grid** – irregular mesh typically composed of tetrahedra, prisms, pyramids, or hexahedra
• Point data
• Polygonal data
• Images
• Multi-block
• Adaptive Mesh Refinement (AMR)
• Time series support
SUPPORTED VISUALIZATION ALGORITHMS

- Isosurfaces
- Cutting planes
- Streamlines
- Glyphs
- Volume rendering
- Clipping
- Height maps
- & more
• Supports derived variables
• Scriptable via Python
• Saves animations
• Can run in parallel / distributed mode for large data visualization
ParaView SUPPORTED FILE FORMATS

- 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)
- CCSM MTSD (.nc, .cdf, .elev, .ncd)
- CCSM STSD (.nc, .cdf, .elev, .ncd)
- CEAucd (.ucd, .inp)
- CMAT (.cmat)
- 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
- ProSTAR (.cel, .vrt)
- Protein Data Bank (.pdb, .ent, .pdb)
- Raw Image Files
- Raw NRRD image files (.nrrd)
- SAMRAI (.samraii)
- SAR (.SAR, .sar)
- SAS (.sasgeom, .sas, .sasdata)
- SESAME Tables
- SLAC netCDF mesh and mode data
- SLAC netCDF particle data
- Silo (.silo, .pdb)
- Spherical (.spherical, .sv)
- SpyPlot CTH
- Spy Plot (.case)
- Stereo Lithography (.stl)
- TFT Files
- TIFF Image Files
- TSurf Files

Many more . . .
• All processing operations (filters) produce data sets
• Can further process the result of every operation to build complex visualizations
  • Extract a cutting plane,
  • Apply glyphs (i.e. vector arrows) to the result
    – Gives a plane of glyphs through your 3D volume
- WRF weather forecast data set
  - Rectilinear grid
  - Multiple scalar and vector variables
  - Time series
- Can show:
  - Clouds
  - Wind
  - Temperature
Three Basic Steps:

• First your data must be read into ParaView

• Next, you may apply any number of filters that process the data to generate, extract, or derive features from the data

• Finally, a viewable image is rendered from the data
ParaView 5.0.0
Let’s get started . . .
Sanity Check

✔ Software Installed?
✔ Data Sets downloaded?
✔ Can you locate the datasets?

http://web.ics.purdue.edu/~vbyrd/trainingData.html
Tutorial Datasets

- RectGrid2.vtk
- headsq.vti

Sample data file
- header.txt
- xCoordinates.txt
- yCoordinates.txt
- zCoordinates.txt
- lookUpTable.txt
What are we going to do?

- Load Data File
- Extract Isosurfaces from the data
- Create contours, clip contours, slice contours
- Volume Rendering
- Saving your Data
- Getting your data into Paraview
- Introduction to ParaView and Python Scripting
- Additional Resources
ParaView User Interface

- Menu Bar
- Tool Bar
- Pipeline Browser
- Object Inspector
- 3D Viewer
Open Data File: RectGrid2.vtk

- Locate RecGtrid2.vtk
- Click File → Open
- Select RectGrid2.vtk
- Click OK
Locate `RectGrid2.vtk`

Click **File → Open**

Select `RectGrid2.vtk`

Click **OK**

Click **Apply**
# vtk DataFile Version 2.0
Sample rectilinear grid
ASCII_DATASET RECTILINEAR_GRID
DIMENSIONS 47 33 11
X_COORDINATES 47 float
-1.22396 -1.17188 -1.11979 -1.06771 -1.01562 -0.963542
-0.911458 -0.859375 -0.807292 -0.755208 -0.703125 -0.651042
-0.598958 -0.546875 -0.494792 -0.442708 -0.390625 -0.338542
-0.286458 -0.234375 -0.182292 -0.130209 -0.078125 -0.026042
0.0260415 0.078125 0.130208 0.182291 0.234375 0.286458
0.338542 0.390625 0.442708 0.494792 0.546875 0.598958
0.651042 0.703125 0.755208 0.807292 0.859375 0.911458
0.963542 1.01562 1.06771 1.11979 1.17188
Y_COORDINATES 33 float
-1.25 -1.17188 -1.09375 -1.01562 -0.9375 -0.859375
-0.78125 -0.703125 -0.625 -0.546875 -0.46875 -0.390625
-0.3125 -0.234375 -0.15625 -0.078125 0 0.078125
0.15625 0.234375 0.3125 0.390625 0.46875 0.546875
0.625 0.703125 0.78125 0.859375 0.9375 1.01562
1.09375 1.17188 1.25
Z_COORDINATES 11 float
0 0.1 0.2 0.3 0.4 0.5
0.6 0.7 0.75 0.8 0.9
POINT_DATA 17061
SCALARS scalars float
LOOKUP_TABLE default
0 0 0 0 0 0
0 0 0 0 0 0
0 0 0 0 0 0
0 0 0 0 0 0
0 0 0 0 0 0
0 0 0 0 0 0

