

GeoVRML 1.0 is an extension to the ISO standard Virtual Reality Modeling Language (VRML) that provides support for geographic applications. It offers the unique capability of an open file format that lets geoscientists integrate their geographic data directly into a three-dimensional (3D) computer graphics scene graph and allows remote users to view the result interactively over the web using freely-available standards-based software. The GeoVRML 1.0 release includes a document specifying a rich suite of extensions to VRML97, as well as an open source sample implementation of these extensions. A number of tools are provided as well as sample content that can be browsed with a standard VRML97 browser plug-in for Netscape Communicator or Internet Explorer. GeoVRML 1.0 was publicly announced as an official recommended practice of the Web3D Consortium in June of 2000. It is the product of work performed within the international GeoVRML working group (<http://www.geovrml.org/>). This article is intended to provide our readers with an overview of the capabilities offered by GeoVRML 1.0 and an appreciation of the status and future direction of the ongoing work.

By Martin Reddy, Lee Iverson and Yvan G. Leclerc

# GeoVRML 1.0

## Adding Geographic Support to VRML



A GeoVRML 1.0 GeoElevationGrid showing a latitude/longitude grid with a vertical exaggeration of 200.

### GeoVRML Features and Capabilities

The following list provides a high-level description of the capabilities that are specifically addressed by GeoVRML 1.0.

- 1. Geographic Coordinate Systems:** Most 3D graphics systems, including VRML, use a simple Cartesian (x,y,z) coordinate system to model all data. GeoVRML provides the ability to also specify locations using geodetic or projected coordinate systems such as those commonly used in the geosciences. Specifically, GeoVRML 1.0 has support for geodetic (latitude/longitude), Universal Transverse Mercator (UTM), and geocentric coordinate systems. In addition, 21 ellipsoids and 1 geoid are supported. This support is built upon the SEDRIS Spatial Reference Model (see <http://www.sedris.org/>).
- 2. Data Fusion:** GeoVRML can handle data from disparate servers across the web, generated from different sources, at different resolutions, and specified in different coordinate systems. These are all fused into a single global context for visualization. For example, you can overlay a Global Positioning System (GPS) track of latitude/longitude coordinates over a UTM-rectified terrain model.
- 3. High Precision:** VRML97 provides only single-precision floating point values.

This is insufficient to represent data on a planetary scale down to around meter resolution or beyond. GeoVRML provides solutions to extending this precision to sub-millimeter by employing the specification of local coordinate systems.

#### 4. Dataset Scalability:

Terrain models often involve large elevation grids or image files. GeoVRML provides various scalability features to manage the streaming of large, multi-resolution models, thus facilitating real-time access and display of arbitrarily large terrain models.

- 5. Metadata Linking:** GeoVRML provides the ability to specify a generic subset of metadata describing geographic objects, including the ability to link to a full metadata description.
- 6. Animation Support** - The ability to perform key frame animations within the supported geographic coordinate systems is provided so that animations can be defined with respect to key points on the surface of the planet.
- 7. Content Tools** - As part of the GeoVRML package, we recommend a small number of tools to aid the creation of VRML

*GeoVRML 1.0 was publicly announced as an official recommended practice of the Web3D Consortium in June of 2000.*

files that use GeoVRML nodes. These include an open source utility to convert US Geological Survey (USGS) DEM files into GeoVRML (see Figure 2), and Chris Thorne's Rez multi-resolution tiling utilities.

### Description of Extensions

Essentially, GeoVRML 1.0 consists of ten new extensions, or nodes, that sit on top of VRML97 (i.e., GeoVRML includes all of VRML97 as a subset). These nodes are defined using VRML's EXTERNPROTO extensibility features. For the purposes of the initial sample implementation, these nodes have been implemented using Java class files that are embedded within the new nodes; although it would be possible for these to be implemented natively by a VRML browser. The following sections detail each of the ten nodes that are provided by GeoVRML 1.0.

