4 Visualizing 3D arrays of data

4.1 Aims of the chapter

The aim of this chapter is for you to learn techniques which are used to visualize 3D data. You are to work through several structured exercises which will demonstrate the visualization process: creating field files, constructing applications and manipulating data in 3D. The solutions to the field files for 3D data are given in Appendix II .

4.2 Exercise 1

The aim of this exercise is to visualize a downsized version of the AVS/Express sample dataset `fin.fld` which shows the flow of air over a blunt fin.

4.2.1 Importing the `fin.dat` data

Below is a description of the data contained in the datafile `fin.dat`:

```
The data file is called fin.dat and is an ASCII file which represents an 10x8x8
irregular grid. Each data element has 5 floating point data components and each
element has some associated coordinate data to position it in 3D space.
```

The data file format is as follows:

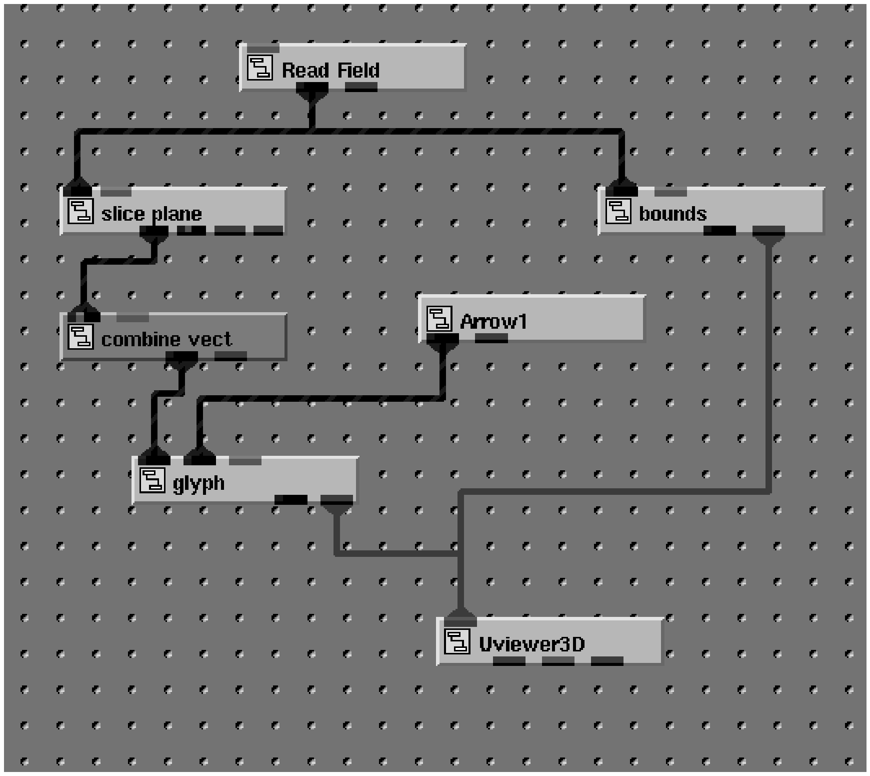
```
density x-momentum y-momentum z-momentum stagnation x-coord y-coord z-coord
```

and a small extract from the file is shown below:

```
2.171 0 0 0 10.7602  0 0 0
2.1523 0 0 0 10.6423  0.0160473 0.129677 0
1.9742 0 0 0 9.61927  0.0757717 0.266559 0
1.6659 0 0 0 7.96548  0.173189 0.379764 0
1.2728 0 0 0 5.94705  0.299646 0.45922 0
0.86119 0 0 0 3.92577  0.443802 0.497866 0
0.7654 0 0 0 3.6417  0.618551 0.501378 0
0.73957 0 0 0 3.52176  1.08654 0.501378 0
0.55002 0 0 0 2.62795  2.47197 0.501378 0
0.45067 0 0 0 2.20615  6.57433 0.501378 0
```

- Create a file called `fin.fld` to describe the data contained in `fin.dat`.
- Use the template field file given on page 26 and the information given above to complete the specification.

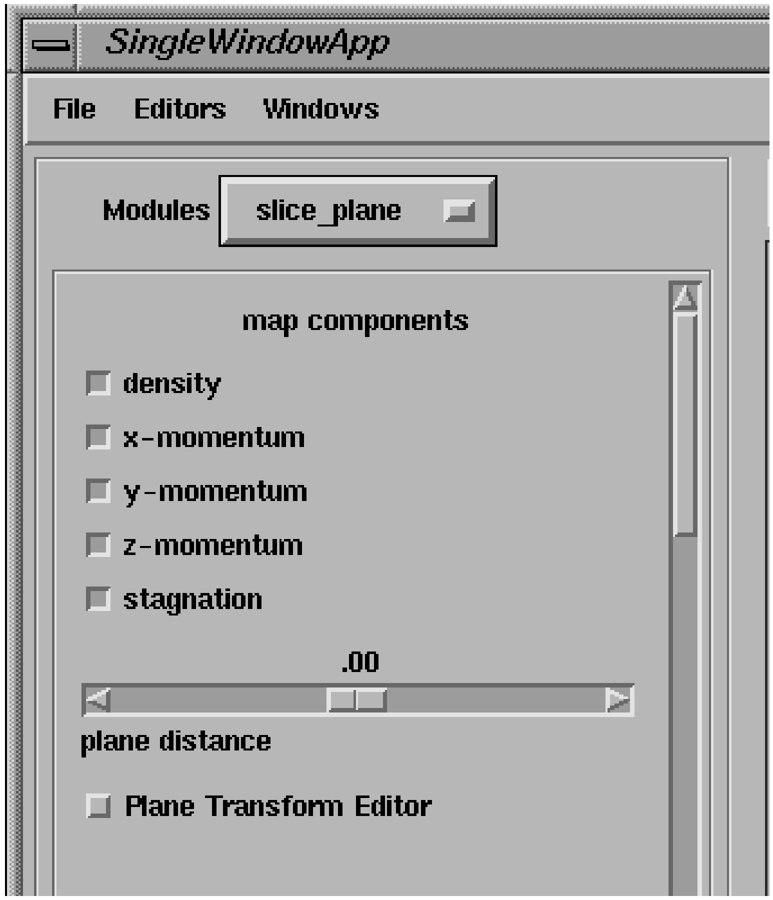
4.2.2 Displaying the vector data



Create the above network using the following modules:

- `Read_Field` (Data Input): reads an AVS field file and converts it into an AVS/Express field.
- `slice_plane` (Mappers): extracts a 2D slice from a 3D field with an arbitrarily positioned slice plane.
- `combine_vect` (Filters): takes all selected scalar data components and combines to output a single vector component
- `bounds` (Mappers): generates a bounding box of a 3D field.
- `Arrow1` (Geometries): creates a wireframe arrow shaped mesh
- `glyph` (Mappers): places a geometrical object at each node of an input field. The glyph is coloured and sized according to the magnitude of the data component at that point.
- `Uviewer3D` (Viewers): creates a 3D image in the DataViewer.

4.2.3 Selecting the map components

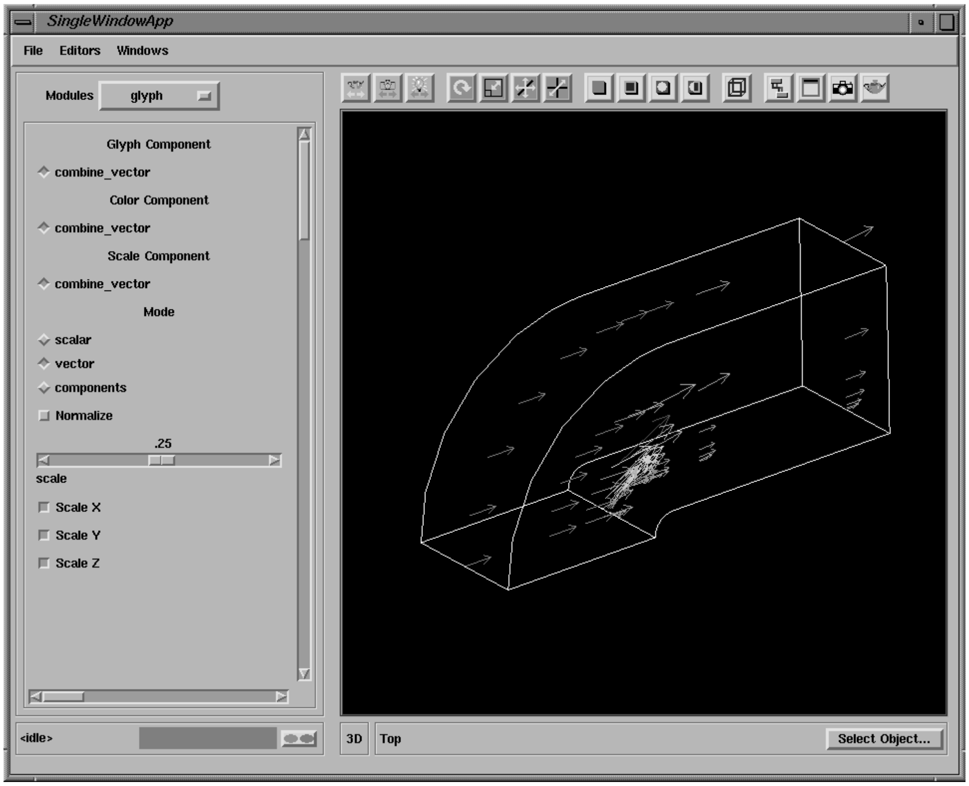


- Select the `Read_Field` module from the **Modules** editor panel and then select the file `fin.fld` which you have just created to import the data using the file browser.
- Select the control panel for the `slice_plane` module and make active all data components, as shown above. For now ignore any error messages which appear.
- This module will now pass on a 3D field with a 5-vector data element consisting of density, x, y and z momentum, and stagnation to the other modules in the AVS/Express network.

