virtual reality transfer protocol (vrtp) Design Rationale

Don Brutzman, Mike Zyda and Kent Watsen
Code UW/Br, CS/Zk, CS/Wa
Naval Postgraduate School
Monterey California 93943-5000 USA
408.656.2149 voice, 408.656.3679 fax
brutzman@nps.navy.mil zyda@siggraph.org
watsen@mbay.net

Mike Macedonia
Fraunhofer Center for Research in
Computer Graphics Inc.
167 Angell Street
Providence Rhode Island 02906 USA
401.453.6363 voice, 401.453.0444 fax
macedoni@crcg.edu

http://www.stl.nps.navy.mil/~brutzman/vrtp/vrtp_design.ps

Abstract

The capabilities of the Virtual Reality Modeling Language (VRML) permit building large-scale virtual environments (LSVEs) using the Internet and the World Wide Web. However the underlying network support provided by the hypertext transfer protocol (http) is insufficient for LSVEs. Additional capabilities for lightweight peer-to-peer communications and network monitoring need to be combined with the client-server capabilities of http. To accomplish this task, we present a detailed design rationale for the virtual reality transfer protocol (vrtp). vrtp is designed to support VRML in the same manner as http was designed to support HTML. Since vrtp must be highly optimized on individual desktops and across the Internet, a Cyberspace Backbone (CBone) is also needed for vrtp development and testing. vrtp appears to be a necessary next step in the deployment of all-encompassing interactive internetworked 3D worlds.

1 Overview and Motivation

It is now possible to construct large-scale virtual environments (LSVEs) comprised of internetworked graphics using the Virtual Reality Modeling Language (VRML) version 2.0 (Carey 96). Such online interactive VRML worlds require diverse and powerful network capabilities to support user demands. The hypertext transfer protocol (http) (Berners-Lee 96) provides effective client-server support for many types of information transfer over the Web (Berners-Lee 94). However other types of network connections are also needed; examples include multicast streaming of real-time audio, video and entity behavior data. Furthermore these many capabilities are directly required by all participants in LSVEs, rather than indirectly accessed via separated client and server platforms.

We propose a new protocol at the application layer, the virtual reality transfer protocol (vrtp). vrtp is intended to significantly extend http in order to provide full support for LSVE requirements by enabling a spectrum of functionality between client-server and peer-peer, all optimized on local desktop computers and across the global Internet.

Years of effort on widely distributed interactive 3D graphics applications have clearly identified networking issues as the critical bottlenecks preventing the creation of LSVEs. Those years have produced numerous important virtual reality technologies (Durlach 95) (Zyda 93). We believe the time is now right to put all of the network pieces together with all of the graphics pieces. vrtp is intended to accomplish this challenge.

This paper presents the planned architecture of vrtp. Section 2 presents pertinent background networking concepts: key network capabilities for LSVEs, multicast and exploiting reality, the Distributed Interactive Simulation (DIS) protocol and the hypertext transfer protocol (http). Section 3 summarizes network considerations for the Virtual Reality Modeling Language (VRML). Section 4 shows how an unrecognized spectrum of functionality exists between client-server and peer-to-peer. Sections 5 and 6 define vrtp and describe vrtp components: client, server, peer-to-peer and network monitoring. Finally Section 7 explains why a Cyberspace Backbone (CBone) will eventually be needed for high-performance vrtp optimization and experimentation.

2 Background Networking Concepts

Of necessity a variety of related networking topics must be summarized as pertinent background for vrtp design. More detailed treatments are found in (Brutzman 95, 96b).

Four key network capabilities. Many types of information can be distributed in a large-scale virtual

environment (LSVE). It is important to be able to group and partition information transfer capabilities into a small set of well-defined primitives. By clearly stating network requirements, protocol design can efficiently and thoroughly support virtual world requirements. We have found that all types of information transfer can be effectively mapped into the categories listed in Figure 1 (Macedonia 94, 95) (Brutzman 95, 96b). The primary usefulness of this approach is to clarify core requirements which need to be well supported by the underlying global network for LSVEs.

Multicast and Exploiting Reality. Multicast is a crucial capability for scaling up network capabilities. Multicast packets have class D Internet Protocol (IP) addresses which permits individual packets to be routed to multiple recipients without duplication on individual LANs (Deering 89). Host machines must intentionally subscribe to a multicast address in order for incoming packets to be passed to the application. Thus multicast packets have particular strengths in minimizing bandwidth and minimizing processor cycles. Multicast capabilities are of fundamental importance when considering networked real-time streams (Macedonia 94).

Since multicast currently uses only the User Datagram Protocol (UDP) and not the Transport Control Protocol (TCP) of the IP suite, multicast streams are a connectionless "unreliable" service. Lost MBone packets stay lost. No setup is required, no acknowledgments are used, and no guarantee of delivery exists for this type of "best effort" service. Ordinarily this is a good thing for most real-time information streams (such as audio, video and behaviors) since it avoids delivery bottlenecks and unwanted overhead. Numerous researchers are experimenting with "reliable multicast" transport protocols which try to achieve a different balance between reliability and potential congestion, typically by gaining occasional retransmission benefits without unacceptable acknowledgment overheads (Crowcroft 96).

