Internetwork Infrastructure Requirements for Virtual Environments

Donald P. Brutzman, Michael R. Macedonia and Michael J. Zyda
Computer Science Department, Naval Postgraduate School
Monterey California 93943-5000 USA
408.656.2149 voice, 408.656.3679 fax
{brutzman, macedonia, zyda}@cs.nps.navy.mil

Abstract. Virtual environments (VEs) are a broad multidisciplinary research area that includes all aspects of computer science, virtual reality, virtual worlds, teleoperation and telepresence. A variety of network elements are required to scale up virtual environments to arbitrarily large sizes, simultaneously connecting thousands of interacting players and all kinds of information objects. Four key communications components for virtual environments are found within the Internet Protocol (IP) suite: light-weight messages, network pointers, heavy-weight objects and real-time streams. Software and hardware shortfalls and successes for internetworked virtual environments provide specific research conclusions and recommendations. Since large-scale networked are intended to include all possible types of content and interaction, they are expected to enable new classes of interdisciplinary research and sophisticated applications that are particularly suitable for implementation using the Virtual Reality Modeling Language (VRML).

1 Overview

Virtual environments (VEs) and virtual reality applications are characterized by human operators interacting with dynamic world models of realistic sophistication and complexity (Zyda 93) (Durlach 95). Current research in large-scale virtual environments can link hundreds of people and artificial agents with interactive three-dimensional (3D) graphics, massive *terrain databases, global hypermedia and scientific datasets.* Related work on teleoperation of robots and devices in remote or hazardous locations further extends the capabilities of human-machine interaction in synthetic computer-generated environments. VE construction can include concepts and components from nearly any subject area. The variety of desired connections between people, artificial entities and information can be summarized by the slogan "connecting everyone to everything." The scope of virtual environment development is so broad that it can be seen as an inclusive superset of all other global information infrastructure applications. As diversity and detail of virtual environments increase without bound, network requirements become the primary bottleneck.

The most noticeable characteristic of virtual environments is interactive 3D graphics, which are ordinarily concerned with coordinating a handful of input devices while placing realistic renderings at fast frame rates on a single screen. Networking permits connecting virtual worlds with realistic distributed models and diverse inputs/outputs on a truly global scale. Graphics and virtual world designers interested in large-scale interactions can now consider the world-wide Internet as a direct extension of their computer. A variety of networking techniques can be combined with traditional interactive 3D graphics to collectively provide almost unlimited connectivity. In particular, the following services are essential for virtual world communications: reliable point-to-point communications, interaction protocols such as the IEEE standard Distributed Interactive Simulation (DIS) protocol, World-Wide Web (WWW) connectivity, and multicast communications.

2 Existing Infrastructure Technologies

Layered models. The integration of networks with large-scale virtual environments occurs by invoking underlying network functions from within applications. Figure 1 shows how the seven layers of the well-known Open Systems Interconnection (OSI) standard network model generally correspond to the effective layers of the Internet Protocol (IP) standard. Functional characteristic definitions of the IP layers follow in Figure 2.

These diagrams and definitions are merely an overview but help illustrate the logical relationship and relative expense of different network interactions. In general, network operations consume proportionately more processor cycles at the higher layers. Minimizing this computational burden is important for minimizing latency and maintaining virtual world responsiveness.

Methods chosen for transfer of information must use either reliable connection-oriented Transport Control Protocol (TCP) or nonguaranteed delivery connectionless User Datagram Protocol (UDP). Each of these complementary protocols is part of the Transport layer. One of the two protocols is used as appropriate for the criticality, timeliness and cost of imposing reliable delivery on the particular stream being distributed. Understanding the precise characteristics of TCP, UDP and other protocols helps the virtual world designer understand the strengths and weaknesses of each network tool employed. Since internetworking considerations impact all components in a large virtual environment,

additional study of network protocols and applications is highly recommended for virtual world designers. Suggested references include (Internet 95) (Stallings 94) and (Comer 91).
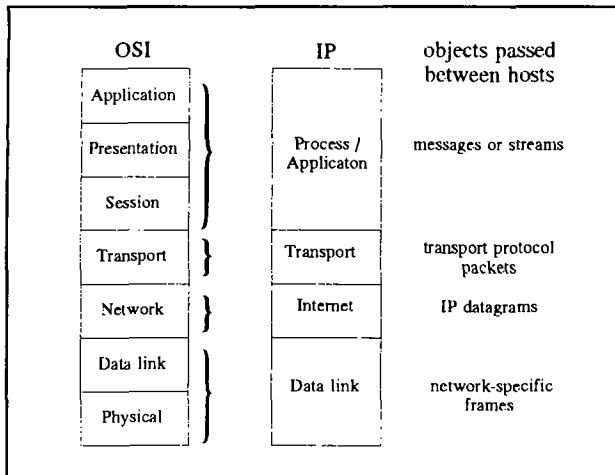

Figure 1. Correspondence between OSI and IP protocol layer models, and objects passed between corresponding host layers.

