

Using GeoVRML for 3D Oceanographic Data Visualizations

Michael P. McCann
Monterey Bay Aquarium Research
Institute
1700 Sandholdt Road
Moss Landing, CA 95039-9644
831-775-1769
mccann@mbari.org

Abstract

The Monterey Bay Aquarium Research Institute operates two Remotely Operated Vehicles that routinely explore the depths of the ocean at sites around the north eastern Pacific Ocean. Our growing archive of observations – which include high frequency navigation and environmental records, video frame grabs, sampling events, annotations of the video record, and multi-beam sonar bathymetric maps of the dive areas – represents a rich resource for scientific studies. Recent advances in commodity 3D computer graphics hardware and tools created by the GeoVRML working group have enabled efficient and easy-to-use visualizations of these data via a standard web browser. Visualizations of MBARI's 3,000-plus ROV dives are now available via our internal web-based Expedition database. The GeoVRML implementation allows for geospatially accurate representation of any ROV dive made anywhere on the globe. The system has been in place at MBARI for over two years and use of it is growing. According to people who have used it the system is not just *cool* but also *useful*.

Keywords

VRML, GeoVRML, Web3D, Oceanography, ROV, underwater, video, scientific visualization.

1. Introduction

The Monterey Bay Aquarium Research Institute operates two Remotely Operated Vehicles in the deep waters of Monterey Bay and the greater eastern Pacific Ocean. MBARI's mission is to develop state-of-the-art systems for conducting scientific research in the deep waters of the ocean. The ROV's act primarily as the eyes and hands of the science teams who operate the vehicle from the support ships; the machines are the observational and operational platforms that often explore areas that have never been seen. Over its fifteen-year history MBARI has routinely logged the variety of data that come from these underwater activities. These data include ship and ROV navigation

information, bathymetric maps of the dive areas, collections of geologic and biologic specimens, video frame grabs, and annotations of the recorded video. It is a rich collection of data that presents interesting challenges for visualization.

The databases that comprise these observations are described in [1]. An architecture for creating standard VRML (*i.e.* not GeoVRML) renderings of MBARI's dives in the area of the Monterey Submarine canyon is also described in [1]. This system was of limited usefulness for several reasons: 1) only dives in the Monterey region could be visualized, 2) ability to bring in other terrain data sets was limited, 3) sample collections and video annotation data could not be visualized, and 4) the original system could view only one dive at a time within a world.

Creating useful and interesting visualizations of the variety of data collected by MBARI presents a challenge whose solution may benefit the greater oceanographic community. The assembly of our various data sets into a common, compelling, efficient, and easy-to-use visualization system has been the goal of this project during its four year history.

This paper describes a system that has been developed using GeoVRML 1.1 [2,3] to provide a general solution for rendering ROV dive data from anywhere on the globe. The system is now in place as part of MBARI's standard archival data management system and several scientists have found it not just cool to visualize their data, but actually useful for integrating disparate data types and visualizing spatial relationships. Section 2 describes the enhancements made to the original system in order to use GeoVRML. Section 3 describes two proposed new GeoVRML nodes. Section 4 describes the implementation and operation on MBARI's internal network, section 5 discusses an evaluation of the system, and section 6 discusses Web3D technologies and outlines some directions for improvement.

2. Converting to GeoVRML

The object-oriented Perl module (EXPD.pm) that was developed for the original system was reused for the new GeoVRML-based system. The EXPD.pm module provides data for each dive and the GeoVRML Post Dive VRML creator (geopdvc.pl) produces the VRML output (.wrl files). Geopdvc.pl was easily adapted from the original standard VRML pdvc.pl script; the main changes being using the GeoVRML nodes – mainly GeoCoordinate for specifying IndexedLineSet geometry and GeoLocation for specifying the geographic locations of objects placed in the world. Many of these changes were encapsulated in the external prototypes GeoPowerDivePlaybackProto and GeoIconImageProto, which are described in section 4.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Web3D Symposium '04.

3. Enhancements TO GeoVRML 1.1

The GeoVRML 1.1 nodes [2,3] provide a good foundation for building a system for visualizing ROV dive data. However there are two features required by 3D Replay that suggested the creation of two new GeoVRML nodes. For geometry composed of GeoCoordinate nodes, such as an IndexedLineSet of an ROV dive track, we need to have the ability to translate the track vertically relative to the terrain. This is because measurement errors in ROV navigation terrain data often put the ROV location under the terrain. With the new GeoTransform node a slider in the 3D Replay user interface now allows for vertical displacement of the ROV dive track relative to the terrain.