4.2.4 Extracting the vector data

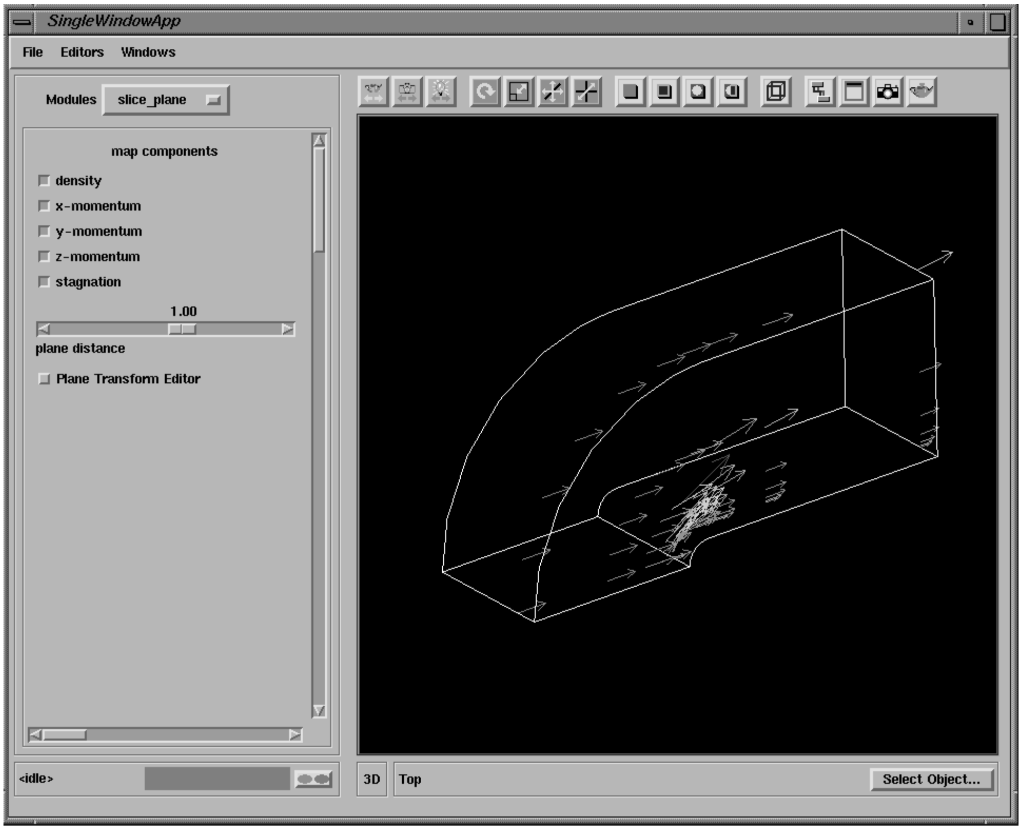
- Select the control panel for the `combine_vect` module and make active the three data components x-momentum, y-momentum and z-momentum, as shown above. Since `veclen` is set to 3 it is only possible to have 3 vector components active at a time.
- This module will now pass on a 3D field with a 3-vector data element consisting of x, y and z momentum to the other modules in the AVS/Express network.

4.2.5 Using the `glyph` module



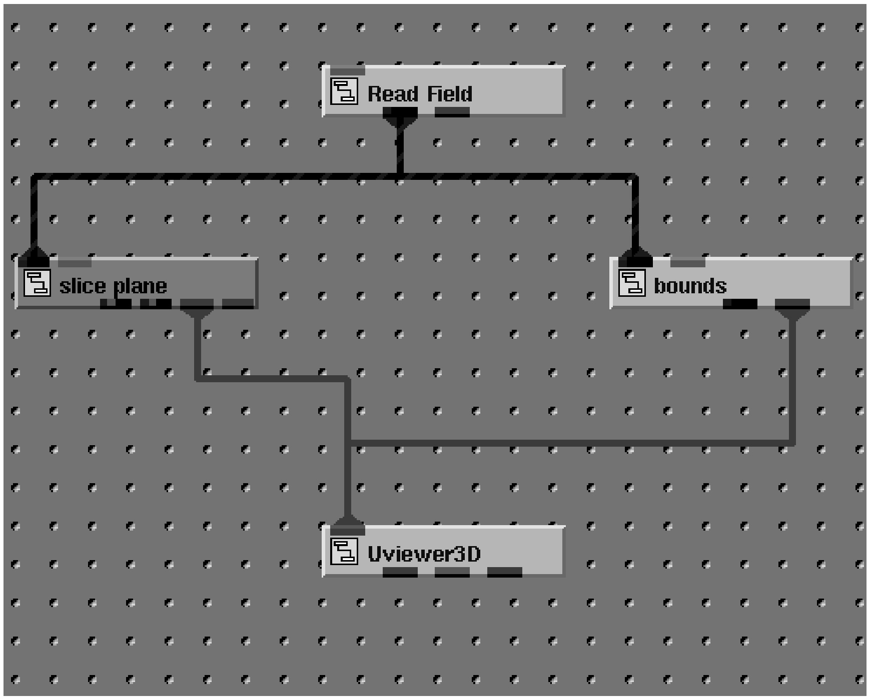
- Select the `glyph` module's control panel.
- Reduce the scale value using the slider to make the arrows a more acceptable size
- Try altering some of the parameters - use the module help facility to understand their effect.

4.2.6 Moving the **`slice_plane`**



- Select the `slice_plane` module's control panel.
- Alter the plane distance value to observe the value of momentum at different positions within the data set.
- Try altering some of the plane transformation parameters to change the orientation of the slice plane.

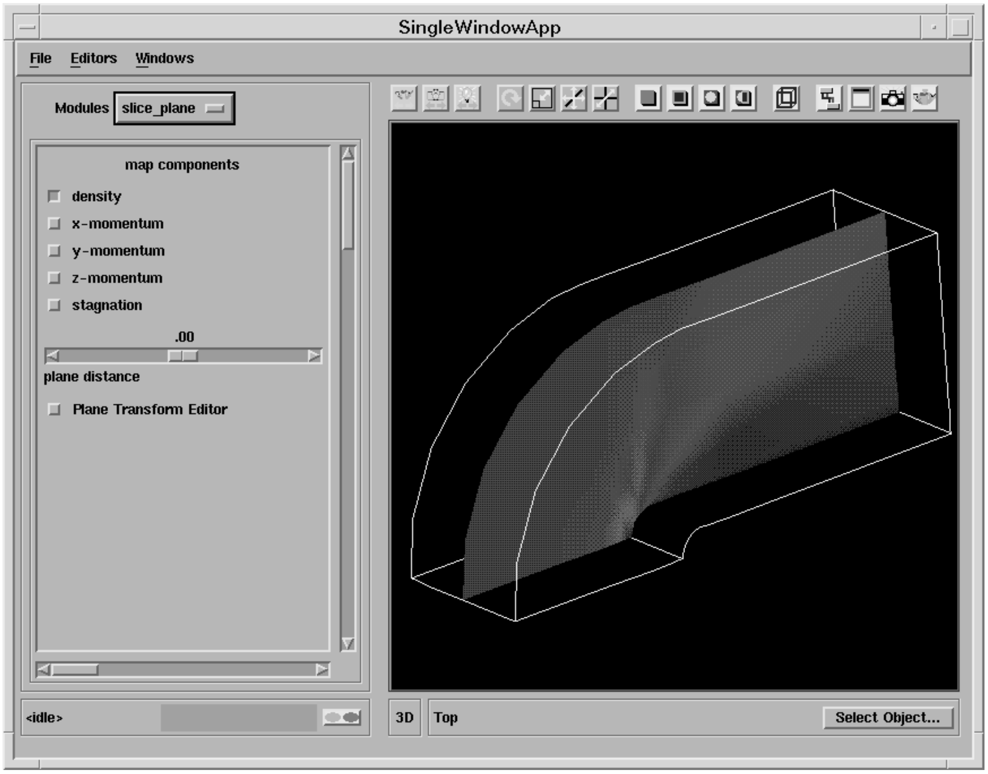
4.2.7 Visualizing the scalar data



Change the network to create the above network:

- Read_Field (Data Input): reads an AVS field file and converts it into an AVS/Express field.
- slice_plane (Mappers): extracts a 2D slice from a 3D field with an arbitrarily positioned slice plane.
- bounds (Mappers): generates a bounding box of a 3D field.
- Uviewer3D (Viewers): creates a 3D image in the DataViewer.

