Stanford Exploration Project, Report 84, May 9, 2001, pages 1–??

SEP AVS User Guide

Robert G. Clapp[1]

ABSTRACT

The last 10 years has seen a steady increase in the number of 3-D seismic surveys. This increase has led to a push to develop innovative tools to deal with the unique challenges that 3-D data present. SEP has chosen to work within the framework provided by Advanced Visual Systems (AVS) for its 3-D work. This paper attempts to summarize current status of SEP's AVS environment; provide some useful hints when problems are encountered running AVS; and give a brief overview of where we are headed in the near future.

INTRODUCTION

As 3-D seismic data becomes the standard rather than the exception, new tools have to be developed to deal with the new challenges that 3-D data presents. Currently, several software packages/companies (GOCAD, Landmark, AVS and GSI) are competing to provide the seismic world with tools to meet these challenges. SEP has chosen to use AVS as its 3-D platform due to its flexibility and large user base inside and outside the seismic industry. AVS offers an interactive 3-D display environment that is not available in standard Unix (SEPLIB), and allows rather straightforward inclusion of Unix batch programs into it's environment, unlike GOCAD. In order to provide people inside and outside of SEP a brief guide to what we can now do with AVS, I have attempted to provide a synopsis of SEP's current AVS abilities. The paper is broken down into three parts. The first part summarizes some useful networks available at SEP and how to use them. The second part is a brief summary of the modules developed at SEP, and the final part attempts to answer some of the more common problems that occur when using AVS.

USEFUL NETWORKS

3-D View

Creates a 3-D object from a 2-D slice. The third dimension is created by transforming the independent values in the 2-D array into the z-coordinate. First, select the dataset using the file browser widget attached to the `Read SEP` module. Choose the `scale` boolean widget if scaling is appropriate (when axis ranges are not comparable). Once the dataset has been chosen a 2-D

[1]**email:** bob@sep.stanford.edu

slice must be selected. The `ortho slicer` performs this function. It allows the user to select a 2-D plane from a 2 or 3-D input array. The user specifies which dimension to slice along, and which plane along that axis to select (using the attached choice and slider widget). The slicer is defaulted to slice along the third dimension and to select the first slice in that dimension.

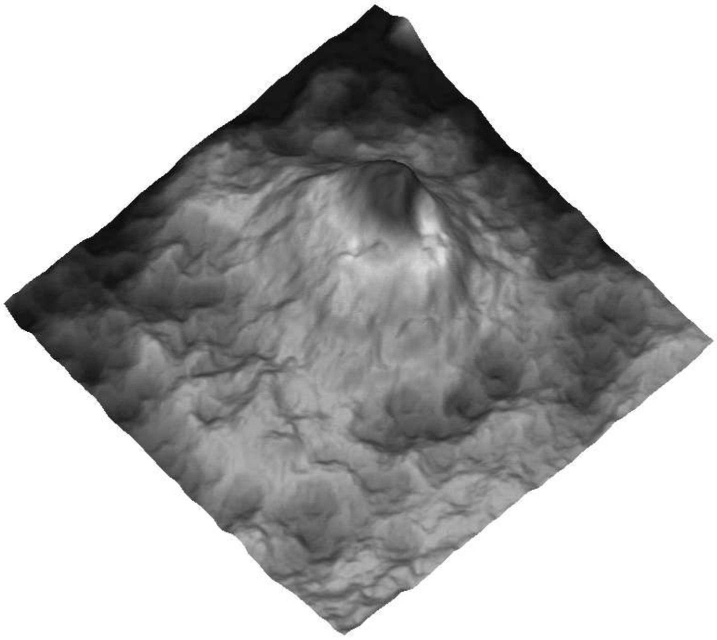

Figure 1: 3-D Phase Unwrapped View of Mt. Vesuvius. bob1-3dview2 [NR]

