BUTTERWORTH HEINEMANN

Virtual reality modeling language in chemistry

H. Vollhardt,* C. Henn,† G. Moeckel,* M. Teschner,† and J. Brickmann*‡

**Institut für Physikalische Chemie and ‡Darmstädter Zentrum für wissenschaftliches Rechnen, Technische Hochschule Darmstadt, Darmstadt, Germany, †Silicon Graphics, Inc., Riehen, Switzerland*

A new concept in the field of molecular modeling using the information transfer mechanism of the World Wide Web (WWW) is presented. The Virtual Reality Modeling Language (VRML) provides an object-oriented method for the description of molecular models. The structure and capabilities of this new language are introduced. It is shown that the transport of molecular models over the WWW using VRML is a very efficient and powerful method for the exchange of molecular information.

Keywords: World Wide Web (WWW), Virtual reality modeling language (VRML), Molecular modeling

INTRODUCTION

Since the early days of the internet, scientists have used computer networks to exchange their knowledge and their experiences. In the first stage, this was achieved by electronic mail. But this medium permits the transport of information only between a few participants. Later, with the introduction of mailing lists and newsgroups questions and answers could be shared among the scientific community more globally.

With the development of the World Wide Web (WWW) in 1989 the situation has changed dramatically. Within the WWW, it is possible to exchange information in various forms. Data retrieval can be done easily. With hyperlinks everything can be referenced from anywhere on the internet. The WWW has turned the entire internet into one large storage for information of any kind.

The new information exchange technology has already led to a rapid growth of electronic publishing media. Not only text and static images can be submitted to an electronic journal, audio sequences and animations can be included as well. The formats used for images and animations are pixel based in nature and the representations are fixed in size and rigid in their behavior. The viewer is not able to change the point of view with respect to the object shown. Moreover, pixel-based formats do not allow the transmission of information in a compact manner. To obtain a three-dimensional (3D) impression of an object, at least two images are needed and the point of view is still preselected by the producer of the scenario. In molecular science it is absolutely necessary to have three dimensions in order to transfer complex structural information. Traditionally, the structural information is transferred via standard file formats (e.g., Brookhaven Protein Data Bank, PDB) containing atom types and 3D coordinates and the visualization and manipulation are done locally with the aid of modeling software packages (such as SYBYL/MOLCAD[1] or others). There was no way up to now to transfer 3D scenario directly. This gap can be filled using the virtual reality modeling language (VRML).[2]

The idea for VRML was born in 1994 at the first WWW conference in Geneva. A group of attendees agreed on the need for a common language to specify 3D scene description and WWW hyperlinks. Immediately after the conference, the "www-vrml" mailing list[3] was established and the discussion on the development of a language specification started. The list members agreed to adapt VRML from a subset of the Open Inventor ASCII File Format[4] from Silicon Graphics, Inc.

This article is organized as follows. In the following section a short description of VRML is given in order to point out the structure of this new language. The next section (VRML in Relation to Other File Formats) deals with a comparison to other file formats, and the specific advantages of VRML are described. Next, some applications are outlined that are contained in the VRML-WWW page of the Institute of Physical Chemistry, Technical University Darmstadt.[5] In the final section some conclusions are drawn and an outline is given.

SHORT DESCRIPTION OF VRML

This section gives an introduction to the Open Inventor File Format and the VRML specification. It provides an over-

Color plates for this article are on pp. 352 and 353

Address reprint requests to Dr. Brickman at the Institut für Physikalische Chemie, Technische Hochschule Darmstadt, Petersenstr. 20, D-65287 Darmstadt, Germany

Journal of Molecular Graphics 13:368–372, 1995

655 Avenue of the Americas, New York, NY 10010

0263-7855/95/$10.00
SSDI 0263-7855(95)00080-1

view of the basic elements of VRML. For a complete definition the reader is referred to Ref. 4.

VRML is based on a subset of the Open Inventor File Format. This subset was extended with networking capabilities, such as WWW hyperlinks. With this feature, VRML is an equivalent to the hypertext markup language (HTML).[6] Just as HTML files describe the layout of 2D text pages to be displayed by WWW browers, VRML file describe the layout of 3D scenarios. The language contains some basic elements that can be characterized as follows.

