AVS Workshop Examples

Instructors

Dr. Paul Alsing

Dr. Andrew Pineda

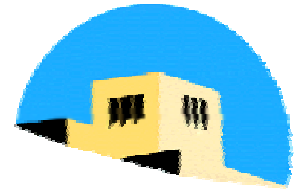


Table of Contents

- Getting Started, Creating/Saving Projects
- Volume Data/Scalar Field Example
  - Using the Network Editor, connecting objects
  - Reading in data using an AVS field file
  - Isosurface, manipulating the image
  - Controlling Datamap/Color Map, Transparency of objects
  - Looping through surfaces
  - Using a 2D viewer to do simple animations
  - Saving your output as figures

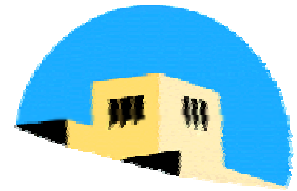


Table of Contents

- Surface/Contour Plots
  - Creating Field Files
  - surfplot, 2D/3D viewers, isoline
  - Creating a custom reader using the File Import Tool, saving custom readers to Workspaces
- Connectivity example - Overlaying Data on a surface
  - Trimesh
  - Combine mesh data
  - Node data

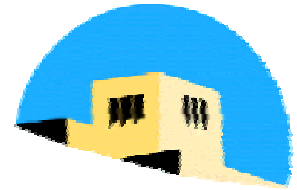
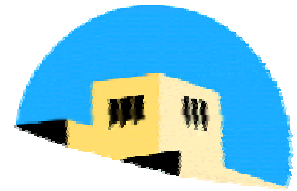


Table of contents

- 2D graph example
  - Using Read Column File, table_to_scattered_field and array extractors to read in the data.
  - Fancy Example using Graph Objects.

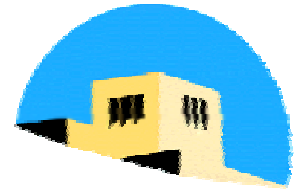


Starting Out

- Starting AVS/Express at AHPCC
  - Set or append

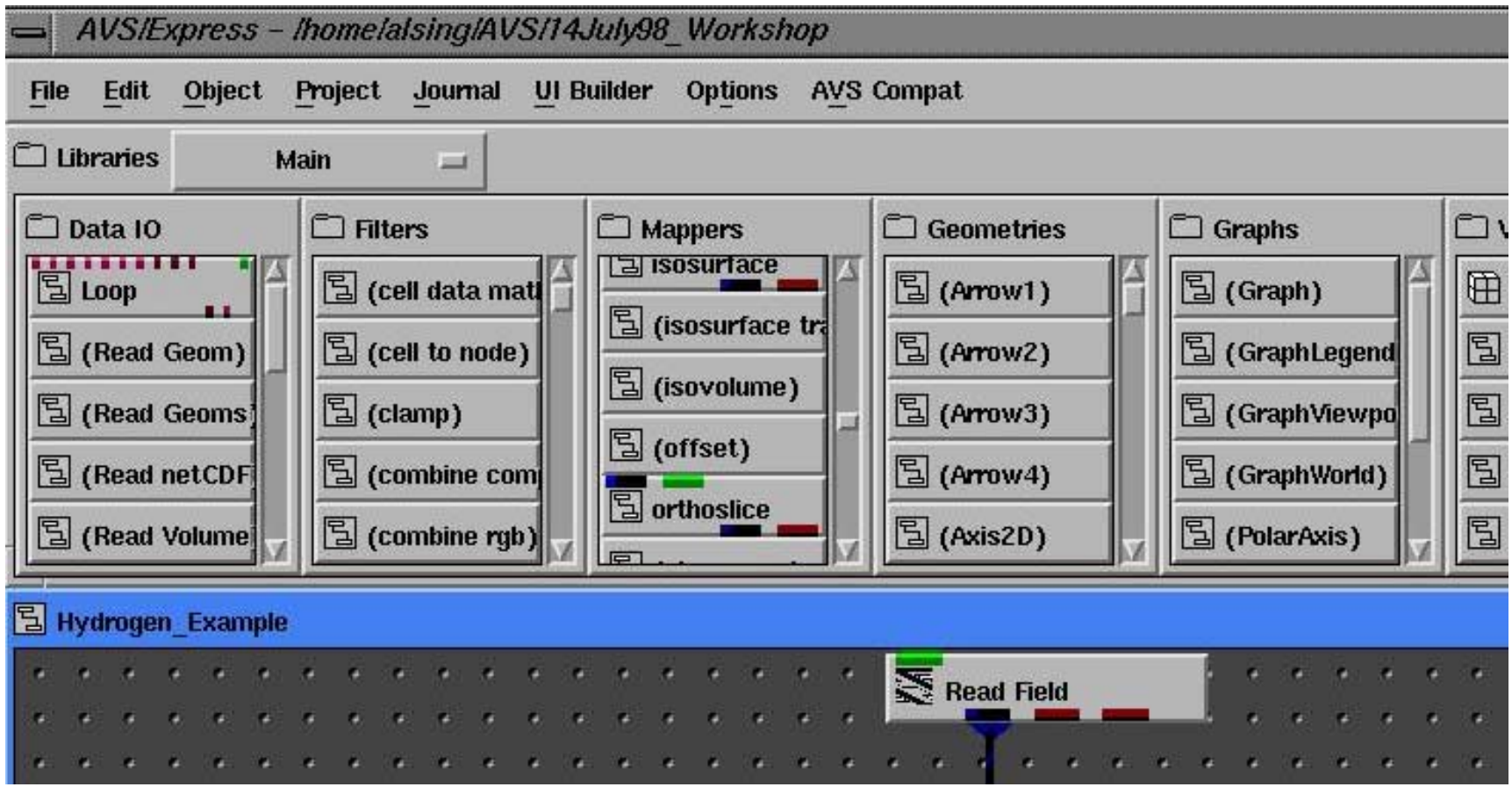
    /usr/local/flexlm/licenses/license.dat

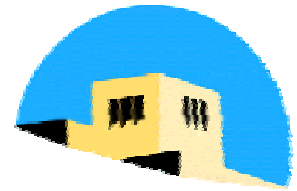
    to your **LM_LICENSE_FILE** environment variable
  - Set or append **/usr/local/AVS/express/bin/sg6** to **LD_LIBRARY_PATH**.
  - Add **/usr/local/AVS/express/bin/sg6** to your path.
  - Unix: type ***vxp*** at prompt. This starts the command line interface.
  - Do not put the command line interface in background or suspend.



