VII. GLOBALLY NETWORKED 3D GRAPHICS AND VIRTUAL WORLDS

A. INTRODUCTION

Three-dimensional interactive graphics 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 distributed models and completely 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, four component services are proposed as being necessary and sufficient for virtual world networking: reliable point-to-point socket communications, multicast communications protocols, interaction protocols such as the IEEE standard Distributed Interactive Simulation (DIS) protocol, and World-Wide Web connectivity.

The key specifications for virtual world networking are the application of appropriate network protocols and careful consideration of bandwidth. Distribution of virtual world components using point-to-point sockets enables upward scalability and real-time response. Multicast protocols permit moderately large bandwidths to be efficiently shared by an unconstrained number of hosts. Applications developed for the Multicast Backbone (MBone) permits open distribution of graphics, video, audio, DIS and other streams worldwide in real time. The DIS protocol enables efficient live interaction between multiple entities in multiple virtual worlds. The coordinated use of hypermedia servers and embedded World-Wide 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. With these four network tools integrated in virtual worlds, 3D computer graphics can be simultaneously available anywhere.

B. NETWORKING BENEFITS

The benefits of networking a virtual world are many and worth enumerating. Any virtual world which attempts to model parts of the real world with nontrivial complexity will soon outstrip the computational capabilities and real-time capacity of any single computer. Heterogeneous processes need to be able to run on heterogenous processors. Massive archived datasets, sensor telemetry, component models, human users and autonomous entities can connect to the virtual world from wherever where they exist in the real world. This approach permits problem scalability, real-time response and interoperability. It also enables economies of scale since the structure of the virtual world can utilize an installed base of computers already connected to the Internet which numbers over twenty million. Since knowledge resource archiving and human access to the Internet is growing phenomenally at a sustained exponential rate of approximately 20% per month, virtual world design must address network connectivity and access efficiency in scalable ways.

C. BANDWIDTH SPECIFICATIONS FOR VIRTUAL WORLD NETWORKING

Three-dimensional computer graphics and network communications are both concerned with the delivery of information streams. In each case an all-encompassing criteria is bandwidth. In computer graphics, bandwidth concerns are manifested by frame rate, image size, level of detail, polygon culling and rendering complexity due to lighting models, texturing etc. The intended net result is delivery of effective visual information to a viewer. In networks, bandwidth is primarily measured by the information capacity of a channel in kilobits per second (Kbps) and is also affected by packet size, delivery latency, network loading, transport reliability and processor capacity. The net result is delivery of a information stream to one or multiple recipients.

It is useful to know the bandwidths of typical information streams since they can vary widely. Uncompressed video bandwidth transmitted on a network can consume as much as 60 Mbps. A 320x240 pixel 8 bit color video or graphics window reproduced by network video tool *nv* requires 128 Kbps for 1-3 frames per second, or 256 Kbps for 3-5 frames per second, where effective frame rate varies inversely with the number of pixels which vary from frame to frame. A telephone-quality audio channel (300-3300 Hz) requires 50-75 Kbps capacity depending upon the encoding algorithm employed. A musical instrument digital interface (MIDI) stream requires 32 Kbps. A representative entity DIS posture stream requires about 1 Kbps. One-time retrieval of data objects over the Internet has highly variable bandwidth which is principally dependent on the capacity of respective host connections and current intermediate network loading.

It is similarly important to know the capacity of various network connections. Most local area networks use Ethernet which has a maximum bandwidth of 10 Mbps. Fiber Distributed Data Interface (FDDI) is 100 Mbps. Microwave wireless bridges used to connect LANs typically have a bandwidth capacity of 1 Mbps. Modems on standard telephone lines can only support 2-20 Kbps. Typical fixed site connections to the Internet are T1 at 1.5 Mbps, or T3 at 45-155 Mbps (depending on whether electrical or optical signaling is used). Integrated services digital network (ISDN) lines are becoming available to business and home users, with line capacities measured in 64 or 128 Kbps increments up to a total of 1.5 Mbps. Frame Relay is a commercially available switching technique that supports best-effort delivery and variable-length data frames at bandwidths up to 2 Mbps. Broadband ISDN (BISDN) refers to Asynchronous Transfer Mode (ATM) (also known as Cell Relay) which uses fixed length data cells for switching bandwidths up to gigabits per second. Depending on contention-handling techniques used by the corresponding protocols, the effective bandwidth of each link type listed above may only be 80-90% of the theoretical maximum before collisions and collision recovery becomes prohibitive.

