To appear in Proceedings of The First Annual Symposium on the Virtual Reality Modeling Language (VRML 95), San Diego, California, December 1995.

# VRweb: A Multi-System VRML Viewer

Michael Pichler, Gerbert Orasche, Keith Andrews
IICM, Graz University of Technology[1]

Ed Grossman
National Center for Supercomputing Applications[2]

Mark McCahill
University of Minnesota[3]

**Abstract**

As VRML becomes *the* standard for describing 3D scenes on the Internet, many VRML viewers are being developed with proprietary interests or for specific target systems or protocols.

VRweb is a VRML viewer providing the same user interface for multiple Web protocols (WWW, Gopher, and Hyper-G) and multiple platforms (Unix, Windows, and Macintosh) and is available as both binary and source code. VRweb source code is copyrighted, but is freely available for non-commercial use, providing a platform for research and experiment. Unlike other VRML viewers available in source code, VRweb does *not* require additional commercial libraries like OpenInventor or Motif, it is based entirely on freely available software components.

A short overview of currently available VRML browsers is followed by a more detailed look at VRweb, including its user interface, multi-system nature, and software architecture.

**Keywords:** VRML, WWW, Gopher, Hyper-G, OpenGL, X11, Windows, Internet, Web.

## 1 Introduction

The Virtual Reality Modeling Language [7, 24], or VRML for short, is evolving as a standard data format for the storage and interchange of 3D models in information systems and modeling packages. VRML is based on the ascii form of Silicon Graphics' Inventor [27] file format, an extensive, proven 3D graphics file format which includes a variety of special-purpose nodes. VRML adopted Inventor's node-based hierarchical structure, field syntax and instancing mechanism, but left out a number of its special-purpose and more esoteric features. VRML defines nodes for geometry (polygon based and primitive shapes), text, grouping, multiple levels of detail, transformations (elementary and arbitrary matrices), coordinates, normal data, properties (materials, textures), rendering attributes (bindings and shape hints), light sources (positional, directional, spot lights), and camera definitions (perspective and orthographic).

The World Wide Web (WWW, W3, or "The Web") [8], currently the most popular Internet information system, is based on the HTML [9] markup language and the HTTP client/server protocol. Navigation is primarily achieved by means of hyperlinks. In contrast, navigation in the Internet Gopher [16] is primarily hierarchical, including its new 3D interface, GopherVR [17]. Both WWW and Gopher support add-on search facilities. Hyper-G [4, 5] combines hierarchical and hyperlink navigation with fully integrated search facilities, whilst remaining interoperable with WWW and Gopher (Hyper-G servers speak WWW and Gopher protocols as well as native Hyper-G). All three systems allow VRML models residing on their respective servers to be accessed by their supported access protocols.

The Web's URL (Universal Resource Locator) mechanism allows documents and files residing on arbitrary servers somewhere on the Internet to be specified using one convenient handle: a document's URL includes its access protocol (for example http, gopher, ftp, or hyperg). VRML's WWWInline and WWWAnchor nodes utilise the URL mechanism. The former allows a scene to be split into parts which may be fetched on demand, for example when navigating a large, complex model components of the model are fetched as one approaches them (important on a slow network connection). The latter supports link anchors attached to arbitrary components of a scene; these can be clicked to load related VRML or other files. Important for the backward-compatibility of extensions to VRML is the possibility for a node to provide a description of its fields and (optionally) the name of a base node that can be used as its substitute. VRML 1.0 supports only *static* scene descriptions; dynamic object behaviour is now under discussion for inclusion in VRML 2.0.

## 2 Related Work

Since the release of the first VRML viewers/browsers in April 1995, there have been a flood of releases and announcements. Here we present an overview of the field as of Au-

[1] Institute for Information Processing and Computer Supported New Media (IICM), Graz University of Technology, A-8010 Graz, Austria. {mpichler,gorasche,kandrews}@iicm.tu-graz.ac.at

[2] National Center for Supercomputing Applications (NCSA), University of Illinois, Urbana-Champaign. egrossma@ncsa.uiuc.edu

[3] Distributed Computing Services, University of Minnesota. mpm@boombox.micro.umn.edu

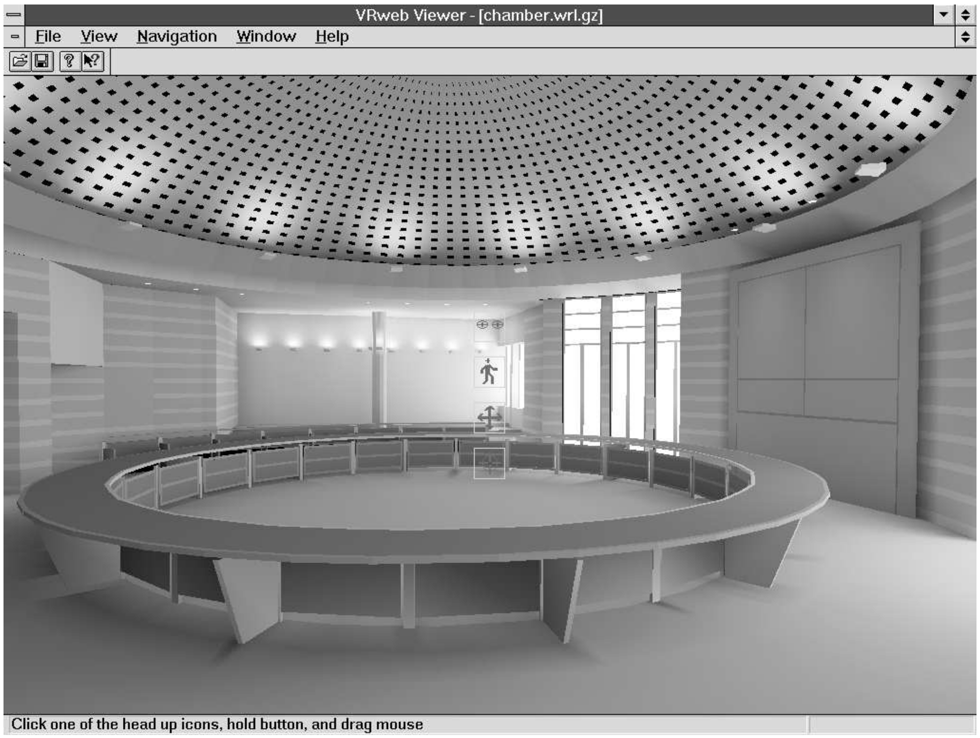


Figure 1: *The VRweb VRML viewer displaying Lightscape's model of Jerusalem City Hall.*