Nodes

A VRML file describes a scene in an object-oriented manner. The basic elements building the objects are called *nodes*. There are five different categories of nodes:

1. *Shape nodes* represent 3D geometry objects. They are the only nodes that are visible.
2. *Property nodes* affect the appearance and characteristics of the other nodes. They can be divided into three subgroups.
   a. *Transform nodes* perform coordinate transformations on the given geometry.
   b. *Appearance nodes* affect the appearance of the object, such as color or roughness.
   c. *Metrics nodes* contain coordinates and other geometric information.
3. *Group nodes* are used to collect child nodes to implement a hierarchical structure. Some of them can isolate the effects of their children from the rest of the scene.
4. *Light nodes* are used to define different types of light sources to illuminate the scene.
5. *Camera nodes* are used to define different points of view, for example, to create a guided tour.

In Table 1 all node types defined in the version 1.0 specification of VRML are listed. Apart from their type, nodes have some further characteristics. They have associated "fields" describing their parameters (e.g., the radius of a sphere or the diffuse color of a material). A name can be assigned to a node, which allows it to be used elsewhere in the scene. All parts can be combined to the following general syntax for a node:

```
DEF name node_type { fields children }
```

The name together with the DEF keyword as well as the fields and the children are optional. Only the node type and the curly braces are required for a node definition.

Table 1. Categories of the VRML nodes

Type	Nodes
Shape	AsciiText,[a] Cone, Cube, Cylinder, IndexedFaceSet, IndexedLineSet, PointSet, Sphere
Property	Coordinate3, FontStyle,[a] Info, Material, MaterialBinding, MatrixTransform, Normal, NormalBinding, Rotation, Scale, ShapeHints, Texture2, Texture2Transform, TextureCoordinate3, Transform, Translation
Light	DirectionalLight, PointLight, SpotLight
Group	Group, LOD,[a] Separator, Switch, TransformSeparator, WWWAnchor,[a] WWWInline[a]
Camera	OrthographicCamera, PerspectiveCamera

[a]Not defined in the Open Inventor File Format.

Scene graph

Nodes are arranged in a hierarchical structure called "scene graph." A scene graph defines the ordering of the nodes and therefore the layout of the 3D scene. The ordering is of importance because nodes earlier in the scene can affect nodes coming up later in the scene. Figure 1 shows an example of a scene graph for a water molecule in CPK representation. Every time the molecular model must be rendered, this scene graph is traversed from top to bottom and from left to right. The different symbols represent the different types of nodes. The following source code contains the complete description of the scene graph of Figure 1.

```
#VRML V1.0 ascii
Separator {
  Separator {
    Material { diffuseColor 1.0 0.0 0.0 }
    Translation { translation -0.085 0.398 -0.103 }
    Sphere { radius 1.6 }
  }
  Separator {
    Material { diffuseColor 1.0 1.0 1.0 }
    Separator {
      Translation { translation -0.829 -0.379 0.072 }
      Sphere { radius 1.3 }
    }
    Separator {
      Translation { translation 0.913 -0.019 0.032 }
      Sphere { radius 1.3 }
    }
  }
}
```

The first line of a VRML file must be "#VRML V1.0 ascii" to identify the contents. Everything after a "#" is interpreted as a comment and will be ignored. After the header line the file contains exactly one node. In most cases this will be a group node holding the rest of the scene.

Extensions

The two most important nodes introduced with VRML are *WWWInline* and *WWWAnchor*. They represent the networking capabilities of the language.

The WWWInline node is used to load additional VRML files from elsewhere on the WWW into the scene. This is accomplished with a field, that defines a Uniform Resource Locator (URL),[7] that is, the location on the internet where the object is stored. The loading may be time delayed until the inlined object is visible on the screen. With this feature,