Once the appropriate parameters have been selected within the `ortho slicer` the data is passed to the `field to mesh` module. The `field to mesh` module then converts the data 2-D array of values into a 3-D surface by mapping each value of the array to given elevation determined by the value at the coordinate location multiplied by the `Z scale` dial widget value. Figure 1 illustrates a sample geometry viewer scene, in this case a phase unwrapped view of Mt. Vesuvius (see (Chemingui et al., 1995)). The network also allows the user to include a color table through the `generate colormap` module (see **Generate Colormap Troubleshooting** if problems are encountered). Further the `image to postscript` module is included to allow scenes in the `geometry viewer` to be captured into a postscript file. The name of the postscript file is determined through the attached `file name` typein widget.

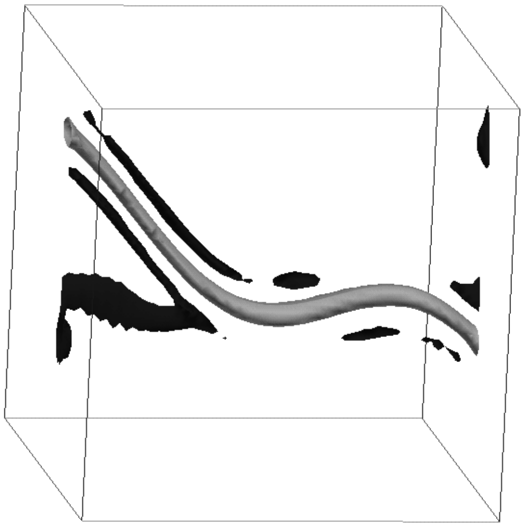

Figure 2: Isosurface of the inversion result for a river channel in a homogeneous medium. bob1-isso [NR]

Isosurface

Constructs an isosurface(s), which can be thought of as a 3-D representation of a 2-D contour line. The cube is selected using the file browser widget within the `Read SEP` module. In the `isosurface` module(s) select the value to map (`level` dial widget). The resulting image should have the form of Figure 2. If it does not appear in the `geometry viewer`, follow the procedure under **Geometry Viewer Troubleshooting**. If you are unsure what to map, the **Generate Colormap Troubleshooting** section might help.

Modeling

Uses Martin Karrenbach's modeling code to perform 2-D acoustic modeling. The user selects the header file of the velocity cube using the file browser widget attached to the `Browser_cm` module. This information is passed along to the `Browser cm` module which reads in the portion of the dataset defined by the user by the slider widgets entitled `(f1, f2, f3, f4, j1, j2, j3, j4, n1, n2, n3, n4)`. Once the user has defined the portion of the dataset they want to read in, one selects the `browser enter` boolean widget, allowing the field to be passed along to the `Slicer cm` module. The `Slicer cm` module selects which plane to model by altering the `HAxis, VAxis`, and `slice number` widgets. In addition to the velocity slice, `Ultimod` requires a wavelet as input. Using the SEPLIB program `Wavelet` or something comparable the user create the desired wavelet.

The next step is to set the desired parameters within `Ultimod`. The source location is modified by the `src_depth` and `src_i` widgets, all other parameters are fairly self explanatory. When the desired parameters have been set, execute the module through the `runit` boolean widget. At this point the time snaps should be fed to the `geometry viewer`. If they do not appear select the `network execution` from `network tools` menu. If the slice is not visible within the `Geometry Viewer` (Figure 3 gives an example of what you see superimposed on a GOCAD surface model (Clapp et al., 1994)) follow the procedure described in **Geometry Viewer Troubleshooting** . If the entire slice appears to be a single color, or has a very limited resolution follow the procedure under **Generate Colormap Troubleshooting**. The image can be additionally improved by modifying the `Gain_cm` module's widgets.