Starting Out

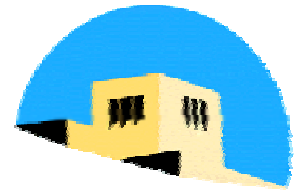
- Create a project directory, start AVS there, and save your project in that directory.
  - This is needed by AVS to store applications, data, and configuration information.
  - Use **Project/Save As...** and remove backslash under **Selection**





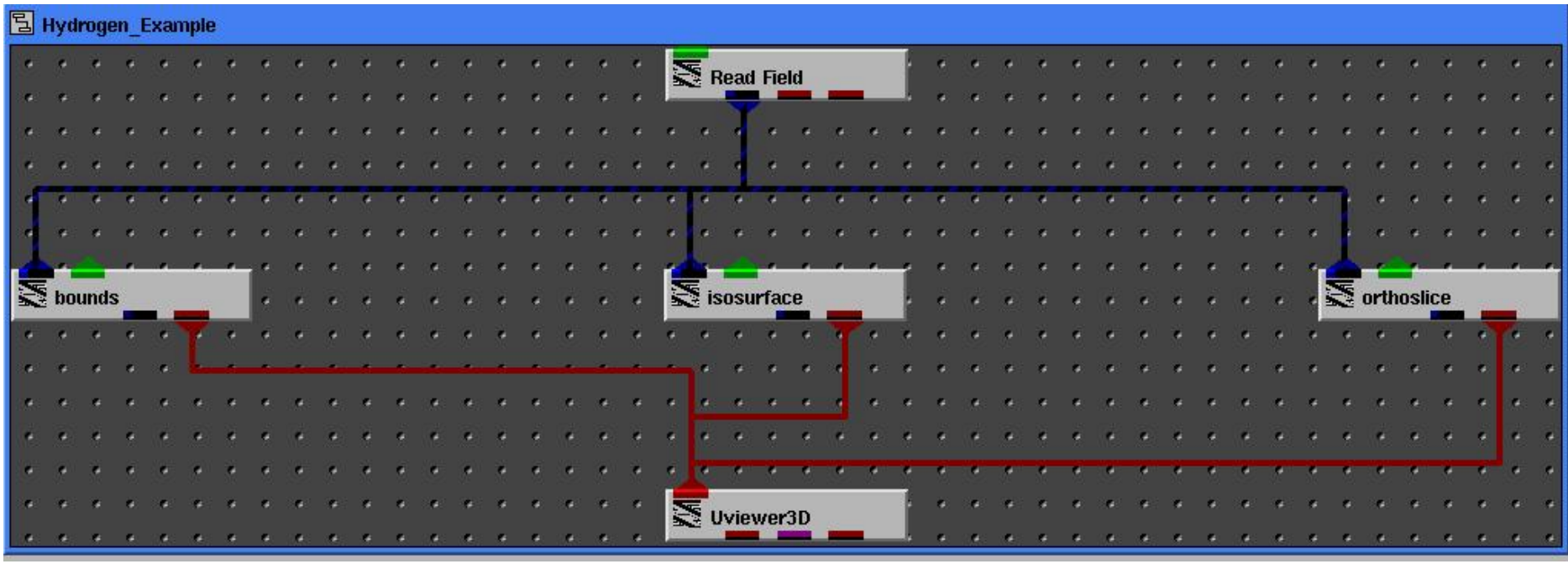
Hydrogen Example - Network Editor

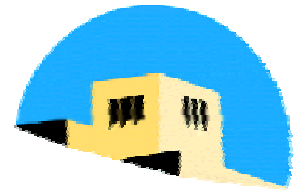
- Create a new single-window application
  - Click **File/New Application…/**Click **Application Type: Single-window Data Viewer button** and **Viewer Type: 3D button/**then Click **Ok.**
  - Go to Network Editor Window, pull down the following objects:
    - **Main.Data IO.Read Field**
    - **Main.Mappers.bounds**
    - **Main.Mappers.isosurface**
    - **Main.Mappers.orthoslice**
    - **Object/Find All Libraries…/type in** Loop<RET> **in Search Pattern/Click Show/Pull down Main.Data IO.Loop, highlighted in blue**



Hydrogen Example - Network Editor

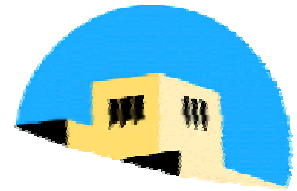
- Connect the objects as shown in the figure on the next page by dragging with the left mouse button.
  - To break connections, place mouse on connection line, press right mouse button, select **Delete connection** from the pull-down menu, release right mouse button.
  - Right click on the grey breadboard area and select **Zoom to Fit/Arrange Icons/Reset Scaling** to arrange and size items.
  - To rename objects, right click on the Item, and select **Rename...** from the pull-down menu. Use this to rename *SingleWindowApp* to *HydrogenExample.*





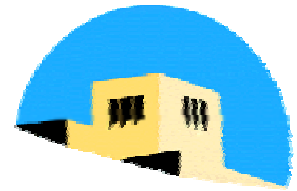
Hydrogen Example - Module Editor

- Go to Single Window Viewer called HydrogenExample
  - Left Click on **Modules** to see a list of GUIs to the objects in Network Editor.
  - Select **Read Field/Click Browse.../Select** *hydrogen.fld*/Click **Ok.**
  - Should see molecule if network is okay.
  - Try rotating the object with the left mouse button. Note the object manipulation buttons on top of viewer. Select one of the scaling or translation options.
  - Select **Modules/orthoslice**. Play with the plane and axis sliders/arrow buttons.
  - Ditto with **Modules/isosurface/iso level**.



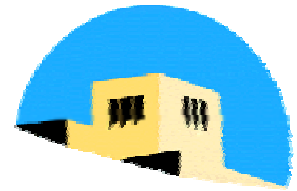
Hydrogen Example

- Saving/Deleting the application.
  - Click on **File/Save Application...** Double click on the v subdirectory. Enter *hydrogen.v*. Click **Ok**. Be sure to save your applications with the viewer window open!!!
  - Right click on the application title bar on the Network Editor and select **Delete**.
- Editing the color map.
  - Go to the Single Window Viewer.
  - Go to **Select Object...** (bottom right corner), Select **isosurface**.
  - Go to **Editors Menu/Datamap.**
  - Play with the sliders