Therefore as part of implementing the ROV 3D Replay system two new GeoVRML nodes have been developed: the GeoTransform node and the GeoProximitySensor node. The GeoProximitySensor node provides a geoCoord_changed eventOut which is routed to the heads up display of the 3D Replay system; this always shows the viewpoint's geographic location. These two new nodes are now included as part of the Windows installer available on the GeoVRML web site [3].

3.1 GeoTransform

A GeoTransform node provides the exposedFields *translation* and *rotation* that allow for the translation and rotation of geometry built using GeoCoordinate nodes within the local VRML coordinate system. Like the VRML97 Transform node it is a grouping node that can be used to define a coordinate system. The X-Z plane of a GeoTransform coordinate system is tangent to the ellipsoid at the *geoCenter* location.

The exposedField *geoCenter* specifies the location at which the local VRML coordinate system is centered. It is specified in the spatial reference frame specified by *geoSystem*. Below are the field definitions for the GeoVRML 1.1 GeoTransform node:

```

EXTERNPROTO GeoTransform [
  exposedField SFString  geoCenter # ""
  exposedField SFVec3f   translation # 0 0 0
  exposedField SFRotation rotation # 0 0 1 0
  field SFNode           geoOrigin # NULL
  field MFString         geoSystem # [ "GD", "WE" ]
]
field MFNode children # []
][ "urn:web3d:geovrml:1.1/protos/GeoTransform.wrl"

"file:///C:/Program%20Files/GeoVRML/1.1/protos/GeoTransform.wrl"

"http://www.geovrml.org/1.1/protos/GeoTransform.wrl"
]

```

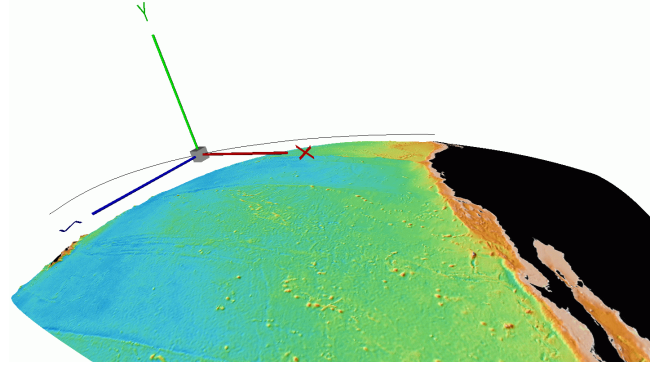


Figure 1. Example of GeoTransform where the Hawaii to California GeoCoordinate specified ship track is translated normal to the ellipsoid at the point of the axis origin (the geoCenter).

3.2 GeoProximitySensor

The GeoProximitySensor is another new GeoVRML node that performs just like a standard VRML ProximitySensor with these additional features: the viewpoint location is available as an eventOut in units of the geoSystem and the boundaries of the sensor's box are defined relative to the ellipsoid. Below are the field definitions for the GeoVRML 1.1 GeoProximitySensor node:

```

EXTERNPROTO GeoProximitySensor [
  exposedField SFString  geoCenter # ""
  exposedField SFVec3f   size # 0 0 0
  field SFNode           geoOrigin # NULL
  field MFString         geoSystem # [ "GD", "WE" ]
  eventOut SFTime       enterTime
  eventOut SFTime       exitTime
  eventOut SFString      geoCoord_changed
  eventOut SFBool       isActive
  eventOut SFRotation    orientation_changed
  eventOut SFVec3f      position_changed
]
][ "urn:web3d:geovrml:1.1/protos/GeoProximitySensor.wrl"

"file:///C:/Program%20Files/GeoVRML/1.1/protos/GeoProximitySensor.wrl"

"http://www.geovrml.org/1.1/protos/GeoProximitySensor.wrl" ]

```

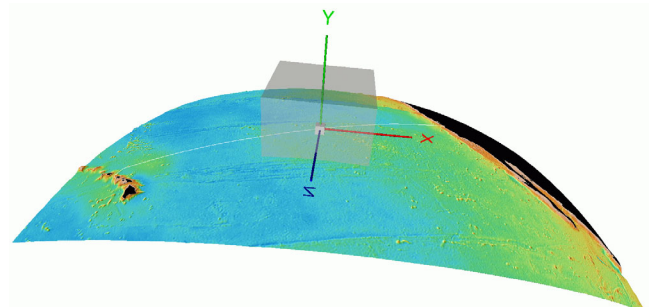


Figure 2. Example of a GeoProximitySensor where the transparent box delineates the sensor's boundary. The upper and lower box faces are aligned tangent to the ellipsoid at the geoCenter location.

