2003

Requirements for Large-Scale Networked Virtual Environments

Don McGregor, Andrzej Kapolka, Zyda, Michael

þÿDon McGregor, Andrzej Kapolka, Michael Zyda and Don Brutzman Requir
þÿLarge-Scale Networked Virtual Environments, Proceedings of the 7th Int

Requirements for Large-Scale Networked Virtual Environments

Don McGregor, Andrzej Kapolka, Michael Zyda, Don Brutzman
The MOVES Institute
Naval Postgraduate School
Monterey, USA
mcgredo{akapolk|zyda|brutzman}@nps.navy.mil

Abstract—**Networked Virtual Environments (NET-VEs) have in the last few years made a significant impact in the commercial and research world. NET-VEs place heavy demands on the full range of network capabilities, including latency and bandwidth. The next generation of NET-VEs will require more than simply more bandwidth; scaling to significantly larger virtual worlds will require new network services, including multicast. Source-specific multicast has particular promise.**

Keywords—Networked virtual environments, multicast, area of interest management, multicast group allocation

I. NETWORKED VIRTUAL ENVIRONMENTS

In recent years a class of applications termed Networked Virtual Environments (NET-VEs) have become popular. These applications often combine graphics with networked communications to let users interact in a shared virtual environment. A classic example is the game Quake Arena, in which users on different hosts control avatars and attempt to shoot avatars controlled by other participants. The feeling of immersion is enhanced by the use of realistic 3D graphics and sound. Other online games, such as EverQuest or The Sims Online, feature thousands of people interacting simultaneously in a single virtual environment. The designers of those games have chosen to use less sophisticated graphics and sound in order to enhance other aspects of game play, but the goal remains the same: allowing users to interact in a shared virtual environment. These games are well-established commercial products, part of a $10 billion computer game industry. [1]

Another major user of NET-VEs is the military, which has long been interested in using them for simulation and training. Systems based on standards such as High Level Architecture (HLA) [2], and the Distributed Interactive Simulation (DIS)[3] have been used by the military to train soldiers in individual tasks, and by commanders in war games to provide a deeper understanding of future battlefields. Pilots have long used simulators for portions of their training rather than expensive, breakable, and dangerous aircraft. Tank and helicopter commanders practice interacting via simulators in a shared virtual battlefield and use only electrons rather than gas and ammunition—an important consideration when real missiles cost tens of thousands of dollars apiece. Since the training environment is entirely virtual, the participants can practice tactics in environments that it would otherwise be impossible for them to access, such as territory that is currently held by the enemy. And they can practice this in the safety of a computer room rather than a firing range, where accidents with today's highly lethal weapons can be deadly.

As NET-VEs have scaled up to larger sizes (as measured by the number of hosts or humans participating) they have run into problems. Practices that work with small, static virtual worlds become unworkable in large and dynamic virtual worlds. Adding a new capability or feature might require a shutdown and restart of the virtual world, something that is difficult to accomplish across multiple time zones and with sites controlled by multiple administrators. In response, researchers have been working on Run-Time Extensible Virtual Environments (RTE-VEs). These combine all the features of NET-VEs, and can, in addition, dynamically add new code at runtime [4,5]. NPSNET-V, developed at the Naval Postgraduate School in the Modeling, Virtual Environments, and Simulation (MOVES) Institute, is one example of the genre. It uses Java as a platform for a dynamically extensible environment that can add new capabilities at runtime. Developers can add new physics, graphics, or networking modules at will while the application is running, without a shutdown and restart. RTE-VEs are for the most part still research systems, but aspects of them will probably migrate to the commercial world over the next few years, just as the commercial game industry in the 90's incorporated research originally conducted by the US Department of Defense a few years previously.

The commercial game industry and the military are real markets that generate billions of dollars a year in products. They want and expect networks that can handle both this and the next generation of NET-VEs. The NET-VE applications and the network services they use are coupled; new applications will demand new network services, and new network services will enable new NET-VE applications. This paper discusses the nature of NET-VE traffic, and the future network technologies it they will demand.