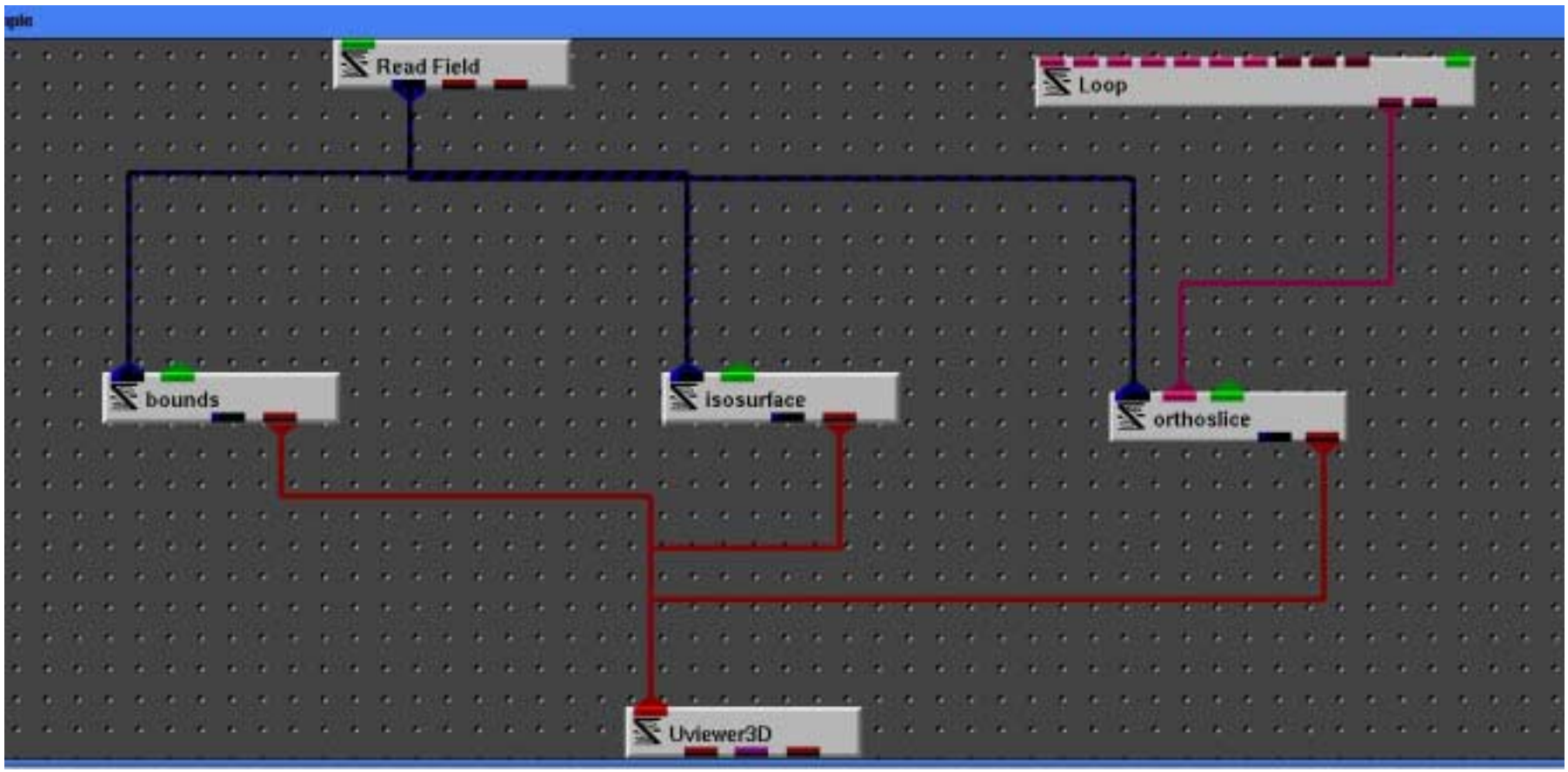
Hydrogen Example

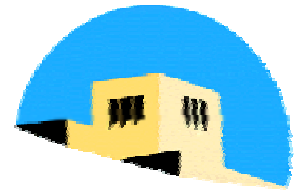
- Controlling the transparency of objects
  - Go To **Editors/Object**.
  - Under Object where is says **Object: General**, change to **Object:Properties**.
  - Change **Type:General** to **Type: Surface**. There are 6 sliders that control opacity, reflectivity, etc. of the surface. Play with the sliders.
- Controlling lighting
  - Go To **Editors/Light**. Change **Type:directional** to **Type:bidirectional**.



Hydrogen Example

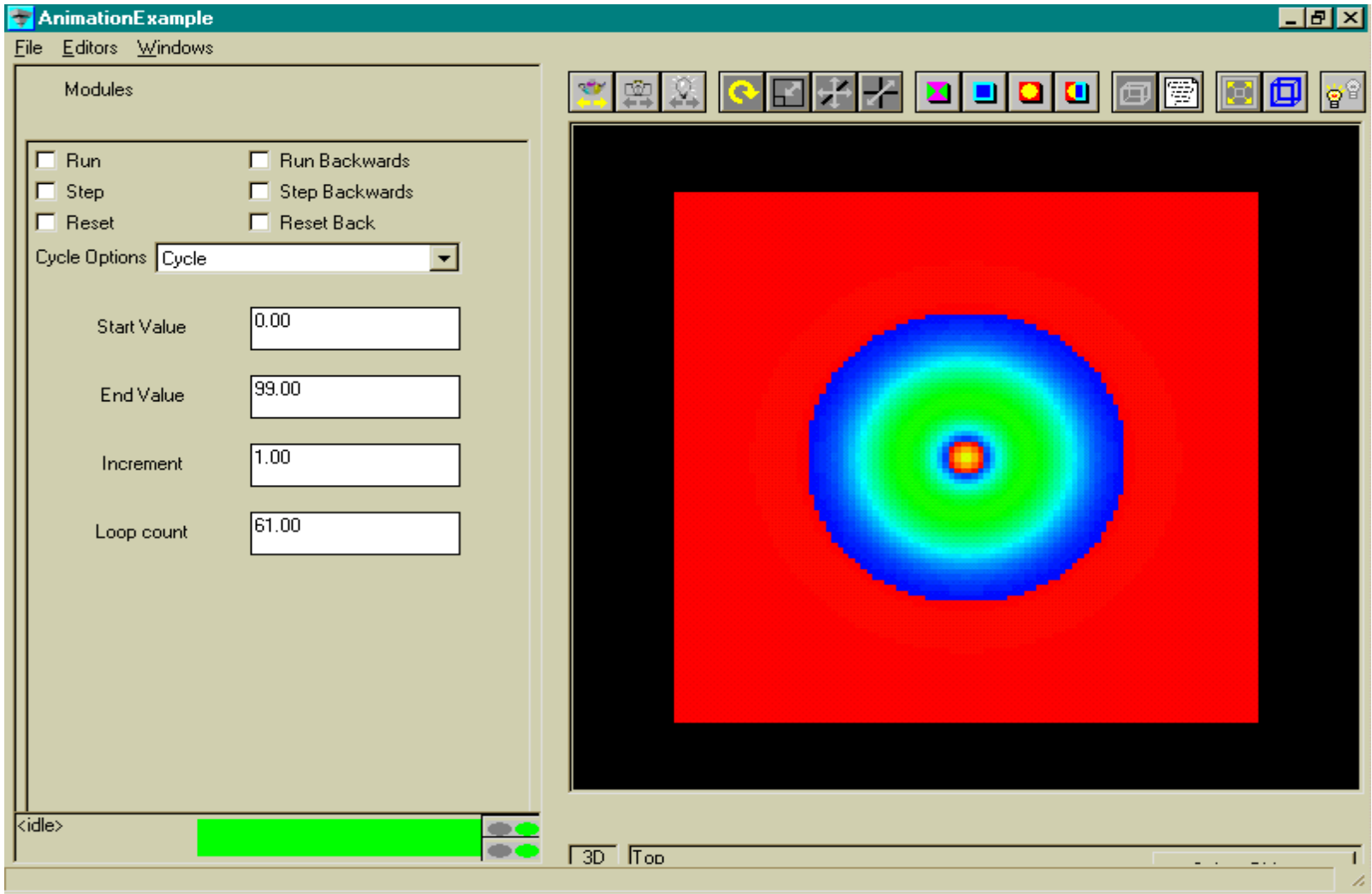
- Looping through isosurfaces, orthoslice planes, etc.
  - Now we will connect the **Loop** object you pulled down to the **orthoslice** object.
    - Right click on **orthoslice**, select **Object Editor**.
    - Left click on **plane**. Click on **Display input port**. In **Connection**, delete the entry there. Click **close**.
    - Right click again on **orthoslice**, select **Display Parameters**. Right click and select close. A new red input tab should appear on orthoslice.
    - Go to **Loop** and right click to select **Display Parameters**.
    - Connect **Count** object to the newly created tab on **orthoslice**.
    - Right click to close **Loop**, and return to single window viewer.
    - Go **Modules/Loop**. Set looping parameters, and select **run**.