The Multicast Backbone (MBone) is one of the Internet's most interesting capabilities since it is used for distribution of live audio, video and other packets (such as DIS) on a global scale. In other words, the MBone interconnects the multicast capabilities of LANs across the Internet. MBone is a virtual network because it shares the same physical media as the Internet and is comprised of a network of routers (mrouters) that support global multicast. Furthermore it is possible to partition the LSVE communications space to exploit reality by assigning different communication channels to different multicast addresses, typically corresponding to geographic, temporal or functional areas of interest (AOI) (Macedonia 95, 96). These considerations for reducing bandwidth and processor cycles while achieving global connectivity are critical when scaling to arbitrarily large numbers of simultaneously interacting users on the network.

- **Light-weight Interactions**. Messages composed of state, event and control information as used in DIS Entity State PDUs or other behavior reports. Implemented using unicast or multicast. Complete message semantics are included in a single packet encapsulation without fragmentation. Light-weight interactions are received completely or not at all.

- **Network Pointers**. Light-weight network resource references, i.e. a global address space that can be multicast to receiving groups. Can be cached so that repeated queries are answered by group members instead of servers. Pointers do not contain a complete object as light-weight interactions do, instead containing only a reference to an object.

- **Heavy-weight Objects**. Large data objects requiring reliable connection-oriented transmission. Typically provided as a web query response to a network pointer request.

- **Real-time Streams**. Live video, audio, DIS behaviors, sequential graphics images or other continuous stream traffic that requires real-time delivery, sequencing and synchronization. Typically implemented using multicast channels.

Figure 1. Four key communications capabilities used in virtual environments (Macedonia 94, 95) (Brutzman 95, 96b).

Distributed Interactive Simulation (DIS) and Behavior Protocols. For the past seven years, a major focus in networked virtual environments has been the Department of Defense's Distributed Interactive Simulation (DIS) standard, also know as IEEE 1278. The DIS standard (IEEE 95) describes the packet format for 27 protocol data units (PDUs) specified for use in military combat simulations. Of these only a few have general applicability. We are most interested in the widely used Entity State PDU (ESPDU). Entity state in this context includes linear and rotational values for posture, velocity and acceleration in combination with efficiently designed algorithms for dead reckoning and track smoothing. The ESPDU is well suited for relaying physically based model information among numerous interacting participants in real time. Meanwhile the DIS specification has been frozen and DoD development work redirected toward a hybrid distributed client-server object broker model, the High-Level Architecture/Run-Time Infrastructure (HLA/RTI) (DMSO 96).

The IEEE DIS protocol is robust and effective in multiuser environments and can be suitable for game scenarios (Katz 96). However a two-year process was

2

typically required for PDU extensions and modifications to be standardized. We believe that a behavior protocol design-test-evaluate-repeat cycle must be interactive in order to achieve the efficiency and scalability improvements needed for true LSVEs. In a separate effort, we are working on a more general solution to extend the advantageous capabilities of DIS ESPDUs using a "dial-a-behavior protocol" approach. Our current design includes an abstract PDU format grammar and Java applets that are able to switch parsed formats on the fly. We expect this will enable useful wide-area behavior protocol testing suitable for inclusion in vrtp. Much lighter and more efficient ESPDUs appear to be possible. A simplified data flow diagram in Figure 2 shows how networked DIS ESPDUs can be processed by a Java applet and passed to a VRML 2.0 scene.

hypertext transfer protocol (http). The hypertext transfer protocol (http) (Berners-Lee 96) used for most Web interactions evolved first by efficiently combining capabilities of precursor protocols (ftp and gopher) and then optimizing performance on client and server machines. Many other capabilities have since been added, but fundamentally http provides a purely client-server relationship: a user can push on a Web resource and get a response, but there are no mechanisms for information sources to independently push back at candidate receivers. Inadequate support is available for light-weight interactions and real-time streams required in Figure 1. The limitations of http are widely known but current http next generation (http-ng) efforts appear focused on optimizing and incrementally extending existing functionality (W3CHTTP 96). Thus even next-generation http appears unlikely to support the LSVE networking requirements identified here. We see http not as an end goal but rather as an important component of a more flexible application layer functionality to be provided by vrtp.