4.2.8 Using the `slice_plane` module



- Select the `Read_Field` module from the **Modules** editor panel and then select the file `fin.fld` to import the data using the file browser.
- Select the control panel for the `slice_plane` module and make active the density data component.
- This module will now pass on a 3D field with a scalar data element to the other modules in the AVS network.
- The viewer window should now show something similar to the above figure.
- Use the module's control panel to change the slice position.
- Try extracting the other scalar components using the module.

4.3 Exercise 2

The aim of this exercise is to visualize some data obtained from a climate modelling experiment. The data contains values of temperature, specific humidity and u, v, & w components of wind velocity, which are sampled over the surface of the globe and at several altitude levels. There are two sets of coordinates for the data sampling: rectangular and spherical.

Firstly, you are to present the scalar data - temperature and specific humidity - using the spherical coordinate system. Secondly you are to present the vector data, the resultant wind velocity, using the rectangular coordinate system. The solutions to the field files for this exercise are given in Appendix II .

4.3.1 Importing the climate data

Below is a description of the data files containing the climate modelling data.

Coordinates: The data is irregularly spaced on a 49x12x40 grid. The coordinates are provided in ASCII files ccm1.rect which gives the coordinates in rectangular space and ccm1.sph which gives the coordinates in spherical space.

Both files have one line of header information and are arranged with all the coordinates of the first dimension followed by all the coordinates of the second dimension, followed by all the coordinates of the third dimension. There are eight values on each line of the files i.e.

```
x1 x2 x3 x4 x5 x6 x7 x8
x9 x10 .............
y1 y2 y3 y4 y5 y6 y7 y8
y9 y10 .............
z1 z2 z3 z4 z5 z6 z7 z8
z9 z10 ............
```

Temperature and Specific Humidity: Temperature and specific humidity data are contained in binary files called temp.dat and humid.dat, respectively. The data consists of a single floating point value at each grid point. There are 16 bytes of header information in each file.

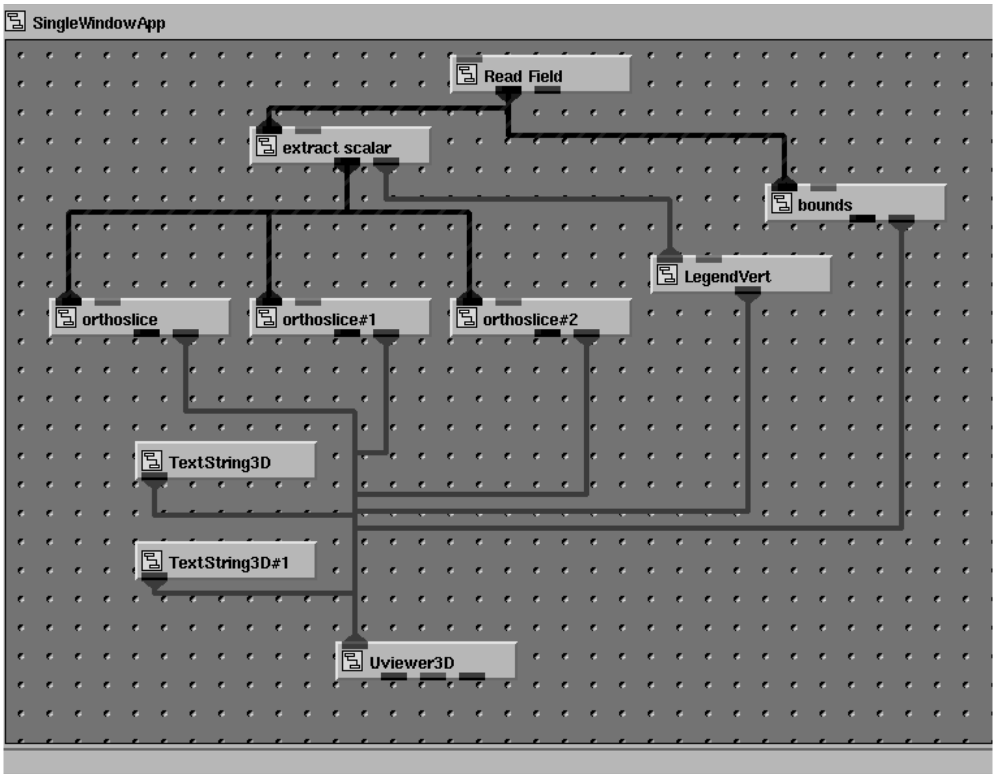
Wind Velocities: The wind velocity data is contained in three binary files of floating point values for the u, v & w components. The files are called bi.u.49, bi.v.49 and bi.w.49, respectively. Each file contains 16 bytes of header information.

World Map: A simple map of the world showing land/ocean outlines is contained in the file world.asc. The file is in ASCII format and irregularly spaced on the same grid as the other files. There is one line of header information.

In order to visualize this information you will need to write three field files:

- Create a field file called `climscal.fld` to describe the scalar data - temperature and specific humidity - using the spherical coordinate system.
- Create a field file called `climvect.fld` to describe the vector data - wind velocity - using the rectangular coordinates
- Create a field file called `worldrec.fld` to describe the world map using the rectangular coordinates.

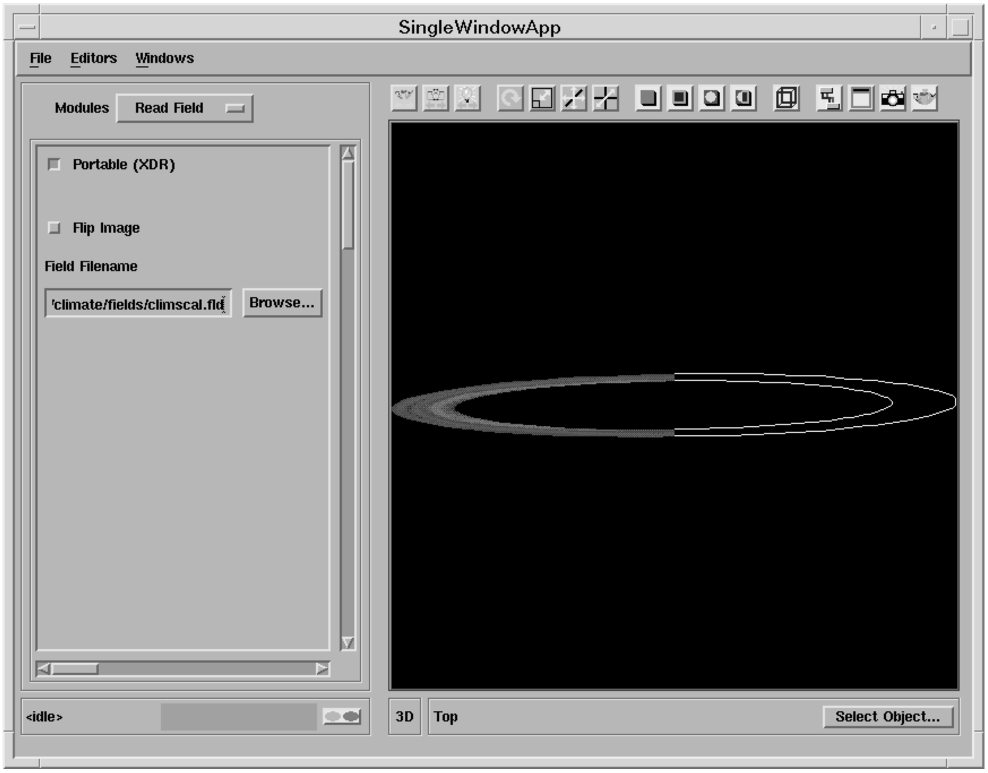
4.3.2 Visualizing the scalar data



Create the above network using the following modules:

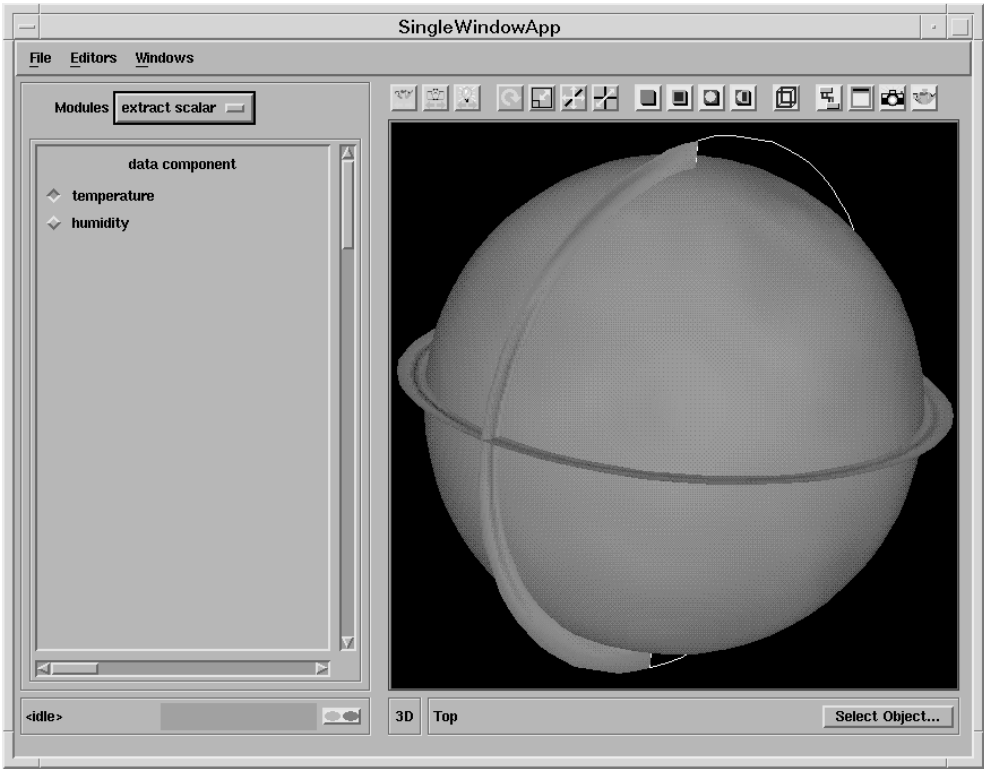
- `Read_Field` (Data Input): reads an AVS field file and converts it into an AVS/Express field.
- `extract_scalar` (Filters): extracts a single scalar data element from a field's vector component.
- `orthoslice` (Mappers): produces a slice of a structured field perpendicular to a selected coordinate axis.
- `bounds` (Mappers): generates a bounding box of a 3D field.
- `LegendVert` (Geometries): produces a color legend which displays the object's datamap.
- `TextString3D` (Geometries): produces a label.
- `Uviewer3D` (Viewers): creates a 3D image in the DataViewer.

4.3.3 Changing the `orthoslice` parameters (1)



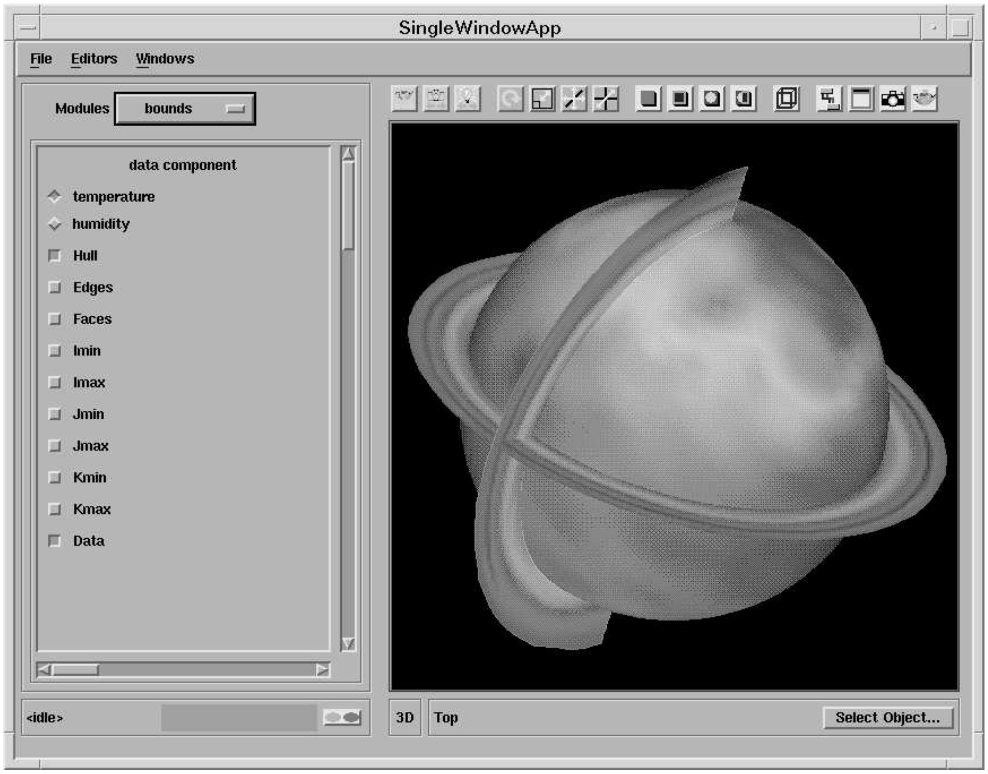
- Select `Read_Field` from the **Module Editors** menu and choose `climscal.fld` from the file browser.
- You should obtain a picture similar to the one above.
- Select the `extract_scalar` module and check that **temperature** is selected.
- Select the `orthoslice`, `orthoslice#1` and `orthoslice#2` modules in turn and set the **axis** values to 0, 1 and 2, respectively.

4.3.4 Changing the legend and label parameters



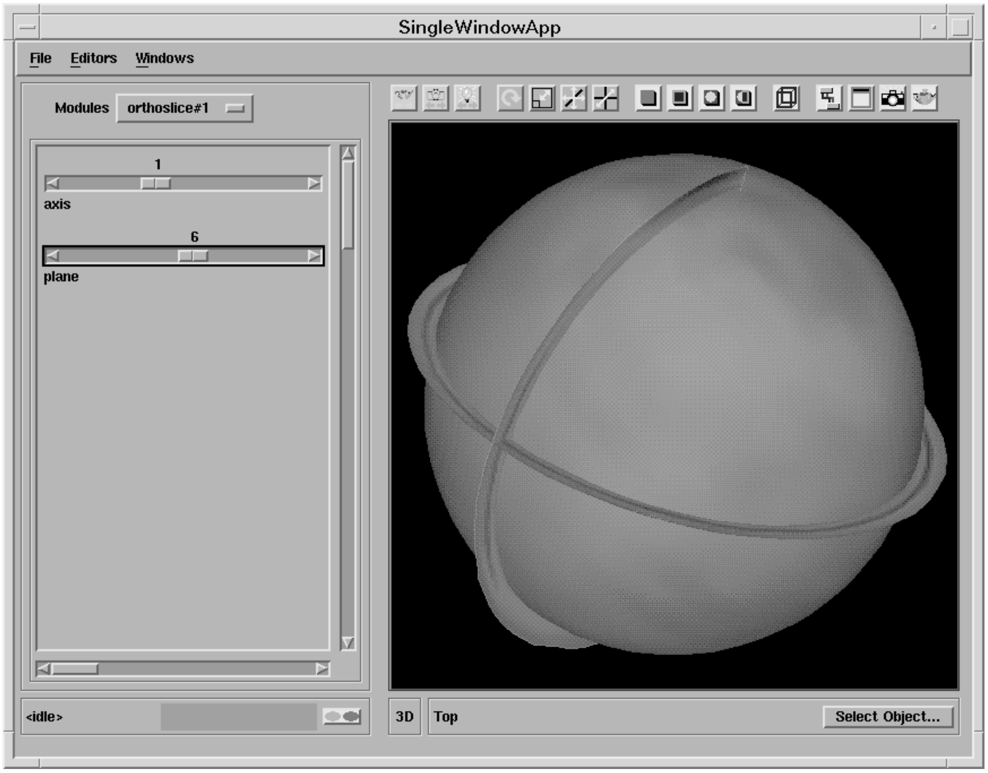
- Rotate the picture so that it appears as above.
- Select the `LegendVert` module and change the parameters to make them look optimal in the picture.
- Select the `TextString3D` modules and enter suitable text into the **string** box. Also change the position variables if necessary.