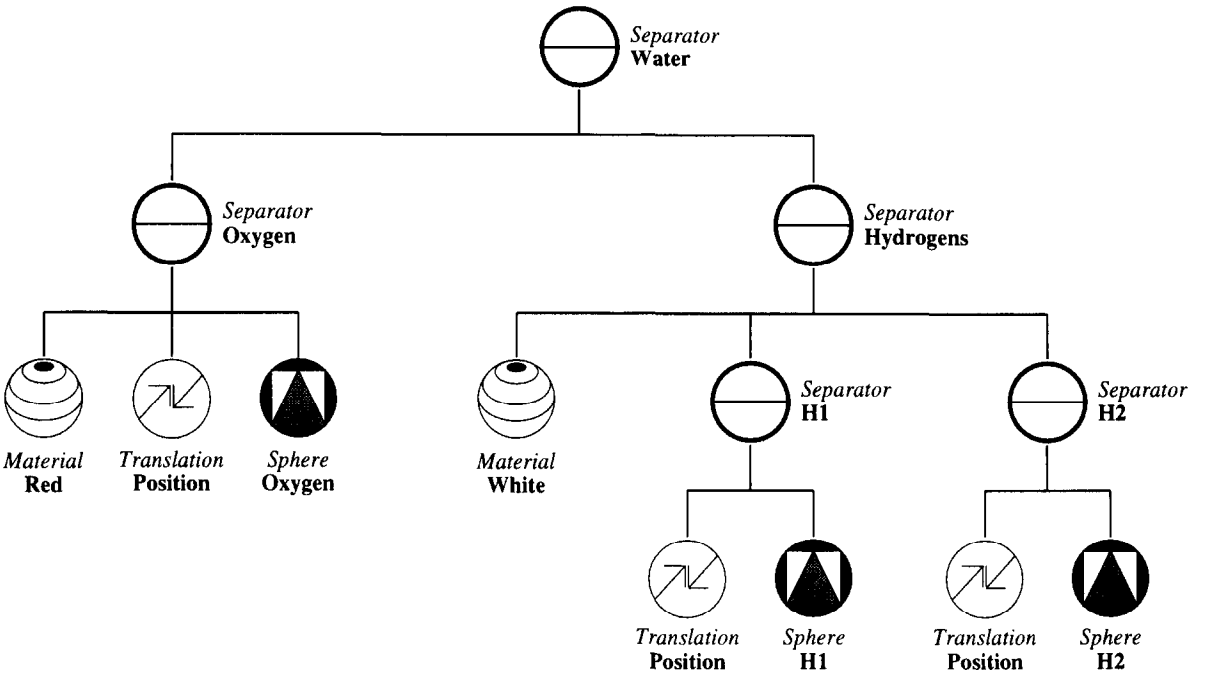

Figure 1. Graphical representation of a VRML scene graph. The different symbols represent the different types of nodes.

large and complex scenes can be composed from a library of smaller objects.

The WWWAnchor node is the equivalent to the anchor field in HTML. Hence, it represents the WWW hyperlink. Because this node is a group node, all its children act as one hyperlink. The URL for this hyperlink is set with a field, analogous to the WWWInline node.

Extensibility

The VRML specification also supports the extension of the node set. Nodes not defined in standard VRML are introduced by *self-describing* nodes. These nodes hold a description of all their fields. If a new node is derived from an existing one, it can define a field containing the name of the original node.

Molecular Inventor,[8] developed at the Basel institution, is an example for an extension of VRML. It is based on the Open Inventor base class set and allows a quick implementation of molecular graphics rendering primitives like high-quality lines, stereo in a window, fast construction of isosurfaces and molecular surfaces, volume rendering, slice planes, property mapping, and so on. With *Molecular Inventor*, the complete information about a given molecule is maintained and can be exchanged locally between applications or globally via the internet.

VRML IN RELATION TO OTHER FILE FORMATS

As mentioned above, the currently used file formats for the transfer of nontextual information are sometimes ineffective. The use of image formats such as GIF[9] or JFIF[10] for molecular structures is insufficient. Because of their 2D nature, single pictures or sequences of pictures are a poor medium for storage and transfer of 3D molecular information. Some authors use small animation sequences; they create a better 3D impression by changing the point of view continuously. But both media are fixed in size and resolution. Objects in VRML files, on the other hand, are stored on the basis of their 3D coordinates. Thus, VRML scenes are in principle infinitely scalable. The files can be compressed to much smaller sizes than files of other formats. This is demonstrated in Table 2, which presents the sizes of files depicting the same object in different formats. The corresponding molecule is shown in Color Plate 1. VRML files are written in ASCII format, and therefore they can be ideally compressed.

The development of chemical MIME[11] (Multipurpose Internet Mail Extensions) types[12] has led to a different approach to the transport of chemical information. MIME types are used to specify the contents of a given file transferred by the WWW (e.g., PDB file, GAUSSIAN output, or MOPAC output). The receiver of the file decides, depending on the particular MIME type, which application should be launched in order to visualize its contents. This method extends the possibilities of chemical information transfer, but also has some disadvantages. First, some of the

Table 2. Comparison of file formats

Format	VRML	GIF image	MPEG stream
Resolution (pixel × pixel)	∞[a]	640 × 480	320 × 256
Colors	∞[a]	256	256
Frames	∞	1	90
File size (KB)	14	27	170
Compressed (KB)	3	27	170

[a]Depends on the hardware.

output files are written in a platform-dependent binary format. Second, in the worst case, for every file format a different viewer must be used. Finally, the author has no control over how the reader is using the data. An unexperienced user may not be able to filter the information correctly. Using VRML, the author controls the layout of the final output. The reader views the result directly without having to learn a number of different and sometimes complicated user interfaces. Additionally, with hyperlinks, the author can add annotations to the objects describing them in various forms, such as text, audio, or video.

