EXTENDED FEATURES

CHAPTER NINE

Overview

AVS 5.5 is primarily a platform maintenance release; its main purpose is to update AVS to run on major new operating system releases and provide critical bug fixes. For an overview of what is specifically in the AVS 5.5 release, please see the "Release Highlights" in Chapter 1.

AVS5.5 NOTE: The AVS_DEMOS have been revived and retested on all platforms and a number of bugs have been fixed. The BTF renderer (AVS/Voxel) has been unlicensed to help simplify AVS 5 licensing. AVS/Graph has been updated to use Toolmaster 7.1 for its underlying graphics support - most platforms previously used Toolmaster 6.5A.

This chapter also describes the Cool CD and UCD Builder materials provided with AVS 5.3 and other features added in earlier releases of AVS 5, such as AVS/Graph, AVS/Voxel, Japanese Online Help, and the AVS Demos. Fixes for known problems are discussed in the introductory UNIX chapter and the chapters for the specific platforms.

Cool CD and UCD Builder

The Cool CD is an extensive collection of public domain and 3rd party modules and data sets, conference proceedings, and other valuable material to expand your use of AVS 5 products.

Much of the Cool CD comes from the International AVS Centre (IAC) through the contribution of users like yourselves who have developed AVS 5 modules and data sets and offered them to the AVS user community. The IAC, now based at the University of Manchester in England, makes these modules available in source form through its World Wide Web site. The Cool CD delivers them to you for convenience but make sure to visit the new IAC Web site for ongoing contributions and updates at *http://www.iavsc.org*.

Additional 3rd party modules are also on the Cool CD, including modules that help you -

- Read and write VRML (Virtual Reality Markup Language), an emerging 3D modeling language for the World Wide Web

- Read input from FEA and CFD applications to create UCD data structures (UCD Builder)

- Read I-DEAS FEM data (FEMbridge) or write data to a V-LAN based video recorder (VIDEOkit), available for evaluation or purchase through KGT, Inc.

Additional information on other third party AVS5 and AVS/Express modules and applications available through other companies is also included.

The Cool CD also included a snapshot of the corporate website for Advanced Visual Systems; this has become substantially outdated since the CD was last updated. Please visit our online website at at *http://www.avs.com*.

The Cool CD is specially formatted to allow you to browse the contents with your favorite Web browser without having to install anything. Just mount, browse, and choose what you want to use; in some cases, the software can be run off the CD with no installation.

Support

Because the Cool CD contents are primarily from third parties and are not part of the AVS 5 product line, AVS Customer Support will not support them or answer questions about them. The contents of the Cool CD are provided "AS IS", with no warranty or support of any kind. We are making them available to you as a service to make it easier for you to explore and experiment with the wide world of public domain AVS 5 modules available.

If you do have problems, you should contact the appropriate third party for help or suggestions. For modules from the International AVS Centre, start by contacting the IAC directly or the original module contributor if they have provided contact information.

For most of the third Party modules (FEMbridge, VIDEOkit, VRML reader and writer), follow instructions on the Cool CD to contact KGT, Inc. for further information.

The UCD Builder product was developed by Scientific Visualization Associates Inc. Working with Advanced Visual Systems, the company is making its product available to you in binary form for free as a contribution to the AVS 5 community. This is an unsupported product and Advanced Visual Systems assumes no responsibility for it; if you do have problems or questions about it, see the related Web page for instructions.

AVS 5.5 NOTE: The UCD Builder has **NOT** been updated since AVS 5.3; however, modules should be upward compatible with the newer OS versions supported by AVS 5.4 and AVS 5.5.

AVS/Graph

AVS 5.5 NOTE: The AVS/Graph module has been updated to the latest version of Toolmaster 7.1.

AVS/Graph is a graphing tool which allows you to plot AVS field data in a variety of ways (curve, scatter, bar, area, polar, pie, images), control the appearance of the graph (titles, axes, logarithmic scales, legends, tick marks) and generate hardcopy output in a variety of formats (PostScript EPS, PostScript Color EPS, CGM Binary, CGM Clear Text, etc.) AVS/Graph consists of the Data Output module named **AVS/Graph** with its own extensive user interface, a set of demonstration scripts, and a set of example networks.

The **AVS/Graph** module is a part of the Supported AVS Module Library. The module is a synchronous coroutine module (i.e., it runs as an independent process that is triggered when one of its inputs changes).