In every case, these various network connections are only of practical use to globally networked 3D graphics when they are compatible with the Internet Protocol (IP) suite. Given current implementations and eventual standardization of IP over

ATM (Armitage 94), IP compatibility exists for all of the listed connection types. Relatively high frame rate graphics can be generated over the Internet by low-end graphics workstations. Simultaneous duplication of graphics-related streams at both high and low bandwidths is feasible and desirable to accommodate these various bandwidth capacities. Duplicate imagery streams permits a variety of users to participate interactively via nearly any of the network connections listed above.

D. TERMINOLOGY AND NETWORK LAYERS

The integration of networks with computer graphics and virtual worlds occurs by invoking underlying network functions from within applications. Figure 7.1 shows

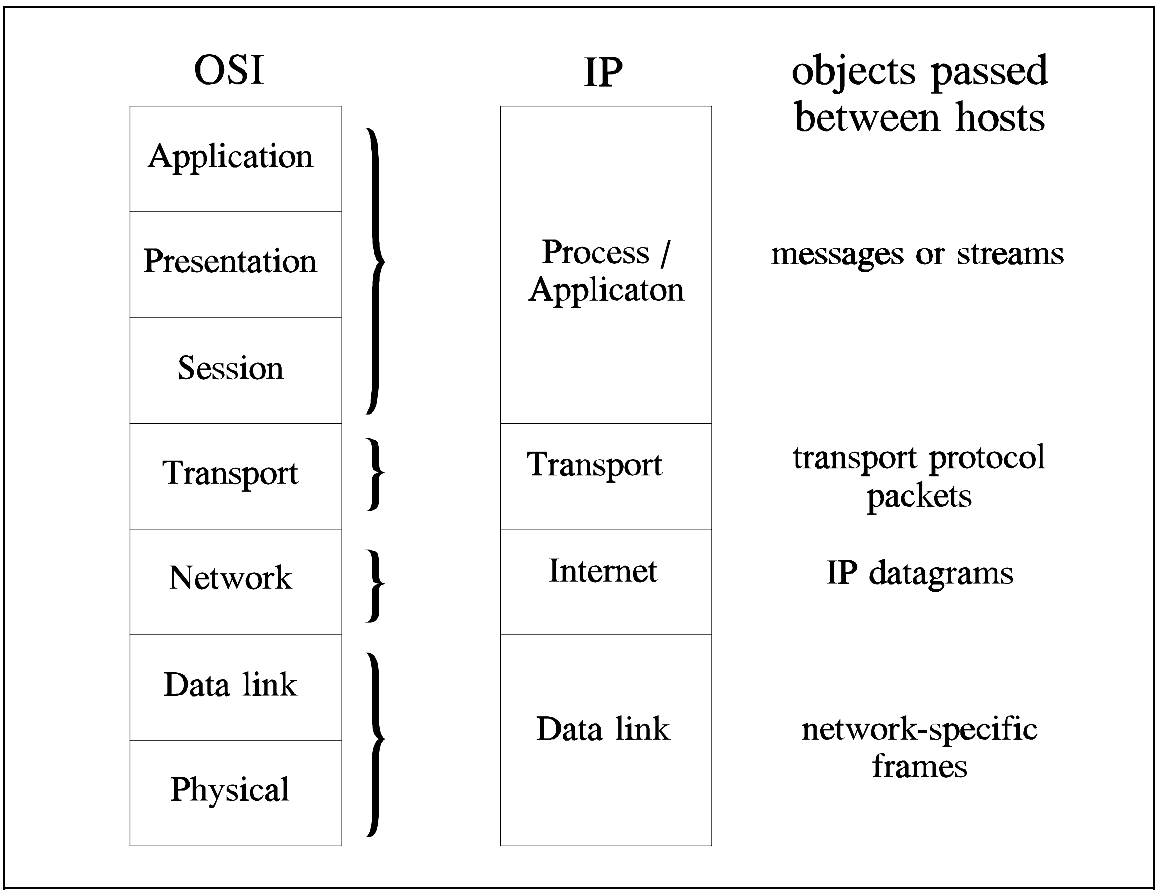

Figure 7.1 Correspondence between OSI and IP protocol layer models, and objects passed between corresponding layers on separate hosts.

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 7.2.

- **Process/Application Layer**. Applications invoke TCP/IP services, sending and receiving messages or streams with other hosts. Delivery can be intermittent or continuous.
- **Transport Layer**. Provide 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.
- **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.
- **Data Link Layer**. Includes 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 7.2. Summary of TCP/IP Internet layers functionality.

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 protocols is part of the Transport layer. One of the two protocols is used as appropriate for the criticality and timeliness of 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. A great deal more can be said about these and related topics. Since internetworking considerations impact all components in a large scale virtual world, additional study of network protocols and applications is highly recommended for virtual world designers. Suggested references include (Internet 94) (Stallings 94) (Comer 91) and (Stevens 90).

E. USE OF SOCKETS FOR VIRTUAL WORLD COMMUNICATION

The most common use of interprocess communications (IPC) among graphics and virtual world component processes is the socket. A socket is not a protocol but rather an application program interface (API) for communication between processes on different hosts (or a single host) via the network layer of the IP suite. Sockets provide a mechanism for passing data that is either reliable connection-oriented stream delivery, or nonguaranteed "best effort" connectionless datagram delivery. Interface details may vary between operating systems but socket syntax remains compatible and reasonably consistent on a variety of platforms.