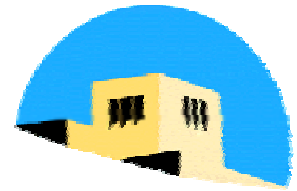




Doing Simple Animations

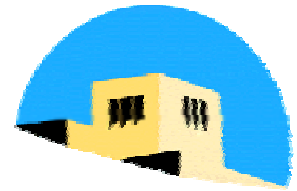
- If one dimension represents time, the previous network can be easily modified to do simple animations.
  - From the viewers section, instance a **Uviewer** (the combined 2D/3D viewer) or **Uviewer2D** object.
  - Delete (or simply break the connections to) the **bounds**, **Uviewer3D**, and **isosurface** objects from the network.
  - Connect the output of **orthoslice** to the right input of **Uviewer**.
  - In the **Read Field** module user interface, change the input file to */home/acpineda/public_html/AHPCC_Vis/AVS/data/wavepack.fld*.
  - Start the loop counter from the **Loop** user interface.





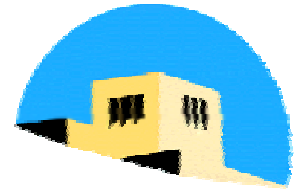
Hydrogen Example

- Printing images/Capturing as graphics files.
  - PC - You can use **Alt-Prt Scr** (Print screen) to capture the current window to the clipboard. Then paste the window into a Windows BMP file using Paint.
  - Unix - You can use a screen capture tool such as **xv**.
  - Printing Postscript. On the single window viewer window, select **Editors/Print**. You can select Postscript, Color Postscript, Postscript Level 2, or a handful of other graphics formats.



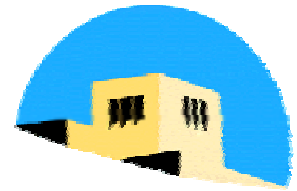
Surface Plot Example

- Create a new single window viewer application (2D/3D viewer) and pull the following objects into the Network Editor:
  - **Read Field**
  - **scale**
  - **surf plot**
  - **Axis3D**
- Copy the file *angres.txt* to your data directory. It is an ASCII text file organized in 8 columns/190 lines. The first 2 columns are x,y coordinates. The other 6 are data. We want to plot the 7th column vs. (x,y).



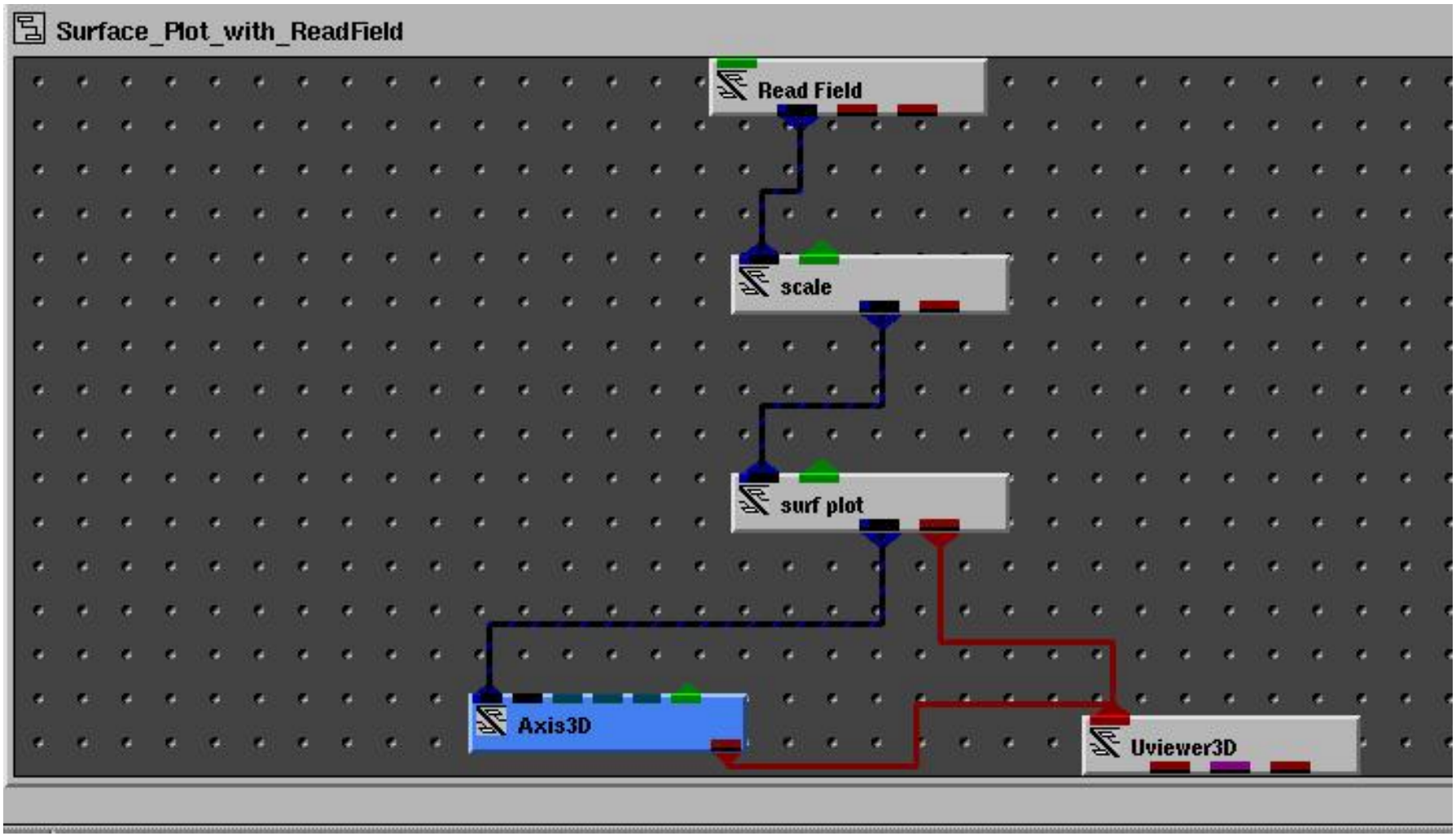
Surface Plot Example - Read Field

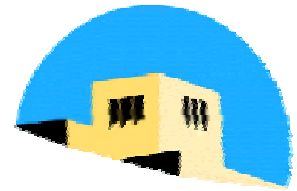
- Create an AVS field file describing the data using the *hydrogen.fld* file as a template. The keywords you need are on the next page. If you get stuck look at *refl.fld*.
- Create the network shown on the page after next.
- Use the Module editor on the single window viewer to tell **Read Field** to read in your field file.