AVS/Graph is implemented using the Advanced Visual Systems product Toolmaster-agX. The source to the module is included with the release. If you also purchase Toolmaster-agX, you can use the module source as a guide to extending AVS/Graph functionality to perform additional tasks such as contouring, data interpolations and smoothing, 3D graphs, and note and arrow annotations.

For complete AVS/Graph documentation, see the *AVS/Graph User's Guide* included in the standard doc set.

Japanese Online Help

This release contains a complete set of AVS 5 online help in Japanese. All of the material in *runtime/help* has been translated, including the module man pages and the help files for the various subsystems. In addition, users can create their own Japanese language online help files to support their applications.

The k14 and a14 X fonts must be present on your system.

To install Japanese Online help:

1. When the *install.avs* script presents its list of installable products, select AVS_JHELP. This one help directory supports all platforms and takes approximately 3 megabytes of disk space.

The installation creates an <*install-dir*>*/avs_jhelp* directory. This directory can be anyway in the file system hierarchy, both within the *avs* installation directory or outside it.

2. With the AVS_PATH environment variable set to point to the AVS directory, execute the script <*install-dir*>*/avs_jhelp/test/SETUP*. If unset, AVS_PATH defaults to */usr/avs*.

This script adds the correct fonts to the *runtime/avs.Xdefaults* file and sets the AVS_HELP_PATH environment variable to point to the Japanese help files.

3. You can check that the installation went correctly by executing the <install-dir>/avs_jhelp/test/RUNME script.

For information on creating your own Japanese online help files, see:

- *test/kanji.eucf*: Man page that describes how to make a module with a Japanese help file. (This file is in Japanese).
- *test/kanji.scr*: AVS script for making a module with a Japanese help file.
- *test/kanjiall.txt*: The table of EUC codes.

AVS/Voxel

AVS/Voxel is a Back-to-Front (BTF) direct volume renderer consisting of a set of modules provided in the Unsupported library for customers to use and experiment with. **NOTE:** The BTF renderer is not provided on Compaq Tru64 UNIX.

The BTF renderer renders volumes one slice at a time, beginning from the back of the volume, and composites each new slice as it renders the volume. It is based on a high performance algorithm that achieves its speed by exploiting memory coherence and by optimizing cases involving transparency. In addition, BTF has support for a binary occupancy volume that increases rendering performance for cases that involve clipping, region growing, and segmentation. This volume renderer is extremely well suited for data sets where much of the data in the volume will be transparent (e.g. seismic interpretation and medical imaging).

The BTF renderer is a set of modules that are currently placed in the Unsupported library. It consists of the following modules:

btf shade

Performs back-to-front (BTF) volume rendering. This module takes a volume, which can be visualized as a block of cubic "voxels" (volume elements), and generates a 2D image using a back-to-front (btf) direct volume rendering technique.

btf anim

Performs back-to-front (BTF) volume rendering like **btf shade** except that it allows the user to select a region of the volume and show intermediate rendering results, layer by layer to more clearly see the internal structure. Provides parameters to select starting and ending layers, number of steps between views, and run control.

btf bitvol

Creates a bit volume for use with **btf volume** renderer. This module generates a bit volume from a 3D scalar byte or 16-bit integer field using the opacity channel from a colormap. The rendering speed of the **btf shade** and **btf anim** modules can be increased substantially by using the **btf bitvol** module. Performance can be increased by a factor of 2-4 depending on the occupancy level of the volume, i.e. the number of non-zero voxels in the volume.

norm8 encode

Computes gradient vectors for 3D data sets and encodes them into bytes. The **norm8 encode** module computes the gradient vector at each point in a 3D field of data. The gradient vector can be used as a "pseudo surface normal" at each point. This vector is then encoded into a byte and packed into a 16-bit integer value along with the original input value.

norm8 table

Generates a lookup table for an encoded gradient. The **norm8 table** module accepts an optional transformation matrix as input. It builds a 256-entry intensity lookup table using the ambient and diffuse coefficients and the light source direction for all possible values of the encoded normal information (gradient vector). This table is then used during the shading process by the **btf shade** and **btf anim** modules to interpret the gradient information that was encoded into a byte by the **norm8 encode** module.

Using AVS/Voxel

On platforms that support AVS/Voxel, the BTF modules are located in the Unsupported library. There are several demonstration scripts that can be found in the *$AVS_PATH/demo/man_scripts* directory, all named with titles beginning with "BTF volume rendering". These are found by selecting the **Help** button and then selecting **Help Demos** to bring up the Script Controller which presents a list of demonstration scripts.