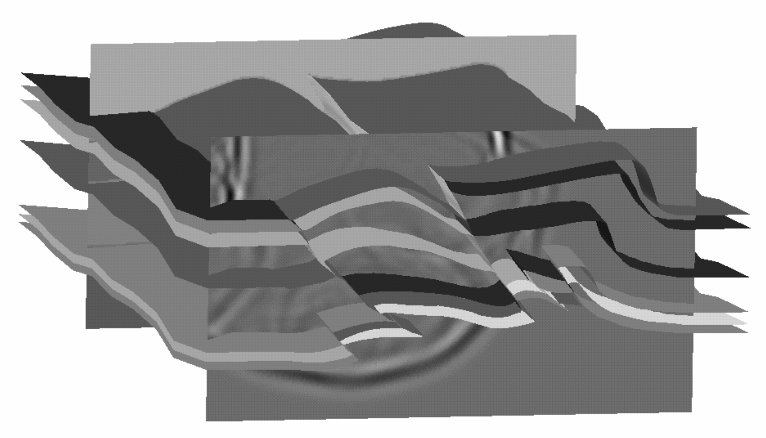

Figure 3: Wavefield propagating through an earth medium. bob1-snap2 [NR]

Pick Info

Allows a line to be drawn on a 2-D slice, and the coordinates of the selected points to be saved into a SEP file. First, select the dataset to be loaded using the `file browser` widget in the `Read SEP` module. If the axes are not compatible (such as one goes from 0 3000, the other 0 to 5), choose the `scale` option. Select the plane to be viewed using the `ortho slicer`. Follow the procedure under **Geometry Viewer Troubleshooting** and **Generate Colormap Troubleshooting**, to bring up and improve the resolution of the image. Then begin selecting points on the image. If you make a mistake you can use the `remove point` boolean widget. If you wish to start a new line select `new line` boolean. Figure 4 illustrates how the geometry viewer scene will look with the superimposed picked line on top of the 2-D slice. Once you have selected the lines choose the `write line` boolean widget. The selection writes the line to the `Write SEP` module. Make sure that the `pick` boolean widget is highlighted, so the module knows to convert the 3-d array into a 2-d array. Additionally finally within the `Write SEP` module choose the dataset name, and then select the `writeit` boolean.

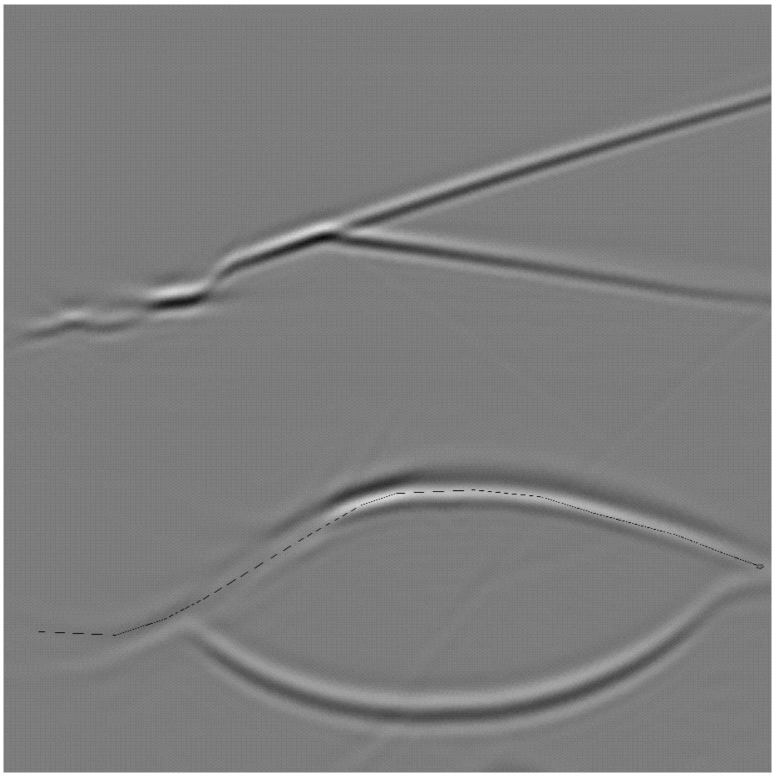

Figure 4: 2-D slice with superimposed picked line. bob1-picking [NR]