Internet Protocol (IP). Although the variety of protocols associated with internetworking are very diverse, there are some unifying concepts. Foremost is "IP on everything" or the principle that every protocol coexists compatibly within the Internet Protocol suite. The global reach and collective momentum of IP-related protocols makes their use essential and also makes incompatible exceptions relatively uninteresting. IP and IP Next Generation (IPng) protocols include a variety of electrical, radio-frequency and optical physical media.

Examination of protocol layers helps clarify current network issues. The lowest layers are reasonably stable with a huge installed base of Ethernet and FDDI systems, augmented by the rapid development of wireless and broadband ISDN solutions (such as ATM). Compatibility with the Internet Protocol (IP) suite is assumed. The middle transport-related layers are a busy research and development area. Addition of real-time reliability, quality of service and other capabilities can all be made to work. Middle-layer transport considerations are being resolved by a variety of working protocols and the competition of intellectual market forces. From the perspective of the year 2000, lower and middle layer problems are essentially solved.

o **Process/Application Layer.** Applications invoke TCP/IP services, sending and receiving messages or streams with other hosts. Delivery can be intermittent or continuous.

o **Transport Layer.** Provides host-host packetized communication between applications, using either reliable delivery connection-oriented TCP or unreliable delivery connectionless UDP. Exchanges packets end-end with other hosts.

o **Internet/Network Layer.** Encapsulates packets with an IP datagram which contains routing information, receives or ignores incoming datagrams as appropriate from other hosts. Checks datagram validity, handles network error and control messages.

o **Data Link/Physical Layer.** Includes physical media signaling and lowest level hardware functions, exchanges network-specific data frames with other devices. Includes capability to screen multicast packets by port number at the hardware level.

Figure 2. Summary of Internet Protocol (IP) suite layer functionality.

Distributed Interactive Simulation (DIS). The DIS protocol is an IEEE standard for logical communication among entities in distributed simulations (IEEE 93). Although initial development was driven by the needs of military users, the protocol formally specifies the communication of physical interactions by any type of physical entity and is adaptable for general use. Information is exchanged via protocol data units (PDUs) which are defined for a large number of interaction types.

The principal PDU type is the Entity State PDU. This PDU encapsulates the position and posture of a given entity at a given time, along with linear and angular velocities and accelerations. Special components of an entity (such as the orientation of moving parts) can also be included in the PDU as articulated parameters. A full set of identifying characteristics uniquely specifies the originating entity. A variety of dead reckoning algorithms permits computationally efficient projection of entity posture by listening hosts. Dozens of additional PDU types are defined for simulation management, sensor or weapon interaction, signals, radio communications, collision detection and logistics support.

Of particular interest to virtual world designers is an open format Message PDU. Message PDUs enable

user-defined extensions to the DIS standard. Such flexibility coupled with the efficiency of Internet-wide multicast delivery permits extension of the object-oriented message-passing paradigm to a distributed system of essentially unlimited scale. It is reasonable to expect that free-format DIS Message PDUs might also provide remote distributed connectivity resembling that of tuples to any information site on the Internet, further extended by use of network pointer mechanisms which already exist for the World-Wide Web. This is a promising area for future work.

World-Wide Web. The World-Wide Web (WWW or Web) project has been defined as a "wide-area hypermedia information retrieval initiative aiming to give universal access to a large universe of documents" (Hughes 94). Fundamentally the Web combines a name space consisting of any information store available on the Internet with a broad set of retrieval clients and servers, all of which can be connected by easily-defined HyperText Markup Language (.html) hypermedia links. This globally-accessible combination of media, client programs, servers and hyperlinks can be conveniently utilized by humans or autonomous entities. The Web has fundamentally shifted the nature of information storage, access and retrieval (Berners-Lee 94). Current Web capabilities are easily used despite rapid growth and change. Directions for future research related to the Web are discussed in (Foley 94). Nevertheless, despite tremendous variety and originality, current Web-based interactions are essentially client-server: a user can push on a Web resource and get a response, but there are no standardized mechanisms for a Web application to independently push back at the user.

Multicast. IP multicasting is the transmission of IP datagrams to an unlimited number of multicast-capable hosts which are connected by multicast-capable routers. Multicast groups are specified by unique IP Class D addresses, which are identified by 1110_2 in the high-order bits and correspond to Internet addresses 224.0.0.0 through 239.255.255.255. Hosts choose to join or leave multicast groups and subsequently inform routers of their membership status. Of great significance is the fact that individual hosts can control which multicast groups they monitor by reconfiguring their network interface hardware at the data link layer. Since datagrams from unsubscribed groups are ignored at the hardware interface, host computers can solely monitor and process packets from groups of interest, remaining unburdened by other network traffic (Comer 91) (Deering 89).