Sockets originated with the Unix operating system as a way to make network communications syntactically similar to input/output, file and other stream operations. Implementing a connection-oriented socket usually requires three stages: open, read/write and close. Such socket use is not symmetric since sockets follow a client/server paradigm, where the server first opens a port and then waits for a client process to connect so that reliable two-way communication can begin. Normally sockets are used point to point between paired processes, such as tightly-coupled distributed virtual world components.

Connectionless sockets differ in that the ultimate destination address of the client need not be known by the server, with a corresponding lack of error detection and error recovery procedures to ensure reliable delivery. A connectionless approach is preferred when the data stream is continuous or in real time, since subsequent packets will automatically supersede and replace previous lost packets.

Broadcast protocols for socket communication are sometimes used for multiple-entity interaction. However such use is usually unacceptable due to indiscriminate consumption of bandwidth and unnecessary demand on processor cycles. The limitations of broadcast are the principal reasons for the current bottleneck in simultaneous communications among many entities. By way of analogy, consider the possibility that you were able to hear (and had to simultaneously listen to) every person speaking in the building where you work. It would be impossible to carry on any type of conversation since your ability to discriminate between speakers and words would be completely overwhelmed. A similar scenario occurs when large numbers of processes communicate indiscriminately via broadcast protocols: every process must receive and interpret every communication at the highest layers of the IP stack, and voluminous entity traffic produces a computational load that can eventually overwhelm processor capacity. Occasionally broadcast can be useful on a dedicated local area network among specific virtual world components, or among a limited number (dozens or perhaps a few hundreds) of entities. For large entity populations, it is necessary to avoid broadcast protocols and instead utilize multicast protocols, in order to logically partition the communication space and eliminate unnecessary interactions (Macedonia 94b).

F. MULTICAST PROTOCOLS AND THE MULTICAST BACKBONE (MBone)

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 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 prevent 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).