Field File keywords

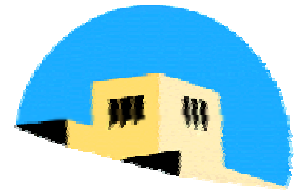
- **ndim** = number of dimensions in computational space
- **dim1, dim2, ...** = size of nth computational dimension
- **nspace** = number of spatial dimensions
- **veclen** = number of data variables per point
- **data** = data type (float, integer, double, etc.)
- **field** = uniform, rectilinear, irregular
- **file** = path to file
- **filetype** = ascii or binary
- **skip** = lines or bytes to skip before 1st data value
- **offset** = number of columns to skip before reading
- **stride** = number of columns
- **variable, coord** = keywords for variables and coordinates





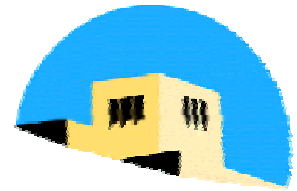
Surface Plot Example Using File Import Tool

- Now we will create a File reader using File Import Tool. Let's start with the coordinates.
  - Go **Libraries/Library Workspaces**. Click on **Workspace 1**.
  - Go To **Object/Add File Import Module...**
    - Enter **ReadSurfData** in the **Module Name** field and press Enter then click Next.
    - Click **Add File Variable**, Under **object name** enter *coords*. Under **base type** click *float*. Click **Object is an array**. Enter *[190][2]* for the **dimensions**. Press Enter and click Next.
    - Click **ASCII**, set *Offset=0,Columns in file=8,* then *check Columns #0 and #1* under **Columns to read**. Click Done. *coords* appears as a new object.



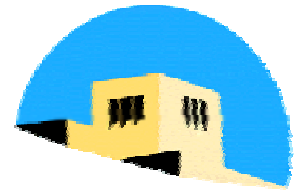
Surface Plot Example - File Import tool

- Now let's add another file variable for the data.
  - Click **Add File Variable** again.
  - Enter *data* for the **Object name**. Select *float* for the **Base type** and **Object is an array**. Enter *[190]* for the **Dimensions**. Click **Next**.
  - Select Variable is *ASCII*, set *Columns=8*, and select *Column #6*. Click **Done**.
- Now we are finished creating file variables.
  - Click **Done**. The dialog should disappear. Click **OK** at the Done Message. Under **Workspace 1**, you should see **ReadSurfData**. Save the Project now!!! Pull an instance of **ReadSurfData** down into the Network Editor.



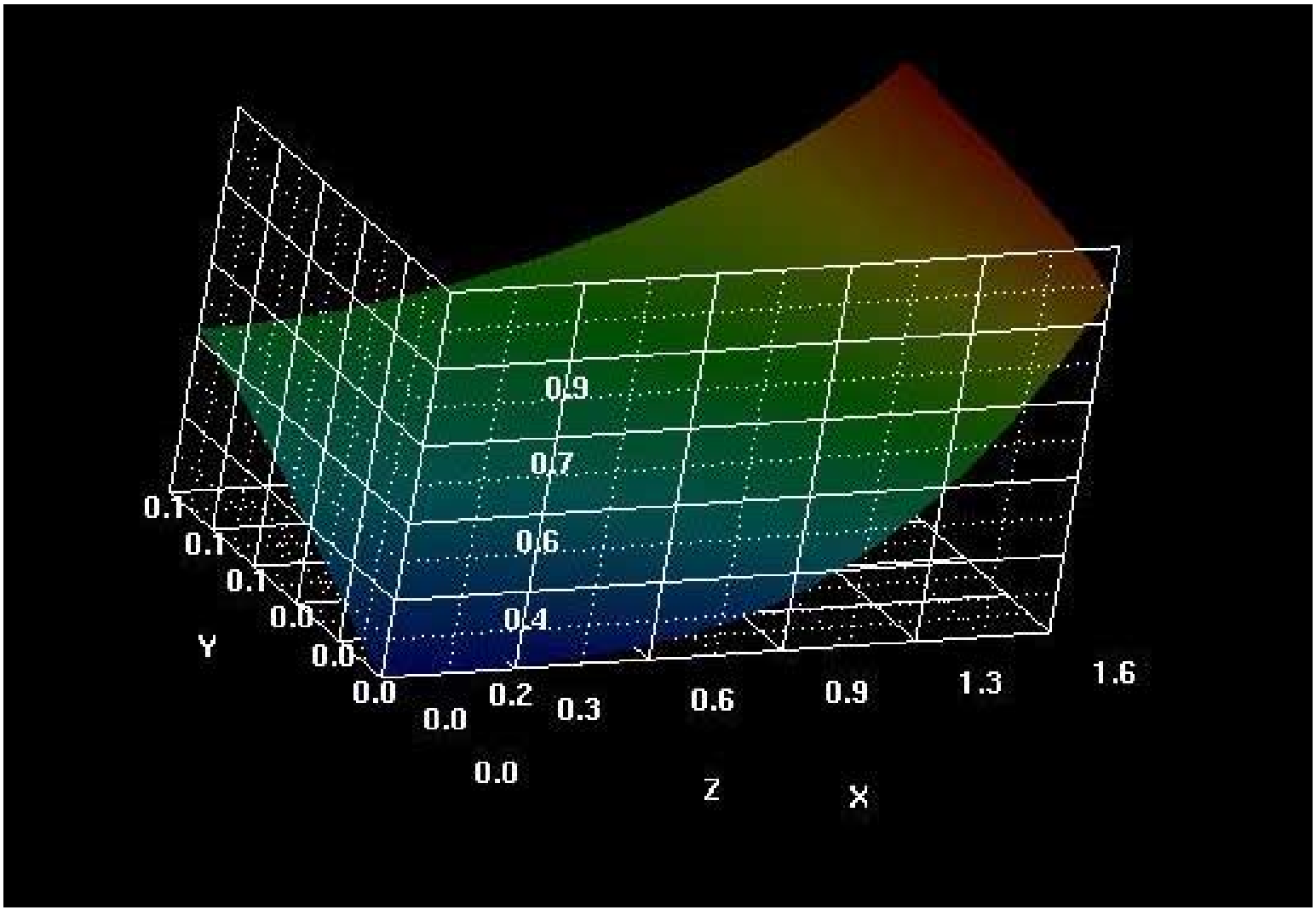
Surface Plot Example - File Import Tool

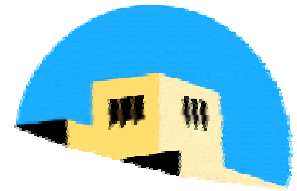
- Go back to the Single Window viewer and bring up **ReadSurfData** in the module editor. Browse for the file *angres.txt* in your data subdirectory.
- Pull down instances of:
  - **struct mesh**
  - **node scalar**
  - **combine mesh data**
- Hook the *coords* from **ReadSurfData** into **struct mesh**.
- Hook the *data* from **ReadSurfData** into **node scalar**.