gust 1995 (which is necessarily neither complete nor up-to-date). We use the term *browser* to refer to software which itself retrieves VRML files across the Internet and *viewer* for software which relies on another supporting application to perform file retrievals.

*WebSpace* [26] from Silicon Graphics in collaboration with Template Graphics was the first VRML browser to be released. WebSpace is based on the Inventor library and has two navigational metaphors (the examiner viewer and walk viewer). WebSpace binaries are freely available (unsupported) with a supported commercial version; no source code is available. WebSpace runs on SGI, IBM AIX, Sun Solaris (with graphics board), and Windows NT; versions for Macintosh, DEC and HP are planned.

*WebView* [25] from the San Diego Supercomputer Center (SDSC) is a publically available VRML browser for SGI systems, available as source code based on the Inventor library. WebView has four viewing styles (examiner, fly, plane, walk) and an integrated editing facility. It can either be run as a standalone browser or as an external viewer with a Web browser. It is intended as a public development and test platform, but is limited to SGI platforms under UNIX.

*WorldView* [12] from InterVista Software is targeted to empower standard PCs for real-time applications. All network communication is built into this standalone browser, which does not rely on a cooperating Web browser. WorldView is available for all Windows platforms (NT, 3.1, 95); versions for Macintosh and UNIX (SGI, Sun OS) are planned.

*WebFX* [21] from Paper Software is a VRML browser for the Windows environment. The binaries are free for non-commercial use and during an evaluation period. It incorporates IRC 3D chatting and physically-based navigation metaphors (including collision detection) as well as VRML authoring facilities. It is available for Windows platforms (NT, 3.1, 95); a Macintosh version is planned.

*i3D* [10] from the Center for Advanced Studies, Research and Development in Sardinia exploits the rendering capabilities of high-end machines. This VRML viewer includes a preprocessing optimisation phase and time-dependent rendering facilities to guarantee constant frame rates even for very large scenes. It is available only for SGI machines.

*WebOOGL* [18] from the Geometry Center of the University of Minnesota is available in source code based on Geomview. Geomview is a program for viewing and manipulating 3D objects and is designed to act as display unit for external modules creating geometry. Multiple control windows exist for motion control, properties and editing. The WebOOGL browser is available for SGI and Sun OS platforms.

*VRweb*, the VRML viewer described in this paper, is a joint project between IICM (home of Hyper-G), NCSA (home of Mosaic), and the University of Minnesota (home of Gopher). It draws on several years of experience with the Harmony 3D Scene Viewer [2, 6] for Hyper-G, which became the base of VRweb. VRweb is designed to work with multiple information systems, namely WWW, Hyper-G, and Gopher, as well as on a variety of platforms. It is currently available for most common UNIX platforms (SGI IRIX, Sun Solaris, Sun OS, DEC Alpha, DEC ULTRIX, HP-UX, and LINUX) as well as Windows (NT, 3.1, 95); a Macintosh version is expected by the time this paper is published. VRweb source code (with copyright) is available free of charge for non-commercial use, providing a platform for research and experiment. Unlike other VRML viewers available in source code, VRweb does *not* require additional commercial libraries like OpenInventor or Motif, it is based entirely on freely available software components (see Section 5 for more detail about VRweb's software architecture). Figure 1 shows VRweb displaying Lightscape's [13] VRML model of the chamber of Jerusalem City Hall.