The BTF modules no longer require any additional licensing from Advanced Visual Systems. You may see a "AVS/Voxel" entry in your AVS5 license but this is no longer needed for AVS 5.5. You may wish to retain this feature line for older versions of AVS 5 still in use at your site.

Documentation

The man pages for the BTF modules are available online in the *$AVS_PATH/runtime/help/modules* directory along with the other AVS 5 modules. Press the module's dimple to get the Module Editor panel and then press "Show Documentation".

Demos

This release provides an optional package of AVS Demos that can be loaded from the CD after AVS has been installed. These demos can be used to see additional ways that AVS can be used and to obtain new demonstration datasets to augment those found in *$AVS_PATH/data*. Most of these datasets and demonstrations have been contributed by AVS users for the benefit of the AVS user community. While our users have permission to use them for demonstration purposes, they are not necessarily public domain and should not be used in products outside of AVS without express written permission from Advanced Visual Systems Inc.

AVS5.5 NOTE: These demos have been rebuilt and retested under all AVS 5.5 platforms and problems have been addressed.

Installing the Demos

You will require approximately 25 Megabytes to unload the Demos from the CD and another 5 to 10 Megabytes to build them for a total of 50 Megabytes. The demos are NOT precompiled but must be built in order to work.

1. Install AVS if you haven't already done so. The Demos will expect to reference a standard AVS product in $AVS_PATH (*/usr/avs* by default).

2. Install the **DEMOS** product from the AVS CD using the *install.avs* script used for installing the standard product. They will be installed into a directory called *avs_demos*.

3. If you are installing the demos from a remote login window you must make sure that your DISPLAY environment variable is set to an open display, e.g. "setenv DISPLAY unix:0.0". The installation procedure will attempt to run *avs* to run some data conversion scripts and this will fail if there is no DISPLAY set. **NOTE:** If the remote system is an SGI and the system you are on is not, you need to set the **__SGI_NO_REMOTE_GL** environment variable to avoid problems running *avs* remotely.

4. Change directory (*cd*) to the *avs_demos* directory and run the *install_demo* script found there to install a reference (link) from the *$AVS_PATH/demosuite* area to the place where the Demos are located. The *install_demo* script will do the following:

    - Move aside the existing *$AVS_PATH/demosuite/demo_menu* file to *demo_menu.orig* and install a new version of the *demo_menu* file with the Demos scripts appended as an "Extended" menu.

    - Create a link from *$AVS_PATH/demosuite/DEMOS* to the *avs_demos* directory. This link is required for making the Demos files since some of them expect to find *DEMOS* in *$AVS_PATH/demosuite*.

    - Offer to "make" the Demos. You should reply "yes" to immediately make the modules and local data files that make up the Demos.

Once the Demos have been "made", the *install_demo* script can be used to have multiple systems point to the same demo area. Just re-run *install_demo* and answer "no" when asked if you want to "make" the Demos again.

NOTE: Many of the demos need to be "made" differently for different platforms. For example, an SGI and a SUN can *not* share the same area. The file *DEMOS/PLATFORM* describes what platform the Demos are currently built for.

Uninstalling the Demos

Change directory (*cd*) to the *avs_demos* directory and run the script *uninstall_demo* which will do the following:

- Remove the Demos version of *$AVS_PATH/demosuite/demo_menu* and move the original *demo_menu.orig* back into place.

- Remove the link from *$AVS_PATH/demosuite/DEMOS* to the *avs_demos* directory.

- Offer to remove the entire current directory (assumed to be the *avs_demos* directory).

Running the Demos

Once installed, the Demos are accessible as a top level menu in the standard AVS demosuite. To run them, run *avs*, select **AVS Applications** and then select **AVS Demo**. The Demos should appear as a new menu called **Extended** and are organized by different market segments. Each demo consists of one or more demonstration scripts.

The following demos are part of the AVS Demos package:

Medical Imaging

Helen's Neck

MRI (Magnetic Resonance Imaging) study of Helen's neck. The data set consists of eleven slices of 256x256 images taken sagittally. On slice 5, you can clearly see her brain, spinal column, and the vertebrae surrounding the spinal cord. This is a real life example based on a study performed on a friend of Advanced Visual Systems Inc.

Brain

Stereotactic Radiosurgery: This demonstration shows the planned treatment for a real patient. The patient was a 2 year old girl with a brain tumor located dangerously close to her brain stem and inoperable by conventional surgical methods. Provided by Dr. Hanne Kooy at the Dana Farber Cancer Institute.

Geographic Information Systems (GIS)

World

This demo shows outlines for the following global features: continental outlines; islands; lakes; rivers; country borders; United States borders.