Surface Plot Example - File Import Tool

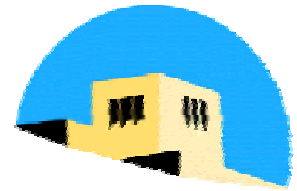
- Go to **struct mesh**, right click on **Object Editor**.
- Select **in dims.** Under **Dimensions** enter *[2]*. Under **Connection** enter *{10,19}* and then **close**.
- Connect blue/black output of **struct mesh** to blue/black port of **combine mesh data**.
- Connect black port of **node scalar** to the black port of **combine mesh data**.
- Break connection from **Read Field** of previous example and replace with blue/black output of **combine mesh data**.





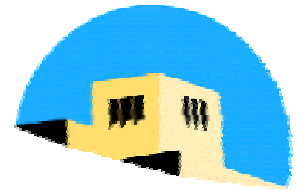
Connectivity Example - Toy Missile

- This example illustrates the rendering of unstructured data. This will require us to provide connectivity information.
- We will also see how to create a flexible file reader with the File Import Tool.
- Copy the following items to your data directory:
  - toy_missile.f (F90 source code)
  - toy_missile.coordsanddata (coords and data)
    - Line 1 is # of entries, rest is 4 columns (3 coords, 1 data)
  - toy_missile.connectivity (connectivity information)
    - Line 1 is # of entries, rest is a single column of connectivity data
    - connectivity data is read as triples of indices (3 row numbers of coordinates) representing the vertices of oriented triangles.



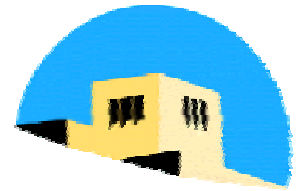
Connectivity Example - Toy Missile

- Pull the following items down into the Network Editor:
  - **trimesh**
  - **node scalar**
  - **combine mesh data**
  - **set null**
- We will tell you how to connect them later.



Connectivity Example - Toy Missile

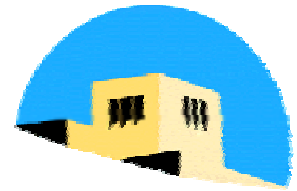
- Now we need to create 2 file readers, 1 for the data and 1 for the connectivity. These readers will dimension arrays according to the sizes found in the input files.
  - Open the File Import Tool. (Go to Libraries/Workspace 1, first.)
    - **Module name**=*ReadConnectivity*. Click Next.
    - **Add file variable**, *dim*. Click Next.
    - Select Variable is *ASCII*, type=*int*, Offset =0. Click Done.
    - **Add file offset variable**. Leave default name, Click Next.
    - Click **skip lines** and **set nlines**=1. Click Done.
    - **Add file variable**, *connectivity*. Click Next.
    - Click **Object is an array**, **Base Type** is *int*. **Dimensions**=*[dim]*. Click Next.



Connectivity Example - File Import Tool

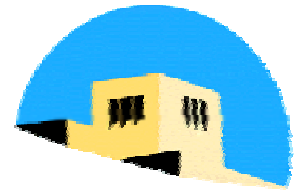
- Click **Variable stored in file as** *ASCII*. Leave **data type** as *int*. Type in *file_offset* for **Offset**, 1 for **Number of columns in file**, and check *Column #0*.
- Click **Done**.

- Instance the **ReadConnectivity** object and browse for the connectivity file.
- Open the File Import Tool again.
  - **Module name**=*ReadCoordsData*. Click **Next**.
  - **Add file variable**, *dim*. Click **Next**.
  - Select **Variable is** *ASCII*, **type** is *int*, **Offset** =0. Click **Done**.



Connectivity Example - File Import Tool

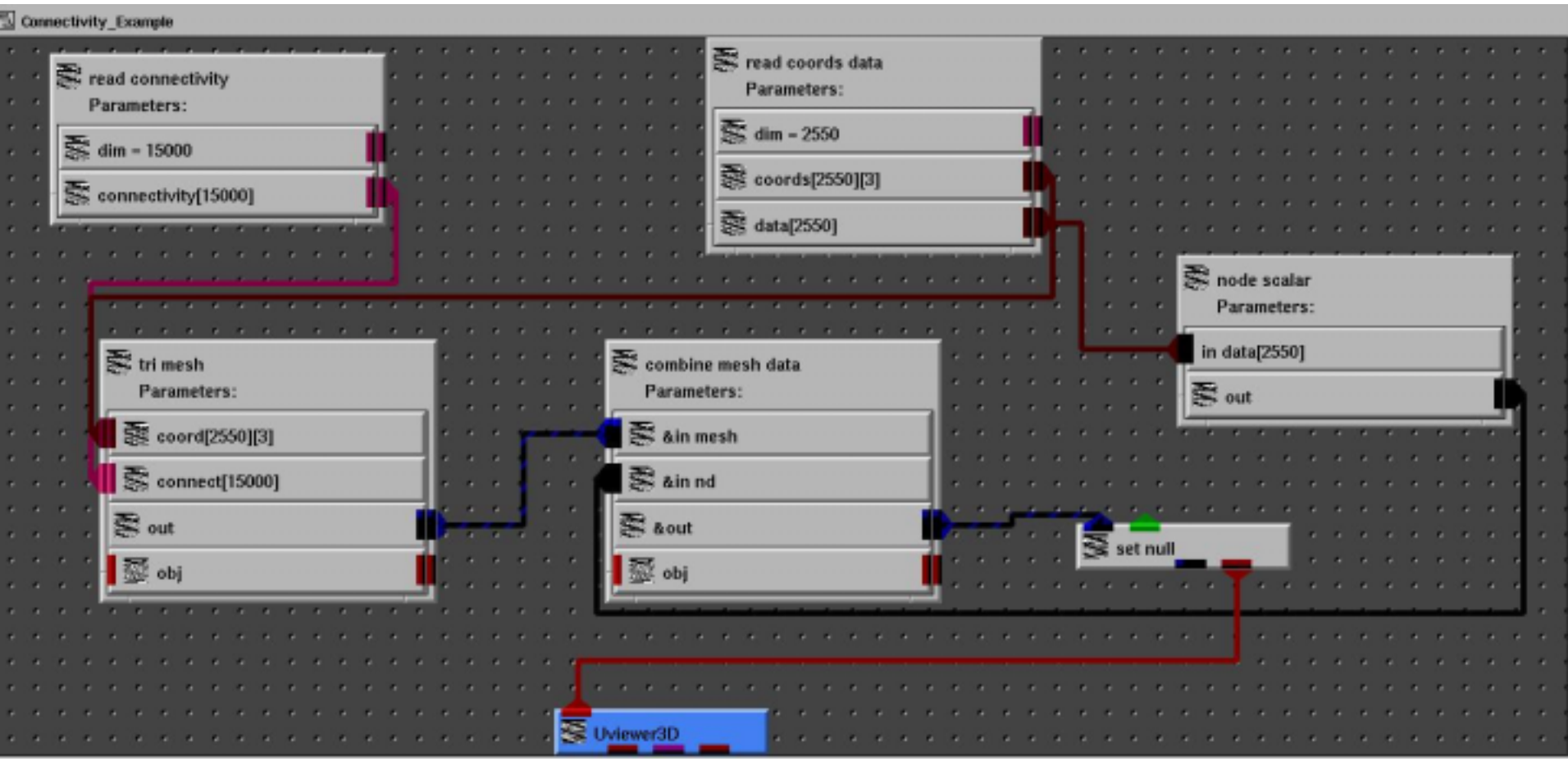
- **Add file offset variable**. Leave default name, Click **Next**.
- Click **skip lines** and set **nlines**=1. Click **Done**.
- **Add file variable**, *coords*. Click **Next**.
- Click **Object is an array**, **Base Type** is *float*. **Dimensions**=*[dim][3]*. Click **Next**.
- Click **Variable stored in file as** *ASCII*. Leave **data type** as *float*. Type in *file_offset* for **Offset**, 4 for **Number of columns in file**, and check *Columns #0,1,2*.
- Click **Done**.