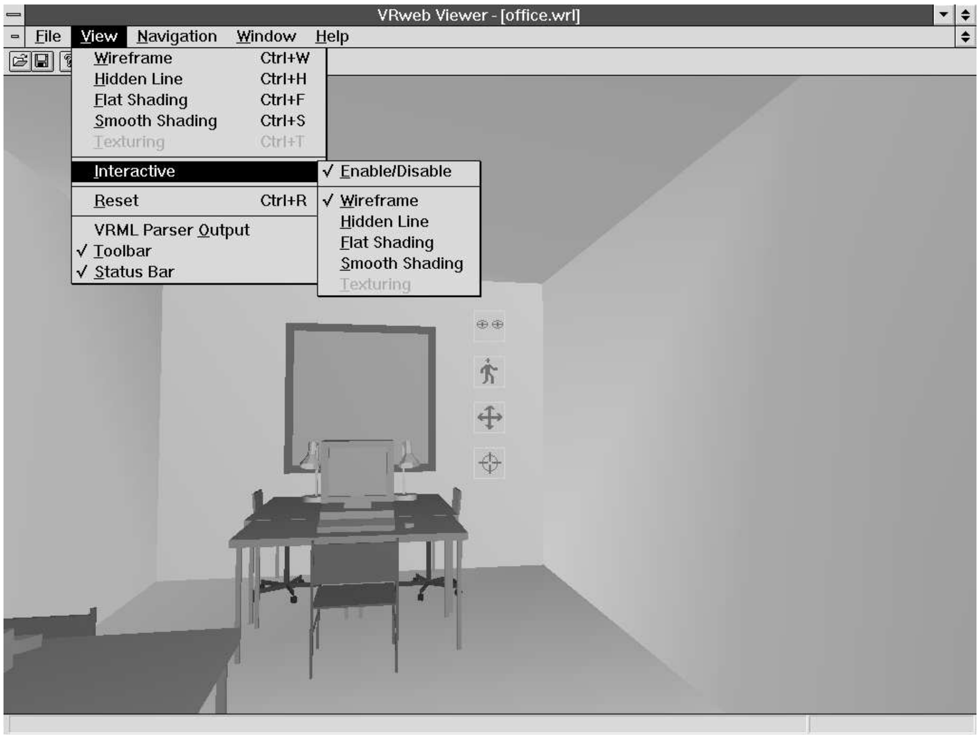


Figure 2: *VRweb for Windows in Heads-Up navigation mode. Icons are overlaid atop the viewing area for looking, walking, vertical/sideways motion, and point-of-interest navigation. The rendering mode may be set separately for faster interactive navigation.*

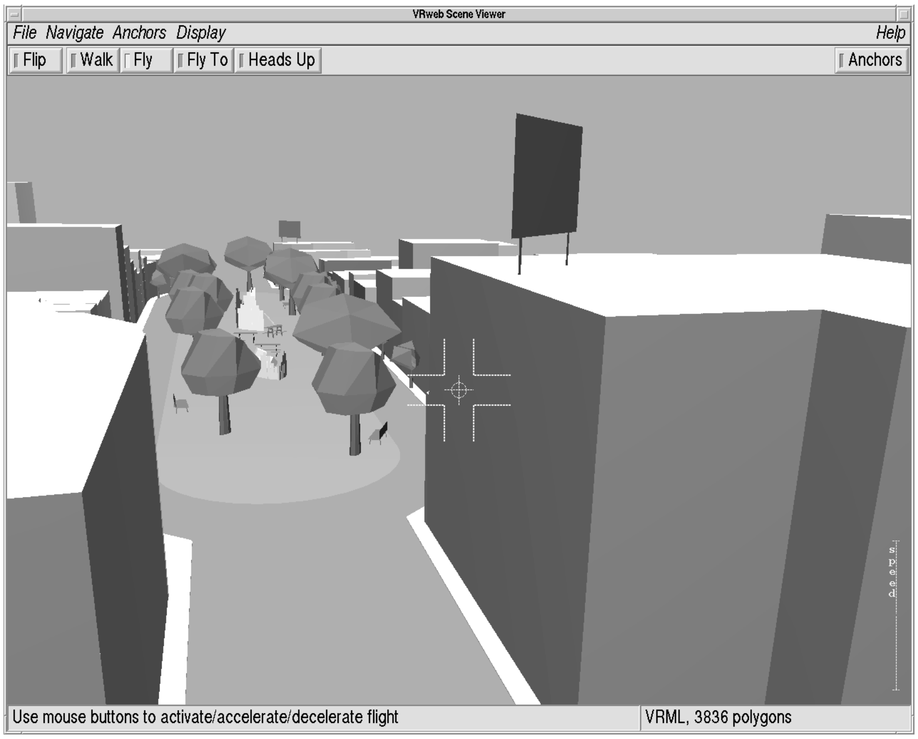


Figure 3: *VRweb for Unix in Fly Mode through VirtualSOMA. The speed of flight is overlaid in the bottom right corner of the viewing area, the direction of flight is determined by the position of the cursor relative to the central target crosshairs.*

## 3 The VRweb User Interface

The VRweb window is divided vertically into four areas: menu bar, tool bar, display area, and status bar. The menu bar provides access to the full functionality of VRweb, the tool bar and additional accelerator keys provide quick access to commonly used functions.

VRweb supports five rendering modes: wire frame, hidden line, flat shading, smooth shading, and texturing; the rendering mode specified in the VRML file can be overridden. A separate rendering mode may be specified for use during interactive navigation. For example, on a display without hardware graphics acceleration where frame rates are slow for textured models, it might be advisable to navigate in wire frame mode and see a textured version only in the pauses between interaction.

Navigation in 3D space is complicated by the need to control (at least) six degrees of freedom at the same time. VRweb does not assume the availability of any special 3D input device, such as a spaceball or position/orientation sensors, and provides a number of natural mappings for a standard 2D mouse. There are basically two classes of viewpoint control metaphor, depending on whether users wish to move the model itself or to move themselves through the model. Models of *objects* are typically moved and examined, models of *scenes* are typically navigated through (walking, flying, etc.). To cope with these different needs, VRweb provides five navigation metaphors.

*Flip* mode is used to examine an object, whilst the viewer remains stationary. The mouse buttons have assignments for translation, rotation, and zooming (the object).

The *Walk* metaphor is used to stroll through a 3D environment. Natural walking motion (forwards and backwards, possibly veering slightly to left or right) is assigned to the left mouse button. Complementary controls for vertical motion and side-stepping, and for turning the head are assigned to the middle and right mouse buttons respectively.

*Fly* is like piloting an aircraft. Flight direction is controlled by the position of the mouse cursor relative to the mid-point of the viewing window (denoted by a cross-hair). The left mouse button activates flight, the other two mouse buttons control acceleration and deceleration. The current direction and speed are indicated in an overlay atop the display area.

*Fly To* mode implements point-of-interest (POI) style navigation [15]. Here, the user first selects a point of interest somewhere in the model and is then able to perform controlled, logarithmic motion towards (and away from) the POI, approaching by the same fractional distance in each time step. Optionally, a rotational component can be activated (with the Shift key), which results in a final approach path to the POI head-on along the surface normal. This mode is very useful for examining details of a scene and complements other navigation metaphors like Walk or Flip, but is not sufficient as a navigation technique per se.