II. NET-VE NETWORK TRAFFIC CHARACTERISTICS

From a networking standpoint the class of applications encompassed by NET-VEs and RTE-VEs share certain qualities. They usually include avatars that are controlled by a user at a host on the network. The attributes of this avatar need to be communicated to other hosts. In the case of Quake, the avatar's position, orientation, and weapon state need to be sent to the other hosts participating in the game so that they can accurately render the information and present it to the user. Humans may control some of the avatars while scripts, artificial intelligence, or agents may control others. Some implementations include streaming audio or text that allows

7th International Conference on Telecommunications - ConTEL 2003
ISBN: 953-184-052-0, June 11-13, 2003, Zagreb, Croatia

users to communicate with each other. NET-VEs sometimes require downloads of new game level information, terrain information, or geometry, and RTE-VEs may require the download of new code modules.

The need for humans to interact in realistic and natural ways within the virtual environment places bounds on the performance requirements of some network traffic. The humans controlling avatars expect their actions to affect the virtual world in ways that correspond to the real world, with very little lag perceived between a user-initiated action and a result. Actions in the environment that affect the avatar should be communicated to other hosts quickly enough to maintain the illusion of direct manipulation. These qualities contribute to what is called a sense of *immersion*. Users should think that they are interacting with other users in real time, and that their actions have an immediate effect on the world [6]. These dual requirements--communicating updates to other hosts in order to create a shared state and shared virtual environment, and doing it quickly enough to let the humans participating in the NET-VE maintain a sense of immersion—are defining characteristics of the network traffic generated by NET-VEs. The packets that communicate these updates are generally speaking small, binary, and sent at a high frequency over unreliable, best-effort protocols. A Quake avatar might send out UDP updates of its state every 50 milliseconds or so. The rate of packet updates can be reduced through various techniques such as motion interpolation (or dead reckoning), but the general nature of the traffic requirements remains [7].

Streaming audio and video are incorporated in some virtual environments, and each has its own set of problems and challenges [8]. Media streams again often uses unreliable, best-effort protocols, often in conjunction with Real-Time Protocol (RTP). Streaming media are sensitive to jitter and as a consequence often use buffering to reduce the impact of variable packet delivery times.

At the start of a game or during the course of execution a NET-VE or RTE-VE might require a download of level information or code. This traffic is more characteristic of classic bulk transfers, and usually requires reliable transport mechanisms. Often multiple clients will download the data from a single server.

In addition, there is miscellaneous network traffic for administration such as participant registration and security features.

NET-VEs and RTE-VEs can exercise the full range of network capabilities, from frequent low latency, unreliable state updates to bulk transfers of data more reminiscent of FTP or a high traffic web site. From a usability standpoint, though, low latency packet delivery is one thing that must work if NET-VE usability is to be maintained. There can be many workarounds at the application layer that can maintain usability despite slow bulk transfers, but the users will revolt if they cannot maintain the sense of immersion that is enabled by real-time interaction. Low latency is harder to fake at higher network layers.

III. NET-VE ARCHITECTURES: CURRENT STATE OF THE ART

The communications architectures for NET-VEs fall into a continuum between two extremes: peer-to-peer (P2P) and client-server. Application designers can choose one extreme or the other, or they can mix and match, using P2P for some aspects of their design while using client-server for others.

In a P2P design, the avatar on each host directly communicates its update information to all other participants in the NET-VE. There is no central server, and also no inherent central point of failure.

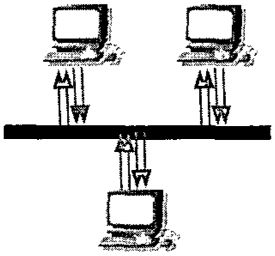

Figure 1. Unicast Peer-to-Peer architecture

The same functionality can be implemented in a client-server architecture using unicast. Each host sends its updates to a server. That server in turn communicates the updates to the other hosts in the NET-VE.

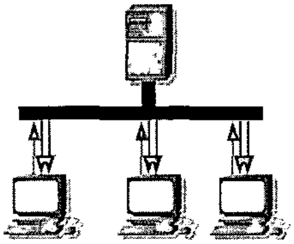

Figure 2. Unicast Client-Server Architecture

Avatar state updates are of course only one aspect of NET-VEs. A system can also require downloads of game level information, terrain data, or code from a central server. Thus, a game may use a P2P design for avatar state updates but a client-server design for game level downloads. The line between P2P and client-server is not always clear. For example, some implementations have a server responsible for a single region of the virtual environment in the game, and hand off clients to another server as the avatar moves to another region. Individual hosts may then be said to have a client-server relationship with their server, but the servers themselves could be communicating in a P2P architecture.