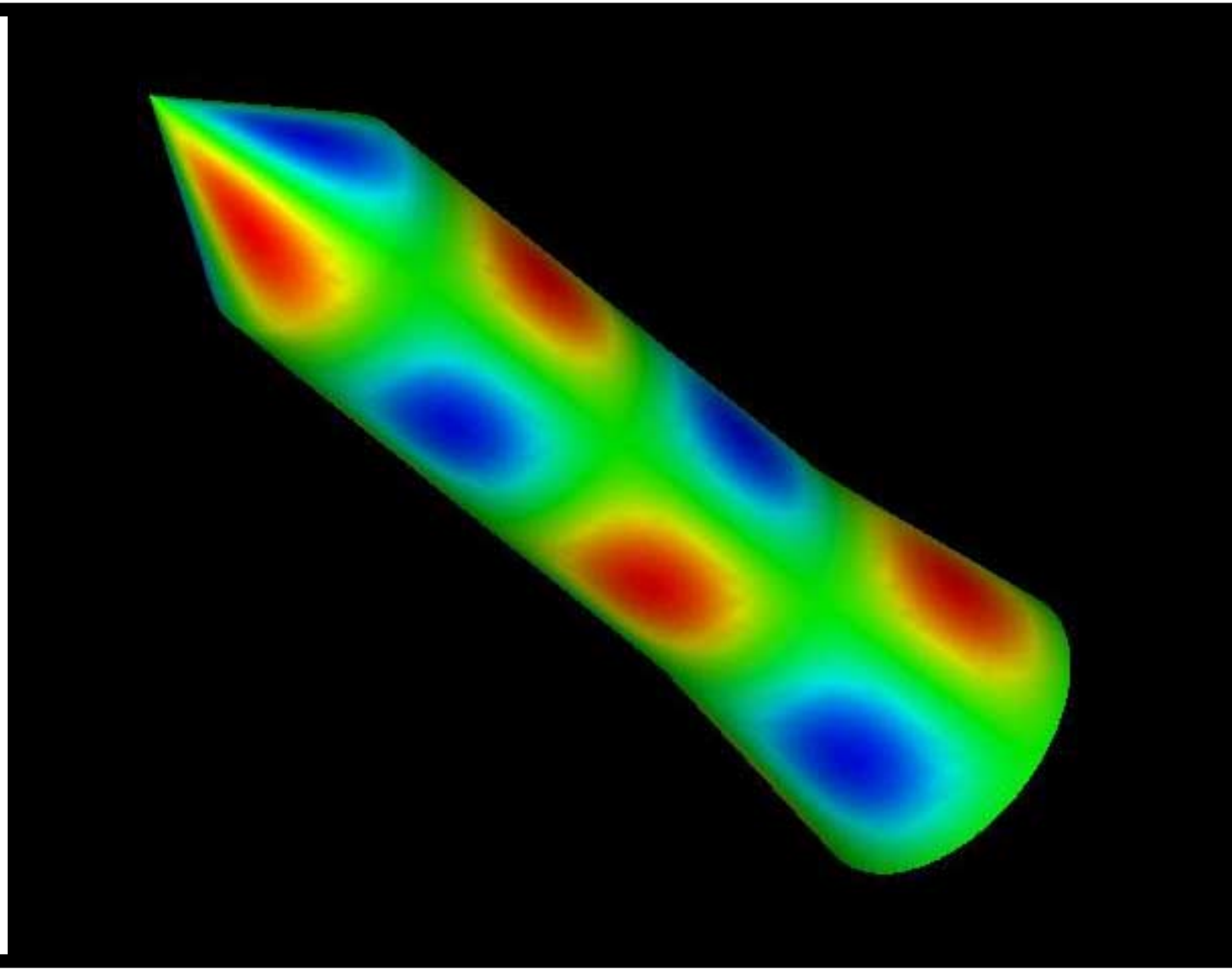


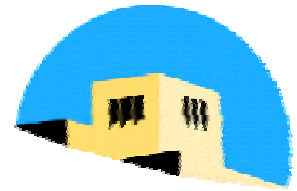
Connectivity Example - File Import Tool

- **Add file variable**, *data*. Click **Next**.
- Click **Object is an array**, **Base Type** is *float*. **Dimensions**=*[dim]*. Click **Next**.
- Click **Variable stored in file as** *ASCII*. Leave **data type** as *float*. Type in *file_offset* for **Offset**, 4 for **Number of columns in file**, and check *Column #3*.
- Click **Done**.
- Click **Done** to Exit the File Import Tool.

- Instance the **ReadCoordsData** object and browse for the coordinates/data file.
- Connect objects as shown on the next slide.
- Play with the null value. Try 127.0 and -1.0.