4.3.5 Changing the `orthoslice` parameters (2)



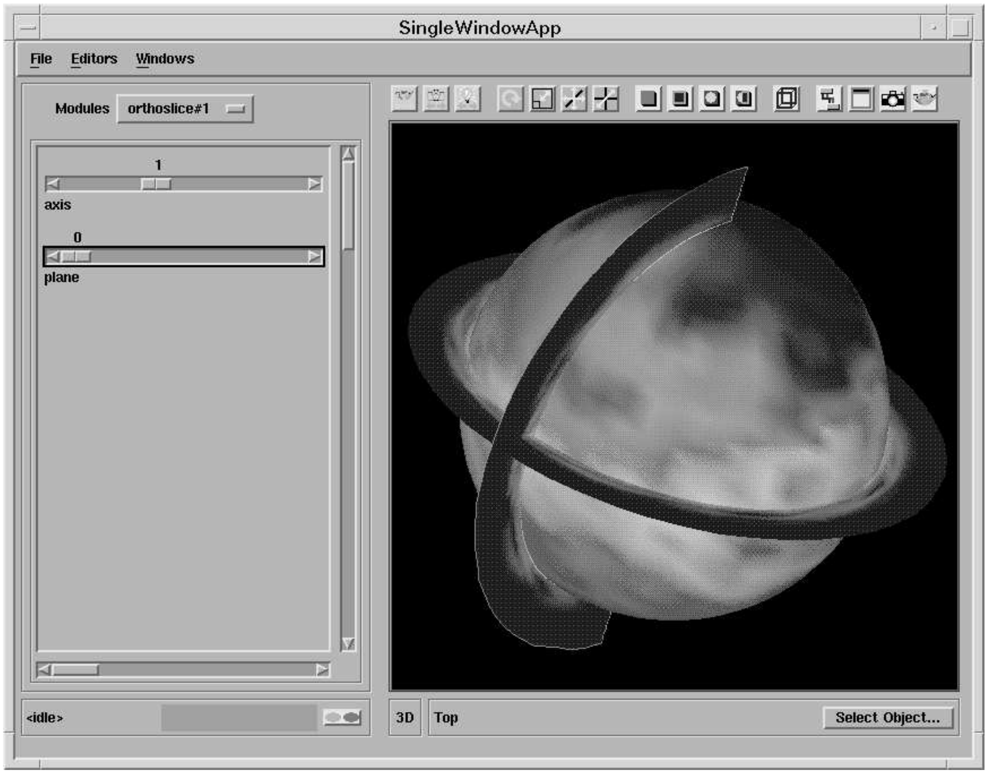
- Select the `orthoslice` and `orthoslice#1` modules, (or the modules with axes set to 0 and 1), and reduce the **plane sliders** to 0.
- You should obtain the above picture.
- You may wish to improve the picture further by selecting the `bounds` module and either removing the **hull** or selecting **data**, to colour the bounds to the boundary value of the data.

4.3.6 Looking at the temperature data



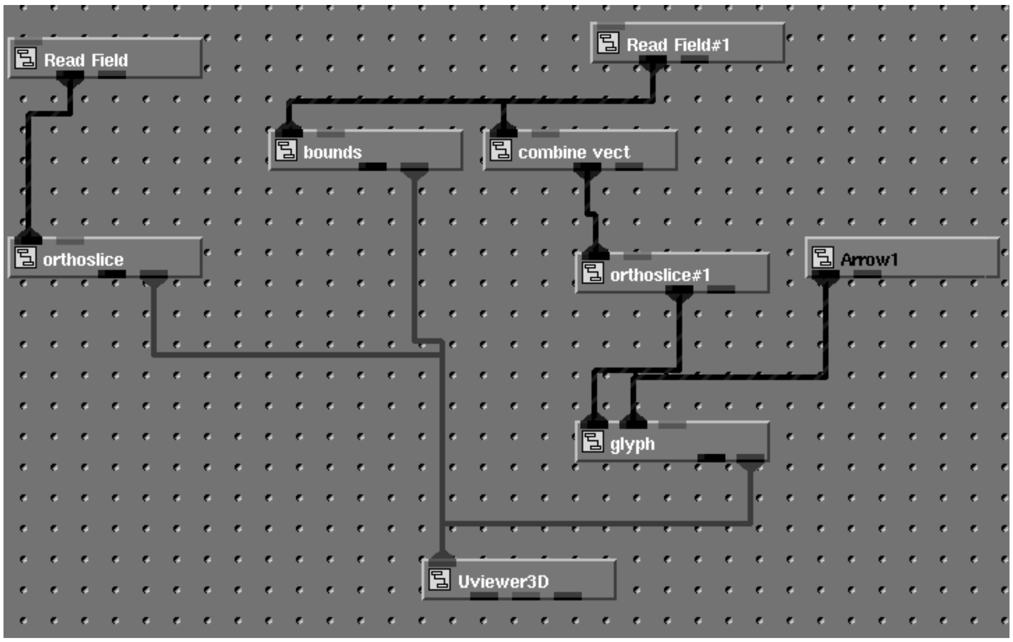
- Select the `orthoslice#1` module whose **axis** is set to 1, you should have just set the **plane slider value** to 0.
- This is the temperature around the surface of the earth.
- Increase the value of the **plane slider** and observe how the temperature of the atmosphere changes as altitude increases.

4.3.7 Looking at the humidity data



- Select the `extract_scalar` module and select **specific humidity**.
- Investigate how specific humidity varies with altitude.

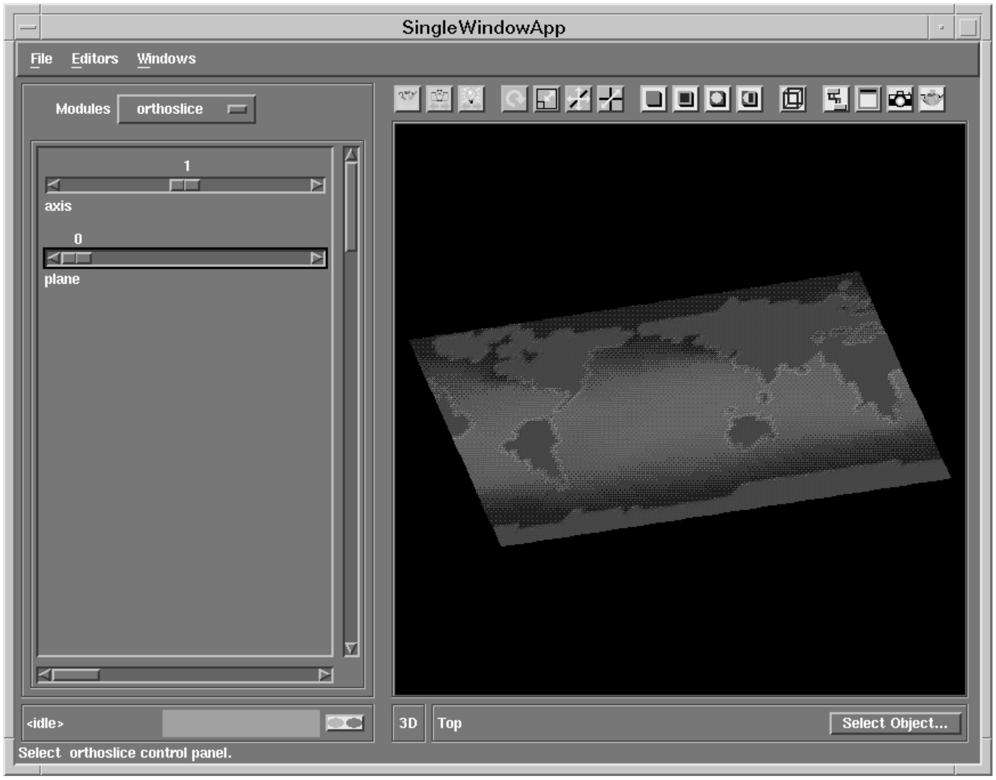
4.3.8 Visualizing the vector data



Create the above network using the following modules:

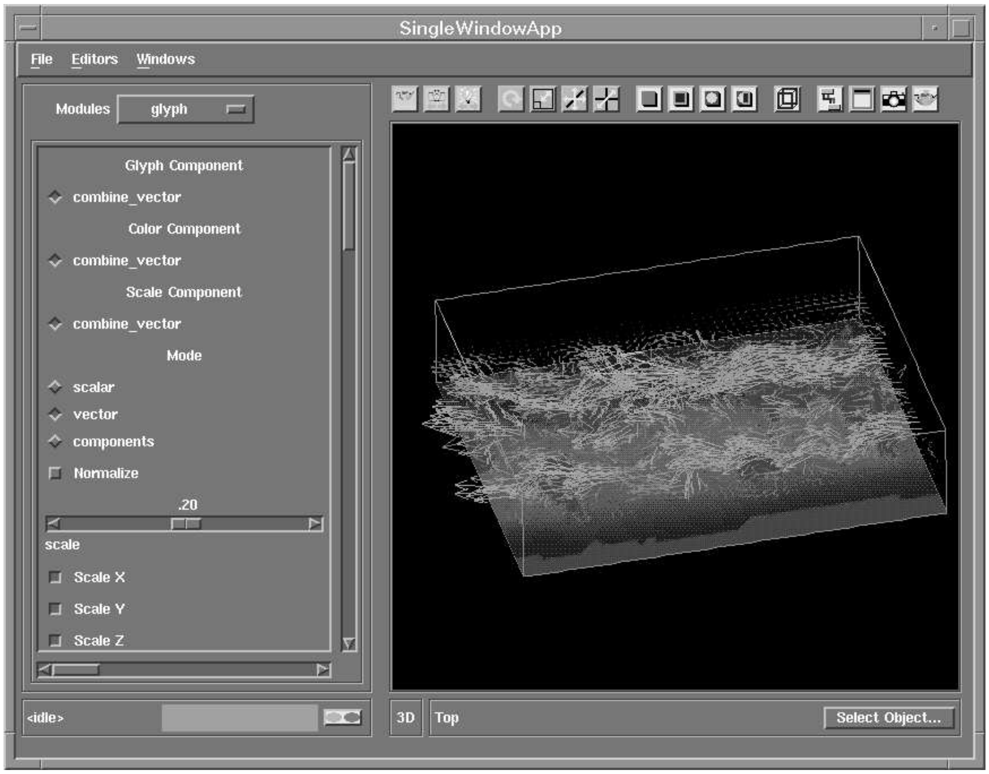
- `Read_Field` (Data Input): reads an AVS field file and converts it into an AVS/Express field.
- `orthoslice` (Mappers): produces a slice of a structured field perpendicular to a selected coordinate axis.
- `combine_vect` (Filters): takes all selected scalar data components and combines to output a single vector component
- `bounds` (Mappers): generates a bounding box of a 3D field.
- `Arrow1` (Geometries): creates a wireframe arrow shaped mesh
- `glyph` (Mappers): places a geometrical object at each node of an input field. The glyph is coloured and sized according to the magnitude of the data component at that point.
- `Uviewer3D` (Viewers): creates a 3D image in the DataViewer.

4.3.9 Reading the world map



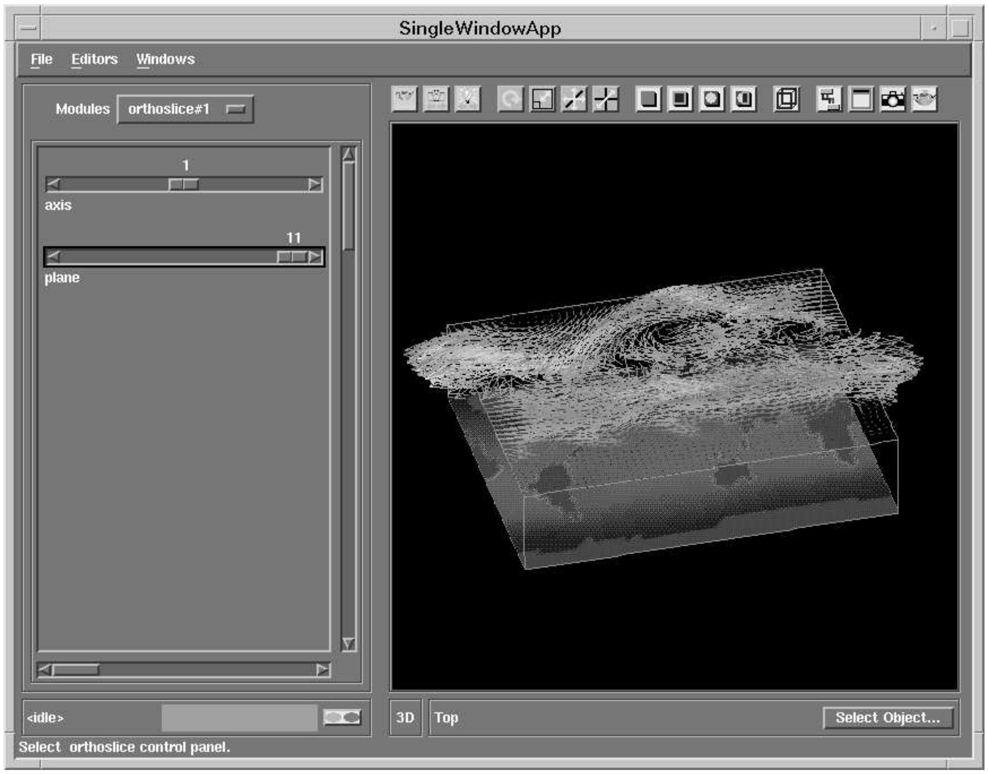
- Select the module `Read_Field` the **Module Editors** menu and choose `worldrec.fld` from the file browser.
- Select the `orthoslice` module which is connected to the `Read_Field` module and set the axis value to 1.
- Reduce the plane slider to 0.
- Using the mouse, rotate the picture until it is orientated as shown above.
- Hopefully the above picture will look familiar.

4.3.10 Reading the velocity data



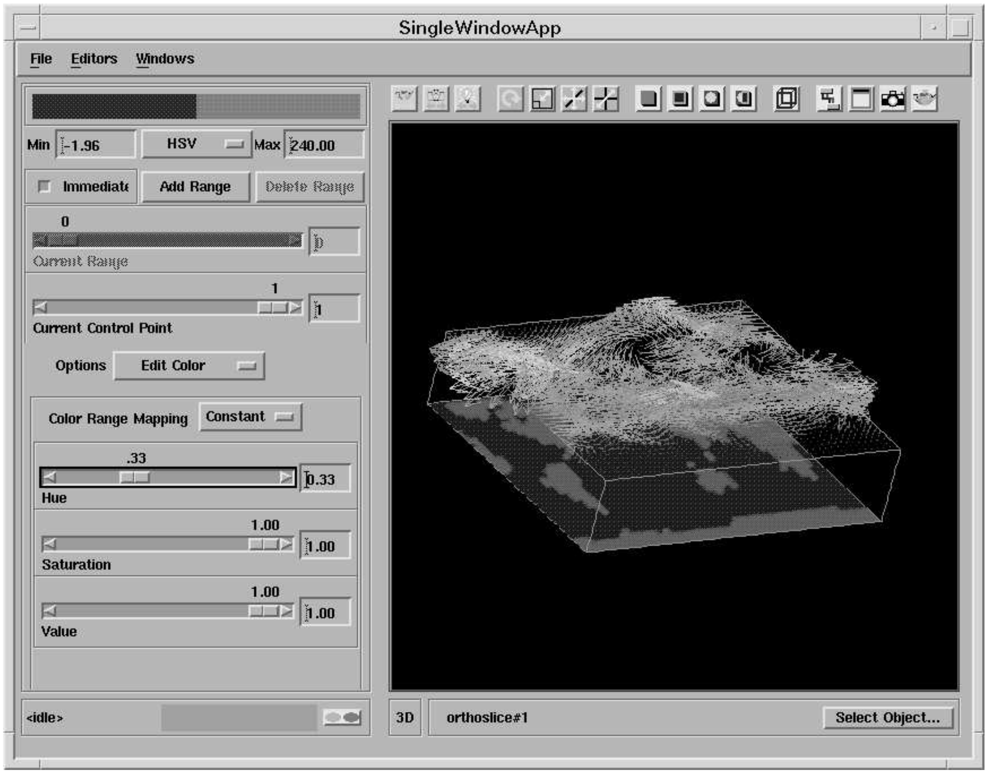
- Select the `Read_Field#1` module and choose `climrect.fld` from the file browser.
- Select the `combine_vect` module and check that **u-comp, v-comp** and **w-comp** are all selected.
- Select the `orthoslice` module which is attached to the `Read_Field` and set the **axis** to 1.
- Select the `bounds` module which is attached to the `orthoslice` module and remove the **hull**.
- Select the other `bounds` module and select **data** to colour the bounds.
- Select the `glyph` module and reduce the **scale** to ~0.2 or until the arrows are a sensible size.
- You should have obtained the above picture.

4.3.11 Looking at the velocity data



- Select the `orthoslice` module concerned with velocities and set the **plane slider** value to 0.
- This shows the velocities on the surface of the earth.
- Increase the plane value and observe how the wind velocities change with increasing altitude.
- The picture above shows the wind velocities at the highest altitiude level.

4.3.12 Changing the datamap



- To improve the appearance of the picture, attempt to alter the **datamap** of the world map so that the land is coloured green and the oceans are coloured blue.
- You may want to use the notes from Section 3.6.5 on page 36 to help you.