:
Create Isosurfaces

Contour – extracts the points, curves, or surfaces where a scalar field is equal to a user-defined value.
Create Isosurface

Click on RectGrid2.vtk

Select
- Filters
- Common
- Contour
Should see

- New object in Pipeline Browser
- One (1) value in the value range box
Select the value in the **Value Range** window
Select the value in the **Value Range** window

Click the minus sign in the value range box to Delete the value
Click the Add a range of values button
Click the Add a range of values button

Click **OK**
Click the Add a range of values button

Click OK
Click Apply
Should see

- A range of values in the Value Range box
- Values range from the min and max values entered in the previous dialog box
Should see

- A range of values in the Value Range box
- Values range from the min and max values entered in the previous dialog box
Should see

- Rendering in the 3D viewer
- Use your mouse to explore (rotate) the output
Step 1

Step 2
Step 1

Step 2
Step 1

Step 2
CREATE ISOSURFACES

SUMMARY OF STEPS

- Click Filters → Common → Contour
- Remove all values in Value Range Box
- Click Add a range of values
- Set values (Or accept default)
- Click Apply
- Set Color By property: vectors
Clip Isosurfaces

Clip – Intersects the geometry with a half space. The effect is to remove all the geometry on one side of a user-defined plane.
- Make sure Contour 1 is selected (highlighted in blue)
- Set View Direction to +Y
Create a new Clip filter:
Create a new Clip filter:
Select:
Create a new Clip filter:
Select:

➢ Filters
Create a new Clip filter:
Select:
- Filters
- Common
Create a new Clip filter:
Select:
- Filters
- Common
- Clip
Should see

- Clip object in Pipeline Browser
- Clipping Plane

If you do not see the clipping plane click the properties button.
• Using the mouse, hover over the arrow head (will change color – red)
• Press the mouse button, keep it pressed and rotate the clipping plane; arrow points out of the screen towards you
➢ Click the **Inside Out** box to switch the clipping plane

➢ Click **Apply**
INSIDE OUT FEATURE
CLIPL ISOSURFACES

SUMMARY OF STEPS

➢ Click +Y view button
➢ Click Filters → Common → Clip
➢ Position the clipping plane (move arrow point to desired position)
➢ Click Apply
➢ Check Inside Out check box to switch the clipping plane view
Slice Isosurfaces

Slice – Intersects the geometry with a plane. The effect is similar to clipping except that all that remains is the geometry where the plane is located.
- Make visible
  - RectGrid2.vtk
  - Clip1

- To make an object visible click the eye icon to the left of the object

- Contour 1 should **NOT** be visible for the next task
Select Clip 1

- Filters
- Common
- Slice
Should see

- **Slice1** object in Pipeline Browser
- Slicing Plane with arrow in 3D window
Position the slicing plane (move arrow point to desired position)

Click **Apply**
You can move the slicing plane along the axis to see a different slice

Click **Apply** to see changes
SLICE ISOSURFACE

SUMMARY OF STEPS

- Select **Clip1 Object**
- Select **Filter**
- Select **Common**
- Select **Slice**
- Position Slicing Plane
- Click **Apply**

**Slice** - Intersects the geometry with a plane. The effect is similar to clipping except that all that remains is the geometry where the plane is located.
Recall Three Basic Steps:

- First your data must be read into ParaView.
- Next, you may apply any number of filters that process the data to generate, extract, or derive features from the data.
- Finally, a viewable image is rendered from the data.

Opened simple data file RectGrid2.vtk

Applied filters:
Contour, Slice, Clip
Let’s try a more complex data set

headsq.vti
Delete all objects in the Pipeline Browser

Select an object in the Pipeline Browser

Click the Delete button (or right click, then Delete)

To select multiple objects press and hold the CTRL key while selecting objects
Locate File: headsq.vti

File
Open

headsq.vti
New Object in Pipeline Browser

Click **Apply**
Should see a bounding box in the 3D viewer window
Create an Isosurface

Select:
→ Filters
→ Common
→ Contour
A new object appeared in the pipeline browser (Contour 1)

**Contour** – Extracts the points, curves, or surfaces where a scalar field is equal to a user-defined value.

The surface is often also called an **isosurface**.
Value Range for the data set is now visible.
- Value Range for the data set is [0, 4095]
- Only one value is showing: 2047.5
- Value Range for the data set is \([0, 4095]\)
- Only one value is showing: 2047.5
- Click **Apply** to see what points, curves, or surfaces in the dataset have a value of 2047.5
If you do not see anything in the 3D window click the eye icon next to Contour1 in the Pipeline Browser.