3 Virtual Reality Modeling Language

VRML has extended the Web to three spatial dimensions (3D). Key contributions of the VRML 1.0 standard (Bell 95) were a core set of object-oriented graphics constructs augmented by hypermedia links, all suitable for cross-platform 3D scene generation. VRML 2.0 (Carey 96) extends 3D scene interactivity by defining language interfaces for JavaScript (Netscape 96) and Java (Sun 96). Of these two languages, only Java permits direct network connections using sockets. VRML-based 3D Web browsers are usually embedded inside 2D browsers or are launched as helper applications when connecting to a 3D site. The typical network interaction model for VRML Web browser scenes remains client-server http via the browser, similar to non-3D Web browsers and helper applications. Each layer

of software can impose additional performance overhead on network communications.
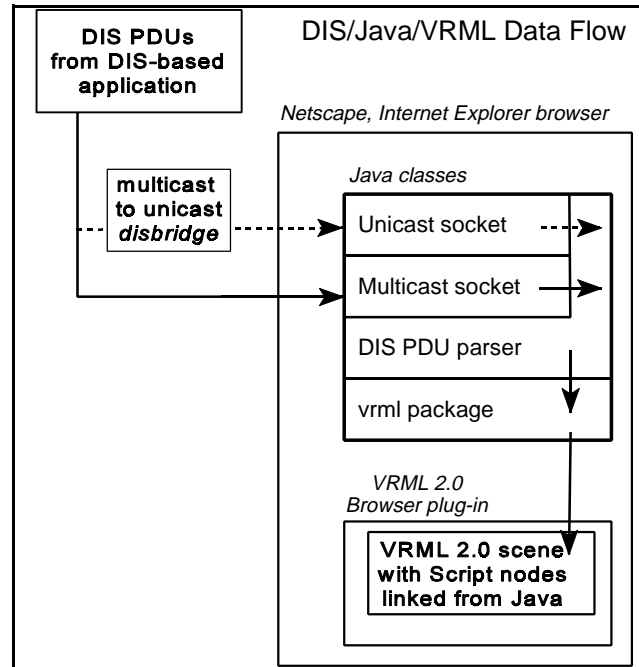

Figure 2.   Example data flow from DIS ESPDUs through Java applet to VRML scene.

Complex issues surrounding real-time animation in VRML 2.0 include implementation of entity behaviors, user-entity interaction and inter-entity coordination. Numerous factors are involved. Many excellent interactive worlds have already been produced using the VRML 2.0 specification (Nadeau 96). However, in order for VRML to scale to many simultaneous users, peer-to-peer interactions are necessary in addition to client-server query-response. Dynamic scene changes will need to be stimulated by a variable combination of scripted actions, message passing, user commands or entity behavior protocols (such as DIS). Thus the forthcoming VRML behaviors standardization will need to simultaneously provide simplicity, security, scalability, generality and open extensions. For these many reasons we expect that networked VRML scenes will demand significant real-time streaming in addition to http-based client-server interactions. This expected shift in network usage by VRML-based LSVEs is a major motivation for development of vrtp.

4 Spectrum of Network Functionality

During many years of dialog about network requirements for LSVEs, we found that most discussions often ended up with the same question: was the approach in question based

on client-server or peer-to-peer? This was a frustratingly simplistic bottom line that often led to unsatisfactory answers. Indeed most application-layer network architectures do take a bipolar view that one or the other approach (client-server or peer-to-peer) must be chosen. This pattern is exemplified by independent configuration requirements and little functional overlap among client-server Web-based applications and peer-to-peer MBone applications respectively. However recent work such as federated http servers at the National Center for Supercomputer Applications (NCSA) and agent-based area-of-interest managers (AOIMs) for partitioning multicast behavior streams (Macedonia 95) indicate that intermediate functionality (somewhere between client-server and peer-to-peer) may be essential for scaling up. Client-server, peer-to-peer and in-between applications are all possible as shown in Figure 3.
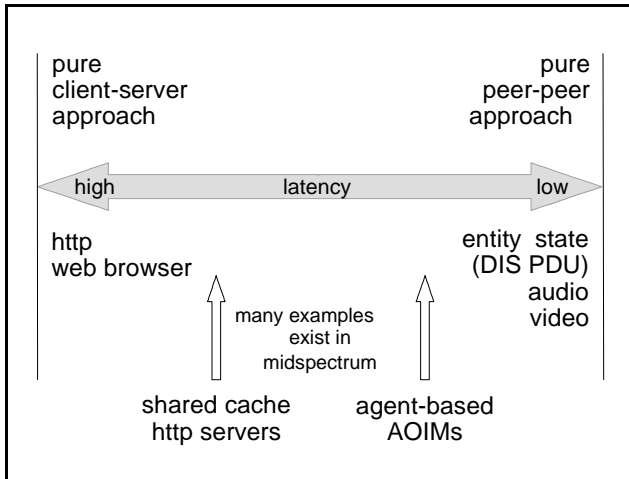
Figure 3. A spectrum of functionality exists between client-server and multicast peer-to-peer. Examples in midspectrum include multiple shared-cache http servers and agent-like entities such as area-of-interest managers (AOIMs).
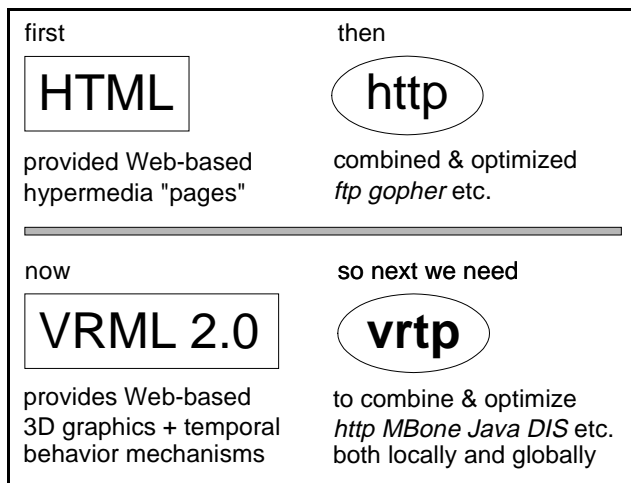
Users and entity applications contribute both static content and dynamic streams within LSVEs. Entity interactions will occur in multiple ways across a spectrum ranging from pure peer-to-peer through pure client-server. Thus entities participating in LSVEs simultaneously need to act as clients, servers and peers. A principal design goal of vrtp is to provide this range of support efficiently. Client, server, peer-to-peer and in-between solutions must all be provided at once by the desktop vrtp architecture.