(Hint: Select the world map as the current object and make the colour map constant. Alter the values of the control points.)

4.4 Using the software renderer

You can use the renderer option to alter the renderer which is used to draw the display. There are several options:

- **Software**: software renderer which uses its own graphics rendering techniques.
- **Hardware**: uses the workstations own software and hardware graphics techniques.

The hardware renderer is used as the default. However, when using the volume renderer to look at 3D images, it is often necessary to use the software renderer. The software renderer is selected in the following way:

- Select the **View editor** from the **Editors** pull down menu.
- In the **Renderer** menu select **Software**.
- Proceed with the example as usual.

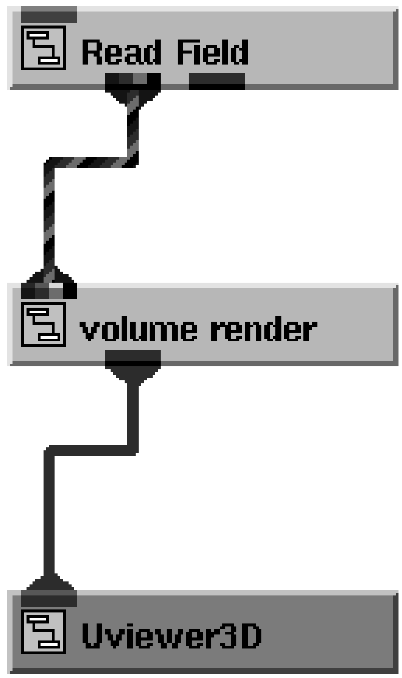
The rest of the exercises in this chapter are concerned with 3D representation of various fields. The examples use the `volume_render` module and all require the use of the software renderer.

Note: Use of the software renderer will significantly slow down the visualization process - but for volume rendering this is unavoidable.

4.5 Exercise 3

The volume to be represented in this example is a 3D image of a human head.

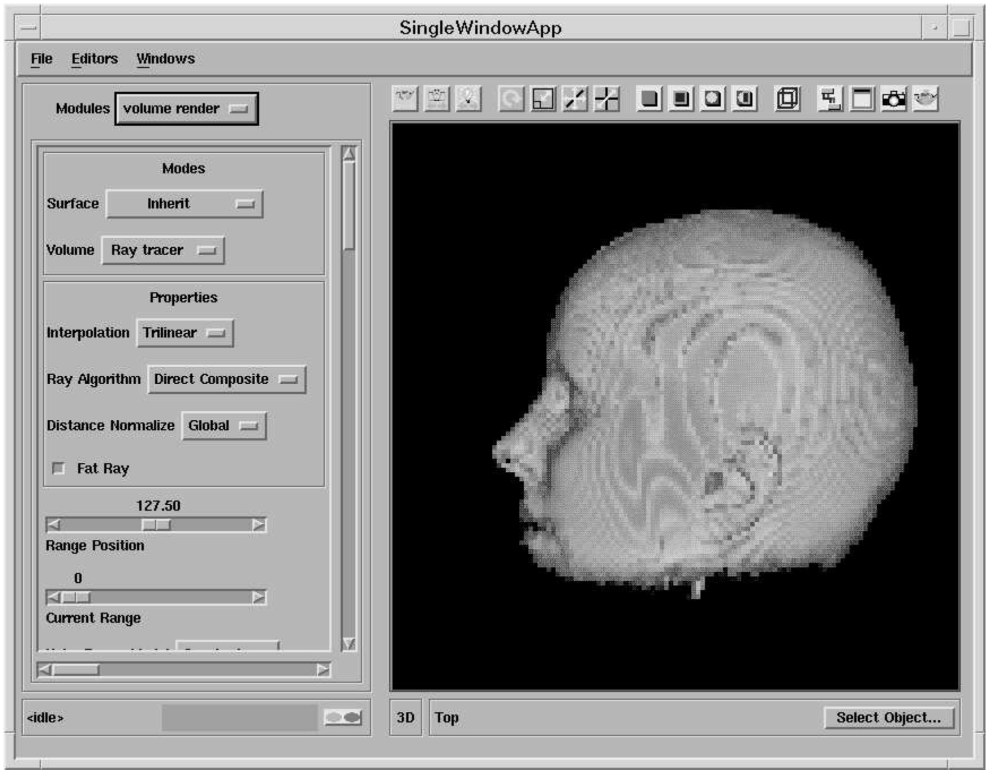
4.5.1 Displaying the `head.fld` data using `volume_render`



Create the above network using the following modules:

- `Read_Field` (Data IO): reads an AVS field file and converts it into an AVS/Express field.
- `volume_render` (Mappers): directly renders a volume.
- `Uviewer3D` (Viewers): creates a 3D image in the Data viewer.

4.5.2 Normalizing the picture



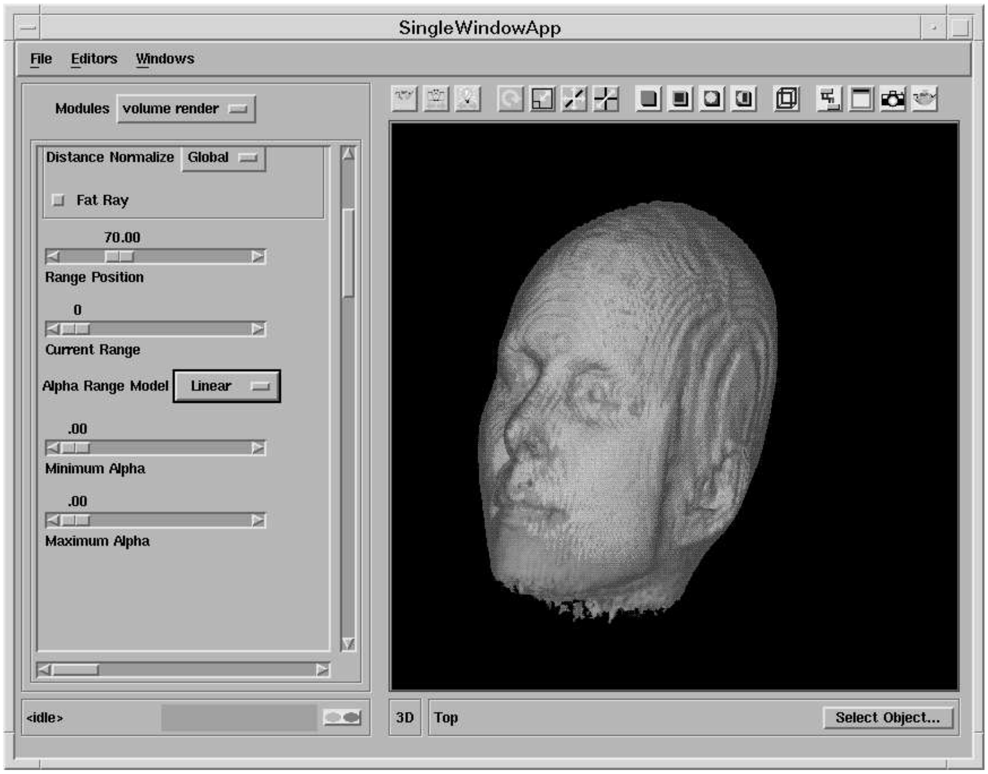
- Select the `Read_Field` module's control panel and select the file `head.fld,` from the file browser, to import the data. The `head.fld` file can be found under the path `/usr/avs/auxdata/...`

Note - if an error message is reported when `Read_Field` is selected, or if no image appears, it is likely that you have forgotten to switch on the software renderer

- Click on the **Reset/Normalize/Center** icon in the Toolbar so you can fit the whole object in the window. The head should now be visible.
- Select the `volume_render` module's control panel as shown above.

The steps listed on the next page will improve the appearence of the image and explain some of the properties of `volume_render.`

4.5.3 Using `volume_render`



- Switch the **surface mode** to **Gouraud**. This selection provides the highest quality rendering - including both lighting and interpolation.
- Switch off the **Fat Ray** option. The resolution of the image should improve.
- The `volume_render` module has two default datamaps. The range to be edited can be selected by altering the value of **Current Range**. The **Range position** value allows the mid-point of the two ranges to be altered. Set the Range position value to approximately 70.
- Set the **Alpha Range Model** to **Linear**. Now the alpha component is interpolated between the maximum and minimum of the current data range. The other option is **constant**, the alpha component for the whole data range is set to the minimum alpha value.
- Rotate the head so it appears as above.
- Try altering some of the other `volume_render` parameters and observe the effect.