View GOCAD

Allows viewing of surfaces defined in GOCAD. Use the file browser widget labeled `project` to select the directory that contains the desired project. Then click on any file within the directory, such as `G.0`, and the list of surfaces contained within the project will appear in the browser labeled `surface`. Select either a single surface, or **all** the surfaces to load. You can either repeat the procedure to load additional surfaces or modify the views by following the **Geometry Viewer Troubleshooting** (Figure 5).

AVS MODULES DEVELOPED BY SEP

AVSGOCAD

The `AVSGOCAD` module operates as the main translator between AVS and GOCAD. It takes as input SEP_1 (which is structure containing the surface name, project name, and associated pick information) and checks for the name of the project and surface. It then uses GOCAD library functions to read the binary form for the selected surface(s). It converts the information into an AVS polytriagonal geometry object. The color of the surface is determined by the `color` browser widget. The module also allows the user to construct an object hierarchy

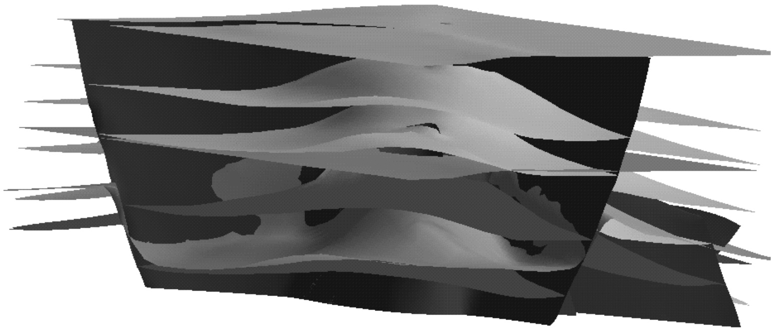

Figure 5: GOCAD defined 3-D Salt Model. bob1-viewgocad [NR]

(parent-child concept), which allows various parts of the model to be grouped together, and given properties as a whole rather than individually.

Browser_cm

`Browser_cm` takes the name of a file as input and then reads some portion of that dataset onto the CM. Which portion of the dataset to read in is controlled by the attached widgets (`n1, n2, n3, n4, j1, j2, j3, j4, o1, o2, o3` and `o4`). All of whose function conform to the SEP conventions. Once the desired portion of the dataset has been selected, the user selects the `browser enter` boolean widget to read the dataset and pass along the resulting array downstream.

Create Line

`Create Line` allows the user to create a line from a series of picks within the `geometry viewer`. The module takes as input the SEP_1 data type (usually from the `Pick Geom` module) from which it extracts the 3-D location of the selected point. When multiple points have been selected it constructs a series of disjoint lines connecting the selected points. In addition it allows the user to: change the color of the line (through the `red, blue`, and `green` widgets); erase a selected point (`delete point` boolean widget); and start another line (`newline`). With each additional selection a 3-D array (3, maximum number of points selected in a line, number of lines) is added too. When the user has finished creating the line(s). The user selects the `writeit` boolean to pass the information downstream.

Gain_cm

`Gain_cm` applies gain to a selected slice/volume that is currently residing on the CM. It has two attached widgets, `gpow and clip` . Both of these widgets operate in the standard manner.

Pick Geom

With `Pick Geom` the user can select any point on any geometry object in the 3-D volume visible in the `geometry viewer`. The module uses the AVS upstream geometry concept to obtain information from the `geometry viewer`. Once the user selects a point in the `geometry viewer` a sphere of size and color defined by the user (the `red, green, blue, and radius` slider widgets) is placed at the selected location. The user has a choice of either selecting the exact point of the mouse or the nearest vertex on the selected geometry object (controlled by the `Choices` widget). Once a point has been selected by the the user the name of the object, the selected point's location, and any primitive data associated with the point is passed downstream through the SEP_1 data type .

