pLVE: SUITABLE NETWORK PROTOCOL SUPPORTING MULTI-USER VIRTUAL ENVIRONMENTS IN EDUCATION

Ch. Bouras [1, 2], Th. Tsiatsos [1, 2]

[1] Computer Technology Institute, Greece
[2] Computer Engineering and Informatics Dept., Univ. of Patras, Greece

GR-26500 Rion, PATRA, GREECE

Tel:+30-61-960375,+30-61-996314

Fax:+30-61-996314

e-mail: bouras@cti.gr, tsiatsos@cti.gr

Abstract

Virtual Reality Modeling Language (VRML) and the broad use of communication networks provide the users with realistic virtual environments. However these environments need and a pioneer application such as distance learning. In order to realize suitable educational virtual environments the desired functionality should be collected, a general architecture should be specified and a suitable communication protocol between the components of the system should be designed. In this paper we present the design rationale for such a communication protocol.

1. Introduction

Research and applications in *Distributed Virtual Environments (DVEs)* can be grouped into two camps regarding the performance of computing and networking [9]. On one side we find government supported research with dedicated super-computers and high-speed networks. On the other side we find a large research community that tries to bring DVEs to the regular user. With developments in computing and networking, the working conditions for these two groups approach each other. It is therefore, and increasingly will be, possible to transfer technologies and concepts from the high-end to the low end. Furthermore, DVEs need to be about something; in other words they need a pioneer application. Along with research in development in VR technologies, the importance of useful applications and convincing content should not be forgotten.

A multi-user virtual educational environment should establish virtual communities with a theme, rules, roles, and moderation where useful services can be employed to facilitate educational procedures. It would be available on the actual global computer network infrastructure in use, currently the Internet, in order to contribute in the realistic deployment and take-up of distributed Virtual Reality. This system should be facilitated by security and management mechanisms in order to be used for meaningful purposes such as remote collaborative learning. With the integration of various servers, the proposed system achieves the guaranteed quality of service, which is essential in learning environments.

In order to design such a system should follows the following steps:

- Definition of a learning virtual environment
- Description of the functionality of these environments
- Definition of a suitable system architecture
- Design of a suitable communication protocol

The remainder of this paper is structured as follows. In the next section we define a virtual learning environment and introduce its requirements. We then present an architecture for a virtual learning environment. Following this, we present a suitable protocol for learning virtual environments and finally we present some concluding remarks.

2. Definition and requirements of a learning virtual environment

A simple *Virtual Environment (VE)* is a computer-generated simulation, which aims to provide its users with a sense of realism. More specifically a VE is a computer system, which generates a 3-dimensional (3D) virtual environment, with which the user can interact, in such a way that he receives real time feedback [12].

If multiple users use the VE and they are able to interact to each other the above definition would extended to multi-user or *Shared VE (SVE)*.

An extension of an SVE would be a *Collaborative VE-system (CVE)* which is an SVE aimed at a collaborative task.

According to the above definitions a simple definition of *Learning Virtual Environment (LVE)* is a CVE aimed not only at a collaborative task but also at additional educational tasks such as synchronous and asynchronous learning. A LVE is a set of virtual worlds or a virtual world, which offers educational functionality to its users. The avatars (the graphical representation) of these users populate the LVE and they are provided with additional behavior such as gestures, interaction, movements and sound.