APPLICATIONS

Some of the capabilities of VRML are demonstrated at the WWW page[5] of the Darmstadt institution with examples from protein research, visualization of molecular structures from 3D databases, and visualization of atomic orbitals.

Prebuild VRML files

***Cytochrome* P-450** Cytochrome *P*-450 enzymes play a central role in carcinogenesis and toxicology. The carcinogenic nitrosamines are activated by hydroxylation at the porphyrin-binding site (Color Plate 1) of the enzyme. To enter this active site located at the center, about 10 Å from the surface, substrates must pass through a kind of channel[13] (Color Plate 2). This substrate channel is demonstrated with a modified 3D structure of cytochrome *P*-450cam (Color Plate 3), which is widely accepted as a model system for *P*-450 enzymes.[14]

In the VRML description, cytochrome *P*-450 has been divided into four parts: protein backbone, channel residues, active site, and substrate. These parts are stored in separate files. To reconstruct the complete enzyme model all files can be combined using the *WWWInline* node. Additionally, each node acts as a hyperlink pointing to the VRML file of the isolated part. Because one hyperlink can reference only one URL, to each model scenario smaller objects (e.g., cones or cubes) have been added that act as additional hyperlinks. Some of them can be used to switch between different representations (wireframe, ball-and-stick, CPK model, etc.). Others are linked to textual information about the given object. In this way, it is easy to navigate through the different components of the enzyme and obtain related data.

The wireframe models are built exclusively out of *IndexLineSet* nodes, whereas the ball-and-stick and CPK representations are made of the more complex shape nodes provided by VRML, spheres and cylinders. VRML can also be used for building user-defined surfaces with *IndexedFaceSet* nodes. This is demonstrated with the solvent-accessible surface[15] of the channel region (Color Plate 4). These surfaces can be color coded with molecular properties (e.g., hydrophobicity). This can be achieved using Gouraud shading[16] or texture mapping[17] techniques.

Interactive generation of VRML files

The HTTP (HyperText Transfer Protocol)[18] provides a second, powerful mechanism for the transfer of data over the WWW. Instead of sending prebuilt files, the data can be generated interactively with CGI[19] (Common Gateway Interface) scripts. The output of these scripts can be controlled with parameters, which again can be set by the user through forms. The flexibility of this interface, provided by most of the WWW browers, is demonstrated with two examples.

Visualization of molecular structures First, an interface to a small database of molecular structures stored in the PDB file format was constructed.[20] The form (Color Plate 5) permits the user to select one of the molecules in the database. Additionally, the representation (e.g., wireframe or ball-and-stick) and the color can be chosen. After sending the request to the WWW server, the user receives the VRML file, generated from the PDB file, within a few seconds.

Visualization of atomic orbitals The second example[21] (Color Plate 6) is the visualization of electron wavefunctions of a hydrogen atom using isocontour surfaces (Color Plate 7). The user selects the orbital type (e.g., $1s$, $2p_x$). The corresponding wavefunction is calculated at points on a 3D grid. The size of the grid and the spacing of the grid points can also be selected. In the second step a isocontour surface with a user-defined contour value is generated using a marching cube algorithm.[22] In the next step the surface file is converted into a VRML file. The user can choose the type of representation (dots, wireframe, or solid). Finally, the VRML file can be compressed by a factor between 4 and 5 before it is returned to the user. All steps together require only a few seconds on a standard workstation.

CONCLUSIONS AND OUTLOOK

In the current state VRML is an excellent visualization tool. It can be used to prepare platform-independent files to visualize the output of experiments or computer simulations, such as chemical structures or *ab initio* calculations. Because VRML is an equivalent to HTML, it is more than a simple file format for the storage of data, like GIF or MPEG. With the WWW hyperlinks information can be referenced from elsewhere on the internet.