Scaling to more avatars in the NET-VE may cause the network to be overwhelmed by bandwidth use. Typically every avatar must send out a state update at least every several seconds. In some implementations even stationary avatars must periodically send out updates to ensure participants know of their continued existence, or to inform new participants that have just joined the NET-VE of their existence. Many commercial game implementations send out avatar state updates every few tens of milliseconds. By the time the NET-VE has several hundred avatars in the virtual world, even modern, high bandwidth local area networks may be taxed, and the lower bandwidth wide-area links that many consumers have will fail even sooner. Both P2P and client-server architectures will fail as the number of avatars increases, though sometimes in different modes. We can look at the bandwidth requirements of NET-VEs from two perspectives: that of the individual host, and that of the network.

Most commercial NET-VEs today use unicast transport protocols for avatar state updates. In a unicast P2P architecture, each host must send its attribute updates to all the other hosts in the NET-VE and receive data from those same hosts. Assuming all avatars send out updates of the same size and at the same rate, and that there is one avatar per host, this means the use of bandwidth at a single host network interface will scale with (N-1) + (N-1) or roughly 2N. This is represented in Figure 1, with red representing updates sent to other machines and green representing updates received from other machines. One failure mode in scaling to more avatars in the NET-VE as a whole thus becomes the bandwidth available to the least well-connected participant in the NET-VE. Asymmetric DSL implementations allow more bandwidth in the download direction than in the upload direction, so another limiting factor may be number of updates the host is sending out.

We can also examine bandwidth used in the same design from the perspective of the network as a whole. If the updates from all the hosts are sent through a single shared medium, such as a classic unswitched Ethernet link, the medium's bandwidth use will scale proportionally to N^2. It may seem that a switched environment would improve on this, but the bottleneck only becomes less obvious. If all the hosts are connected on a single switch, all traffic will go through the switch backplane, and use of that resource will still scale with N^2. Somewhat paradoxically, highly distributed P2P applications may scale more effectively than an application at a single site. Several Quake players connected through separate ISPs may in fact be communicating over separate point-to-point links with no (or few) shared network media among them, and therefore side-step the N^2 scaling problem. Overall bandwidth use in the network is still scaling with N^2, but that bandwidth is being distributed across multiple independent network links.

In a unicast client-server architecture, the bandwidth used at a client host network interface scales linearly with the number of participants in the NET-VE. The host has to send only a single avatar update to the server for itself, no matter how many other hosts are participating in the NET-VE, and receives updates from the other hosts that have been relayed by the server. But since every host is sending data to the server for distribution, bandwidth use at the server scales with the square of the number of participants.

From the standpoint of the network as a whole, client-server unicast architectures see the same problems with shared network media as P2P architectures, and see the same advantages in distributed applications as P2P architectures. A shared network media that connects all clients to the server will see bandwidth use scale with N^2. But a client host at a user's home communicating with a server on another network will see bandwidth scale only proportionally to N. Unlike the P2P unicast architecture, there is guaranteed to be at least one host that sees bandwidth use scale with N^2: the server.

Client-server unicast architectures can have two failure modes when scaling to more avatars: either the server runs out of bandwidth, or the host participants run out of bandwidth. Note, however, that individual clients use about half the bandwidth compared to the unicast P2P alternative, and that the server may be placed on a high bandwidth link. The client's network traffic is primarily download direction, the direction in which asymmetric networks have the most bandwidth. These considerations help account for the popularity of client-server architectures in the commercial game industry. NET-VEs with up to a few tens of participants and high playability can be created using only network technology that is widely deployed, even with weak home network connections. While the server's bandwidth use scales with the square of the number of participants, often the client network connection will be saturated first.