Multicasting has existed for several years on local-area networks such as Ethernet and Fiber Distributed Data Interface (FDDI). However, with Internet Protocol

multicast addressing at the network layer, group communication can be established across the Internet. Since multicast streams are typically connectionless UDP datagrams, there is no guaranteed delivery and lost packets stay lost. This best-effort unreliable delivery behavior is actually desirable when streams are high bandwidth and frequently recurring, in order to minimize network congestion and packet collisions. Example multicast streams include video, graphics, audio and DIS. The ability of a single multicast packet to connect with every host on a local-area network is good since it minimizes the overall bandwidth needed for large-scale communication. Note however that the same multicast packet is ordinarily prevented from crossing network boundaries such as routers. If a multicast stream that can touch every workstation were able to jump from network to network without restriction, topological loops might cause the entire Internet to become saturated by such streams. Routing controls are necessary to prevent such a disaster, and are provided by the recommended multicast standard (Deering 89) and other experimental standards. Collectively the resulting internetwork of communicating multicast networks is called the Multicast Backbone (MBone) (Macedonia, Brutzman 94).

Improved real-time delivery schemes are also being evaluated using the Real-time Transport Protocol (RTP) which is eventually expected to work independently of TCP and UDP (Schulzrinne 93). Other real-time protocols are also under development. The end result available today is that even with a time-critical application such as an audio tool, participants normally perceive conversations as if they are in ordinary real time. This behavior is possible because there is actually a small buffering delay to synchronize and resequence the arriving voice packets. Research efforts on real-time protocols and numerous related issues are ongoing, since every bottleneck conquered results in a new bottleneck revealed.

The MBone community must manage the MBone topology and the scheduling of multicast sessions to minimize congestion. Currently over 1800 subnets are connected world-wide, with a corresponding host count equivalent to the size of Internet in 1990. Topology changes for new nodes are added by consensus: a new site announces itself to the MBone mail list, and the nearest potential providers decide who can establish the most logical connection path to minimize regional Internet loading. Scheduling MBone events is handled similarly. Special programs are announced in advance on an electronic mail list and a forms-fed schedule home page. Advance announcements usually prevent overloaded scheduling of Internet-wide events and alert potential participants. Cooperation is key. Newcomers are often surprised to learn that no single person or

authority is "in charge" of either topology changes or event scheduling.

3 Software Infrastructure Needs

The "grand challenges" of computing today are not large static gridded simulations such as computational fluid dynamics or finite element modeling. Similarly traditional supercomputers are not the most powerful or significant platforms. Adding hardware and dollars to incrementally improve existing expensive computer designs is a well-understood exercise. What is more challenging and potentially more rewarding is the interconnection of all computers in ways that support global interaction of people and processes. In this respect, the Internet is the ultimate supercomputer, the Web is the ultimate database, and any networked equipment in the world is a potential input/output device. Large-scale virtual environments attempt to simultaneously connect many of these computing resources in order to recreate the functionality of the real world in meaningful ways. Network software is the key to solving VE grand challenges.

Four key communication methods. Large-scale virtual world internetworking is possible through the application of appropriate network protocols. Both bandwidth and latency must be carefully considered. Distribution of virtual world components using point-to-point sockets can be used for tight coupling and real-time response of physics-based models. The DIS protocol enables efficient live interaction between multiple entities in multiple virtual worlds. The coordinated use of hypermedia servers and embedded Web browsers allows virtual worlds global input/output access to pertinent archived images, papers, datasets, software, sound clips, text or any other computer-storable media. Multicast protocols permit moderately large real-time bandwidths to be efficiently shared by an unconstrained number of hosts. Applications developed for multicast permit open distribution of graphics, video, audio, DIS and other streams world-wide in real time. Together these example components provide the functionality of light-weight messages, network pointers, heavy-weight objects and real-time streams (Figure 3). Integrating these network tools in virtual worlds produces realistic, interactive and interconnected 3D graphics that can be simultaneously available anywhere (Brutzman 94)(Macedonia 95a 95b).

Application layer interactivity. It is application layer networking that needs the greatest attention in preparing for the information infrastructure of the near future. DIS combined with multicast transport provides solutions for many application-to-application

communications requirements. Nevertheless DIS is insufficiently broad and not adaptable enough to meet general virtual environment requirements. To date, most of the money spent on networked virtual environments has been by, for and about the military. Most remaining work has been in (poorly) networked games. Neither is reality. There is a real danger that specialized high-end military applications and chaotic low-end game hacks will dominate entity interaction models. Such a situation might well prevent networked virtual environments from enjoying the sustainable and compatible exponential growth needed to keep pace with other cornerstones of the information infrastructure.

- **Light-weight Interactions.** Messages composed of state, event and control information as used in DIS Entity State PDUs. Implemented using multicast. Complete message semantics is 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, 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 WWW query response to a network pointer request.

- **Real-time Streams.** Live video, audio, DIS, 3D graphics images or other continuous stream traffic that requires real-time delivery, sequencing and synchronization. Implemented using multicast channels.

Figure 3. Four key communications components used in virtual environments.

Next-generation DIS. A successor to DIS is needed which is simpler, open, extensible and dynamically modifiable. DIS has proven capabilities in dealing with position and posture dead reckoning updates, physically-based modeling, hostile entity interactions and variable latency over wide-area networks. DIS also has several difficulties: awkward extendibility, requiring nontrivial computations to decipher bit patterns, and being a very "big" standard.