#### • GeoCoordinate

The GeoCoordinate node enables the specification of coordinates by using geographic coordinate systems. This node can be used within standard VRML geometry nodes such as IndexedFaceSet, IndexedLineSet, or PointSet, enabling the modeler to specify coordinates in a system such as UTM. For example, a GPS will normally output location as a latitude/longitude coordinate. With the GeoCoordinate node, we can insert these coordinates directly into a VRML file and have them integrated with any other geospatial data.

#### • GeoElevationGrid

The GeoElevationGrid node provides the capability to define a grid of height values offset from the ellipsoid or geoid used to model the planet. It supports the specification of height fields in latitude/longitude or UTM coordinate systems. VRML97 already

provides an ElevationGrid node; however, in this node all values are offset from a single flat plane. This is acceptable if the area being modeled is small in extent, for example, less than 1 km<sup>2</sup>, but for larger areas the curvature of the earth becomes significant. Figure 1 illustrates an example where we have created a GeoElevationGrid that spans the entire earth with all data specified in terms of a latitude/longitude grid. The grid of latitude/longitude heights is transparently converted into a Cartesian frame by the GeoElevationGrid node and accurately displayed with the correct degree of curvature for the earth. We have exaggerated the vertical heights by a factor of 200

*GeoVRML can handle data from disparate servers across the web, generated from different sources, at different resolutions, and specified in different coordinate systems. These are all fused into a single global context for visualization.*

to accentuate areas of high elevation. As a result, large mountain ranges such as the Himalayas are clearly distinguishable.

#### • GeoInline

The GeoInline node is a grouping node that reads its children data from a location in the web. The point at which its children are read and displayed is defined by the value of a field called load. If the load field is set to TRUE (the default field value), then the VRML file specified by the url field is loaded immediately. If the load field is set to FALSE, then no action is taken, but a subsequent TRUE event sent to the field will cause the file to be loaded. Such an event can be caused, for example, by a ProximitySensor event (e.g., load a scene

as the user approaches it) or a VisibilitySensor (e.g., unload the scene when it is not visible). This node provides improved scalability over the standard VRML97 Inline node because the VRML97 specification does not define when the scene for an Inline node should be loaded.

#### • GeoLocation

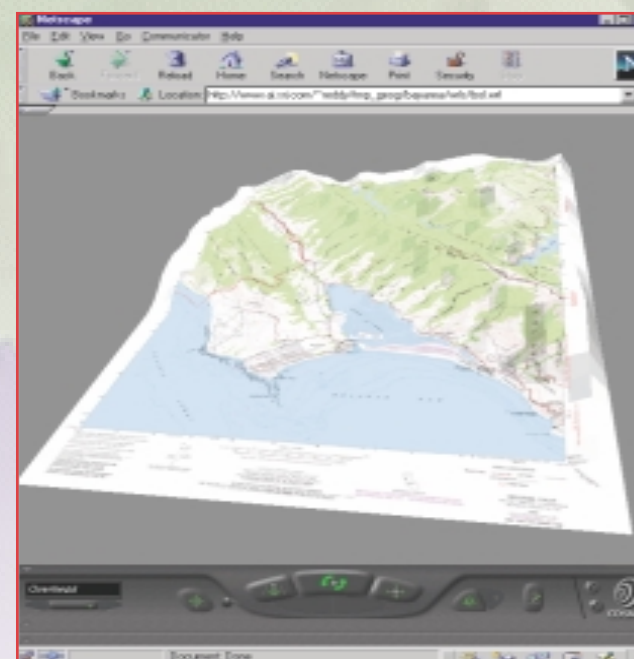
The GeoLocation node lets the user georeference an arbitrary VRML model, that is, locate it at a specific point on the earth. It also orients the model correctly, depending upon its position on the earth, so that +Y is aligned with gravitational up, +Z points true north, and +X points east. This ensures that a model built using the standard VRML right-handed coordinate system will be placed on the earth so that its base is aligned with the surface of the planet. Figure 3 illustrates the capabilities of the GeoLocation node by georeferencing models of individual buildings of the SRI International main campus to an underlying terrain model of Menlo Park, CA.