This allows you to toggle between views in the 3D Viewer.
Visually Explore Dataset
Visually Explore Dataset
You should be here

Pipeline Browser → Two objects → Value Range [0, 4095]
Isosurfaces

- Select 2047.5 showing in Value Range
- Delete that value (click the minus button to remove all values)
Click the button below the minus button to: **Add a Range Of Values**

- Should see the Add Range Window
  - Use this window to set the range of values
  - For this tutorial
    - Min: 0
    - Max: 4095
  - Feel free to pay around with the range (between 0 and 4905)
Click the button below the minus button to: Add a Range Of Values

Should see the Add Range Window
  - Use this window to set the range of values
  - For this tutorial
    - Min: 0
    - Max: 4095
  - Feel free to pay around with the range (between 0 and 4905)
  - Click OK
Notice the Value Range: [0, 4095]

There are 10 values (steps) showing values between 0 and 4095
This may take a few seconds to render...
Explore the rendering . . .
Contour – Extracts the points, curves, or surfaces where a scalar field is equal to a user-defined value.

The surface is often also called an isosurface.
Set Value Range

<table>
<thead>
<tr>
<th>From</th>
<th>To</th>
<th>Step</th>
<th>What do you get?</th>
</tr>
</thead>
<tbody>
<tr>
<td>1500</td>
<td>2500</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>2500</td>
<td>3500</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>2999</td>
<td>3500</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>3500</td>
<td>4000</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>2000</td>
<td>2999</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>3000</td>
<td>3999</td>
<td>10</td>
<td></td>
</tr>
</tbody>
</table>
Click red X – Remove all entries

Click Add a range of values button

Click **OK**

Click **Apply**
Reset Values

Click red X – Remove all entries

Click Add a range of values button

Click OK
Click Apply
Reset Values

Click red X – Remove all entries

Click Add a range of values button

Click OK

Click Apply
Clip Isosurface

- CLIP - Intersects the geometry with a half space.
- The effect is to remove all the geometry on one side of a user-defined plane.
Select:
→ Contour 1 *(in pipeline Browser)*
→ Filters
→ Common
→ Clip
Clip Isosurface

- A new object appeared in the pipeline browser (Clip 1)
- Clipping plane (see red vertical line and horizontal arrow)
- Select the arrow point (arrow turns red)

- Rotate (drag) the arrow point until the arrow is pointing out of the screen toward you
Clip Isosurface

- Make sure the eye icon is not greyed out on the Clip1 object in the Pipeline Browser.

Depending on where you placed the clipping plane the results may be easily seen: see clipped ears; and area round neck.
Rotating the view reveals the clipped isosurface.
If this property is set to 0, then clip filter will return that portion of the dataset that lies within the clip function.

If set to 1, the portions of the dataset that lie outside the clip function will be returned instead.
Click the eye icon next to Clip1 in the pipeline browser (hide the clip plot)

Select Contour 1
Select

- Filters
- Common
- Slice
Drag arrow point around to point out of screen toward you.
SLICE – Intersects the geometry with a plane. The effect is similar to clipping except that all that remains is the geometry where the plane is located.
Q: How do we combine (show) the Clip and Slice views at the same time?
Many Options to Save Your Work

- Open...
- Recent Files
- Load State...
- Save State...
- Save Data...
- Save Screenshot...
- Export Scene...
- Save Animation...
- Save Geometry...
- Load Window Layout...
- Save Window Layout...
- Connect...
- Disconnect
- Exit
- Export Cinema...
Save Your Work

Save State

save the state of the visualization pipeline itself, including all the pipeline modules, views, their layout, and their properties.

This is referred to as the application state or, just, state.

In paraview, you can save the state using the File Save State... menu option.

Conversely, to load a saved state file, you can use File Load State...
Save Data

You can save the dataset produced by any pipeline module in ParaView, including sources, readers, and filters.

To save the dataset in paraview, begin by selecting the pipeline module in the Pipeline browser to make it the active source.

For modules with multiple output ports, select the output port producing the dataset of interest.

To save the dataset, use the File Save Data menu or the button in the Main Controls toolbar. You can also use the keyboard shortcut Ctrl + S (or + S).

The Save File dialog will allow you to select the filename and the file format. The available list of file formats depends on the type of the dataset you are trying to save.
Save Your Work

Save Screenshot

To save the render image from a view in paraview, use the File Save Screenshot menu option. This will pop up the Save Screenshot Options dialog (Figure 8.3).