Limiting the network traffic so that each host receives only the data that is of use to it has helped NET-VEs scale to larger sizes despite limited bandwidth. Often users are only able to interact with nearby avatars--for example other avatars that are in the same room. They can safely ignore network traffic from avatars that are outside the room, or better yet never have updates for avatars outside the room sent to them in the first place. The technique is called Area of Interest Management (AOIM), sometimes also called Data Distribution Management (DDM). [9, 10, 11]

NET-VEs are deployed in both LAN and WAN environments. Players who have never seen each other often play commercial entertainment games across the Internet. But even in this market segment gamers will often gather for "LAN parties," in which users meet, set up a temporary network, and play in an environment with high, dedicated bandwidth, low latency and low jitter, and consistent performance for all users. The purpose for the LAN parties is not entirely technical; the social factor is at least as important. The military can often co-locate hosts used in NET-VEs since the users are often already at one site. NET-VEs with high training value can also sometimes be deployed on dedicated WAN networks that are used exclusively for the purposes of that NET-VE. This reduces uncertainty in network performance by eliminating other network traffic, but requires special network configuration that would be impractical in general purpose networks.

In some situations, broadcast may be a workable solution. Using broadcast with a P2P architecture halves the bandwidth used at a host network interface compared to the unicast alternative—only one avatar state update has to be sent out, rather than N-1. In client-server broadcast architectures, the client has identical bandwidth use at the host network interface. The server's bandwidth use in a client-server architecture using broadcast will be proportional to 2N rather than N^2; the server receives an avatar state update and then sends a single copy back out again. The network as a whole will likewise see bandwidth use scale with 2N.

Broadcast is less than an ideal solution. It does not transit multiple networks well (if at all). In switched environments, broadcast will flood an entire broadcast domain with traffic. Hosts not participating in the NET-VE will process broadcast packets only to throw them away, which wastes CPU cycles. For these reasons, using broadcast in large NET-VEs is only workable on dedicated networks, such as those that might exist at a LAN party or at a single military training site.

Game level downloads, or, in the case of RTE-VEs, code downloads, are primarily bulk transfer oriented. Servers may be overwhelmed when many clients simultaneously request the same information from the same server. The servers may fail either due to lack of CPU or disk on the server, or due to lack of network bandwidth to the server. The first situation can be addressed by the use of conventional load balancing or layer 4 switching to distribute the load to several replicated servers. The bandwidth problem is somewhat more difficult, since bandwidth to the server site itself is limited. The most likely solution to this problem lies at the application layer and the use of replicated, distributed servers at multiple sites.

IV. NET-VEs: Future Network Technologies

As mentioned, most current commercial applications use unicast and a client-server architecture, and in a few instances broadcast. This is primarily due to the current state of network services. As a new generation of NET-VEs are deployed they will generate demand for new network services, and the new services will enable a new generation of applications.

1. Broadband

More bandwidth is the most obvious improvement to networks. As home users go from V.90 modems to cable or DSL connections, NET-VEs will have more bandwidth to utilize, therefore enabling them to implement higher fidelity worlds, worlds with more participants, or both. Broadband connections also have less inherent latency than modems. While client-server architectures can scale to perhaps a few tens of participants when the clients are connected with a v.90 modem, it does not follow that increasing the client bandwidth by an order of magnitude or more will allow dramatically larger NET-VEs—the server's bandwidth use is likely to become the new limiting factor. As pointed out above, unicast architectures do not scale to large numbers of participants very well, and broadcast is not viable for large networks.

The underlying problem is that bandwidth is being used inefficiently in unicast architectures; the data being sent is massively redundant. Eliminating the redundancy through more efficient network transport mechanisms would allow NET-VEs to scale to larger numbers of participants. Broadcast, the most widely deployed technology that can address this problem, does not scale to large networks well. More bandwidth alone, without the corresponding deployment of other network services, will not enable the next generation of NET-VEs.

2. Any-Source Multicast

The same basic architectures—P2P and client-server—will probably still be used in future NET-VEs. The decision about which architecture to use involves much more than calculations of bandwidth use, and many companies have significant investments in server architectures. An alternative to broadcast that avoids many of that technology's problems is Any-Source Multicast (ASM) [12]. ASM allows a host to send data to only those hosts that have joined a specified multicast group. When compared to broadcast, it has much superior performance within a single LAN, in a single campus, and in WANs. Most hosts can handle multicast more efficiently than broadcast by filtering traffic at the Ethernet card level rather than in the operating system's network stack. Many modern switches can perform "IGMP snooping," so that a multicast group's traffic is sent out on only those switch ports with a host that has subscribed to the group. Campus networks and wide area networks benefit from ASM because routers send only a single copy of data on any link, while networks that have no hosts subscribed to the multicast group receive no data.