*Heads Up* is perhaps the easiest navigation mode for beginners, since its controls are clearly visible. Icons overlaid across the centre of the viewing window (like a pilot's heads-up display) symbolise the individual navigation tools: eyes to look around, a walking person for walking, crossed arrows for vertical and sideways motion, and a crosshair symbol to activate point-of-interest motion.

Should users ever get lost, a "Reset View" function is available to restore the initial view of the scene. A "Level View" function makes the current view horizontal. Other functions include opening local files, saving files, setting preferences, changing colours, etc. The current frame rate can be displayed in the status area. The Hyper-G version of VRweb also supports interactive, point-and-click link creation.

Figure 2 shows the VRweb user interface in Heads-Up navigation mode. Figure 3 shows a user is navigating in Fly mode through VirtualSOMA [23], the VRML model of the South of Market area of San Francisco.

## 4 VRweb as a Multi-System VRML Viewer

VRweb can either be used both as a standalone viewer to view local VRML files (e.g. before putting them onto a server) or in conjunction with a WWW, Hyper-G, or Gopher client to view VRML files on the Net. Depending on the particular configuration, VRweb either functions as a passive helper application unable to resolve external URLs or as a partner application connecting back to the particular browser to resolve URLs as the need arises (for example, when a link anchor is clicked). A bidirectional connection to the client is necessary to retrieve anchor destinations (WWWAnchor nodes) and inlined scenes (WWWInline nodes), since VRweb does not itself speak HTTP, Hyper-G, or Gopher protocol. A bidirectional connection is also useful for general browsing functionality, such as opening a document of arbitrary type, going back and forward in the history, retrieving help files, and setting hotlist entries.

### 4.1 VRweb and WWW

VRweb can use NCSA Mosaic [1, 19] to resolve WWWInline and WWWAnchor requests. The Common Client Interface (CCI) allows external applications such as VRweb to communicate with running sessions of NCSA Mosaic via TCP/IP (or OLE). VRweb uses Mosaic to resolve requests for VRML scenes from an httpd server, and to manage the display of non-VRML documents which might be referenced in a WWWAnchor node.

When VRweb is invoked (generally as a helper application, with a temporary file containing the scene file as a parameter) it makes a CCI connection to Mosaic, and requests that all documents with the VRML MIME type be sent directly to it. It also asks Mosaic to notify it any time a new anchor is activated. Finally, it asks Mosaic to give it the URL corresponding to the file name with which it was invoked, which is displayed to the user in the status box, and displays the scene.

Subsequent scenes are passed directly to VRweb through the CCI socket. Because VRweb is notified whenever an anchor is activated via Mosaic, it can determine any scene's corresponding URL and, depending on how it was requested, display it as either an inline node or an entirely new scene. Figure 4 shows VRweb running with NCSA Mosaic for Windows.

A similar mechanism is being investigated to implement inline and anchor requests over the Netscape client API.

### 4.2 VRweb and Hyper-G

Harmony is the Hyper-G client for Unix/X11. A number of navigational tools are provided, including a hierarchical overview (collection browser), a bidirectional link overview (local map), and a 3D information landscape. Harmony supports the whole range of Hyper-G features, including link creation in any document type. VRweb's communication with Harmony [3] goes through the Harmony Session Manager, which holds a connection to a Hyper-G server. The