4. Implementation and operation

From a user's perspective, all of MBARI's ROV dives are available as virtual 3D replays via our Expedition database. A user simply searches for a dive (or dives) of interest and clicks on links to bring up an interactive 3D world of the dive data. The user interface for the 3D world is shown in Figure 3. The system has been implemented using Perl scripts for generation of the GeoVRML files, a relational database for storage of the references to these files and their relationships to Expedition metadata, and server-side Perl asp scripts for dynamic generation of web pages. Some specific implementation details concerning level of detail management and the abstraction of behavior in external prototypes are described below.

4.1 Terrain generation

4.1.1 MB-System macro *mbm_grd2geovrml*

Oceanographers map the seafloor with multi-beam swath sonar systems. The data from these systems are processed, gridded and stored as geographically registered GMT grid files [4]. From these bathymetric data we produce geographically registered artificially illuminated image files which are often used in GIS systems for 2 Dimensional data visualization. To view the sonar measured bathymetric data in 3D we combine the bathymetric data with a co-registered image, which may be the artificially illuminated image, a slope-magnitude shaded image, or a shaded image of the acoustic backscatter intensity.

Programs from the open source *tsmApi* [5] and *GMT* [4] toolkits were used to build GeoVRML format terrain tiles for use in the 3D Replay system. The programs were consolidated into an easy to use macro (*mbm_grd2geovrml*) which is now distributed as part of the MB-System package [6]. This macro takes as input a

GMT geographic-coordinate bathymetry grid file and produces a GeoVRML formatted terrain tile set. The bathymetry (geometry) is paired with imagery (texture) to produce artificially illuminated terrain or slope shaded terrain. Alternatively, a co-registered image may be provided as a substitute to the latter imagery which is generated from the bathymetric data. This may be used to, for instance, drape an acoustic backscatter intensity image over the bathymetry.

Before conducting ROV dive operations in an area MBARI acquires high-resolution bathymetric survey data for that area. We currently possess over 80 such surveys for sites throughout the northeast Pacific Ocean. *Mbm_grd2geovrml* has been used to create 3 sets of GeoVRML tile sets for each of these areas (same geometry, different textures): an artificially illuminated texture, a sloped shaded texture, and a side scan image texture.

In constructing the GeoVRML terrain tile sets two factors must be balanced in order to achieve acceptable performance with these high-resolution bathymetric data sets. First, the size of the geometry and texture files must be small enough to be transferred and quickly rendered on the user's desktop. Second, the complexity (*i.e.* number of polygons and textures) must be small enough to achieve better than 10 frames per second animation performance. Tests were conducted using commodity 3D Graphics hardware in the fall of 2001. In balancing these two factors we opted for a tile size of 16 by 16 polygons and textures of 128 by 128 pixels. This results in geometry file sizes of about 20 KB (which may be compressed to about 5 KB) and texture sizes of 5 KB. Two years ago these choices provided a good balance between responsiveness in network transmissions, graphics hardware performance, and the number of tiles that must be loaded to accurately represent the terrain for an area of interest.