DIS protocol development continues through a large and active standards community. However the urgent military requirements driving the DIS standard remain narrower than general virtual environment networking requirements.

A common theme which runs through all network protocol development is that realistic testing and evaluation are essential, because initial performance of distributed applications never matches expectations or theory. A next-generation DIS research project ought to develop a "dial-a-protocol" capability, permitting dynamic modifications to the DIS specification to be transmitted to all hosts during an exercise. Such a dynamically adjustable protocol is a necessity for interactively testing and evaluating both global and local efficiency of distributed entity interactions.

Other interaction models. Many other techniques for entity interaction are being investigated, although not always in relation to virtual environments. Intelligent agent interactions are an active area of research being driven by artificial intelligence and user interface communities. Rule-based agents typically communicate via a message-passing paradigm that is a natural extension of object-oriented programming methods. Common Gateway Interface (cgi) scripts function similarly, usually using HyperText Transfer Protocol (http) (Berners-Lee 94) query extensions as inputs. Ongoing research by the Linda project uses "tuples" as the communications unit for logical entity interaction, with particular emphasis on scaling up (Gelernter 92). MUDs (multi-user dungeons) and MOOs (MUDs Object-Oriented) provide a powerful server architecture and text-based interaction paradigm that is well suited to support a variety of virtual environment scenarios (Curtis 94). Passing scripts and interpretable source code over the network for automatic client use has been widely demonstrated for the multiplatform Tool Control Language (Tcl) (Ousterhout 94). Recently the *Java* language has provoked interest over the possibility of simple and secure passing of precompiled program object files for multiplatform execution (Sun 95).

Virtual Reality Modeling Language (VRML). The Web is being extended to three spatial dimensions thanks to VRML, a specification based on the Silicon Graphics Inc. (SGI) *OpenInventor* scene description language (Wernicke 94). Key contributions of the VRML 1.0 standard are a core set of object-oriented graphics constructs augmented by hypermedia links, all suitable for scene generation by browsers on PCs, Macintoshes and Unix workstations. The current interaction model for VRML browsers is client-server, similar to most other Web browsers. Specification development has been effectively coordinated by mail

list, enabling consensus by a large, active and open membership (Pesce 94) (Bell 94).

Discussion on incorporating interaction, coordination and entity behaviors into VRML 2.0 has been very active. A great number of issues are involved. In order to scale to arbitrarily large sizes, peer-to-peer interactions will be necessary in addition to client-server query-response. Although behaviors are not yet formally specified, the following possible view of behaviors extends the syntax of existing "engine" functionality in *OpenInventor* (Figure 4). Two key points in this representation follow. First, engine outputs only operate on the virtual scene graph, so behaviors do not have any explicit control over the host machine (unlike CGI scripts). Second, behaviors are engine drivers, while engines are scene graph interfaces. This means that a wide variety of behavior mechanisms might stimulate engine inputs, including *OpenInventor* sensors and calculators, scripted actions, message passing, command line parameters or DIS. Thus it appears that forthcoming VRML behaviors might simultaneously provide simplicity, security, scalability, generality and open extensions. Terminology will change as consensus is reached but it appears that most or all proposed behavior mechanisms can compatibly share this common level of functionality. Finally, as the demanding bandwidth and latency requirements of virtual environments begin to be exercised by VRML, the client-server design assumptions of the HyperText Transfer Protocol (http) will no longer be valid. Users will not be satisfied with network mechanisms that fail to accommodate high-bandwidth information streams or break down after a few hundred players. A Virtual Reality Transfer Protocol (vrtp) will be needed to take advantage of available transport layer functionality and overcome bottlenecks in http. Experimentation and quantitative evaluation will be essential to better understand how to practically deal with the new dynamic requirements of diverse interentity virtual environment communications.

Vertical interoperability. A striking trend in public domain and commercial software tools for DIS, MBone and the Web is that they can seamlessly operate on a variety of software architectures. The hardware side of vertical interoperability for virtual environments is simple: access to IP/Internet and the ability to render real-time 3D graphics. The software side is that information content and even applications can be found that run equivalently under PC, Macintosh and a wide variety of Unix architectures. One important goal for any virtual environment is that human users, artificial entities, information streams and content sources can interoperate over a range which includes highest performance machines to least common denominator machines. Here are some success metrics for vertical

interoperability: "Will it run on my supercomputer?" Yes. "Will it run on my Unix workstation?" Yes. "Will it also run on my Macintosh or PC?" Yes. This approach has been shown to be a practical (and even preferable) software requirement. Vertical