Conventional P2P and client-server NET-VE architectures can be moved to multicast environments fairly directly. The benefits are significant. In unicast P2P implementations, hosts have to send redundant data to all other hosts that express an interest in that host's avatar. In a multicast implementation the host can simply send one copy of the data, no matter how many hosts are interested. In contrast to unicast implementations, bandwidth use scales proportionally to N rather than 2N, allowing a host to join virtual environments twice the size. In asymmetric network connections, the traffic will be primarily in the download direction, the direction with the greatest bandwidth. From the standpoint of the network as a whole, bandwidth use will scale with N, rather than N^2, and any shared network mediums will see a corresponding drop in bandwidth requirements.

In a client-server implementation the client will see the same bandwidth use as with unicast. The server will use bandwidth in proportion to 2N, rather than N^2, as will the network as a whole.

ASM also provides a convenient mechanism for implementing AOIM. Suppose the virtual space in the NET-VE is divided into smaller regions, and each region assigned its own multicast group, as seen in figure 3 below. Updates to the avatar's state are sent out to that area's multicast group only.

239.1.2.	239.1.2.
239.1.2.	239.1.2.

Figure 3. Using Multicast to Implement AOIM

Network traffic can be cleanly partitioned and each host receives data only for the regions for which an avatar on that host is present. In effect, network cards, switches, and routers are being used to partition the NET-VE traffic into smaller, more manageable data streams.

Deployed multicast routers alone are not sufficient to enable the next generation of NET-VEs. Multicast groups need some way to be uniquely allocated to applications, so that one person's Quake game doesn't interfere with another person's Unreal game. A group of participants in one NET-VE need some way to gain exclusive control of a set of multicast addresses for the duration of the application, and then relinquish control once the NET-VE terminates. The IPv4 class D network multicast range includes 28 bits of address space, or about 268 million unique multicast addresses (some of which are reserved for other uses).

There are several techniques for addressing the multicast allocation problem. Sometimes it is possible to manage multicast group allocations manually within one site if an exclusive range of multicast addresses is made available. Multicast traffic can be made site-local either through the manipulation of packet TTL values or by using the administratively scoped 239/8 address range [13]. Since the multicast traffic is distributed over a much smaller domain under the control of a single authority, administrators may plausibly manually allocate addresses to applications.

Each site that has an Autonomous System Number (ASN) has been allocated the exclusive control of 256 multicast addresses in the 233/8 address range [14]. These addresses can have worldwide scope, rather than site-local scope of the 239/8 address range. Again, it is possible that an administrator could manually allocate multicast groups to applications at that site for those groups the site controls.

Solving the multicast group allocation problem on an internet-wide scale is more difficult. Standards have been proposed to solve the problem, including MADCAP [15]. Session Announcement Protocol (SAP) [16] is also sometimes used to manually reserve multicast addresses.

IPv6 increases the size of the multicast address space to 112 bits. This makes collisions of groups allocated by a random allocation algorithm less likely, but does not lessen the severity of collisions if they do occur.

NET-VEs that use multicast have not been deployed very widely in WAN environments, so the multicast group allocation problem has not been resolved by accepted practice. NET-VEs that use multicast have primarily been research systems, and for the most part researchers have simply manually allocated multicast addresses.

ASM has been deployed on some networks, notably Internet2. But it has not yet been widely deployed in consumer environments. Where native multicast is not available, some have implemented multicast overlay networks [17].

3. Source-Specific Multicast

While ASM is a valuable technology for NET-VEs, it has limited deployment on the consumer-oriented Internet. Source-Specific Multicast (SSM) is an attempt to retain most of the functionality of ASM, but also substantially simplify routing protocols [18]. In SSM, hosts specify the both the multicast group they wish to receive data from and the hosts from which they will receive the data. This corresponds to a very common case in multicast applications, one in which there are few senders but many receivers.