5 vrtp defined

HTML required http, now VRML requires vrtp. The two component technologies that have enabled rapid exponential growth of the Web are HTML and http. VRML is now extending the flat 2D functionality of HTML into four dimensions: 3D geometric space plus temporal behaviors. We believe that a new network protocol is needed to extend the Web into fully interactive 3D LSVEs: vrtp, the virtual reality transfer protocol. vrtp is needed to support multiple simultaneous users of shared large-scale Web-based interactive 3D graphics, just as http was needed to support large-scale use of HTML. vrtp is an optimized combination of essential and available (but currently separate) components. vrtp will efficiently combine multiple existing dissimilar protocols (such as http, DIS, multicast streaming, Java agents, network monitoring etc.) just as http efficiently combined multiple pre-existing dissimilar protocols. This relationship is shown in Figure 4.

Networked interactive 3D graphics rendering is an essential human interface technology. VRML graphics primitives provide the raw material for anyone to put together a sophisticated 3D scene viewable on any platform. Addition of VRML 2.0 script connections via Java, JavaScript and links to behavior protocols (such as DIS) enables animation of scenes both locally and globally. Internet application layer support for VRML (currently provided by http) is insufficient for the demands about to be made by users who will start connecting every imaginable geometry to every imaginable information stream. vrtp will efficiently provide all four essential network capabilities identified in Figure 1 earlier.

Scaling up. Eventually we expect that VRML/vrtp applications will provide exemplars which evolve into the next-generation Web, where LSVEs are all-inclusive supersets of all media, protocols, behaviors, interfaces and content. Using VRML, vrtp, existing Internet infrastructure and Web-scalable approaches, we intend to help establish the technical foundation and underlying network mechanisms for cyberspace. Such research products will eventually connect everybody to everything. vrtp will enable the underlying network infrastructure for implementing distributed LSVEs as predicted by "Gibsonian" cyberspace, i.e. the futuristic vision of unconstrained virtual reality prevalent in popular science fiction (Gibson 84). In the case of vrtp, we are designing with science and without fiction. An open architecture will use standards-based public-domain software and repeatable exemplar implementations, all taking advantage of the sustained exponential growth of internetworked global information.

4

first	then
HTML	http
provided Web-based hypermedia "pages"	combined & optimized *ftp gopher* etc.
now	so next we need
VRML 2.0	vrtp
provides Web-based 3D graphics + temporal behavior mechanisms	to combine & optimize *http MBone Java DIS* etc. both locally and globally

Figure 4. Just as http was designed to support the hypermedia networking needs of HTML, vrtp is designed to support the large-scale virtual environment (LSVE) networking needs of VRML.

LSVEs require scaling up graphics and networking in every direction without bound. Adding entities and content to LSVEs must be no harder than contributing other forms of content to the Web. In other words, adding interactive 3D entities must be no harder than building a home page. Global participation in LSVEs must be feasible without single-server or single-virtual-circuit bottlenecks. vrtp is a necessary enabling technology for cyberspace, just as http was a necessary enabler for the Web. http takes a paired client-server view of interactions that essentially ignores global network considerations. Unlike http, vrtp must be designed with regional and global internetworks in mind.

Geography-based protocol extensions may also be necessary to permit scaling up in a navigable fashion. An open research issue is whether "real estate" assignments are needed to permit consistent, compatible and unmediated links among extremely large numbers of virtual 3D spaces. The assumption here is that the real world is the best model for organizing the massive volume of content in virtual worlds.