interoperability is typically supported by open nonproprietary specifications developed by standardization groups such as the Internet Engineering Task Force (IETF).
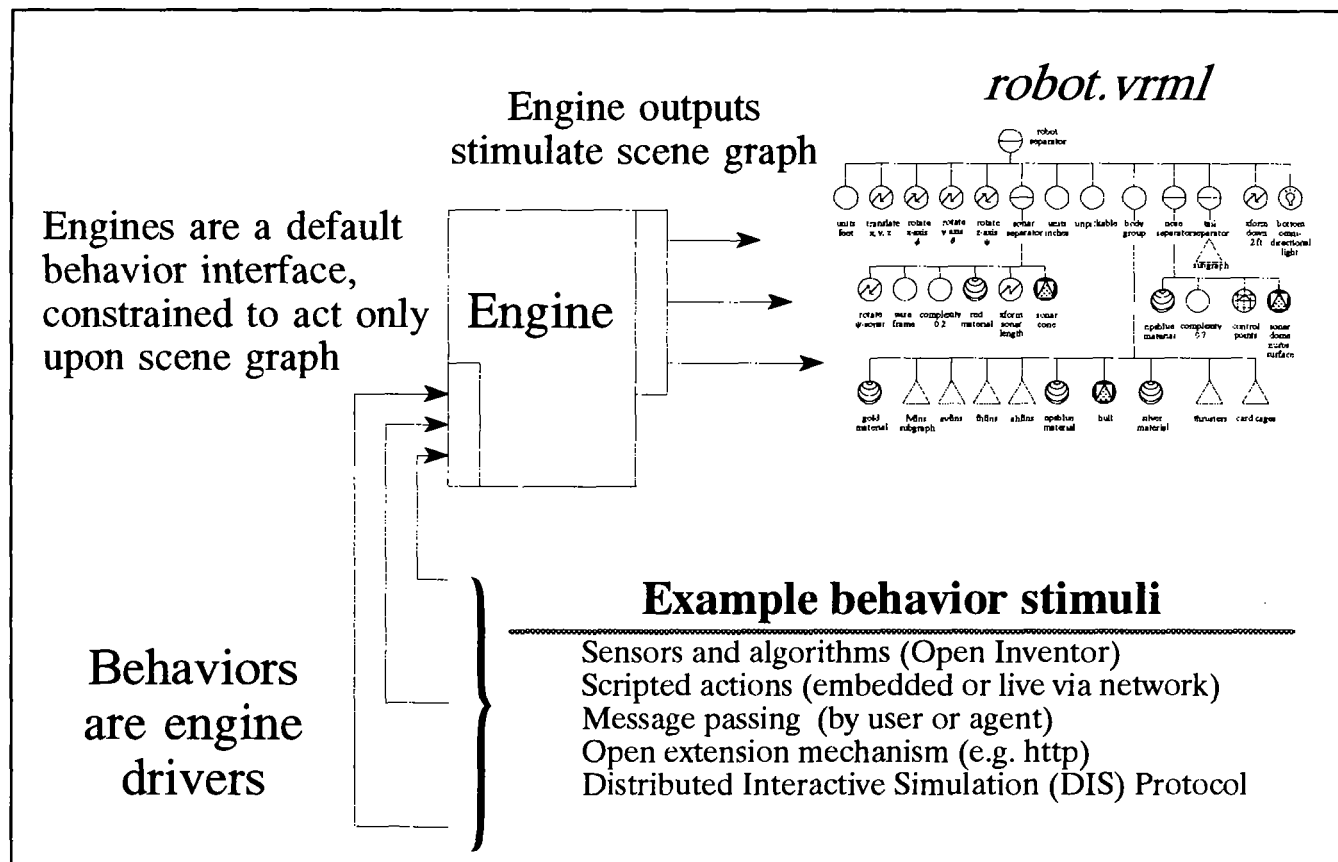

Figure 4. Proposed behavior interaction model for Virtual Reality Modeling Language (VRML).

4 Hardware Infrastructure Needs

Research testbed. The National Research Council report on Virtual Reality (Durlach 95) made few recommendations for funding virtual environment hardware research due to active commercial development in most critical technologies. However the report also has a notable hardware-related recommendation regarding networks:

"RECOMMENDATION: The committee recommends that the federal government provide funding for a program (to be conducted with industry and academia in collaboration) aimed at developing network standards that support the requirements for implementing distributed VEs on a large scale. Furthermore, we recommend funding of an open VE network that can be used by researchers, at a reasonable cost, to experiment with various VE network software developments and applications." (Durlach 95, p. 83)

The cost of high-speed network connections has precluded most academic institutions from conducting basic research in high-performance network applications. Those sites with high-performance connections are rarely free from the reliability requirements of day-to-day network operations. A national Virtual Environment Network Testbed for academia and industry is proposed as a feasible collaboration mechanism. If rapid progress is expected in the near future, it is clearly necessary to decouple experimental network research from campus electronic mail and other essential services. The International Wide-Area Year (I-WAY) project is a proposed experimental national network that is applications-driven and ATM-based (I-WAY 95). It will connect a number of high-performance computing centers and supercomputers together. I-WAY may well serve as a first step in the direction of a national testbed, but additional efforts will be needed to connect institutions

100

with lesser research budgets. Finally it must be noted that design progress and market competition are bringing the startup costs of high-speed local-area networks (e.g. FDDI, ATM) within reach of institutional budgets. At most schools, it is the off-campus links to the Internet that need upgrading and funding for sustained use. Numerous commercial efforts are targeting low-bandwidth dialup connections so scarce research community dollars are not needed in that arena.