In order to implement an integrated LVE, it should satisfy some basic requirements. These requirements are listed below:

- *High level of presence*. A LVE should offer to its users a high level of presence. The user should represented by an avatar of his choice, which would simulate some basic realistic actions of the users, such as gestures and movement, giving them a shared sense of space, presence and time [13]. The users should have the ability to navigate in a 3D shared space in order to access the content provided, to examine their knowledge, to interact with each other, to exercise their skills and to receive the information provided. Furthermore the user should be informed for the presence of other users (avatars) their arrival in the LVE and their leave.
- *Interaction*. Another important task of a LVE is to provide their users with many types of interaction in order to enhance the development of users as autonomous active learners both in the immediate learning context and in the longer term. Two types of interaction would be defined in a LVE:
    - Multimodal user-to-user interaction: chat, voice communication and gestures. This type of interaction could be supported by manipulation of shared 3D objects.
    - User-system interaction, which would be based on navigational aid and commands that the system should provide to the user for a specific function as well as the manipulation of 3D objects. Furthermore, the users should be able to insert and change objects in the 3D world, sharing these activities with the other users. This type of interaction would offer the user the capability to customize the total design and outlook of the VEs according to the needs of their specific themes. Therefore the user-system type of interaction satisfies the need for customization.
- *Immersiveness*. Regarding the immersiveness, it is true that the immersive applications are more effective in the use of VR technology. However the main feature of educational VR applications is the interactivity and not the immersion [18]. Moreover, a VR application, which is designed for educational use should be suitable for widespread use and mature in the part of the technology. Considering these requirements, immersive VR technology is not mature and it is expensive. On the other hand desktop VR is more suitable for widespread use regarding the hardware and software requirements [3].

- *Scalability.* The LVE must be scalable to a large number of users in order to support large virtual educational communities. This number of users would be divided in each virtual world that is a part of a virtual educational community. Each of these worlds should be able to support a maximum number of simultaneous users.
- *Consistency*. Consistency should be realized by distributing and synchronizing user input as well as user independent behavior in order to achieve the impression of a single shared world.
- *Coherence*. Coherence with the sense of a uniform structure of the provided services, concerning mainly the functional and operational characteristics rather than its visual representation in the VEs.

To achieve these goals, we need to develop a software architecture for learning virtual environments to offer educational services to the users in a sufficient way. To realize this, both a learning virtual environment architecture and a suitable protocol should be specified.

3.Architecture for learning virtual environments

Many SVEs have been proposed [1], [2], [4], [7] and [15]. These SVEs uses different network architectures that have been categorized at [4].
Having in mind the existing networking technologies and models, their advantages and disadvantages as well as the goals and the limitations that an educational system points in the design, we propose a hybrid multi-server communication model, where the task of each server is to offer a specific service.

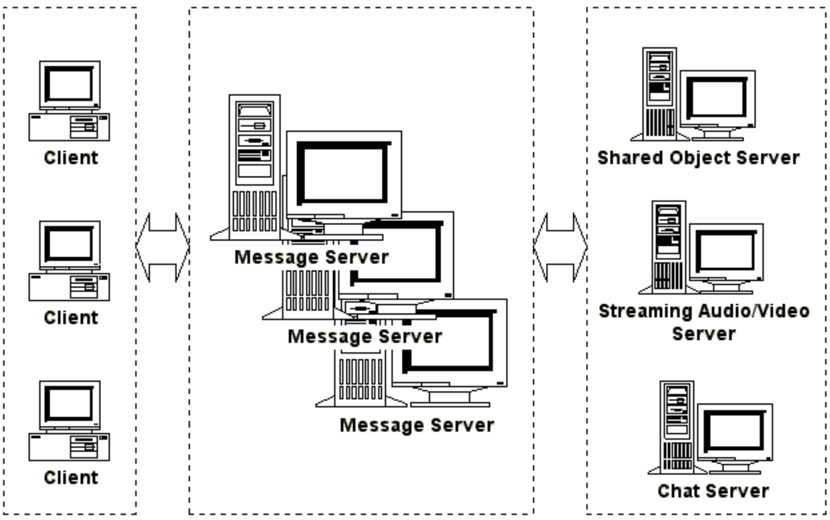

Figure 1: A high level overview of the proposed architecture

This communication model is not a pure client - server or a peer to peer model. A high level overview of the proposed architecture is depicted in the *Figure 1*.