4.6 Exercise 4

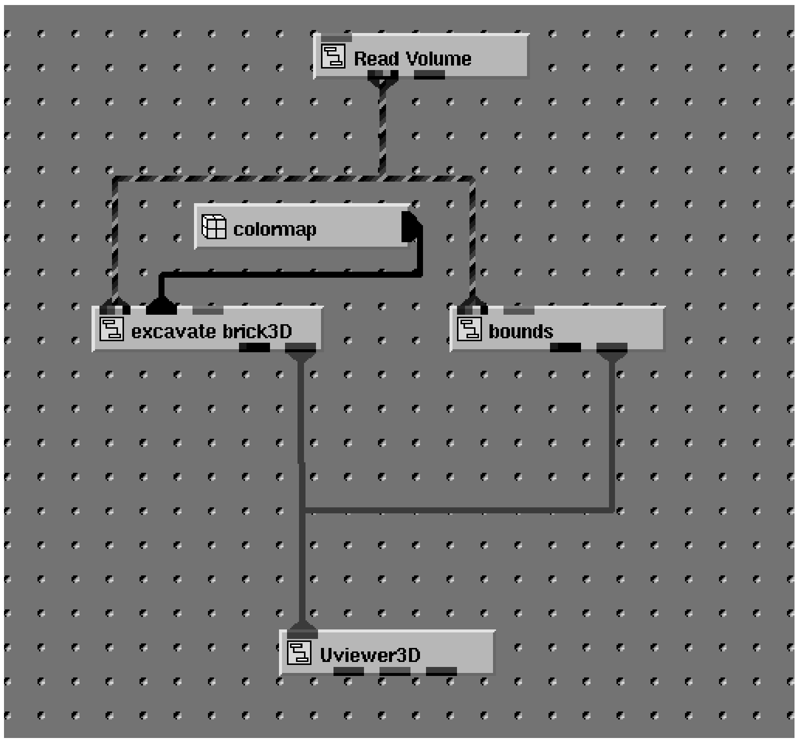
In this Exercise the electron density of the hydrogen atom will be represented, this time in 3D.

- Use the network constructed in Exercise 3.
- Select the `Read_Volume` module's control panel and select the file `hydrogen.dat,` in the file browser, to import the data.
- Click on the **Reset/Normalize/Center** icon in the Toolbar so you can fit the whole object in the window.
- Change some of the parameters of `volume_render` to improve the appearance of the image. Use the explanations of the properties of the `volume_render` module given in the previous exercise.
- Using the mouse, transform the object.
- Finally, try adding a `slice_plane` module to the network and investigate the results obtained.

4.7 Exercise 5

This example will use the `excavate_brick` module to observe the 3D image of a lobster encased in a brick.

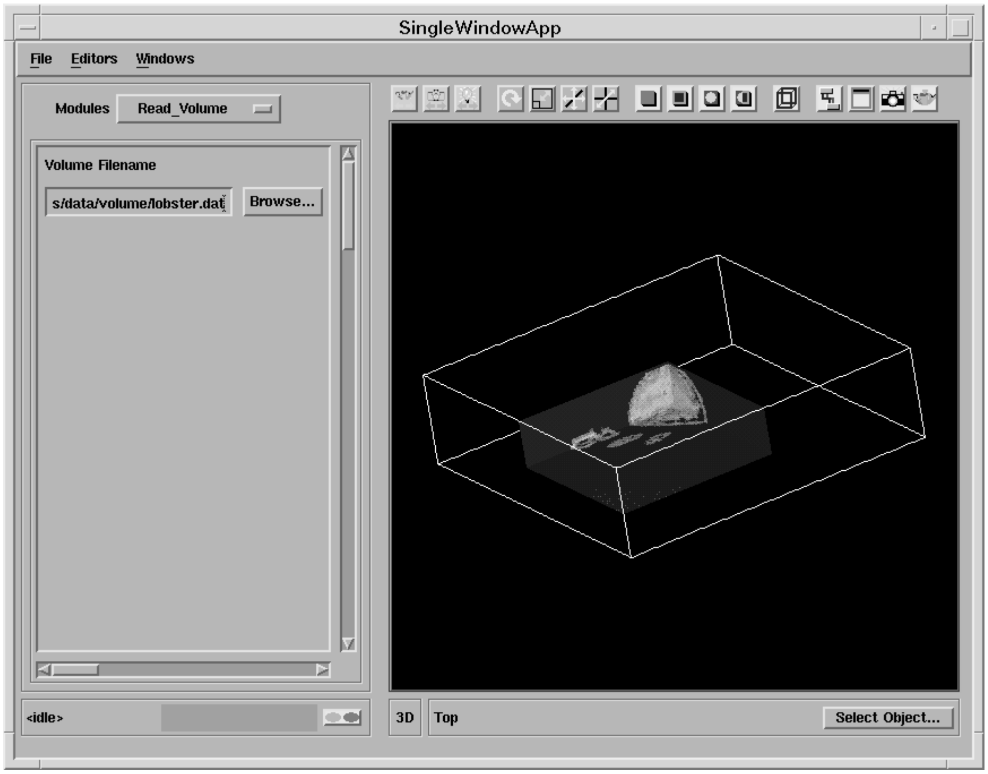
4.7.1 Displaying the `lobster.dat` data using `excavate_brick`



Create the above network using the following modules:

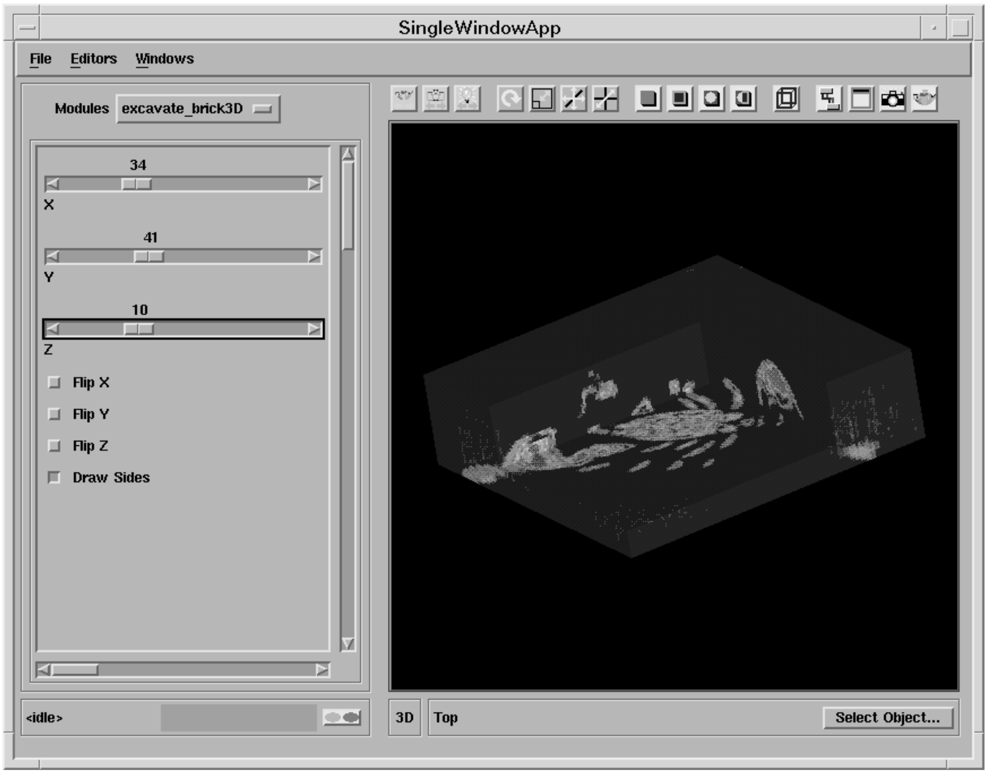
- `Read_Volume` (Data IO): reads a volume format file and outputs an AVS/Express field.
- `colormap` (Data IO): generates a coloured image.
- `excavate_brick3D` (Mappers): technique for visualizing 3D uniform volume. Volume is displayed with an X, Y & Z slice plane, which removes a rectangular sub-volume of the field, revealing the internal structure.
- `bounds` (Mappers): generates a bounding box of a 3D field.
- `Uviewer3D` (Viewers): produces a 3D image in the Data Viewer.

4.7.2 Normalizing the picture



- Select the `Read_Volume` module's control panel and select the file `lobster.dat` to import the data.
- Click on the **Reset/Normalize/Center** icon in the Toolbar so you can fit the whole object in the window.
- Now, using the mouse, transform the object so it appears as above.

4.7.3 Using `excavate_brick`



- Select the `excavate_brick` control panel from the **Modules** menu.
- Turn on **Draw Sides** - this allow the sides of the brick to be drawn.
- Try altering some of the other parameters and observe their effect.
- Select the `bounds` control panel and remove the **Hull**.