#### • GeoLOD

The GeoLOD node provides the capability to browse multi-resolution, tiled terrain data that are streamed over the web. It automatically manages the progressive loading of higher-resolution data as the user approaches the terrain, and also unloads terrain data that the user has flown past. These are essential memory management and scalability operations for browsing massive terrain datasets. For example, Figure 5 illustrates a large multi-resolution dataset that is built using the GeoLOD node. We illustrate the capability to fly down through several levels of detail while higher-resolution data are streamed over the Internet to the user's display.

#### • GeoMetadata

The GeoMetadata node aims to specify metadata describing any number of GeoVRML nodes. This can be thought of as similar to a VRML97 WorldInfo node, but specifically for describing geographic information. There is a number of organizations that are already working on standards and representations for geographic metadata, such as the ISO TC211, FGDC, CEN TC287, OpenGIS Consortium, and others. Rather than adopt any particular standard, the purpose of the GeoMetadata node is to provide links to any of these complete metadata descriptions, with the option to also supply a short, human-readable summary.



A USGS DEM that has been converted to a GeoElevationGrid using the dem2geoeg utility, and then draped with a DRG image for the same area.





A texture-mapped site model of the SRI International campus that has been georeferenced onto an underlying terrain model. Note the accurate alignment of the latitude/longitude-specified buildings with the underlying 1 m resolution USGS DOQ satellite imagery in UTM coordinates.

#### • GeoOrigin

VRML97 uses single-precision (32-bit floats) to model and render all geometry. However, we require greater precision to model geographic (i.e., geocentric) coordinates beyond a resolution of several meters. That is, in order to represent objects at the resolution of individual mailboxes (or smaller) over the range of the entire planet. We solve this problem by defining an absolute geographic origin (defined by a GeoOrigin node) in double-precision using MFStrings. Then for all double-precision geographic coordinates we take the difference between each coordinate and the GeoOrigin. The result gives a single-precision offset that can be used for faithful rendering (assuming a good origin was chosen). In effect, we specify the geographic location that will be used by the graphics system as its (0,0,0) point for rendering the scene. All of the GeoVRML nodes that deal with coordinates support the use of a GeoOrigin in this fashion. Figure 4 illustrates the single-precision rounding artifacts that can occur when geographic coordinates are represented to only single-precision.

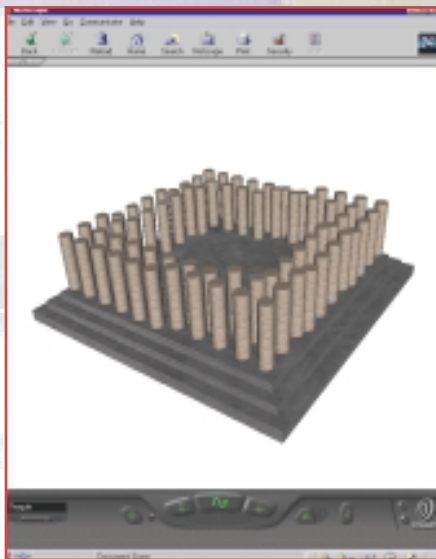
#### • GeoPositionInterpolator

One of the many strengths of VRML is its ability to model dynamic systems. We therefore incorporate the capability to animate models using geographic coordinates. This is implemented through the GeoPositionInterpolator node, which functions in much the same way as the stan-

dard VRML97 Position-Interpolator node, except that the key frame values can be specified using geographic coordinates. For example, if a GeoPosition-Interpolator is created and given two latitude/longitude coordinates, (40.669, -73.944, 10000) and (48.865, 2.35, 10000) deg, and the output is routed to a VRML model of a Boeing 777, then this aircraft would fly over the surface of the planet, from Paris to New York, at a constant altitude of 10,000 m. Using this capability, a vehicle could be animated based upon the list of latitude/longitude coordinates from a prerecorded GPS track, or from a live feed of Distributed Interactive Simulation (DIS) Protocol Data Units (PDUs).