The number of applications for VRML will grow significantly in the course of time, like the HTML field. One of the next steps in the development of the VRML specification will be the implementation of *dynamic behavior*. With this feature, motion can be applied to parts of the scene, allowing the visualization of vibrations or histories from molecular dynamics simulations. Another requirement for VRML is *interactivity*, which enables the user to change the contents of a scene. With this set of features VRML could become a standard data exchange format for collaborative applications. A modeling language with all these capabilities may lead to a complete shift of molecular modeling technology.

ACKNOWLEDGMENTS

We thank H. Look (Silicon Graphics, Inc.) for providing the early versions of the VRML viewer WebSpace[23] and B. Spiegelhalder (Deutsches Krebsforschungszentrum, Heidelberg, Germany) for helpful discussions on protein

biophysics. This work is partly supported by the Fonds der chemischen Industrie (Frankfurt, Germany).

REFERENCES

1 Brickmann, J., Goetze, T., Heiden, W., Moeckel, G., Reiling, S., Vollhardt, H., and Zachmann, C.-D. Interactive visualization of molecular scenarios with MOLCAD/SYBYL. In: *Data Visualization in Molecular Science* (Bowie, J.E., ed.). Addison-Wesley, Reading, Massachusetts, 1995, pp. 84–97

2 http://vrml.wired.com/vrml.tech/vrml10-3.html

3 www-vrml@wired.com

4 Wernecke, J. *The Inventor Mentor*. Addison-Wessley, Reading, Massachusetts, 1994

5 http://www.pc.chemie.th-darmstadt.de/vrml

6 Berners-Lee, T. and Connolly, D. Hypertext Markup Language version 2.0. (1995); (http://www.w3.org/hypertext/WWW/MarkUp/html-spec/html-spec_toc.html)

7 Berners-Lee, T., Masinter, L., and McCahill, M. Uniform Resource Locators (URL). RFC 1738 (1994); (ftp://ds.internic.net/rfc/rfc1738.txt)

8 Henn, C., Benzel, M., Engel, A., and Teschner, M. Submitted, 1995

9 CompuServe, Inc., Graphics Interchange Format (1992); (http://www.w3.org/hypertext/WWW/Graphics/GIF/spec-gif89a.txt)

10 Hamilton, E. JPEG File Interchange Format (1992); (http://www.w3.org/hypertext/WWW/Graphics/JPEG/jfif.txt)

11 Borenstein, N. and Freed, N. MIME (Multipurpose Internet Mail Extensions). Part One. Mechanism for Specifying and Describing the Format of Internet Message Bodies. RFC 1521 (1993); (ftp://ds.internic.net/rfc/rfc1521.txt)

12 Rzepa, H., Murray-Rust, P., and Whitaker, B. A Chemical Primary Context Type for Multipurpose Internet Mail Extensions. Internet draft (1995); (http://www.ch.ic.ac.uk/chemime2.html)

13 Moeckel, G. Theoretische Methoden zur Untersuchung molekularer Grundlagen der Carcinogenese von Nitrosaminen. Ph.D. thesis. University of Darmstadt, Darmstadt, Germany, 1994

14 Poulos, T.L., Finzel, B.C., and Howard, A.J. High-resolution crystal structure of cytochrome P450cam. *J. Mol. Biol.* 1987, **195,** 687–700

15 Connolly, M. Solvent accessible surfaces of proteins and nucleic acids. *Science* 1983, **211,** 709–713

16 Gouraud, H. Continuous shading of curved surfaces. *IEEE Trans. Comput.* 1971, **20,** 623–628

17 Teschner, M., Henn, C., Vollhardt, H., Reiling, S., and Brickmann, J. Texture mapping: A new tool for molecular graphics. *J. Mol. Graphics* 1994, **12,** 98–105

18 Berners-Lee, T., Fielding, T., and Frystyk Nielsen, H. HyperText Transfer Protocol—HTTP/1.0 Internet draft (1995); (http://www.w3.org/hypertext/WWW/Protocols/HTTP1.0/draft-ietf-http-spec.html)

19 http://hoohoo.ncsa.uiuc.edu/cgi/interface.html

20 http:/www.pc.chemie.th-darmstadt.de/vrml/pdbvis.html

21 http://www.pc.chemie.th-darmstadt.de/vrml/wave.html

22 Heiden, W., Goetze, T., and Brickmann, J. Fast generation of molecular surfaces from 3D data fields with an enhanced "marching cube" algorithm. *J. Comput. Chem.* 1993, **14,** 246–250

23 http://www.sgi.com/Products/WebFORCE/WebSpace