6 vrtp Components

Client, server, peer-to-peer, monitoring. The design of vrtp is not driven by abstract network capabilities but rather by the needs of highly populated VRML-based virtual environments. Players in LSVEs will act as clients when examining other world databases, will act as servers when making their own local world-view models available, and will act as peers when participating among large groups of active entities that scalably exchange behaviors through logically partitioned multicast channels. These client/server/peer capabilities must all be present on individual users' desktops and are not split across separate machines (as is currently required by the http client-server approach). Finally, since vrtp must simultaneously interoperate on multiple user desktops distributed across the Internet, diagnosis and correction of network problems is essential for practical usability and protocol optimization. Thus network monitoring capabilities are a fourth vrtp component. Figure 5 shows how these four components are used from the perspective of the LSVE participant's desktop.

Software integration. Example software for all four vrtp components exists and can be integrated to build vrtp. Widely used public-domain client, server, peer-to-peer and monitoring code will be utilized whenever possible. We hope to deploy vrtp on all common computer architectures. Our approach is to take advantage of open source code distributions while avoiding proprietary ownership issues which hamper open participation. In this way we expect to build a public domain reference version of vrtp. We expect that commercial versions of vrtp will also be produced to provide greater performance and user support.

Client components. The client component of vrtp is reasonably straightforward since numerous Web browsers exist. So much rapid work is ongoing in browser development that duplication of effort is counterproductive. The most likely client solution for vrtp is to provide efficient hooks to commercial Web browser application program interfaces (APIs) so that vrtp client capabilities track with browser capabilities and their associated plug-in programs. Public domain browsers are also candidates, such as *Arena* and *Amaya* by the World Wide Web Consortium (W3C 96) or *Mosaic* by NCSA (NCSA 96). Mime types permit customized handoffs to appropriate viewers for arbitrary data types, thus permitting extensibility regardless of browser types (Borenstein 93). It will also be important for vrtp to include a Java virtual machine (Sun 96) for a variety of reasons: the Java API for VRML (Carey 96), on-the-fly behavioral dial-a-protocol updates, enabling distributed network monitoring programmability, and providing automatic vrtp upgrade capabilities.

Server components. The server component of vrtp is also relatively straightforward to include, due to much work accomplished in this area. We expect to use a proven http server as the baseline software for vrtp. Apache is an excellent public domain candidate (Apache 96). Server extensibility via Common Gateway Interface (CGI) scripting using *perl* and other languages will also be valuable to ensure all commonly used server capabilities are present.

5

What does the networked LSVE desktop look like?
Participants need multiple capabilities simultaneously:

- **client**
    - looking at many other people's worlds

- **server**
    - showing other people your personal, changeable and persistent world

- **peer-to-peer**
    - send and receive multiple real-time streams of audio, video and behavior interactions
    - rapidly join or leave multicast peer groups

- **network monitoring**
    - everything simply "works" for ordinary users
    - must diagnose and correct network problems
    - vrtp optimization on desktop, across Internet

Figure 5. vrtp components from desktop perspective: client, server, peer-to-peer and monitoring.

Peer-to-peer components. Embedding efficient native multicast support is the first peer-to-peer capability to be added to vrtp. Multicast Backbone (MBone) connectivity is also essential for scalable distributed network solutions (Macedonia 94). Including up-to-date MBone applications for audio, video, whiteboard, DIS and other behavior protocols will provide a broad set of highly efficient cross-platform solutions for streaming data. Reading and creating global session directory advertisements is another important native capability. Finally we expect that ongoing work in "reliable multicast" protocols and resource reservation will present other peer-to-peer transport protocol possibilities besides connectionless UDP (Crowcroft 96).

Monitoring components. Scaling up to arbitrarily large groups of users means that the underlying network infrastructure must be robust. Optimizing vrtp on local machines and across the global Internet means that measurements must be able to accurately determine when protocol changes actually improve performance. Each of these cases implies that network monitoring is needed. Currently Internet-wide network monitoring is neither frequently performed nor well analyzed; much work needs to be done in this area (Paxson 96a, 96b). For example, (Paxson 96b) reports that over a ten-year period, approximately 14,000 papers were published relating to network queuing theory while about 44 papers were published on network monitoring. Fortunately a variety of new capabilities are becoming possible. Clock

synchronization using the Simple Network Time Protocol (SNTP) (Mills 95) will permit accurate test measurements and meaningful use of time stamps among LSVE participants. Query capabilities using the Simple Network Monitoring Protocol (SNMP) (Case 96) will permit open interoperable communications methods for site-to-site monitoring queries, and inclusion of multicast trace tools such as *mtrace* and *mrinfo* will enable MBone connectivity evaluation and diagnosis (Edwards 96) (Erdogan 96). Network monitoring, diagnosis and correction appears to be a rich and important area for future work.