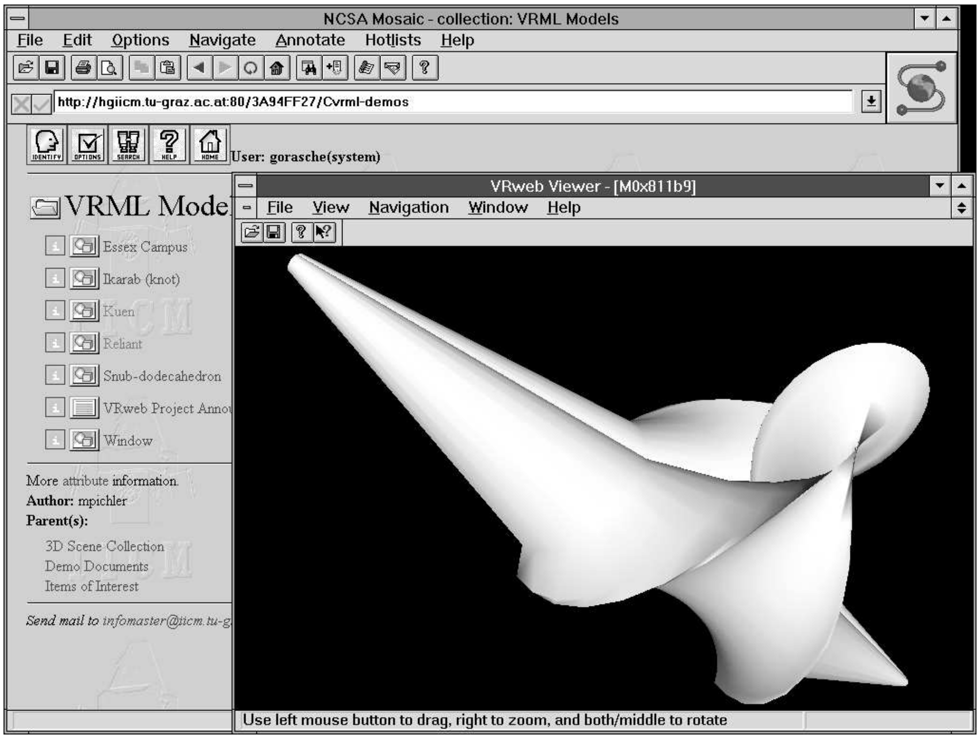


Figure 4: *VRweb as a helper application with Mosaic for Windows.*

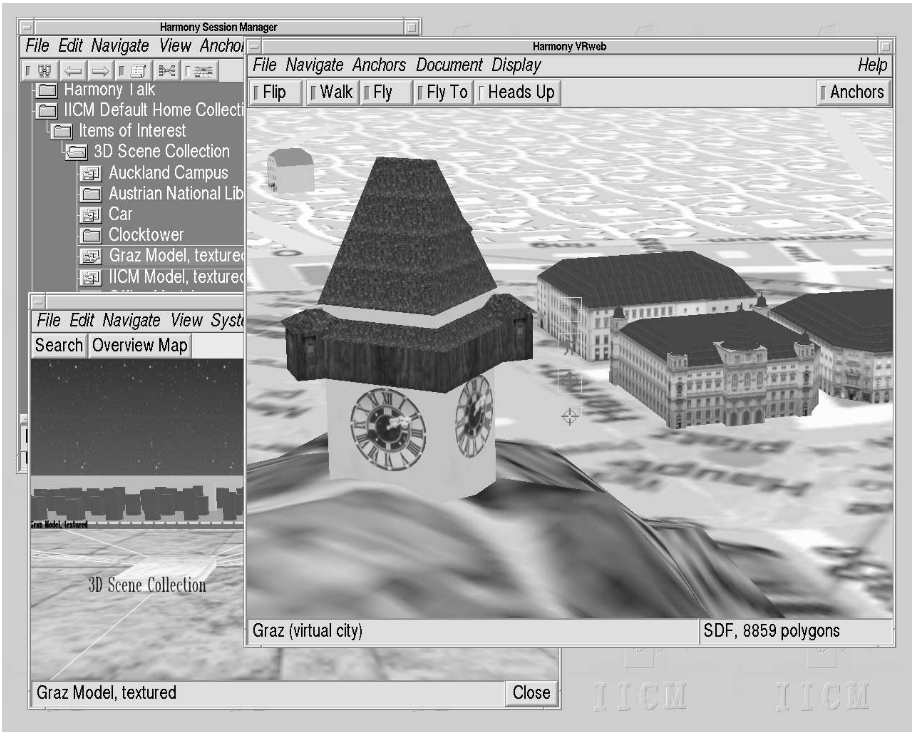


Figure 5: *VRweb and the Harmony client for Hyper-G. The user is navigating through a virtual model of the city of Graz. In the background are the Harmony Session Manager and 3D Information Landscape.*

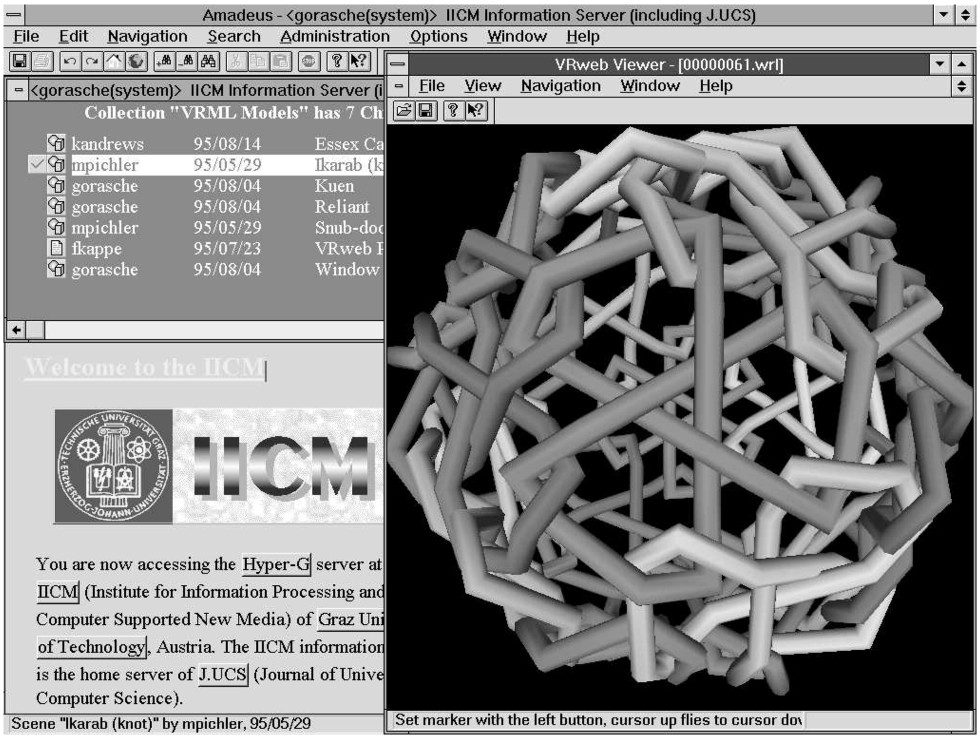


Figure 6: *VRweb and the Amadeus client for Hyper-G.*

Session Manager retrieves metadata and hyperlinks directly, documents themselves are retrieved along a direct connection between the document viewer (VRweb in the case of VRML scenes) and the Hyper-G server. All documents, whether they come from a Hyper-G server or through another protocol like WWW or Gopher, go via the Hyper-G server, which allows for a single document-viewer protocol and organisation-wide caching of documents. Figure 5 shows Harmony and VRweb with a virtual tour through through the city of Graz.

Amadeus is the Windows client for Hyper-G. Besides viewing all types of Hyper-G media it provides easy to use authoring facilities using standard Windows applications and file formats. In the Windows environment all communication with the server is done by Amadeus, the Hyper-G client for Windows. It provides a link list, holding all anchors for this document, which is generated at document selection time. Inline scenes are handled by Amadeus using asynchronous requests. Link creation is done the same way as for other media types, e.g. text, images, PostScript. Figure 6 shows Amadeus and the VRweb for Windows.