The MBone is a virtual network since it shares the same physical media as the Internet. A specially configured set of multicast-capable routers (mrouters) enables multicast packets to reach networks that have arranged for multicast connectivity. These mrouters can be upgraded commercial routers, or dedicated workstations running with modified kernels in parallel with standard routers. They are augmented by "tunneling," a scheme to encapsulate and forward multicast packets among the islands of MBone subnets through Internet IP routers that do not yet support IP multicast. The net effect of each routing scheme is identical for end users and applications: they can send and receive continuous multicast data streams throughout the MBone, and thus most of the Internet.

The MBone controls multicast packet distribution across the Internet in two ways: multicast packet hops through routers can be limited at the source using an attached time-to-live parameter, and sophisticated experimental mrouter pruning algorithms can adaptively restrict multicast transmission. Network administrators can also logically constrain the threshold capacity of multicast routes to avoid overloading physical link capacity. Multicast packet truncation is performed by decrementing the time-to-live (ttl) field each time the packet passes though an mrouter. A ttl value of 16 might logically limit a multicast stream to a campus, as opposed to values of 127 or 255 which might send a multicast stream to every subnet on the MBone (currently about 15 countries). A ttl field is sometimes decremented by large values under a global thresholding scheme provided to limit multicasts to sites and regions if desired.

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 1500 subnets are connected worldwide. 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. 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. Figure 7.3 shows a typical session directory (*sd*) list of programs available on the MBone.

Figure 7.3 Session directory (*sd*) programs available on the MBone. Note DIS packets for NPS AUV Underwater Virtual World are sent over the whiteboard address (orientation: *dis-auv-uvw*).

Note session specifications in the advertisement window are used to automatically launch and connect video, audio, whiteboard, and DIS-compatible graphics viewer applications.

The MBone community is active and open. Work on tools, protocols, standards, applications, and events is very much a cooperative and international effort. Such cooperation is essential due to the limited bandwidth of many networks, particularly transoceanic links. So far, no hierarchical scheme has been necessary for resolving potentially contentious issues such as topology changes or event scheduling. Interestingly, distributed problem solving and decision making has worked on a human level just as successfully as on the network protocol level. Hopefully this decentralized approach will continue to be successful, even with the rapid addition of new users (Macedonia, Brutzman 94).

G. DISTRIBUTED INTERACTIVE SIMULATION (DIS) PROTOCOL USAGE

The Distributed Interactive Simulation (DIS) protocol is an IEEE standard for communication among entities in distributed simulations (IEEE 93, 94a, 94b). 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 well-suited for general use. Information is exchanged using protocol data units (PDUs) which are defined for a large number of interaction types.

Multicast and broadcast DIS implementations are freely available and have been successfully utilized in real-time virtual battlefield exercises containing hundreds of active human and autonomous entities (Zeswitz 93) (Pratt 93, 94a) (Zyda 93b). Exploiting the features of multicast to logically partition DIS interactions in a manner similar to real world interactions is expected to permit scaling up virtual worlds to include 10,000 or more players (Macedonia 95a, 95b, 95c).

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 can uniquely and completely specify the originating entity. A variety of dead reckoning algorithms permits computationally efficient projection of entity posture by listening hosts. Several dozen additional PDU types are also 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 optionally-addressable open format message PDU type. Message PDUs allow user-specified 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. Of related interest is ongoing research by the Linda project into the use of "tuples" as the communications unit for logical entity interaction (Gelernter 92a, 92b) (Carriero 90). 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 using mechanisms which already exist for the World-Wide Web. This is a promising area for future work.

H. INTERNET-WIDE DISTRIBUTED HYPERMEDIA VIA THE WORLD-WIDE WEB (WWW)

The World-Wide Web (WWW) 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 WWW 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*) multimedia 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 94a, 94b) (Hughes 94) (Vetter 94).