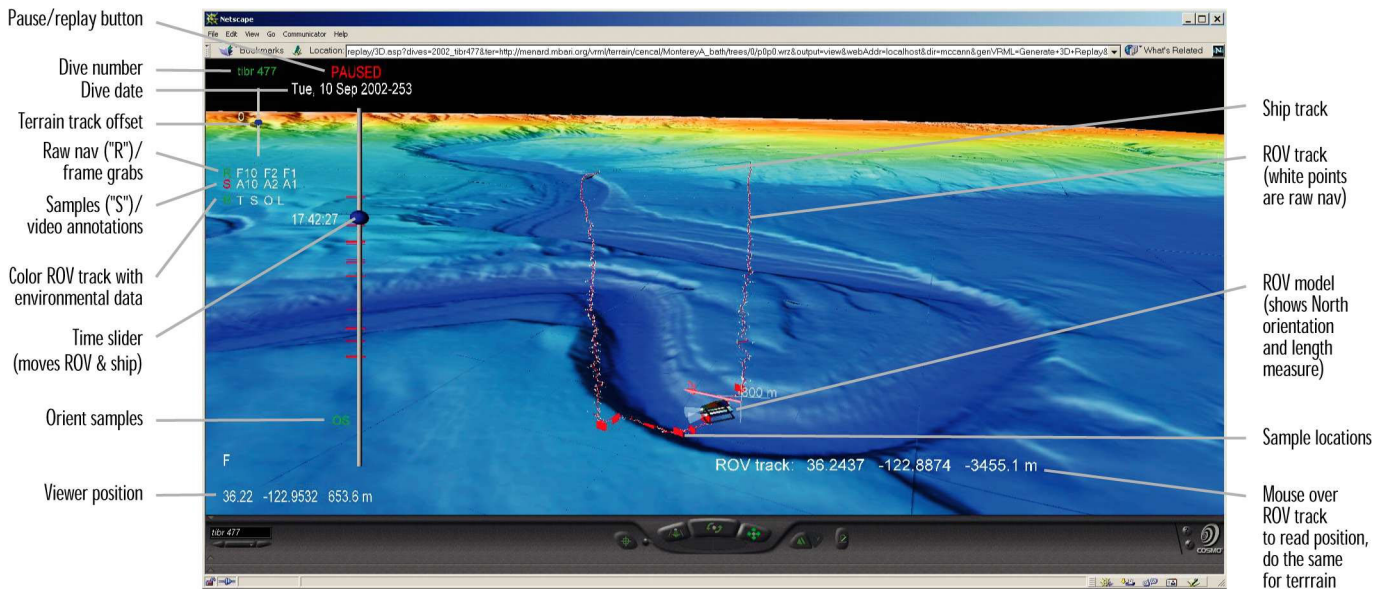


Figure 3. A screen shot from the 3D ROV dive replay system shows sample locations from a dive on Shepard Meander of the Monterey submarine canyon. The user interface is a heads up display which includes the features that are described in the figure.

4.1.2 Level of detail management

The standard VRML level of detail node (LOD) is inadequate for multi-resolution terrain visualization. It chooses to load higher resolution child nodes based only on a distance measurement. It does not consider visibility within the view frustum or the speed of the viewpoint. Rather than spending time developing a more efficient LOD algorithm for the 3D Replay system (see [7]) we opted for a user-controllable terrain LOD management. For a typical bathymetric terrain map there will be just a few square kilometers of area in which a ROV may explore. Yet the terrain tile set may encompass thousands of square kilometers. The user is mainly interested in high-resolution terrain data only in the few square kilometers of the dive area. The adjacent area is important to provide context for the visualization, but only the highest resolution data is needed in the dive area.

The make_geovrml utility of the tsmApi toolbox was modified to only load the child nodes of a GeoLOD node when a GeoTouchSensor on the parent node is touched. With terrain tile sets produced by this utility the user may select the areas for which high resolution terrain is desired. In the current implementation the Monterey Bay Simrad EM300 GeoVRML tile set covers 40 thousand square kilometers and has nine levels of detail whose highest resolution tile has a resolution of 40 meters in latitude. The size of the original GMT bathymetry grid file is 60 MB. The GeoVRML tile set is 76 MB and is comprised of 39,860 files. To view the highest resolution tiles the terrain must be clicked on three times (two levels are loaded with each click). The process takes less than one minute to load and render on a 1 GHz laptop with a 100 Mbps intranet connection. The resulting world animates at better than 12 fps with an Nvidia GeForce2 Go graphics processor unit.

4.2 GeoPowerDivePlaybackProto

Most of the work for rendering the dive data is accomplished by the GeoPowerDivePlaybackProto which is basically the same as the PowerDivePlaybackProto that is described in [1]. The geopdvc.pl script generates GeoVRML formatted files for each ROV dive: a navigation file, a raw navigation file, sample collection events, and files for video frame grabs and video annotations. For the video frame grabs and annotations there are files for every tenth, every other, and every single object. These are made available so that the user may preferentially load images and annotations appropriately for each dive. Some dives have just a few video frame grabs while others have hundreds. A few hundred frame grabs loaded into a world can significantly slow performance, both in waiting for them to load and in rendering them in the world. The files that place geo-located objects in world make use of the GeoIconImageProto, which is described in the next section.

4.3 GeoIconImageProto

The GeoIconImageProto is used to display images inside a GeoVRML world. It has been developed for use in visualizing images captured on remotely operated submersible dives, but it is general enough to be used for any geo-located image or text. It locates an IndexedFaceSet on which an image is placed as a texture. A TouchSensor on this bit of geometry may point to a URL that will be loaded in another window. This is useful, for instance to bring up a database query for an object that has been

placed in the world. The 3D Replay system uses this feature to bring up details for a particular Sample or video frame grab.