#### • GeoTouchSensor

The GeoTouchSensor node was added to the GeoVRML 1.0 nodeset by John Brecht of SRI International. It provides a means to discover the geographic location of the part of a scene under the mouse cursor. For example, an intelligence analyst may wish to point at a building and find out its latitude/longitude coordinate, or a database-driven VRML application could display some location-specific information



An illustration of the rounding errors that can occur in geographic models due to single-precision inaccuracies. Both images show a model of a virtual temple structure where each column and slab has been georeferenced to meter accuracy under the UTM projection. In (a)-left image we have not used a GeoOrigin node so the absolute double-precision coordinates have just been rounded to give single-precision. The result is a number of visual artifacts where objects are not correctly co-positioned and camera jitter is experienced during movements. In (b)-right image, the same coordinates are specified, except that this time we use a GeoOrigin node to establish a local single-precision coordinate system around the center of the model. All objects are now faithfully located and no camera jitter is evident during navigation.

once the user clicks over certain regions on the surface of a GeoElevationGrid globe.

#### • GeoViewpoint

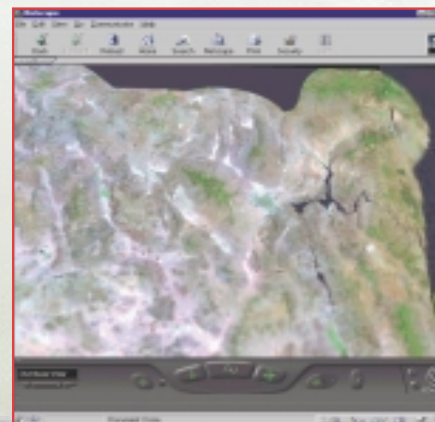
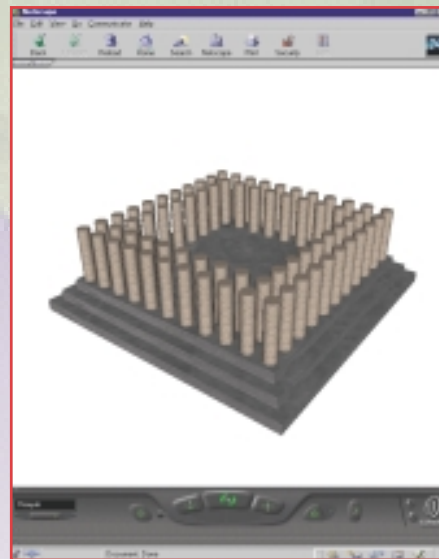
The GeoViewpoint node allows you to specify a viewpoint in terms of a geographic coordinate. As part of this capability, the camera orientation can be specified relative

*We can insert GPS coordinates directly into a VRML file and have them integrated with any other geospatial data.*

to the local up vector for that location, i.e. specifying a rotation of 180 deg about the X axis will always make the camera point down. The GeoViewpoint location could be animated based upon the output of a GeoPositionInterpolator in order to create geospatially-defined flight paths.

#### Status and Directions

GeoVRML 1.0 is currently an accepted recommended practice of the Web3D Consortium. In addition to this, the GeoVRML nodes have been submitted to ISO as part of an amendment to the VRML97 specification. The intention is



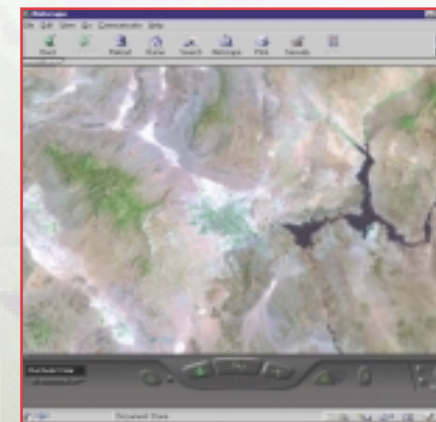
An example showing the scalability features provided by the GeoLOD node. Images (a) - (c) show a user's descent into a terrain model of the Mojave desert in southern California. The source imagery and elevation data for this terrain model total 1.3 GB.

that these capabilities will eventually become an optionally-implemented part of the current VRML ISO specification, and also an additional profile in any future revisions of the specification.

The GeoVRML work is of major relevance to several exciting initiatives that are currently evolving. For example, we note that this work provides the technical foundation to build the vision of a highly accurate and large-scale model of the earth into which we can embed massive quantities of georeferenced data: a vision that was offered recently as a challenge to the scientific community by U.S. Vice President Al Gore. SRI is implementing one version of this vision in its Digital Earth project (<http://www.ai.sri.com/digital-earth/>) as part of the Federal Digital Earth initiative (<http://www.digitalearth.gov/>).