Even though links in documents on a Hyper-G server are stored in a link database separate from the documents themselves, link anchors contained within VRML files residing on other servers are respected by VRweb, so as to retain compatibility with the rest of the Web. In the other direction, VRML files on a Hyper-G server have their links merged in on-the-fly, when being accessed from a WWW client.

### 4.3 VRweb and Gopher

VRweb is in the process of being integrated into Gopher clients, including the new three-dimensional GopherVR. Figure 7 shows VRweb running with GopherVR.

## 5 VRweb's Software Architecture

Figure 8 shows the basic software architecture of VRweb. The WWW, Gopher, or Hyper-G client communicates via an interface layer with VRweb. The arrows indicate the flow of data and control signals. The VRML data stream is parsed and an internal data structure constructed. Commands for inline scenes and anchor activations are initiated by the link management module and passed on to the clients for retrieval. The rendering component manages user input and visualises the data structure. Graphics output is done via an abstract interface (GE3D) to one of several underlying graphics libraries.

For parsing the VRML input stream, VRweb utilises the freely available QvLib parser by Paul Strauss of Silicon Graphics. It supports VRML to the extent of the 1.0 specification and was extracted from the Inventor library. The parser has object-oriented structure and is implemented in C++. Each node and data type of VRML corresponds to a class, which has generic methods to read the data. Methods for rendering and selecting objects have been added for use with VRweb.

To keep VRweb source code independent from any particular graphics library, an abstract interface layer called GE3D (Graphics Engine for 3D) is placed between the rendering code and the graphics library. GE3D provides functionality at a slightly higher level of abstraction than, say, OpenGL [20] (a single GE3D call typically resulting in several lower-level OpenGL calls). For example, GE3D has single functions for geometric transformations, drawing polyhedra (a set of faces with related properties), defining light sources, etc. Interface functions to window systems, like window mapping and event handling, are kept out of GE3D, so as to achieve a high level of portability. These routines

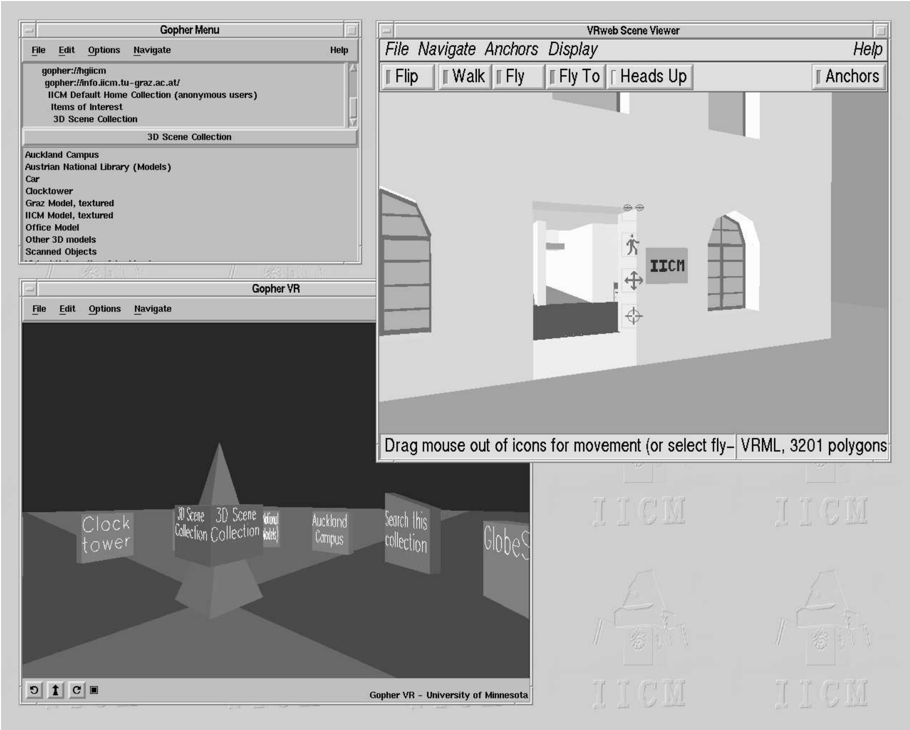


Figure 7: *VRweb running alongside GopherVR.*

are handled in a platform-specific manner within the application.

GE3D is currently implemented for four OpenGL-like graphics libraries: OpenGL, IrisGL, Mesa, and VOGL (an earlier version of GE3D was also ported to Hewlett-Packard's Starbase). OpenGL is the most prominent 3D graphics API, a vendor-neutral standard endorsed by many major suppliers. Its functions are at a relatively low level and designed for hardware acceleration. Its predecessor, GL (now referred to as IrisGL), was Silicon Graphics' original proprietary 3D graphics API. Mesa [22] is a freely available graphics library with an API very similar to that of OpenGL. Mesa maps graphics calls to both the X-protocol and to Windows, allowing 3D graphics rendering without any special hardware. Another library with similar characteristics is VOGL: its API is similar to IrisGL, but it lacks functions for hidden surface elimination and smooth shading in its original implementation.

## 5.1 VRweb for Unix/X11

The Unix version of VRweb evolved from the Harmony 3D Scene Viewer for Hyper-G. The design follows object-oriented principles, and is implemented in C++. The base class is the Scene which provides abstract methods for loading, drawing, picking, link management, and selection. Data management, including parsing and traversal to implement the functions, is handled by a class dependent on the scene file format.

For VRML scenes, the data structure built by the browser is kept and has to be traversed for drawing. As the structure of VRML is very much tailored to the capabilities of OpenGL, the mapping of VRML primitives to the drawing commands of GE3D is straightforward. Using the Inventor library for this purpose would have been easier, but would also have limited the availability of the viewer to platforms on which Inventor is available. The approach taken allows for the optimisation and tuning of the graphics library to the scope of VRML compared to the more general-purpose Inventor library.