Other problems. In order to achieve broad vertical integration, it is recommended that proprietary and vendor-specific hardware be avoided. Videoteleconferencing (VTC) systems are an example of a market fragmented by competing proprietary specifications. Broad interoperability and Internet compatibility are essential. Closed solutions are dead ends. In the area of new network services such as Asynchronous Transfer Mode (ATM) and Integrated Services Digital Network (ISDN), some disturbing trends are commonplace. Supposedly standardized protocol implementations often do not work as advertised, particularly when run between hardware from different vendors. Effective throughput is often far less than maximum bit rate. Latency performance is highly touted and rarely tested. Working applications are difficult to find. Network operating costs are often hidden or ignored. Application developers are advised to plan and budget for lengthy delays and costly troubleshooting when working with these new services.

5 Applications

Working applications will drive progress, not theories and not hype. In this section we present feasible applications that are exciting possibilities or existing works in progress. Many new projects are possible and likely to occur in the near future if virtual environment requirements are adequately supported in the information infrastructure. Several conjectured scenarios follow.

Sports: live 3D stadium with instrumented players. Imagine that all of the ballplayers in a sports stadium wear a small device which senses location (through the Global Positioning System or local electrical field sensing) and transmits DIS packets over a wireless network. Similar sensors are embedded in gloves, balls, bats and even shoes. A computer server in the stadium feeds telemetry inputs into an physically-based articulated human model which extrapolates individual body and limb motions. The server also maintains a scene database for the stadium complete with textured images of the edifice, current weather and representative pictures of fans in the stands. Meanwhile, Internet users have browsers which can navigate and view the stadium from

any perspective. Users can also tune to multicast channels providing updated player positions and postures along with live audio and video. Statistics, background information and multimedia home pages are available for each player. Online fan clubs and electronic mail lists let fans trade opinions and even send messages to the players. Thus any number of remote fans might supplement traditional television coverage with a live interactive computer-generated view. Perhaps the most surprising aspect of this scenario is that all component software and hardware technologies exist today.

Military: 100,000 player problem. "Exploiting Reality with Multicast Groups" describes ground-breaking research on increasing the number of active entities within a virtual environment by several orders of magnitude (Macedonia 95a, 95b). Multicast addressing and the DIS protocol are used to logically partition network traffic according to spatial, temporal and functionally related entity classes. "Exploiting Reality" further explains virtual environment network concepts and includes experimental results. This work has fundamentally changed the distributed simulation community, showing that very large numbers of live and simulated networked players in real-world exercises are feasible.

Science: virtual worlds as experimental laboratories for robots and people. In separate work, we have shown how an underwater virtual world can comprehensively model all salient functional characteristics of the real world for an Autonomous Underwater Vehicle (AUV) in real time. This virtual world is designed from the perspective of the robot, enabling realistic AUV evaluation and testing in the laboratory. Real-time 3D computer graphics are our window into that virtual world. Visualization of robot interactions within a virtual world permits sophisticated analyses of robot performance that are otherwise unavailable. Sonar visualization permits researchers to accurately "look over the robot's shoulder" or even "see through the robot's eyes" to intuitively understand sensor-environment interactions. Theoretical derivation of six-degree-of-freedom hydrodynamics equations has provided a general physics-based model capable of replicating highly nonlinear (yet experimentally verifiable) response in real time. Distribution of underwater virtual world components enables scalability and rapid response. Networking allows remote access, demonstrated via MBone audio and video collaboration with researchers at distant locations. Integrating the World-Wide Web allows rapid access to resources distributed across the Internet. Ongoing work consists primarily of scaling up the types of interactions, datasets

101

and live streams that can be coordinated within the virtual world (Brutzman 94).

Interaction: Multiple CAVEs using ATM and VRML. A CAVE is a type of walk-in synthetic environment which replaces the four walls of a room with rear-projection screens, all driven by real-time 3D computer graphics (Cruz-Neira 93). These devices can accommodate 10-15 people comfortably and render high-resolution 3D stereo graphics at 15 Hz update rates. The principal costs of a CAVE are in high-performance graphics hardware. We wish to demonstrate affordable linked CAVEs for remote group interaction. The basic idea is to send graphics streams from a master CAVE through a high-speed low-latency ATM link to a less expensive slave CAVE which contains only rear-projection screens. Automatic generation of VRML scene graphs and simultaneous replication of state information over standard multicast links will permit both CAVEs and networked computers to interactively view results generated in real time by a supercomputer. Our initial application domain is a gridded virtual environment model of the oceanographic and biological characteristics of Chesapeake Bay. To better incorporate networked sensors and agents into this virtual world, we are also investigating extensions to IP using underwater acoustics (Brutzman 95c). As a final component, we are helping establish an ambitious regional education and research network which connects scientists, students from kindergartens through universities, libraries and the general public. Vertically integrated Web and MBone applications and a common theme of live networked environmental science are expected to provide many possible virtual world connections (Brutzman 95a, 95b).