Universal Resource Locators (URLs) are a key WWW innovation (Figure 7.4). A block of information might contain text, document, image, sound clip, video clip, executable program, archived dataset or arbitrary stream. If that block of information exists on the Internet, it can be uniquely identified by host machine IP address,

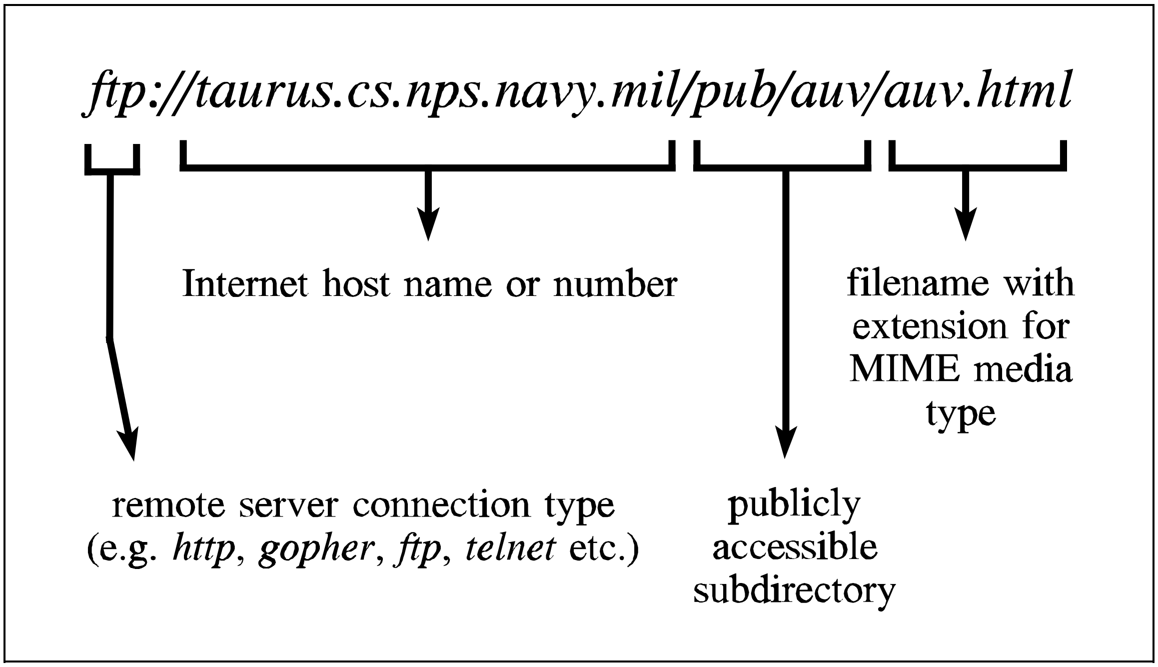

Figure 7.4. Example Universal Resource Locator (URL) components.

publicly visible local directory, local file name, and type of client needed for retrieval (such as anonymous ftp, hypertext browser or gopher). Ordinarily the local file name also includes an extension which identifies the media type (such as *.ps* for PostScript file or *.rgb* for an image). File type extensions are ordinarily specified by Multipurpose Internet Mail Extensions (MIME) (Borenstein 93). Thus the URL completely specifies everything needed to retrieve any type of electronic information resource. Example URLs appear in the list of references, e.g. (Hughes 94).

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 we do not lack content. Mountains of content have become accessible. What is needed now is context, some way to locate and retrieve related pieces of information or knowledge that a user needs in a timely fashion.

The World-Wide Web provides an open and easy way for any individual to provide context for the mass of content available on the Internet. For virtual world designers this is a particularly inviting capability. Virtual worlds are intended to model or extend the real world (Zyda 93a). Access to any media available world-wide can now be embedded in virtual worlds, enabling much greater realism and timeliness for virtual world inputs.

What about scaling up? Fortunately there already exists a model for this growing mountain 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. Furthermore, the structure and scope of a virtual world relationships can be dynamicly extended by passing WWW references over multicast network channels (e.g. as a DIS message PDU). This efficient distribution of information lets any remote user or component in a virtual world participate and interact in increasingly meaningful ways.