While ASM may be used in both P2P and client-server architectures, SSM is of most use in client-server architectures. It can be used there as a drop-in replacement for broadcast or ASM, with identical performance characteristics. Clients will scale bandwidth use in proportion to N (with most of that in the download direction) and servers will scale bandwidth use proportionally to 2N. SSM limits the number of hosts that can be designated as senders, which makes it less useful for large-scale P2P architectures.

SSM has another important advantage beyond simplified routing protocols: multicast group allocation is far easier. In SSM, receivers subscribe to *both* a source IP *and* a multicast group. Two NET-VEs can use the same multicast group so long as they specify different sources for the data, which greatly reduces the need for multicast group allocation implementations. Combined with the more efficient use of bandwidth, simplified routing compared to ASM, and the ability to distribute applications across multiple networks, this makes a compelling case for NET-VEs in SSM when compared to unicast, broadcast, or ASM transport mechanisms.

V. SUMMARY

NET-VEs are an important application category that will demand much from networks in the coming years. Low latency is important, but not by itself sufficient to enable the next generation of applications. Nor will increased consumer bandwidth by itself allow larger network games. If large scale, distributed NET-VEs are to expand to the next level, they will require the deployment of new network services, including ASM and SSM.

[1] Joseph Moran (2002, November 25). NPD Funworld Releases Video Game Industry Sales Figures. *Game Market Watch* Retrieved January 17 from. http://www.gamemarketwatch.com/news/item.asp?nid=2604.

[2] High Level Architecture IEEE Standard 1516

[3] Institute of Electrical and Electronics Engineers, International Standard, ANSI/IEEE Std 1278-1995, Standard for Information Technology, Protocols for Distributed Interactive Simulation

[4] Michael Capps, Don McGregor, Don Brutzman, Michael Zyda "NPSNET-V: A New Beginning for Dynamically Extensible Virtual Environments", IEEE Computer Graphics, Sept/Oct 2000.

[5] Don Brutzman, Manuel Oliveira, Jon Crowcroft Mel Slater, "Components for Distributed Virtual Environments", Symposium on Virtual Reality Software and Technology 1999

[6] Brooks, Fred, "What's Real About Virtual Reality?", IEEE Computer Graphics and Applications, 19, 6:16-27

[7] Singhal, Sandeep, Zyda, Michael, *Networked Virtual Environments: Design and Implementation*, pp 122-143, Addison-Wesley 1999

[8] Habib, Ibrahim, Saadawi, Tarek, "Multimedia Traffic Characteristics in Broadband Networks", IEEE Communications, July 1992

[9] Macedonia, Michael, Zyda, Michael, Pratt, David, Brutzman, Don, Braham, Paul, "Exploiting Reality with Multicast Groups: A Network Architecture for Large-Scale Virtual Environments", IEEE Computer Graphics & Applications, September 1995.

[10] Morse, Katherine, Zyda, Michael, "Multicast Grouping for Distributed Data Management", SIMPRA-Journal of Simulation Practice and Theory, Elsevier, Fall 2001

[11] Benford, Steve, Greenhalgh, Chris, Lloyd, David, "Crowded Collaborative Virtual Environments", Computer-Human Interaction 1997 Proceedings

[12] Deering, Steve, "Host Extensions for IP Multicast", Internet Request for Comments, RFC 1112, August 1989

[13] D. Meyer, "Administratively Scoped IP Multicast", Internet Request for Comment, RFC 2365, July 1998

[14] D. Meyer, P. Lothberg, "GLOP Addressing in 233/8", Internet Request for Comment, RFC 3180, September 2001

[15] Hanna, S, Patel, B, Shah, M, "Multicast Address Dynamic Client Allocation Protocol (MDACAP)" Internet Request for Comment, RFC 2730, December 1999

[16] Handley, M, Perkins, C., Whelan, E., "Session Announcement Protocol", Internet Request for Comment, RFC 2974, October 2000

[17] Chu, Yang-hua, Rao, Sanjay, Zhang, Hui, "A Case for End System Multicast", Proceedings of ACM SIGMETRICS, Santa Clara CA June 2000

[18] Holbrook, H, Cain, B., "Source-Specific Multicast for IP", Internet-Draft, November 3 2002