Read SEP

Read SEP mimics the standard input Unix command. Through an attached file browser a user selects a standard SEP header file. The validity of the header files is checked and the values for (`n1, n2, n3, n4, d1, d2, d3, d4, o1, o2, o3, o4`, and `esize`) are read in. The module then creates an AVS field with the correct dimensions and calculates an extent array. If the `scale` boolean widget is active it sets the minimum and maximum extents array so that each pixel represents one position in modeling space and that the model is centered around (0,0). If scaling is not chosen the d1,o1,d2,o2,etc parameters are used to construct an extent array that accurately reflects the model.

Slicer_cm

`Slicer_cm` performs the same basic function as the standard AVS module `ortho slicer`. It takes a 2+-D volume that resides on the CM and selects a single 2-D slice. It has two attached typein widgets `HAxis and VAxis` to select which plane to preserve and slider widgets along the other dimensions.

Surface Name

`Surface Name` is used to select the GOCAD project and surface(s) to load. The user selects the desired project using the file browser widget labeled `project`. The module then scans the `.file2gobj` in the project directory for the list of surfaces contained within and outputs the list to the browser widget labeled `surface`. Once the user selects the surface the module passes the name of the project and surface (through the SEP_1 data type) on (usually) to the *AVSGOCAD* module.

TraceGeom

`TraceGeom` displays survey acquisition parameters. It takes as input two AVS fields (one for source location and one receiver location) and outputs a 1-D, 3-vector array. It allows the user to display either source-receiver, midpoint-offset, bin-fold, or bin-offset using the `Display Type` choice widget. The user can reduce the amount of data displayed by modifying the (`N Mid x, N mid Y, O Mid x, O Mid y, D Mid x, D mid Y, N Traces, First Trace, J Mix X, and J Mid Y` widgets). The specified output is then sent downstream, normally to the `scatter arrow` module.

Ultimod

`Ultimod` is based upon Karrenbach's elastic wave propagation modeling program (**?**). It can perform 2-D acoustic modeling in real time. It takes as input a 2-D velocity slice and a wavelet. It allows the user to modify the source location, maximum frequency, number of time slices, slice interval, etc.

Write SEP

`Write SEP` operates similar to the standard output convention in Unix. It uses a file browser widget to select the output path for the header file. If the scaling pipe is connected, it checks to see if scaling has been applied. It then checks to see whether the input data is from a 2-D picking routine (`pick` boolean widget) in which case the data is transformed into (data in the SEP format esize=8) data containing the x and y coordinates. The module obtains the dimensions and extents from the input AVS field, converts them to the SEP conventions (`n1,d1,o1`,etc.), and finally uses auxputch and srite, to write out the dataset. The dataset is written when the user selects `writeit`.

OTHER USEFUL HINTS

Getting Started

- Loading pre-created networks IF you are at SEP, and on SEP's HP730, Spur:
  - at prompt: avs
  - select *Network Editor*
  - select *Read Network*
  - select *demos*

  – select desired network

- Network working environment

  The left portion of the network environment, the Network Control Panel, contains the list of modules currently in the network and for the selected module the attached widgets (see Figure 6). The right hand portion, the Network Construction Window, is the main working environment. The top, right column of the Network Construction Window, the Network Editor Menu, contains general flow controls. To the right is the list of modules, the Module Palette. They are generally sorted into a series of module libraries (`SEP, public, supported, imaging, etc`) and then more specifically categorized by type (`Data Input, Filters, Mappers, and Data Output`).