Extensions to the World-Wide Web to support globally distributed virtual reality and virtual world functionality are the subject of active investigation (Pesce 94). A Virtual Reality Modeling Language (VRML) specification and implementation is being developed by a large and informal working group (Bell 94). This group hopes to produce public browsers for the exploration of easily and consistently defined virtual worlds. The key components of VRML are likely to be a scene description language (e.g. modified *Open Inventor* file format), existing World-Wide Web functionality (e.g. *.html*), and entity behavior descriptions (e.g. *Open Inventor* engines), augmented by multicast communications (e.g. MBone) and active entity interaction protocols (e.g. DIS).

I. NETWORK APPLICATION IMPLEMENTATION EXAMPLES

Examples of the networked communication methods discussed here have been implemented in a distributed underwater virtual world, designed to support a single networked autonomous underwater robot while permitting any number of human observers. Remote participants use 3D real-time interactive computer graphics as a window into the underwater virtual world. Robot to virtual world communications are performed using a reliable stream socket. The virtual world provides real-time physically-based modeling of six degree-of-freedom vehicle hydrodynamics and sonar. Vehicle position and posture are output using multicast DIS 2.0.3 entity state PDUs. Remote graphics viewers can receive PDUs from any location on the MBone to render robot motion and virtual world interaction, again in real time, seen from whatever viewpoint each individual user might choose. Graphics windows and audio can also be multicast using standard MBone video and voice applications. A diagram of virtual world communication flows appears in Figure 7.5.

On the fly text-to-speech data sonification is provided using a WWW client which relays mission script commands to a sound server in the Netherlands (Belinfante 94). That remote sound server parses arbitrary text strings into phonemes and then generates a corresponding audio file, which is returned to the virtual world for local play. Text-to-speech sound queries are played and saved locally using a filename matching the original text, ensuring that network bandwidth consumption is minimized during repetitive queries.

A WWW home page provides free access to source code, binary executable programs, installation and help guides, reference papers and pertinent images to anyone with Internet access (Brutzman 94a, 94b, 94e). Modifications to the standard MBone session directory configuration file are also provided which enable remote MBone users to participate using the graphics viewer, DIS communications, default video stream, virtual world audio output and *Mosaic* display of the virtual world home page. All of these applications can be launched in concert with the click of a single button on the MBone session directory. As participation in remote virtual worlds