6 Projections

If one considers the evolving nature of the global information infrastructure, it is clear that there is no shortage of basic information. Quite the opposite is true. Merely by reading the *New York Times* daily, any individual can have more information about the world than was available to any world leader throughout most of human history! Multiply that single information stream by the millions of other information sources becoming openly available on the Internet, and it is clear that the Internet does not lack content. Mountains of content have become accessible. What is needed now is context, a way to interactively locate, retrieve and display the related pieces of information and knowledge that a user needs in a timely manner. Context further permits selectively culling large volumes of information that can otherwise overwhelm any person or process.

Within two lifetimes we have seen several paradigm shifts in the ways that people record and exchange information. Handwriting gave way to typing, and then typing to word processing. It was only a short while afterwards that preparing text with graphic images was easily accessible, enabling individuals to perform desktop publishing. Currently people can use 3D real-time interactive graphics simulations and dynamic "documents" with multimedia hooks to record and communicate information. Furthermore such documents can be directly distributed on demand to anyone connected to the Internet. In virtual environments a further paradigm shift becoming possible. The long-term potential of virtual environments is to serve as an archive and interaction medium, combining massive and dissimilar data sets and data streams of every conceivable type. Virtual environments will then enable comprehensive and consistent interaction by humans, robots and software agents within those massive data sets, data streams and models that recreate reality. Virtual environments can provide meaningful context to the mountains of content which currently exist in isolation without roads, links or order.

What about scaling up? Fortunately there already exists a model for these growing mountains of information content: the real world. Virtual worlds can address the context issue by providing information links similar to those that exist in our understanding of the real world. When our virtual constructs cumulatively approach realistic levels of depth and sophistication, our understanding of the real world will deepen correspondingly. In support of this goal, we have shown how the structure and scope of virtual environment relationships can be dynamically extended using feasible network communications methods. This efficient distribution of information will let any remote user or entity in a virtual environment participate and interact in increasingly meaningful ways.

Open access to any type of live or archived information resource is becoming available for everyday use by individuals, programs, collaborative groups and even robots. Virtual environments are a natural way to provide order and context to these massive amounts of information. World-wide collaboration works, both for people and machines. Finally, the network is more than a computer, and even more than your computer. The Internet becomes our computer as we learn how to share resources, collaborate and interact on a global scale.

7 Conclusions

- Virtual environments (VEs) are an all-inclusive superset of the exponentially growing global information infrastructure.

- Any workable VE solution must address scaling up to arbitrarily large sizes.

102

- Lower and middle network layer problems are basically solved.

- Four key communications components are used in virtual environments:
  - Light-weight entity interactions (e.g. DIS PDUs)
  - Network pointers (e.g. URLs), usually passed within a light-weight entity interaction that includes identifying context
  - Heavy-weight objects (e.g. terrain databases or large textured scene graphs) which require reliable connections for accurate retrieval
  - Real-time streams (e.g. audio and video), usually via MBone

- The Virtual Reality Modeling Language (VRML) will take the Web into 3D and make powerful applications available across platforms which were previously restricted to expensive graphics workstations.

8 Recommendations

- It is the upper network application layers relating to entity interactions that most need active research work.

- An international Virtual Environment Network Testbed is needed to enable research schools to cheaply connect and test high-bandwidth experimental services without jeopardizing production networks.

- Behaviors for VRML will allow 3D graphics objects to be driven by
  - Internal algorithms (e.g. *OpenInventor* sensors)
  - Scripted actions (embedded or live via network streams)
  - Message passing (by user or agent, e.g. MUDs/MOOs)
  - Open extension mechanism (e.g. *http attributes*)
  - DIS (or superior) entity interaction protocol

- Virtual Reality Transport Protocol (vrtp) will soon be needed
  - Bandwidth, latency and efficiency requirements will change relative to the design assumptions of current *http* servers
  - Light-weight objects, heavy-weight objects, multicast real-time streams
  - Small sets of servers will combine to serve very large client/peer base

- Compelling applications will drive progress, not protocols. Some examples:
  - Sports: live 3D stadium with instrumented players, user chooses view

  - Military: 100,000 player problem
  - Science: virtual worlds as experimental laboratories for robots, people
  - Interaction: affordable linked CAVEs for remote group interaction

References

Bell, Gavin, Parisi, Anthony and Pesce, Mark, "The Virtual Reality Modeling Language (VRML) Version 1.0 Specification," 26 May 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.

Brutzman, Donald P., *A Virtual World for an Autonomous Underwater Vehicle*, Ph.D. Dissertation, Naval Postgraduate School, Monterey California, December 1994. Available at *http://www.stl.nps.navy.mil/~brutzman/dissertation/*

Brutzman, Don, "Remote Collaboration with Monterey Bay Educators," *Visual Proceedings, Association for Computing Machinery (ACM) Special Interest Group on Computer Graphics (SIGGRAPH) 95*, Los Angeles California, August 7-11 1995, p. 145.