Atop this layer, one class is responsible for user interaction. For the X11 version of VRweb, the InterViews [14] toolkit and a library of widgets built on top of it are used to provide menus and other user interface elements. All events, like keystrokes, mouse clicks and movements, are handled and result in function activation or are mapped onto elementary navigational functions provided by the Camera class, depending on the current navigation metaphor. This modular approach allows for a relatively simple port to another window system or platform, as was done for MS-Windows with Visual C++.

The interface to WWW, Gopher, and Hyper-G clients is encapsulated in a separate module. Although they vary in their functionality and capabilities, the different clients are accessed via a common abstract interface.

## 5.2 VRweb for Windows

The design goal of the Microsoft Windows version of VRweb was to have a single source tree and one executable for each platform (Windows 3.x, Windows 95, Windows NT on Intel, Alpha, PowerPC, and MIPS). Therefore, VRweb for Windows is based on Microsoft's WIN32s 32-bit call interface. Using WIN32s has a number of other advantages over the 16-bit Windows programming interface, such as its flat memory model, higher stability, and better device performance.

To render the three dimensional scenes we used OpenGL at a low level and the GE3D library at a higher level, thus

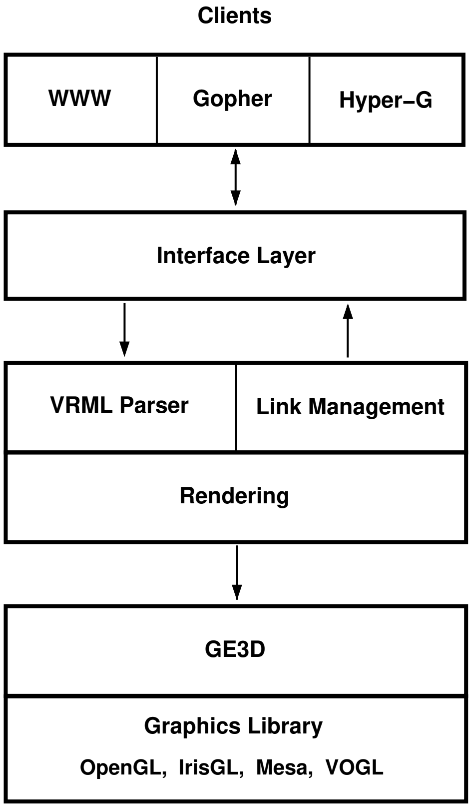


Figure 8: *VRweb's software architecture.*

staying compatible with the UNIX versions of VRweb. Depending on the system capabilities, either original Microsoft OpenGL (for Windows NT and Windows 95, possibly with hardware acceleration) and/or the public domain libraries Mesa and VOGL (software-only rendering) may be used. The University of Minnesota extended the VOGL library with a binary space partitioning (BSP) algorithm for faster hidden surface elimination (for details on BSP see [11]). The Mesa library was enhanced by an additional rendering mode for the Windows driver using RGB colours and a 256 colour palette with index addressing. This mode uses the WING library which was designed especially for fast bitmap display.

The architecture of VRweb for Windows is based on the Windows Multiple Document Interface (MDI) and the Microsoft Foundation Classes (MFC). This makes integration into other environments very easy. The VRweb viewer is fully integrated into Amadeus, the Hyper-G client for Windows. It will also interface with Mosaic and the Gopher client. A stand-alone version is available using the same code but not providing any link or inline facilities. Hooks for these functions are provided in the source code, so anyone can implement them for a proprietary target platform. The user interface is similar to that of VRweb for X11, but has Windows look and feel.

### 5.3 VRweb for Macintosh

A port of VRweb for the Macintosh is underway and is expected to be available by the time this paper is published.

## 6 Availability

VRweb is available both in binary and in source code for a variety of platforms. The source code is copyrighted, but is freely available for non-commercial use (see the copyright notice with the distribution for full details). VRweb code is available by anonymous ftp from:

```
ftp://ftp.iicm.tu-graz.ac.at/pub/Hyper-G/VRweb
ftp://ftp.ncsa.uiuc.edu/Hyper-G/VRweb
ftp://boombox.micro.umn.edu/pub/Hyper-G/VRweb
```

as well as from a number of other mirror sites. There is a VRweb technical home page with up-to-date technical information at:

```
http://info.iicm.tu-graz.ac.at/Cvrweb
gopher://info.iicm.tu-graz.ac.at/
hyperg://info.iicm.tu-graz.ac.at/vrweb
```

The VRweb mailing list is the place for questions, suggestions, and discussion of VRweb. To subscribe to the mailing list, send email to:

```
listproc@iicm.tu-graz.ac.at
```

with message body (no subject needed):

```
subscribe vrweb FirstName LastName
```

To unsubscribe, mail "`unsubscribe vrweb`" to the same address. To send mail to all members of the list, simply compose your message or question and send it to:

```
vrweb@iicm.tu-graz.ac.at
```

## 7 Concluding Remarks

In order to firmly establish VRML as *the* standard for three-dimensional scenes on the Internet, it is not sufficient to have a good language specification. Viewers capable of many protocols and implemented on many platforms are necessary. Another crucial point is the free availability of source code for non-commercial organisations to allow research and experimentation with this new type of media. These were the design goals of VRweb.

Future areas of research will include the incorporation of collision detection for more realistic navigation, visibility preprocessing to speed up rendering (especially on slower machines), and editing functionality for home space construction. We hope that the availability of an open development platform will contribute significantly to the evolution of the VRML standard.