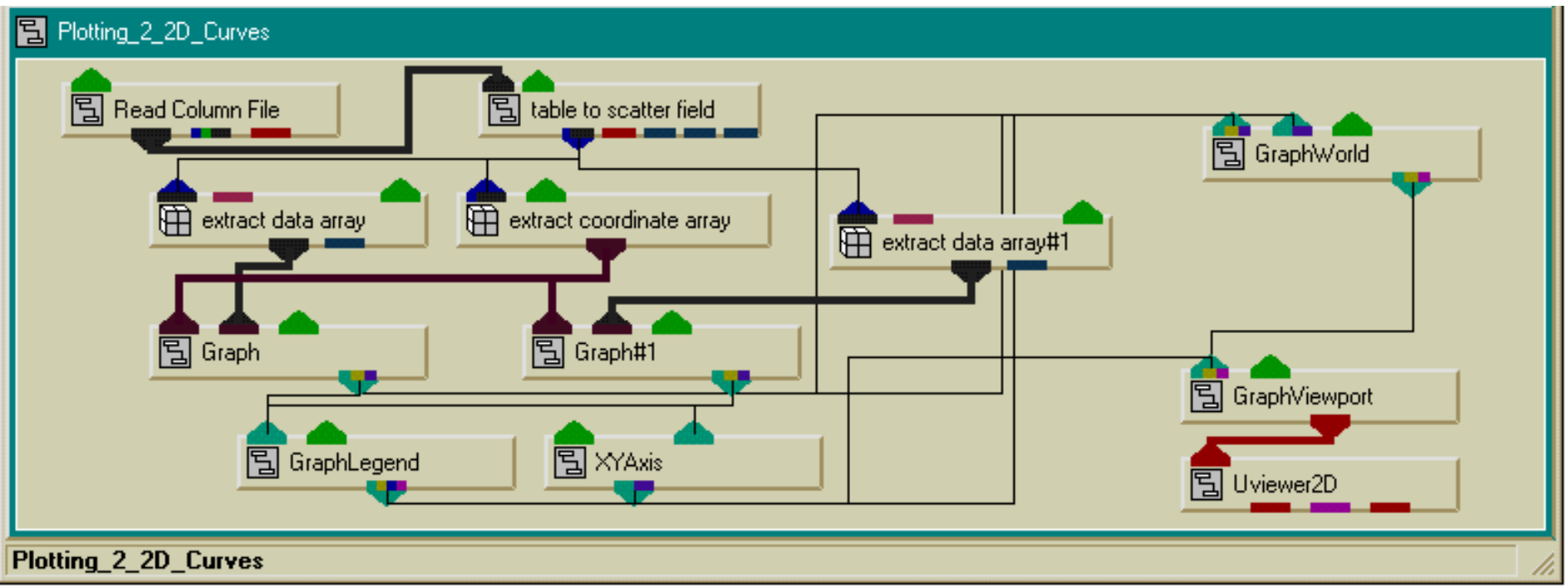


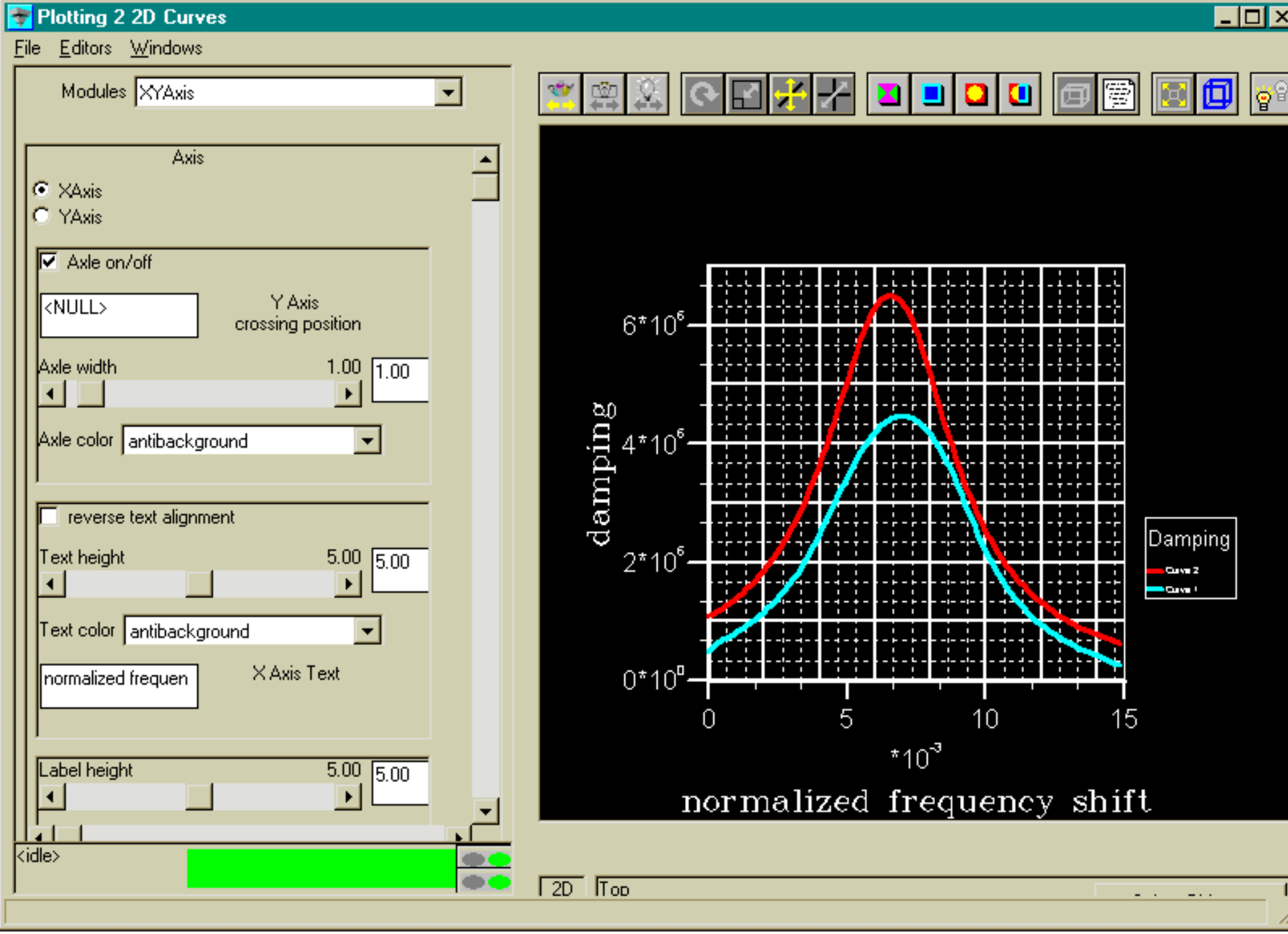


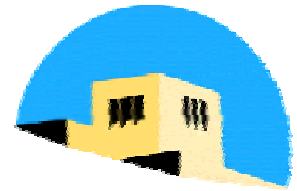


2D graph example

- This is a case where using AVS/Express is harder than using a 2D graphing program, but here goes…
- Download the file **/u/demo/AHPCC_Vis/AVS/data/FIGURE7.DAT** to your account.
  - FIGURE7.DAT is an ASCII text file containing 3 columns of data. The first column is the x axis; the second and third are y values for different curves.
- The network on the next page uses **Read Column File** to read in x and two y columns. **Table to scatter field** converts this to a field. Array extractors are used to obtain x and y as arrays, which are the inputs to the graph objects.

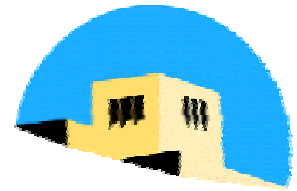



Plotting 2 2D Curves
File
Editors
Windows
Modules
XYAxis
Axis
XAxis
YAxis
Axle on/off
<NULL>
Y Axis
crossing position
Axle width
1.00
1.00
Axle color
antibackground
reverse text alignment
Text height
5.00
5.00
Text color
antibackground
normalized frequen
X Axis Text
Label height
5.00
5.00
<idle>
damping
6*10^6
4*10^6
2*10^6
0*10^0
0
5
10
15
*10^-3
normalized frequency shift
Damping
2D
Top



Taking your workshop examples home

- To move your examples to another machine, you will need to edit some of your project files to make sure saved directory and file paths are correct for the new location. The relevant project files are:
  - avsenv
  - application files (*.v)
  - AVS field files
- PC Users
  - On the desktop (C:\WINDOWS\DESKTOP), you will find free PC versions of ZIP/UNZIP and TAR programs.



Additional AVS References

- Materials on the WWW
  - http://www.sdsc.edu/~mjb/avs
  - http://www.iavsc.org/express/courses/intro-xp
  - http://www.iavsc.org/express/courses/intro-xp/workbook
  - http://www.man.ac.uk/MVCtraining/
- AVS Manuals
  - Getting Started
  - Visualizing Your Data with AVS/Express
  - User's Guide