The main idea is to divide the services and not only the users or the virtual worlds as described in the previous paragraph. In addition, the whole system will serve as a virtual representation of the relevant theme and a presentation mean for the available material. This implies that the 3D community, that will be implemented and use this communication model, will consist of a number of smaller VEs. This offers a "segmentation" of the virtual community and led us to design a communication model that consists of a number of message servers. Each message server hosts some specific VEs and it is back-up server for the rest of VEs. The set of message servers constitutes a locus of control of all the system.

This model is well suited for an educational environment because: (a) it offers scalability due to the fact that the load is divided, and servers for additional services can be added, without to affect the end user (b) it offers concerted management and authentication procedure, (c) the clients have not excessive system and network requirements, (d) there is no central point of failure, and (c) it is flexible, because if the number of users is small, some of the dedicated servers to one service can be consolidate in a message server.

In order to support the above architecture a suitable protocol should be designed and implemented.

4.Requirements for learning virtual environment protocols

According to the described functionality a multi-user environment protocol should be able to serve various types of communication data. The main tasks of a multi-user environment protocol according to [5] and [8] are the following:

- The transmission of the virtual world to each user: the transfer of the VE contents to all other users at the first time that they visit a specific world
- To keep the world consistent: the transfer of the changes in specific objects in a VE. With this way the users feel that participate in a single and shared virtual world

The above requirements should be surrounded by additional requirements in order to design a suitable protocol to support educational virtual environments. Such requirements are the following:

- The transfer of the user representation (avatar)
- The transfer educational material in different formats (e.g. pictures, 3D objects, presentations etc.)
- The transmission of audio and video streams

- The transmission of the privileges and the roles of each user in a specific VE

Many protocols for shared virtual environments have been presented. Some of these protocols are the *Distributed Interactive Protocol (DIS)* [10], [11], the *Distributed Worlds Transfer and communication Protocol (DWTP)* [5], the *Interactive Sharing Transfer Protocol (ISTP)* [13], *VS Server Client Protocol (VSCP)* [8], the *VRML Interchange Protocol (VIP)* [17] which is used by the VNet [16] and the *Virtual Reality Transfer Protocol (VRTP* [6]. None of the above protocols satisfies all of the above requirements. The next section introduces a suitable protocol, which specifies the type of interaction between the components of the above described architecture and satisfies the requirements for learning virtual environment protocols.

5. pLVE overview

According the discussion in the previous sections a pLVE should support the following data types:

- Triggers: triggers are messages, which need little bandwidth to be transmitted. They invoke procedures in the corresponding servers, such as predefined object manipulation, viewpoints and additional animations.
- Streams: streams are used for the transmission of audio and video data.
- State update messages: these messages are responsible for the consistency of the virtual world and it includes, among others, the avatar movement and the non-predefined object manipulation.
- Files: files are the 3D virtual worlds, the users' avatars and the additional educational material (3D objects, texts, pictures, etc) provided by the users.

In order to support the above types of data and the previous described architecture the pLVE should be based on the following different components:

- *Message Server (MS)*: The message server has three main tasks: to transmit virtual world contents, to offer scalability to the system and to keep the 3D world consistent. There would be a set of MS that comprise a locus of control of the whole system.
- *Audio/Video (A/V) server*: The A/V server is responsible to provide real-time streaming audio/video capabilities to the whole system.
- *Chat server*: the chat server is responsible for the chat capability.
- *Shared object server*: this server contains all the specific objects that are shared in the 3D virtual environment. Such objects could be a slide presenter, shared files and educational material. The additional educational material, which is provided by the user, is uploaded at this server.

- *The clients*: The client of the system interacts only with the message server and the shared object server.

5.1.The interaction of the pLVE components

The components of the pLVE could be summarized at three main categories: The client, the set of message servers and the application servers (shared object server, chat server and A/V server). This implies that there are two main interactions: message server - application server and client - message server. In addition three interfaces must be described: the client interface, the message server interface and the application server interface.