GeoIconImage handles scaling of the IndexedFaceSet so that the image or icon is always visible. When the viewpoint is far away the image texture is replaced by an icon. The transition distance for this switch is available for changing via an eventIn. The icon or image may also be scaled via an eventIn. If the orientation for the image is provided then another eventIn flag may be used to set the orientation of the IndexedFaceSets, otherwise the images are contained in a Billboard with an AxisOfRotation of (0,0,0).

The 3D Replay system makes use of these eventIns by presenting slider controls in the heads up display for adjusting icon transition distance, image scaling and orientation. (Note that this feature was added after Figure 3 was produced.)

Navigating a 3D world with a mouse that moves in 2D is always a challenge. For viewing 3D Dive Replays, the CosmoPlayer VRML browser is preferred because of the utility of its Examine functions. The user can position the ROV to an area of interest along the dive track and “Seek” on it. This action resets the center of rotation to the seek point and moves the viewpoint closer. He or she can then rotate about this point to visualize nearby features. To facilitate this mode of operation, the default behavior of the Shift key should be changed. In CosmoPlayer’s preferences, Keyboard tab, select the “Turbo Mode and Continuous Seek” option. With this setting the Shift key now modifies the action of the mouse. The user holds the Shift key down and clicks once to Seek to a point. To rotate about this point the user holds the Shift key down, clicks and holds the mouse button down while moving the mouse. With the Shift key held down, the Alt and Ctrl keys can be used to switch to the Pan and Zoom controls. The Shift key also disables TouchSensors within the world. A little experimenting and practice will give the user confidence in navigating the 3D world to provide whatever view of the data is desired. If the user gets lost in the world then one of the set viewpoints can be selected to reset the view.

5. Evaluation

In the 18 months that the GeoVRML implementation of the 3D Replay system has been generating virtual dive replays there have been over 200 uses of the system to view one or more dives. Most of the uses have been by the Information Applications Group at MBARI to assure the quality of the data we archive. Several scientists have requested visualization of dive data that could not be accomplished with standard 2D visualization tools such as commercial GIS applications. These requests have been simple to satisfy as the GeoVRML files for each dive are automatically generated following the collection of each day’s worth of data. As the 3D navigation interface is difficult to master it is often necessary for an experienced user to operate the controls while the interested user directs the interaction.

Interest in using the system was so high in the summer of 2002 that we were asked to implement its operation on the R/V *Western Flyer’s* computer system. Its use aboard the ship during dive operations produced mixed results. The 3D Replay system was designed with post-dive data visualization in mind. Aboard ship during ROV dive expeditions the science teams are so busy conducting dive operations, dealing with sample collections, and preparing for the next day’s activities that very little time and energy is available for 3D visualization of the dive data.

However, when the effort is expended, validation of successful data archiving and co-registration of disparate data sets have proven valuable in operations aboard the *Western Flyer*.

The most extensive uses of the 3D Replay system have been for marine geology applications, where spatial correlations between geologic features as seen in the multi-beam sonar data and the images collected by the ROV's camera are important to relate.

5.1 Cleft segment of the Juan de Fuca Ridge

In the summers of 2000 and 2002 the ROV *Tiburón* conducted transverse transects of the Cleft segment of the Juan de Fuca Ridge off the coast of Washington – an area where new oceanic crust is created by submarine volcanism. In 2003 a DVD video was produced using the 3D Replay system (Figure 4). The video was produced in real-time by direct digital video capture of the computer's video signal. The video has been shown at scientific conferences and has been successful in quickly communicating the observations of off-axis volcanism. The chief scientist of these expeditions reported the following experience with the software:

"The 3D Replay system has proven to be a utilitarian tool for integrating visual and laboratory observations as well as a remarkable presentation vehicle for observational data. Within marine geology, much of our seafloor investigations are grounded in interpretations of visual phenomenon. We see varieties of volcanic forms and structures that appear in regular configurations. Laboratory investigations suggest the underlying linkages among the rocks and we infer the fundamental controls.

However, integrating the laboratory, field and intuitive connections has been difficult because of the intractable nature of the observational database. Traditional publication and presentation forms (papers and posters) are limited to small snapshots bogged down in technical terminology and not immediately understood without supplemental explanation. Using the 3D Replay software created a dynamic presentation format that effectively exploits the visual nature of field work and is immediately graspable by students and collaborators. We plan to use this DVD as a supplement to a more traditional manuscript and we expect excellent results."

5.2 Submarine terraces off of Hawaii

Also in 2003, a post doctoral scientist used the 3D Replay system to interactively investigate data from dives we made two years earlier off the northwest coast of Hawaii (Figure 5). He commented:

"The 3D visualization software has proved to be very useful for my research on submerged reefs around Hawaii. With this software, I am able to very quickly get a sense of; (1) the spatial context of each sample, (2) whether the samples are in place (critical for reef studies), while (3) seamlessly and simultaneously browsing the MBARI sample database. I was not involved in the original Hawaii dives, so for me, this software provides a quick and easy introduction to each dive and the spatial context of each sample, allowing me to better focus my interest when going back to the original dive tapes".