FlightPath

The **flight path** module takes a scatter field position list and animates the camera path along the trajectory. The modules is derived from *$AVS_PATH/examples/camera.c*. The scatter path field is in the form of "field 1D 3-space irregular float" where the coordinate information in the field specifies the path. The data values are ignored. It moves %top, not the camera, so multiple camera views can be set up to watch the path. The **file_descriptor** or **read_field** modules can be used to get the path from an external file. The **animated integer** module can be used to control time, for a moving sequence.

Terrain

Terrain Reconstruction from Digital Elevation data. This demo shows an elevation map and a landsat image (full color) of Orange County, with a perspective view. The image is rendered with a synthetic light source direction. The texture processing is performed by the AVS software renderer, or hardware adapters that support this operation.

Seal Tracks

Southern Elephant Seal movement and dive data in the South Atlantic. Provided by the SEA Mammal Research Unit, British Antartic Survey, NERC, Cambridge, UK. Ref: Dr. Ollie Cox and team.

Meteorology

Weather

Weather Satellite Images: If you have ever watched TV weather forecasts, you probably have seen the short "movies" of the clouds moving over the earth, especially when there is a hurricane off the coast. A new satellite photo is available every hour via the Internet. We have put together a series of 12 such photos from daytime on July 21, 1992. The images register infrared light and therefore indicate temperature. These images were taken by the GOES-7 satellite and retrieved from a machine at University of Illinois at Urbana-Champaign, Illinois.

Oil and Gas

Velocity 3D

This is a demo of a synthetic 3D velocity profile field, with a "plumb line" interactive fence diagram picking module.

Intera

This is a series of representative models of oil and gas exploration projects produced by InteraView, a product of Intera Information Technologies Limited, Petroleum Production Division that uses AVS to deliver end user solutions.

Computational Fluid Dynamics (CFD)

Phoenics

PHOENICS fluid flow simulation, courtesy of CHAM (UK). The PHOENICS CFD code is interfaced to AVS for both pre-processing and post-processing. Includes Hot Electric Box, Smoke Spread in House (two versions), Cooling Tower and Turbine Blade.

Vortex

CFD RIBBONS - generate ribbon representation for streamlines. The **ribbons** module generates a set of geometric ribbons by taking the polyline output of the **streamlines** module and replacing them with finite width colored and textured polytriangle ribbons. The data set shows a 3D vortex field, simulated by researchers at NCSA.

Mechanical

Automobile

Max vonMises Stress on a Mercedes: In this demonstration, NASTRAN was used to compute the Max vonMises stresses for a Mercedes-Benz automobile. This finite element dataset contains 15,843 cells. Provided by Keith Redner, Scientific Visualization Associates.

Crankshaft

Displacement on a Crankshaft: In these demonstrations, stress is applied to a crankshaft. This finite element dataset contains 609 cells. Provided courtesy of Pafec, Ltd.

Chemistry

Fast Animate

Water over Clay: Dr. Keith Refson, University of Oxford, UK uses a molecular dynamics program to simulate the interaction of water and cat ions with layers of clay. The simulation requires significant processing per time step, so it is run offline with the molecules' positions at each time step saved into files. This demonstration shows Dr. Refson's AVS animation module.

BGF Ribbon

The code in this directory is for the **PROTEIN** module. This module reads a Biograf file and creates various displays of the molecule.

Statistics

City Scape

City Scape Data Analysis Display: Network Line Statistics are displayed using a 3D Bar-Chart technique, called "City Scape". The height and color of each cell shows two variables, while row and column averages are computed dynamically and shown as projected side panels. Provided by Advanced Visual Systems, Inc.

Jigsaw

The **jigsaw** module is a tool to show irregular boundary regions, defined in 2D, like map boundaries, and apply data to loft them into 3D with data defined height, and to apply color to indicate a second variable. Provided by Advanced Visual Systems, Inc.

Magnetics

Vector Fields

Vector Fields Magnetic Flux Modeling: This structure represents the result of an electro-magnetic flux analysis using "Tosca" from Vector Fields Ltd. The magnet core is shown as a ring structure, with air cells surrounding, with interpretive vector display modules. The model is axial-symmetric, so a geometry duplication method is used to construct the complete model for viewing. The Vector Fields interface, this network, and the visualization project is by Janet Haswell, Rutherford Appleton Labs, Oxfordshire, UK.

Texture

Texture Sampler

Sample texture image files: A set of image files containing textures such as rock, sand, and clouds. Contributed by Evans and Sutherland.