7 Cyberspace Backbone (CBone)

Need for OpenVE network recognized. In the 1995 report entitled Virtual Reality: Scientific and Technological Challenges (Durlach 95), the National Research Council recommends establishment of a nationwide, open virtual environment (OpenVE) research network. The purpose of that network is to allow universities, academia and government agencies to examine the technological issues involved with networking LSVEs. Proposed work includes the study of design issues behind virtual environment protocols, the development of tools for rapid development of virtual environment protocols, and the development of networked software architectures capable of supporting LSVEs. A cyberspace LSVE network is an inevitable development as access to practically all technologies and disciplines converges via the Internet and begin to take advantage of the capabilities offered by VRML.

CBone. In order to effectively test and optimize vrtp, a network with moderately high bandwidth and guaranteed low latency is needed. We intend to establish the high-speed OpenVE network described in the NRC report. We call this network the Cyberspace Backbone (CBone) (Brutzman 96a) in homage to the Multicast Backbone (MBone) currently used for VE network testing. Although much is currently possible using the MBone, a more carefully engineered backbone service that includes dedicated multicast capabilities, dedicated bandwidth and dedicated latency is needed for the experimental development of LSVEs. We view the CBone as an essential step for experimentally optimizing vrtp to work efficiently despite the variety of latency and congestion conditions experienced across the Internet.

Internet Protocol Next Generation (IPng, IPv6). The basic infrastructure upon which vrtp and CBone will operate is changing. From (Nierle 96):

"The next-generation Internet Protocol (IPng), now formally called IP version 6 (IPv6), is an evolutionary enhancement of IPv4. IPv6 is designed to redress IPv4's shortfalls, retain IPv4's strong points, and accommodate

6

the expected future growth and diversity of the global Internet. The Internet Engineering Task Force (IETF) has defined the formal structure of IPv6 and this new protocol is on track to become an Internet Standard. IPv6 will eventually replace IPv4 throughout the global Internet and in most private TCP/IP networks around the world. However, the Internet community does not intend for this transition to take place precipitously. Rather, it is expected that IPv6 and IPv4 will coexist for years with transition rates driven by user requirements." (Nierle 96)

IPv6 is approved as a Standards Track protocol (Huitema 96). We intend to take advantage of IPv6 capabilities as they become publicly available, especially regarding router resource reservation, quality of service, mobility and security. Most importantly we will build and deploy vrtp using today's technology (IPv4) while designing with tomorrow's capabilities (IPv6) in mind.

Conclusions

vrtp appears to be an essential requirement to support the needs of LSVEs built using VRML. We expect this combination of client, server, peer-to-peer and monitoring capabilities will enable exciting new applications that scale with the growth of the Internet and the World Wide Web.

Acknowledgments

References

Apache Project, "Apache http Server," Web page, October 1996. Available at *http://www.apache.org*

Bell, Gavin, Parisi, Anthony and Pesce, Mark, editors, "The Virtual Reality Modeling Language (VRML) Version 1.0 Specification," May 26 1995. Available via the VRML Repository at *http://www.sdsc.edu/vrml*

Berners-Lee, Tim, Cailliau, Luotonen, Ari, Nielsen, Henrik Frystyk, and Secret, Arthur, "The World-Wide Web," *Communications of the ACM*, vol. 37 no. 8, August 1994, pp. 76-82.

Berners-Lee, T. and Connolly, D., "Hypertext Markup Language - 2.0," Request for Comments (RFC) 1866, November 1995. Available at *http://ds.internic.net/rfc/rfc1866.txt*

Berners-Lee, Tim, Fielding, R., and Henrik Frystyk, "Hypertext Transfer Protocol - HTTP/1.0," Request for Comments (RFC) 1945, May 1996. Available at *http://ds.internic.net/rfc/rfc1945.txt*

Borenstein N. and Freed N., "MIME (Multipurpose Internet Mail Extensions) Part 1," Request for Comments (RFC) 1521, September 1993. Available at *http://ds.internic.net/rfc/rfc1521.txt*

Brutzman, Donald P., Zyda, Michael and Macedonia, Michael, "Internetwork Infrastructure Requirements for Virtual Environments," *First Annual Symposium of the Virtual Reality Modeling Language (VRML)*, *Association for Computing Machinery (ACM) Special Interest Group on Computer Graphics (SIGGRAPH)*, San Diego Supercomputing Center, December 13-15 1995, pp. 95-104. Available at *http://www.stl.nps.navy.mil/~brutzman/vrml/vrml_95.html* and *vrml_95.slides.html*

Brutzman, Don, Zyda, Michael and Macedonia, Michael, "Cyberspace Backbone (CBone) Design Rationale," *Fifteenth Distributed Interactive Simulation (DIS) Standards Workshop*, Orlando Florida, September 16-20 1996, paper 96-15-099, pp.657-665. Available at *http://www.stl.nps.navy.mil/~brutzman/vrml/CBoneDIS.ps* and *CBoneDISslides.ps*