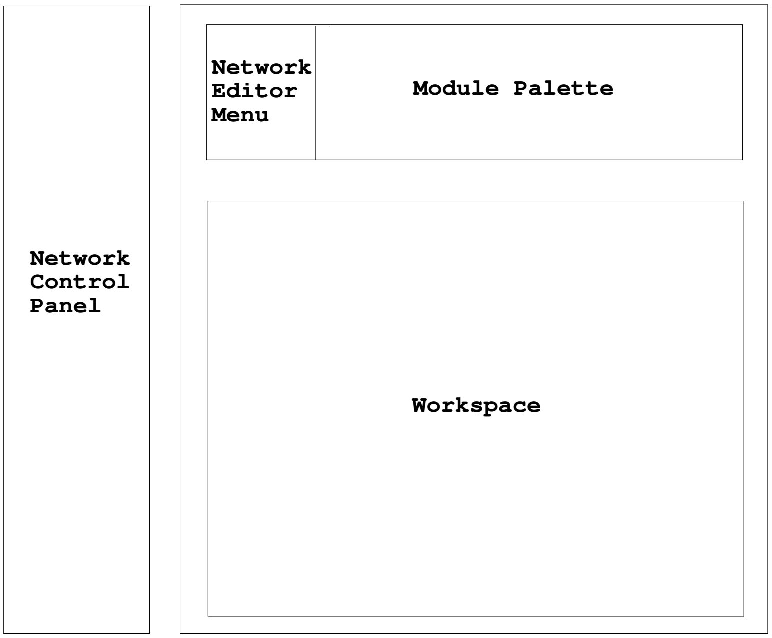

Figure 6: AVS working environment bob1-overview [NR]

  Beneath the module library is the Workspace. It contains the flow that the network will execute. Modules are connected through color coded pipes, corresponding to the type of data that they pass. Modules can be brought into the working area by dragging them down from the module library while depressing the left button. They can be connected by first selecting an input/output port with the middle button and than highlighting the desired connection. To destroy a connection the right button is selected while the mouse is positioned upon either the input or output port of the desired pipe. The user then highlights the desired pipe to remove the connection.

- AVS widgets

  AVS uses several different types of widgets, which are comparable to UNIX parameters, the most common of these are shown in Figure 7.

  – typein: User highlights the widget and types in a number or string.

Figure 7: Example of the several types of widgets available within AVS: `Edit-Load-Save` shows a choice widgets; `projectin` is an example of a file browser widget; `color`, a choice browser; `newgroup` a typein widget; `remove object` a boolean widget; `red` is a slider widget; and `ohor` a radio dial. bob1-widgets [NR]

- radio buttons: User moves the dial to choose the desired float/integer. By clicking on the center of the dial the user can bring up a menu that will allow direct typing in of a desired value.
- choice browser: User selects one of a list of choices provided by the module by highlighting the desired choice and selecting with the left mouse button.
- file browser: User selects a file. The user can either search the directory tree or by using the new directory/new file buttons to go directly to the desired file.
- slider: The user selects the desired float/integer by moving the slider between the minimum and maximum.
- choice: User selects one of several options by selecting a desired settings (choices look like booleans).
- boolean: User has two operating modes to chose from, selected or not selected (1 or 0). When highlighted in selected mode. Change mode by selecting with left mouse button.

Geometry Viewer Troubleshooting

Learning to effectively operate the `geometry viewer` is probably the most confusing aspect of AVS. Rather than going into detail in the operation of the `geometry viewer`, it is probably better to summarize some of the most common encountered problems and their solutions.

- Nothing appears in the `geometry viewer`

  Make sure that the `geometry viewer` is set to `Transform Object`, and the object selected is "top". If another object is selected click in a area of the `geometry viewer` where no objects are present. Once these two conditions have been met, select the `normalize` button. The viewer should now contain all objects that have been created. Switch to `Transform Camera` before you proceed any further.

- An object appears to be incorrectly placed in space

  The most common cause is that at one point `Transform Object` mode was selected and only one geometry object was modified instead of the entire scene. The easiest solution is to recreate the object in question. If the misplaced object does not disappear, follow the procedure under **removing objects** .