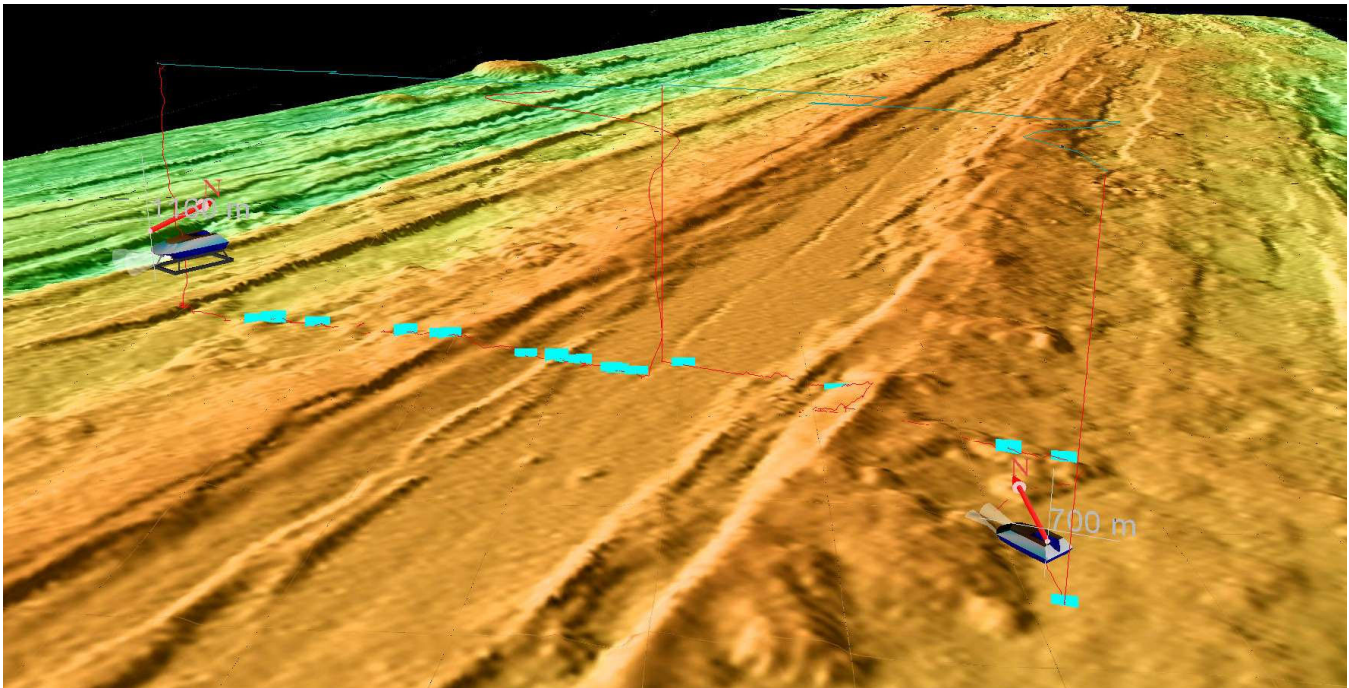


Figure 4. Visualization of two ROV dives made on the cleft segment of the Juan de Fuca mid ocean ridge where new oceanic crust is being formed along the central axis of the ridge. The blue rectangles are locations of selected video frame grabs. The rectangle icons are hyperlinked to the frame grabs allowing for easy relation to the surrounding geomorphology.