This dialog allows you to select various image parameters, such as image resolution and image quality (which depends on the file format chosen).
The dialog also provides an option to change the color palette to use to save the image using Override Color Palette.

By default, paraview will save rendered results from the active view. Optionally, you can save an image comprising of all the views laid out exactly as on the screen by unchecking the Save only selected view button.
In preparation for the next section

- Delete all objects in the Pipeline Browser
- Select an object in the Pipeline Browser
- Click the Delete button (or right click, then Delete)
- To select multiple objects press and hold the CTRL key while selecting objects
Getting Your Data Into VTK File Format

Sample File
Supported Data Formats

- EnSight
- Plot3D
- Various polygonal formats
- Users can write data readers to extend support to other formats
- Conversion to the VTK format is straightforward
VTK Simple Legacy Format

VTK simple legacy format ([http://www.vtk.org/VTK/img/file-formats.pdf](http://www.vtk.org/VTK/img/file-formats.pdf))

- ASCII or binary
- Supports all VTK grid types
- Easiest for data conversion

```
# vtk DataFile Version 2.0
Really cool data
ASCII | BINARY
DATASET type
...
POINT_DATA n
...
CELL_DATA n
...
```

Part 1: Header

Part 2: Title (256 characters maximum, terminated with newline \n character)

Part 3: Data type, either ASCII or BINARY

Part 5: Dataset attributes. The number of data items n of each type must match the number of points or cells in the dataset. (If type is FIELD, point and cell data should be omitted.)

Part 4: Geometry/topology. Type is one of:
- STRUCTURED_POINTS
- STRUCTURED_GRID
- UNSTRUCTURED_GRID
- POLYDATA
- RECTILINEAR_GRID
- FIELD
The data

- Simulated temperature values
- Sample size: 100 x 100
- Rectilinear Grid
# vtk DataFile Version 2.0
Rectilinear grid of temperature values
ASCII
DATASET RECTILINEAR_GRID

Part 1: Header

Part 2: Title (256 characters maximum, terminated with newline \n character)

Part 3: Data type, either ASCII or BINARY

Part 4: Geometry/topology. *Type* is one of:

  - STRUCTURED_POINTS
  - STRUCTURED_GRID
  - UNSTRUCTURED_GRID
  - POLYDATA
  - RECTILINEAR_GRID
  - FIELD

Part 5: Dataset attributes. The number of data items *n* of each type must match the number of points or cells in the dataset. (If *type* is FIELD, point and cell data should be omitted.)
# vtk DataFile Version 2.0
Rectilinear grid of temperature values
ASCII
DATASET RECTILINEAR_GRID

**DIMENSIONS 100 100 1**

*X_COORDINATES 100 float
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99

Y_COORDINATES 100 float
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99

Z_COORDINATES 1 float
0

*Although this is a 2D grid, the z-coordinate must be included and represented in the DIMENSIONS*
# vtk DataFile Version 2.0
Rectilinear grid of temperature values
ASCII
DATASET RECTILINEAR_GRID
DIMENSIONS 100 100 1
X_COORDINATES 100 float
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28
29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52
53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76
77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99
Y_COORDINATES 100 float
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28
29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52
53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76
77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99
Z_COORDINATES 1 float
0
POINT_DATA 10000
SCALARS temperature float
LOOKUP_TABLE default
* x-dimension * y-dimension * z-dimension
# vtk DataFile Version 2.0
Rectilinear grid of temperature values
ASCII
DATASET RECTILINEAR_GRID
DIMENSIONS 100 100 1
X_COORDINATES 100 float
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99
Y_COORDINATES 100 float
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99
Z_COORDINATES 1 float
0
POINT_DATA 10000
SCALARS temperature float
LOOKUP_TABLE default
20.18 20.36 20.54 20.73 20.93 21.13 21.35 21.58 21.82 22.09 22.38 22.70 23.06 23.46 23.92 24.44 25.05 25.77 26.63 27.68 28.99 30.68 32.90 35.99 40.50 47.61 60.00 84.65 142.03 300.00 300.00 300.00 300.00 300.00 300.00 300.00 300.00 300.00 300.00 289.04 288.50 287.82
Data Files

Header.txt
xCoordinates.txt
yCoordinates.txt
zCoordinates.txt
lookUpTable.txt

Task:
Combine these files into one file and save SampleData.vkt
- Open data file (the file that you just created and saved)
- Click **Apply**
<table>
<thead>
<tr>
<th>Add Contour Plot</th>
<th>Set the range of values</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>From 20.01</td>
</tr>
<tr>
<td></td>
<td>To: 300</td>
</tr>
<tr>
<td></td>
<td>Step 10</td>
</tr>
</tbody>
</table>
What should the pipeline browser look like?

Split Window