- Modifying the view in the `geometry viewer`

  – left button-select an object
  – middle button- changes the location of the camera in relation to scene (rotate)
  – right button- moves the scene with the `geometry viewer` (translate)
  – middle button and shift- zoom in (up with the mouse) and out (down with the mouse)

- Scene is poorly lighted

  As a first approach select `Lights` and than select `Bi-Directional`. If the scene still appears poorly lighted the direction of lights can be modified by first selecting `Transform Light` and than using the standard `geometry viewer` commands to change the direction and location of the lighting. Make sure to change back to `Transform Camera` when finished.

- Unwanted object in the scene

  Select the box next to the name of the currently selected object, located above the miniature view of the scene. A dialog box will then appear containing the names of all the objects in the scene. Select the unwanted object, select `Objects`, and select `Delete Object`.

Generate Colormap Troubleshooting

A problem that is commonly encountered when displaying a slice is that it appears to be a single color, or has an extremely limited resolution. This is most likely caused by an error in the definition of the color table. The `generate colormap` contains two dial widgets, `min and max` variables, which controls what values of the slice will be mapped to a specific color. By changing these values, the resolution of the image can be improved. If the user is unsure of what values to choose for the min and max, the `statistics` module, found under the `Data Output` category, in the `Supported`, subgrouping, can be connected to the field socket leaving the orthogonal slicer.

FUTURE WORK

As with SEPLIB itself, SEP's AVS environment is always expanding and changing. Currently we see two main fronts for improvement, model building and data viewing, both areas that current SEPLIB tools have proved ineffective and/or inefficient in dealing with.

Model Building

With model building we are attempting to bring the entire process into the AVS environment. Currently in AVS we have the ability to convert a GOCAD surface model into an AVS geometry object, to view SEPLIB defined velocity models simultaneously, and to do 2-D acoustic modeling based on the input velocity cube (Clapp et al., 1994). In addition within SEPLIB we have the ability to convert a GOCAD velocity model into a SEPLIB cube. In the future we plan to:

- to be able to modify surfaces inside AVS (by moving vertexes and implementing GOCAD's DSI smoothing);
- to be able to transfer an AVS model back into a GOCAD format;
- to be able to grid a GOCAD model into a SEPLIB and AVS field format (currently only available as a batch program in SEPLIB environment (Berlioux, 1993));
- to be able to do 3-D modeling all within the AVS environment.

With these features, the entire modeling process will be brought into one user friendly environment.

Data Viewing

Our second main front for expansion is due to our recent experience in handling data that consists of tracks (**?**),(Crawley, 1995), and (Fomel and Claerbout, 1995). Current SEP tools proved to be ineffective in dealing with information that could not be described on the standard SEPLIB cube. AVS offers a solution. With AVS, the data space can be composed of an arbitrary number of points, which make up an arbitrary number of tracks, which can be viewed by the *Geometry Viewer* in 3-D as a series of lines, surfaces (for seismic data where each point represents a trace), or points. With the interactive nature of AVS, it is possible to differentiate between the tracks by either color coding each track based on predetermined characteristics or by allowing the user to select specific tracks. In addition the model space can be viewed simultaneously making finding anomalous points and/or tracks a much simpler matter.

REFERENCES

Berlioux, A., 1993, 3-D grid with GOCAD: SEP–**79**, 301–318.

Chemingui, N., Clapp, R. G., and Claerbout, J., 1995, 2-D Phase unwrapping: SEP–**84**, **??**–**??**.

Clapp, R. G., Biondi, B., and Karrenbach, M., 1994, AVS as a 3-D seismic data visualizing platform: SEP–**82**, 97–106.

Crawley, S., 1995, Multigrid nonlinear SeaBeam interpolation: SEP–**84**, **??**–**??**.

Fomel, S., and Claerbout, J., 1995, Searching the Sea of Galilee : The splendors and miseries of iteratively reweighted least squares: SEP–**84**, **??**–**??**.