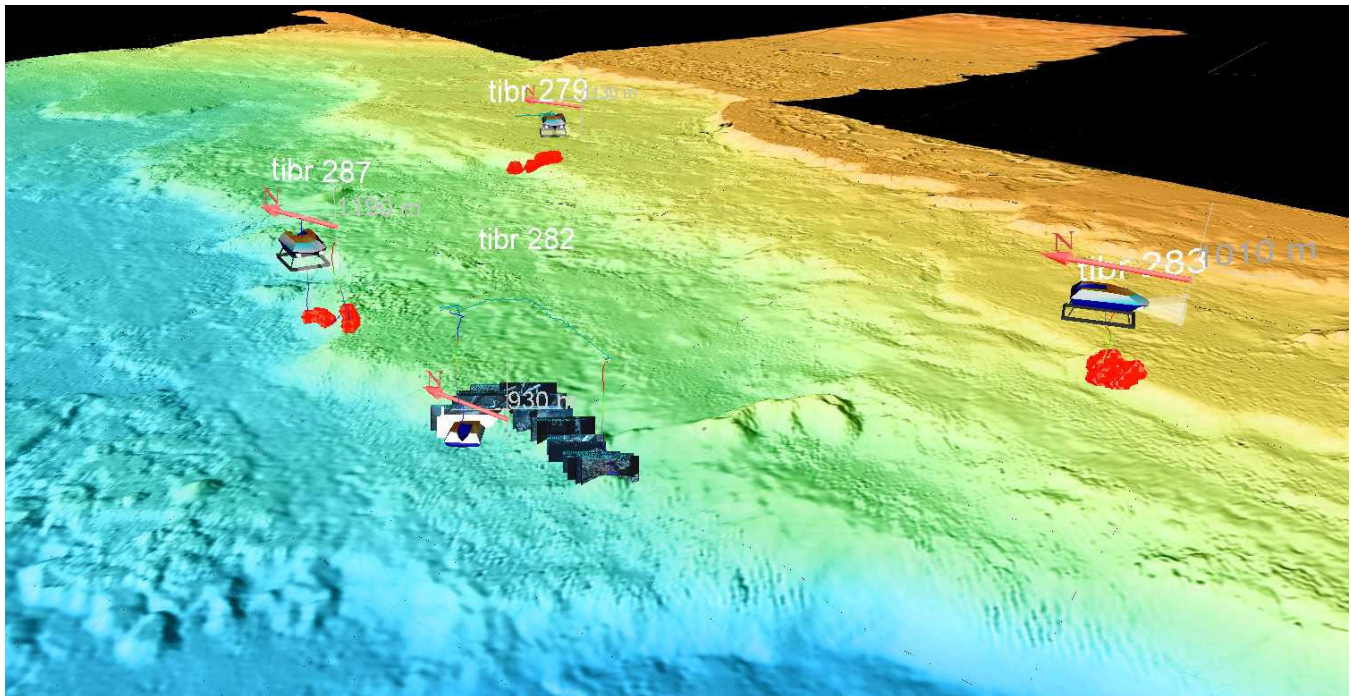


Figure 5. Four *Tiburon* dives made on submerged reefs off the northwest coast of the big island of Hawaii in 2001. The red icons show the location of samples. The samples for dive 282 are shown as the video frame grab of the collection event as the viewpoint is with the transitionDist for the GeoIconImageProto.

6. Discussion of Web 3D

The vision of web accessible interactive 3D graphics on every desktop drives most of the people working in this field. The VRML97 specification provides standard node definitions and ability to implement behavior, yet does not specify how the user navigates a 3D world. The actual methods of navigating a world on a desktop computer are left to the VRML browser. These methods were expected to improve and become more capable as computer human interfaces evolved. The problem is that the WIMP interface (Windows, Icons, Mouse, Pointer) is still the dominant computer human interface and two-dimensional mouse motion does not easily translate to three-dimensional motion in a virtual world.

The X3D [8] effort aims to improve the robustness of browser implementations by providing application specific profiles that have limited sets of nodes that need to be supported. Application specific profiles and the validation of X3D content against a DTD promises to solve some of the deficiencies of VRML97 – namely the difficulty of implementing a browser that is fully compliant. The extensible nature of the XML encoding in X3D also promises quicker adoption of new rendering features such as multi-texturing. Again, world navigation techniques are left to the implementation of the X3D browser.

For the 3D Replay application end users want to easily navigate around the ROV data objects and zoom into interesting areas for more detailed exploration. Consistent navigation techniques across different browsers on common desktop computers will aid in the use of this tool and adoption of Web3D. In addition to standardization of node syntax there should also be an effort to

adopt a recommended practice for a common set of navigation methods.

6.1 VRML Browser issues

The 3D Replay system works best with the CosmoPlayer plugin – even though it is years old it still provides the best navigation methods for our ROV dive virtual worlds. Because GeoVRML requires the execution of Java in the Script node, CosmoPlayer with GeoVRML requires Netscape 4.7 (whose embedded Java Virtual Machine accepts Java Script node execution with CosmoPlayer). Unfortunately later versions of Netscape do not support Java in Script node with CosmoPlayer. Fortunately, Netscape 4.7 and CosmoPlayer 2.1 are still available and work in Windows 2000 and Windows XP.