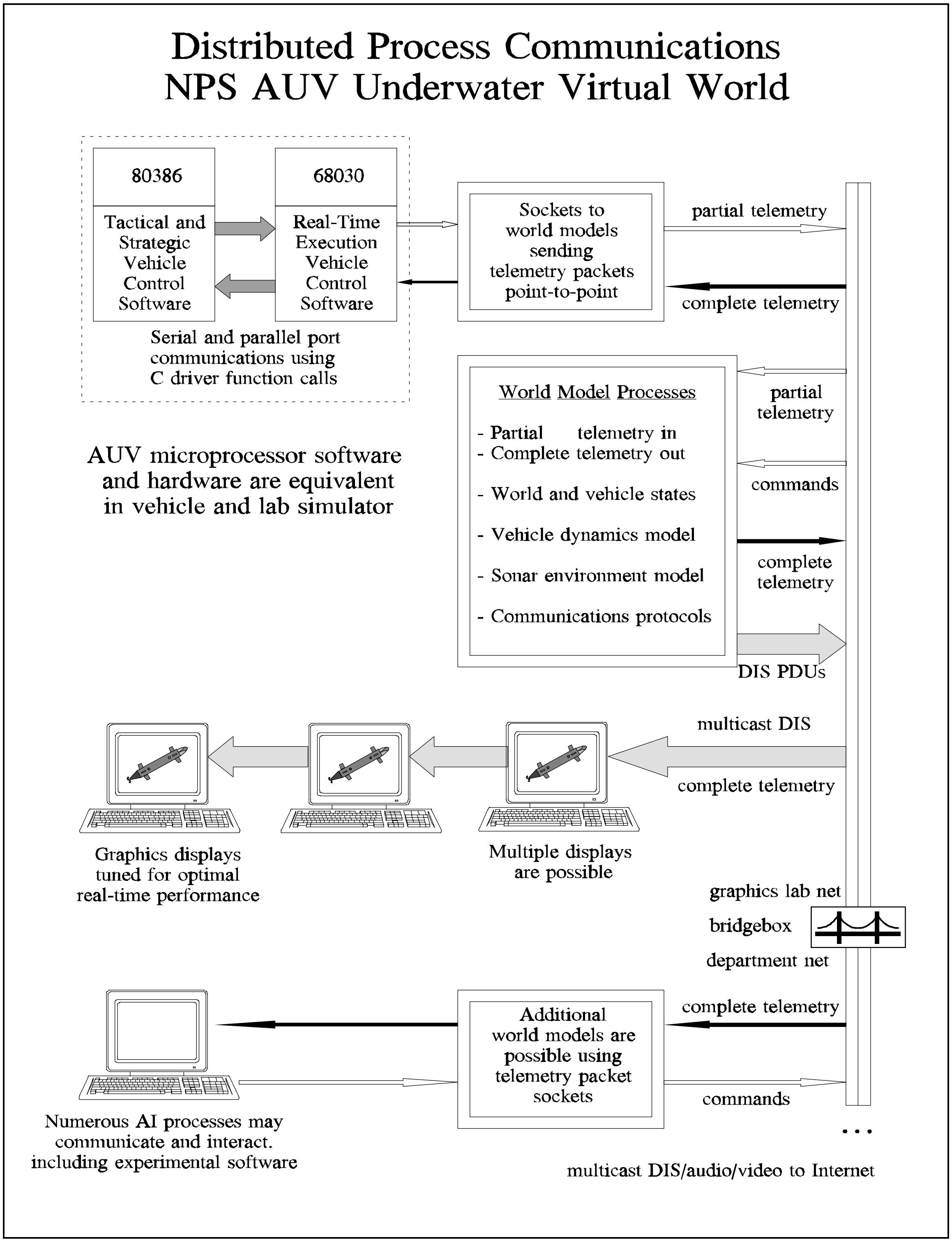

Figure 7.5. Distributed communications in NPS AUV Underwater Virtual World.

approaches the ease of use of a telephone, collaboration and participation in computer graphics-enhanced virtual worlds are expected to grow dramatically (Brutzman 94c, 94d) (Rhyne 94).

It is perhaps startling to hear someone say, "Here is an interactive multimedia television station that you can use to send out computer graphics and virtual world interactions between your desktop and the world." These are powerful concepts and powerful tools that extend our ability to communicate and collaborate tremendously.

J. SUMMARY AND FUTURE WORK

Four network components are proposed as being sufficient for global-distributed virtual world networking: sockets, multicast communications protocols, the Distributed Interactive Simulation (DIS) protocol and World-Wide Web connectivity. Sockets are best used for direct communication among tightly-coupled virtual world components and not for participants. Multicast protocols and the MBone provide efficient Internet-wide distribution of graphics, video, audio and DIS entity state information in a way that permits scaling up to very large numbers of active participants. DIS provides well-defined and standardized ways for physical interaction communications among multiple distributed entities in real time. The World-Wide Web enables virtual worlds to utilize as much of the real world as can be connected to the Internet, both as inputs and outputs.

A myriad of opportunities previously considered impossible are now becoming accessible. MBone, DIS and the World-Wide Web are changing the fundamental nature of the Internet. A distributed approach works both on a human level and a technical level. Scientific collaboration, shared experiences, simulation, training, education, virtual environments, high-bandwidth networked graphics, remote presence and telerobotics are all affected by these capabilities. Implementation of these concepts in an underwater virtual world has demonstrated their feasibility and value.

Open access to any type of live or archived information resource is available for use by individuals, programs, collaborative groups and even robots. Virtual worlds 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 network becomes our computer as we learn how to share resources, collaborate and interact on a global scale.