Brutzman, Don, "Graphics Internetworking: Bottlenecks and Breakthroughs," invited chapter, *Digital Illusion*, Clark Dodsworth editor, Addison-Wesley, to appear 1996. Available at *http://www.stl.nps.navy.mil/~brutzman/breakthroughs.html*

Brutzman, Don, Pesce, Mark, Bell, Gavin, van Dam, Andries and AbiEzzi, Salim, "VRML: Prelude and Future," *Proceedings, Association for Computing Machinery (ACM), Special Interest Group on Computer Graphics (SIGGRAPH 96)*, New Orleans Louisiana, August 4-9 1996. Available at *http:// www.stl.nps.navy.mil/~brutzman/vrml/siggraph96panel.html*

Case, J., McCloghrie, K., Rose, M. and Waldbusser, S., "Introduction to Community-based SNMPv2," Request for Comments (RFC) 1902, January 1996. Available at *ftp://ds.internic.net/rfc/rfc1902.txt*

Carey, Rikk, Marrin, Chris and Bell, Gavin, editors, "The Virtual Reality Modeling Language (VRML) Version 2.0 Specification," International Standards Organization/ International Electrotechnical Commission (ISO/IEC) draft standard 14772, August 4 1996. Available via the VRML Consortium at *http://www.vrml.org*

Crowcroft, Jon, moderator, "Workshop on Matters MBone," *Special Interest Group on Data Communication (SIGCOMM) 96*, Stanford University, Palo Alto California, 26 August 96.

Defense Modeling and Simulation Office (DMSO), "DoD High-Level Architecture (HLA)," overview Web page, October 1996. Available at *http://www.dmso.mil/projects/hla/*

Deering, Steve, "Host Extensions for IP Multicasting," Request for Comments (RFC) 1112, August 1989. Available at *ftp://ds.internic.net/rfc/rfc1112.txt*

Durlach, Nathaniel I. and Mavor, Anne S., ed., *Virtual Reality: Scientific and Technological Challenges,* National Research Council, National Academy Press, Washington DC, 1995.

Edwards, Evan, *Internetworking: Automated Local and Global Network Monitoring*, Master's Thesis, Naval Postgraduate School, Monterey California, September 1996. Available at *http://www.stl.nps.navy.mil/~iirg/edwards*

Erdogan, Ridvan, *Internetworking: Implementation of Multicast and MBone over Frame Relay Networks*, Master's Thesis, Naval Postgraduate School, Monterey Califorinia, September 1996. Available at *http://www.stl.nps.navy.mil/~iirg/erdogan*

Gibson William, *Neuromancer*, Ace Books, New York, 1984.

Huitema, Christian, *IPv6: The New Internet Protocol*, Prentice-Hall PTR, Upper Saddle River, New Jersey, 1996.

IEEE Standard for Distributed Interactive Simulation (DIS) -- Communication Service and Profiles, IEEE Standard P1278.1, Institute of Electrical and Electronic Engineers, New York, 1995. Information available at *http://www.sc.ist.ucf.edu/~STDS*

Katz, Warren, organizer, "Bayou Sauvage," Digital Bayou Multiplayer Simulation Demonstration, *Visual Proceedings, Association for Computing Machinery (ACM) Special Interest Group on Computer Graphics (SIGGRAPH 96)*, New Orleans Louisiana, August 4-9 1996, p. 60.

Macedonia, Michael R. and Brutzman, Donald P., "MBone Provides Audio and Video Across the Internet," IEEE *COMPUTER*, vol. 27 no. 4, April 1994, pp. 30-36. Available at *ftp://taurus.cs.nps.navy.mil/pub/i3la/mbone.html*

Macedonia, Michael R., *A Network Software Architecture for Large Scale Virtual Environments*, Ph.D. Dissertation, Naval Postgraduate School, Monterey California, June 1995. Available via *http://www.npsnet.nps.navy.mil/*

Macedonia, Michael R., Zyda, Michael J., Pratt, David R., Brutzman, Donald P. and Barham, Paul T., "Exploiting Reality with Multicast Groups: A Network Architecture for Large-Scale Virtual Environments," *IEEE Computer Graphics and Applications*, vol. 15 no. 5, September 1995, pp. 38-45. Available via *http://www.npsnet.nps.navy.mil/*

Mills, D., "Simple Network Time Protocol (SNTP)," Request for Comments (RFC) 1769, March 1995. Available at *ftp://ds.internic.net/rfc/rfc1769.txt*

Nadeau, Dave, Moreland, John and McKeon, Rosemarie, "VRML Repository," Web site, October 1996. Available at *http://www.sdsc.edu/vrml*

National Center for Supercomputer Applications (NCSA), "What is Mosaic?" Web page, October 1996. Available at *http://www.ncsa.uiuc.edu/Indices/Discover/WhatIsMosaic.html*

Netscape Corporation, "JavaScript Introduction," Web page, October 1996. Available at *http://home.netscape.com/eng/mozilla/3.0/handbook/javascript/introd.html*