## 8 Acknowledgements

The authors are grateful to Brian Paul for helpful discussions regarding the Mesa library and would like to thank all of the members of the Hyper-G, NCSA, and Gopher teams for their dedication and fruitful collaboration on the VRweb project.

The IICM work on VRweb is partially financed by the Anniversary Fund of the Austrian National Bank, under project number 5334, "Virtual Information Spaces".

## REFERENCES


[1] Marc Andreessen. NCSA Mosaic Technical Summary. Available at ftp://ftp.ncsa.uiuc.edu/Mosaic/Papers/mosaic.ps.Z, May 1993.

[2] Keith Andrews. Using 3D Scenes as Hypermedia Nodes. Poster at ECHT'92, Milan, Italy, December 1992.

[3] Keith Andrews and Frank Kappe. Soaring Through Hyperspace: A Snapshot of Hyper-G and its Harmony Client. In Wolfgang Herzner and Frank Kappe, editors, *Proc. of Eurographics Symposium on Multimedia/Hypermedia in Open Distributed Environments*, pages 181–191, Graz, Austria, June 1994. Springer.

[4] Keith Andrews, Frank Kappe, and Hermann Maurer. Hyper-G: Towards the Next Generation of Network Information Technology. *Journal of Universal Computer Science*, 1(4):206–220, April 1995. Special Issue: Selected Proceedings of the Workshop on Distributed Multimedia Systems, Graz, Austria, Nov. 1994. Available at http://info.iicm.tu-graz.ac.at/Cjucs_root.

[5] Keith Andrews, Frank Kappe, and Hermann Maurer. Serving Information to the Web with Hyper-G. *Computer Networks and ISDN Systems*, 27(6):919–926, April 1995. Proc. Third International World-Wide Web Conference, WWW'95, Darmstadt, Germany.

[6] Keith Andrews and Michael Pichler. Hooking Up 3-Space: Three-Dimensional Models as Fully-Fledged Hypermedia Documents. In *Proc. of East-West International Conference on Multimedia, Hypermedia, and Virtual Reality (MHVR'94)*, pages 11–18, Moscow, Russia, September 1994.

[7] Gavin Bell, Anthony Parisi, and Mark Pesce. The Virtual Reality Modeling Language – Version 1.0 Specification. Available at http://vrml.wired.com/vrml.tech/vrml10-3.html.

[8] Tim Berners-Lee, Robert Cailliau, Ari Luotonen, Henrik Frystyk Nielsen, and Arthur Secret. The World-Wide Web. *Communications of the ACM*, 37(8):76–82, August 1994.

[9] Tim Berners-Lee and Dan Conolly. Hypertext Markup Language (HTML). Available at http://info.cern.ch/hypertext/WWW/Markup/HTML.html, June 1993.

[10] Center for Advanced Studies, Research and Development. i3D - A New Dimension to Hypermedia. Available at http://www.crs4.it/~3diadm/.

[11] James D. Foley, Andries van Dam, Steven K. Feiner, and John F. Hughes. *Computer Graphics: Principles and Practice*. Addison-Wesley, Reading, Massachusetts, second edition, 1990.

[12] InterVista Software, Inc. WorldView. Available at http://www.webmaster.com/vrml/.

[13] Lightscape Technologies, Inc. Lightscape Visualization System. See http://www.lightscape.com/.

[14] Mark A. Linton, John M. Vlissides, and Paul R. Calder. Composing User Interfaces with InterViews. *IEEE Computer*, 22(2):8–22, February 1989.

[15] Jock D. Mackinlay, Stuart K. Card, and George G. Robertson. Rapid Controlled Movement Through a Virtual 3D Workspace. In *Proc. SIGGRAPH '90, Dallas, Texas*, pages 171–176, New York, August 1990. ACM.

[16] Mark P. McCahill and Farhad X. Anklesaria. Evolution of Internet Gopher. *Journal of Universal Computer Science*, 1(4):235–246, April 1995. Selected Proceedings of the Workshop on Distributed Multimedia Systems, Graz, Austria, Nov. 1994. Available at http://info.iicm.tu-graz.ac.at/Cjucs_root.

[17] Mark P. McCahill and Thomas Erickson. Design for a 3D Spatial User Interface for Internet Gopher. In *Proc. of ED-MEDIA 95*, pages 39–44, Graz, Austria, June 1995. AACE.

[18] Tamara Munzner, Daeron Meyer, Paul Burchard, and Stuart Levy. WebOOGL: Integrating 3D graphics and the Web, The Geometry Center, University of Minnesota. Available at http://www.geom.umn.edu/software/weboogl/.

[19] National Center for Supercomputing Applications. NCSA Mosaic Home Page, University of Illinois. See http://www.ncsa.uiuc.edu/SDG/Software/Mosaic/NCSAMosaicHome.html.

[20] Jackie Neider, Tom Davis, and Mason Woo. *OpenGL Programming Guide*. Addison-Wesley, 1993.

[21] Paper Software, Inc. WebFX. Available at http://www.paperinc.com/webfx.html.

[22] Brian Paul. The Mesa 3-D Graphics Library, Space Science and Engineering Center, University of Wisconsin. Available at http://www.ssec.wisc.edu/~brianp/Mesa.html.

[23] Planet 9 Studios. VirtualSOMA. Available at http://www.hyperion.com/planet9/vrsoma.htm.

[24] San Diego Supercomputer Center. VRML Repository. Available at http://www.sdsc.edu/vrml/.

[25] San Diego Supercomputer Center. WebView. Available at http://www.sdsc.edu/EnablingTech/Visualization/vrml/webview.html.

[26] Silicon Graphics, Inc. WebSpace. Available at http://www.sgi.com/Products/WebFORCE/WebSpace/.

[27] Josie Wernecke. *The Inventor Mentor*. Addison-Wesley, 1994.