The 3D Replay system also works with the Cortona browser (in more modern web browsers such as Internet Explorer 6. Though Cortona has a native implementation of GeoVRML, the 3D Replay system does not use it, as Cortona's native GeoVRML requires different signatures for the External Prototypes and hence a separate set of all the 3D Replay content.

The 3D Replay system produces sensor rich worlds. All of the terrain tiles have touch sensors which must be disabled in order to use the pointing device for navigation. CosmoPlayer allows this to be done by pressing the Shift key. Cortona toggles touch sensor activation with the Scroll Lock key. Examine is the preferred navigation mode and browsers that support this mode well are the preferred ones to use for 3D Replay.

Even though the VRML browser technology is years old, the dramatic increase in graphics processor performance over the last few years has greatly improved the facility and usefulness of the

system. Video memory, rendering performance, and network bandwidth have increased to the point where the terrain generated for 3D Replay can be tuned again, perhaps increasing the tile size from 16x16 polygons to a size that reduces the number of different files that must be loaded to accurately render an area of interest. Other suggestions have also been made such as providing a video inset window to show the time evolution of the video imagery as the ROV is dragged along the track. VRML and GeoVRML have proven to be very flexible in building this system for visualizing ROV dive data.

6.2 Future directions

A natural evolutionary path for the 3D Replay system is to convert the content to X3D and use a browser such as xj3d, Flux, or Contact for integration with our Web-based Expedition database system. Taking this step would be attractive if additional functionality is achieved with X3D. An example of advanced functionality that would motivate this conversion is a shadow mapping capability such that an ROV shadow can appear on the color textured terrain. This feature would help with depth perception and improve the realism of the visualization. More robust operation promised by active development efforts of X3D browsers would be another motivation to do the conversion.

Another approach is to keep the content as VRML97, continue using the existing VRML browsers, and implement the user-requested features such as larger terrain tiles and the video inset. These features can be easily implemented with VRML97 and GeoVRML.

A third approach is to develop a stand-alone application (not a web browser plugin based system) that would provide access to and visualization of our ROV dive data. This approach can be successful as demonstrated by applications such as the Integrated Data Viewer [9]. Though this is not a Web3D-based approach, all options must be considered should we undertake an effort to revamp the existing 3D Replay system.

7. Conclusion

We have shown that 3D visualizations of oceanographic data can be routinely provided to the desktops of scientists. The scientific usefulness of the GeoVRML implementation is beginning to be fully realized. Two new nodes proposed for the GeoVRML specification provide useful features for visualizing and interacting with ROV dive data. Major development of this system has been completed and its continued operation can be supported with existing VRML browsers.

8. Acknowledgements

I'd like to thank Dr. Martin Reddy and other members of the GeoVRML working group for providing truly useful tools for representing geospatial data in VRML. Appreciation is also given to Dr. Bruce Robison, Dr. Dave Clague, Jenny Paduan, and John Graybeal for supporting the vision of this project. Kirsten Carlson produced Figure 3 and Kevin Gomes implemented the ship-based version of the 3D Replay software. Dr. Debra Stakes and Dr. Jody Webster provided comments for the Evaluation section. Lastly, I'd like to thank the MBARI intern program and interns Jenny de la Hoz and Alex Derbes for their contributions to my education and to this project. This research was supported by the Monterey Bay Aquarium Research Institute through funding from the David and Lucile Packard Foundation.

References

- [1] McCann, M. P. 2002. Creating 3D Oceanographic Data Visualizations for the Web. In Web3D-VRML 2002 Seventh Symposium Proceedings. ACM. 179-184.
- [2] GeoVRML web site <http://www.govrml.org>.
- [3] Reddy, M., Iverson, L., Leclerc, Y. G. 2000. Under the hood of GeoVRML 1.0. In Web3D-VRML 2000 Fifth Symposium Proceedings. ACM. 23-28.
- [4] GMT - The Generic Mapping Tools. <http://gmt.soest.hawaii.edu/>.
- [5] tsmApi – Tile Set Manager Application Programming Interface. <http://www.tsmapi.com/>.
- [6] MB-System. Software for the Processing and Display of Swath Sonar Data. <http://www.ldeo.columbia.edu/MB-System/>.
- [7] Luebke, D., M. Reddy, J. Cohen, A. Varshney, B. Watson, and R. Huebner. 2003. *Level of Detail for 3D Graphics*. Morgan Kaufmann.
- [8] Bullard, L. Extensible 3D: XML Meets VRML. <http://www.xml.com/pub/a/2003/08/06/x3d.html>.
- [9] IDV - Integrated Data Viewer. <http://my.unidata.ucar.edu/content/software/metapps>.