Paxson, Vern, "End-to-End Routing Behavior in the Internet," *Proceedings of the ACM Special Interest Group on Data Communication (SIGCOMM) 96*, Stanford University, Palo Alto California, 26-30 August 96, published in *Computer Communication Review*, vol. 26. no. 4, pp. 25-38. Available at *http://www.acm.org/sigcomm/sigcomm96/papers/paxson.html*

Paxson, Vern, "An Introduction to Internet Measurement and Modeling," tutorial, *Special Interest Group on Data Communication (SIGCOMM) 96*, Stanford University, Palo Alto California, 26 August 96. Abstract available at *http://www.acm.org/sigcomm/sigcomm96/tutorials.html#tue2*

Schulzrinne, H., Casner, S., Frederick, R. and Jacobson, V., "RTP: A Transport Protocol for Real-Time Applications," Request for Comments (RFC) 1889, January 1996. Available at *http://ds.internic.net/rfc/rfc1889.txt*

Sun Microsystems Corporation, *Java* language Web page, October 1996. Available at *http://java.sun.com/*

World Wide Web Consortium (W3C), overview Web page, October 1996. Available at *http://www.w3.org/*

World Wide Web Consortium HTTP (W3CHTTP) working group, "Hypertext Transfer Protocol - Next Generation," overview Web page, October 1996. Available at *http://www.w3.org/pub/WWW/Protocols/HTTP-NG/*

Zyda, Michael J., Pratt, David R., Falby, John S., Barham, Paul T., Lombardo, Chuck and Kelleher, Kristen M., "The Software Required for the Computer Generation of Virtual Environments," *PRESENCE: Teleoperators and Virtual Environments*, vol. 2 no. 2, MIT Press, Cambridge Massachusetts, Spring 1993, pp. 130-140.

Author Information

Don Brutzman is a computer scientist working in the Interdisciplinary Academic Group at the Naval Postgraduate School. His research interests include underwater robotics, real-time 3D computer graphics, artificial intelligence and high-performance networking. He is a member of the Institute of Electrical and Electronic Engineers (IEEE), the Association for Computing Machinery (ACM) Special Interest Groups on Graphics (SIGGRAPH) and Communications (SIGCOMM), the American Association for Artificial Intelligence (AAAI) and the Virtual Reality Modeling Language (VRML) Consortium board of directors and technical review board.

Mike Zyda is a Professor in the Department of Computer Science at the Naval Postgraduate School, Monterey, California. Professor Zyda is also the Academic Associate and Chair of the NPS Modeling, Virtual Environments and Simulation curriculum. He has been at NPS since February of 1984. Professor Zyda's main focus in research is in the area of computer graphics, specifically the development of large-scale, networked 3D virtual environments. Professor Zyda was a member of the National Research Council's Committee on Virtual Reality Research and Development. Professor Zyda is the chair of the National Research Council's Computer Science and Telecommunications Board Committee on Modeling and Simulation: Linking Entertainment & Defense. Professor Zyda is also the Senior Editor for Virtual Environments for the MIT Press quarterly PRESENCE, the journal of teleoperation and virtual environments. He is a member of the Editorial Advisory Board of the journal Computers & Graphics. Professor Zyda is also a member of the Technical Advisory Board of the Fraunhofer Center for Research in Computer Graphics, Providence, Rhode Island. Professor Zyda has been active with the Symposium on Interactive 3D Graphics and was the chair of the 1990 conference, held at Snowbird, Utah and the chair of the 1995 Symposium, held in Monterey, California. Professor Zyda began his career in Computer Graphics in 1973 as part of an undergraduate research group, the Senses Bureau, at the University of California, San Diego. Professor Zyda received a B.A. in Bioengineering from the University of California, San Diego in La Jolla in 1976, an M.S. in Computer Science/Neurocybernetics from the University of Massachusetts, Amherst in 1978 & a D.Sc. in Computer Science from Washington University, St. Louis, Missouri in 1984.

Kent Watsen is pursuing a Ph.D. under Mike Zyda while acting as a staff member of the NPSNET Research Group in the Computer Science department at the Naval Postgraduate School. He is now lead architect of the next-generation NPSNET. His experience includes the design and development of the character animation and 3D ocean modules for EasyScene, a virtual environment toolkit which he also helped develop while working for Coryphaeus Software; as well as three years of experience building DIS simulators. He holds Computer Science and Applied Mathematics engineering degrees from the University of Virginia.

Mike Macedonia received his Ph.D. in computer science from the Naval Postgraduate School. He received the M.S. in telecommunications from the University of Pittsburgh and B.S. from the U.S. Military Academy, West Point. His research interests include multicast data networks, real-time computer graphics and large-scale virtual environments. He is currently Vice President of the Fraunhofer CRCG in Providence Rhode Island and is responsible for virtual environments and global network research