On a related topic, the OpenGIS Consortium recently announced its Web Mapping Testbed technology as a means to revolutionize the use of geospatial data on the web. Its efforts to date have focused largely on two-dimensional (2D) presentations and also on the cataloging of geographic data. As such, the GeoVRML work is well positioned to contribute to this effort and bring web-based 3D visualization capabilities to the initiative.

In addition, the GeoVRML working group is tracking the evolving X3D development and hopes to provide extensions to enable the support for geographic applications in X3D. In this regard, Don Brutzman of the Naval Postgraduate School has already produced a document type definition (DTD) for GeoVRML which is available from the X3D web page at <http://www.web3d.org/x3d.html>.



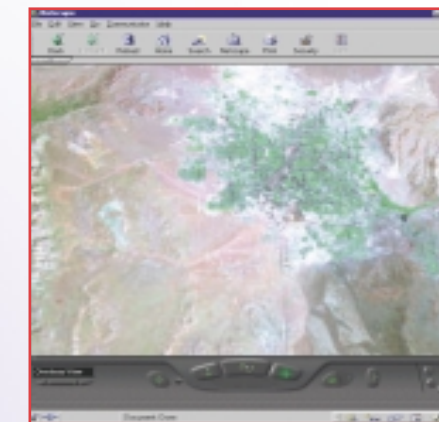
Finally, the GeoVRML group is currently compiling issues for a future version of GeoVRML, including support for more coor-

*GeoVRML automatically manages the progressive loading of higher-resolution data as the user approaches the terrain, and also unloads terrain data that the user has flown past. These are essential memory management and scalability operations for browsing massive terrain datasets.*

dinate systems, ellipsoid definitions for all other planets in the solar system, new GeoVRML nodes, and additions to various current nodes.

#### Conclusions

GeoVRML 1.0 provides geoscientists with a rich suite of enabling capabilities that cannot be found elsewhere as a web browser plug-in. That is, the ability to model dynamic 3D geographic data that can be distributed over the web and interactively visualized using a standard browser configuration. GeoVRML includes nodes for VRML97 that perform this task; addressing issues such as coordinate systems, scalability, animation, accuracy, and preservation of the original geographic data. The implementation is released as



open source and includes various tools for generating GeoVRML data. All these facilities provide geoscientists with an excellent medium to present complex 3D geographic data in a dynamic, interactive, and web-accessible format. For further details, see <http://www.geovrml.org/>.

Dr. Martin Reddy ([reddy@ai.sri.com](mailto:reddy@ai.sri.com)) is a computer scientist in the Artificial Intelligence Center's Perception Program at SRI International. His work focuses on the real-time display of massive 3D terrain databases distributed over wide-area networks.

Dr. Lee Iverson ([leei@ai.sri.com](mailto:leei@ai.sri.com)) is a computer scientist at SRI's Artificial Intelligence Center. His areas of interest include computer vision (especially edge detection, relaxation labeling, and stereo) and graphical environments for geographical information systems.

Dr. Yvan G. Leclerc ([leclerc@ai.sri.com](mailto:leclerc@ai.sri.com)) is a senior computer scientist at SRI's Artificial Intelligence Center, doing research in computer vision and network-based terrain visualization since 1985.

SRI International, Menlo Park, CA 94025, USA.  
<http://www.ai.sri.com/>

#### Acknowledgements

Many people have contributed to the development of GeoVRML 1.0. In particular, the authors would like to thank Farid Mamaghani and the SEDRIS project for algorithms to support geographic transformations; Dan Toms for producing a Java package based upon these algorithms; Jay Feuquay and Brian Davis of the USGS for providing terrain data; and Nat Bletter, Aaron Heller, and Doug Gordin for providing resources for the various application examples. The work performed by SRI International was funded under the DARPA-sponsored projects BADD (contract no. MDA972-97C-0037), MAGIC II (subcontract 12165SRI under contract no. F19628-95-C-0215), and Digital Earth (contract no. MDA972-99-C-0011).