Brutzman, Don, "Networked Ocean Science Research and Education, Monterey Bay California," *Proceedings of International Networking (INET) 95 Conference*, Internet Society, Honolulu Hawaii, June 27-30 1995. Available at *http://inet.nttam.com/HMP/PAPER/039*

Brutzman, Don and Reimers, Stephen, "Internet Protocol over Seawater: Towards Interoperable Underwater Networks," *Unmanned Untethered Submersibles Technology 95*, Northeastern University, Nahant Massachusetts, September 25-27 1995, to appear.

Comer, Douglas E., *Internetworking with TCP/IP Volume I: Principles, Protocols and Architecture*, second edition, Prentice Hall, Englewood Cliffs New Jersey, 1991.

Cruz-Neira, Carolina, Leigh, Jason, Papka, Michael, Barnes, Craig, Cohen, Steven M., Das, Sumit, Engelmann, Roger, Hudson, Randy, Roy, Trina, Siegel, Lewis, Vasilakis, Christina, DeFanti, Thomas A. and Sandin, Daniel J., "Scientists in Wonderland: A Report on Visualization Applications in the CAVE Virtual Reality Environment," *IEEE 1993 Symposium on Research Frontiers in Virtual Reality*, San Jose California, October 25-26 1993, pp. 59-66 and CP-3.

Curtis, Pavel and Nichols, David A., "MUDs Grow Up: Social Virtual Reality in the Real World," *Proceedings of the IEEE Computer Conference*, IEEE Computer Society Press, Los Alamitos California, 1994, pp. 193-200. Available at *ftp://ftp.parc.xerox.com/pub/MOO/papers/MUDsGrowUp.ps*

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., editors, *Virtual Reality: Scientific and Technological Challenges*, National Research Council, National Academy Press, Washington DC, 1995.

Foley, Jim and Pitkow, James, ed., *Research Priorities for the World-Wide Web*, National Science Foundation (NSF) Information, Robotics and Intelligent Systems Division Workshop, Arlington Virginia, October 31 1994. Available at *http://www.cc.gatech.edu/gvu/nsf-ws/report/Report.html*

Gelernter, David, *Mirror Worlds - or the Day Software Puts the Universe in a Shoebox... How It Will Happen and What It Will Mean*, Oxford University Press, New York, 1992.

Hughes, Kevin, "Entering the World-Wide Web (WWW): A Guide to Cyberspace," Enterprise Integration Technology Inc., May 1994. Available at *http://www.eit.com/web/www.guide/*

IEEE Standard for Information Technology - Protocols for Distributed Interactive Simulation (DIS) Applications, version 2.0, Institute for Simulation and Training report IST-CR-93-15, University of Central Florida, Orlando Florida, May 28 1993.

Internet Network Information Center (NIC), *Request For Comments (RFC)* archive, *ftp://ds.internic.net*, 1994.

International Wide-Area Year (I-WAY) project, 1995, information available at *http://www.iway.org*

Macedonia, Michael R. and Brutzman, Donald P., "MBone Provides Audio and Video Across the Internet," IEEE *COMPUTER*, 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.

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.

Ousterhout, John K., *Tcl and the Tk Toolkit*, Addison-Wesley, Reading Massachusetts, 1994.

Pesce, Mark and Behlendorf, Brian, moderators, "Virtual Reality Modeling Language (VRML)," working group mail list, 1994-1995. Archived at *http://vrml.wired.com/*

Schulzrinne, Henning and Casner, Stephen, "RTP: A Transport Protocol for Real-Time Applications," Audio-Video Transport Working Group, Internet Engineering Task Force, working draft, Oct. 20, 1993. Available at *ftp://nic.ddn.mil/internet-drafts/draft-ietf-avt-rtp-04.ps*

Stallings, William, *Data and Computer Communications*, fourth edition, Macmillan Publishing, New York, 1994.

Sun Microsystems Corporation, *Java* language home page, 1995, *http://java.sun.com/*

Wernicke, Josie, *The Inventor Mentor: Programming Object-Oriented 3D Graphics with OpenInventor™, Release 2*, Addison-Wesley Publishing, Reading Massachusetts, 1994.

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 Group on Graphics (SIGGRAPH), the American Association for Artificial Intelligence (AAAI), the Marine Technology Society (MTS) and the Internet Society (ISOC).

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. His research interests include multicast data networks, real-time computer graphics and large-scale virtual environments. A retired Army officer, his leadership on the Joint Electronic Warfare Center and CENTCOM staffs was instrumental in successfully deploying and integrating advanced computers, networks and telecommunications systems during Operations Desert Shield and Desert Storm. He is currently working at Fraunhofer CRCG in Providence Rhode Island.

Mike Zyda is Professor of Computer Science at the Naval Postgraduate School. His research interests include computer graphics, virtual world systems and visual simulation systems. He is Executive Editor of the journal *PRESENCE: Teleoperators and Virtual Environments*, published by MIT Press. His recent accomplishments include organizing and serving as General Chair for the 1995 ACM SIGGRAPH Symposium on Interactive 3D Graphics.