- *The client interface*: The client would allow the users to connect and interact with other users in the 3D educational environment. The client interface should allow the users to connect to 3D virtual worlds of the 3D educational community, to send triggers to these worlds in order to invoke predefined events, to receive these events, to send/receive state update messages and streaming data, to send their privileges and roles and to send avatar and 3D object descriptions and to upload educational material
- *The message server interface*: Each message server has to interact with the clients, the other message servers and the corresponding application servers. The message server acts as a reflector and a filter that reflects all the messages, which it receives from the application servers to the corresponding users according to their preferences and privileges. As it already stated, all message servers hosts all the VEs that compose the whole educational virtual environments, but each message server is responsible for a small number of these VEs and the rest of message servers are used as back-up servers for these VEs. As a result the message server should be able to send/receive files, triggers and state update messages. In addition each message server should update the rest of message servers for the state of their worlds.
- *The application server interface*: The application servers mainly interact with message servers. These servers should be able to receive/send shared objects, audio video streams and chat messages.

5.2.How pLVE works

In order to describe the operation of the pLVE, the following scenario is presented.
Assume that,

- There are n clients
- The educational environment consists of 9 virtual environments: VE_1 - VE_9
- There are 3 message servers (MS): MS_1 1, MS_2 and MS_3 and each of them is responsible for VE_1 - VE_3, VE_4 - VE_6, and VE_7 - VE_9 respectively. There are 3 application servers: the Shared Object server, the chat server and the A/V server
- The clients 2, 3 and 4 are multicast capable and the rest of them not
- The clients 1, 2, 3, 4, 6 and n wants to participate in the VE_1

The first step for the participation of a VE is the entrance of the users to the world. The client connects to the responsible server using a UDP unicast connection. The first time that the client connects to the responsible MS, it notifies the MS according to its privileges, and network capabilities: unicast or multicast capable. The MS sends the VE contents to the user using the multicast group, which is responsible for the specific VE or using UDP unicasting if the client is not multicast capable. This realized by sending to the user the VRML code. In addition, the MS holds the last events and triggers on specific object in the world, in order to serve new users. When a new user enters the VE the MS sends to it not only the VRML code but also the last events in the world.

In order to reduce the network load of a specific MS and to minimize the probability to be a bottleneck in the network, each MS can reply to a maximum number of no-multicast capable clients. This realized by connecting the rest of MS in the multicast group for the specific VE and using these MSs (the MS_2 for the client 6 and the MS_3 for the client n) to reply to the additional clients. The alternative MSs are selected in turn and controlling their load in order to succeed a balanced system.

The above described interaction may cause some problems in the synchronization of the user, because the latency in the connection of clients that receive the events from the alternative MSs and the clients that receive the events from the responsible MS is bigger. In order to avoid this there are three solutions:

- To refresh periodically the whole world: this may periodically cause an abrupt transmission of data.
- To periodically refresh each of the shared objects in the world, in a round robin way: this has the problem that if there are many objects would be periods that the VE would not consistent.
- When an illegal action (e.g. a request to open a book, which is removed by another client in the VE) in an object occurs to refresh this object: this implies that the clients should sent to the MS acknowledge messages (ACK) which contribute to the network overload.

The best solution is the combination of the last two solutions. The second solution is suitable for a LVE because there is not so many shared objects in a specific VE and if it supported by notifying the illegal actions, the VE will be always consistent.

The communication between the shared object server and the MS as well as the communication between the chat server and the A/V server is based on a 1-1 unicast connection. The client sends a chat message or a request for a specific object to the MS. The MS passes them to the responsible server, they serve the request and sends the results back to the server.

The A/V communication is somehow different. The client sends an A/V stream to the A/V server and a message to the MS. The MS informs the A/V server about the corresponding multicast group. The A/V server send the stream to this multicast group and to the MS in order to send the A/V stream to no multicast capable clients.

All the above-described interactions are depicted in the *Figure 2*.

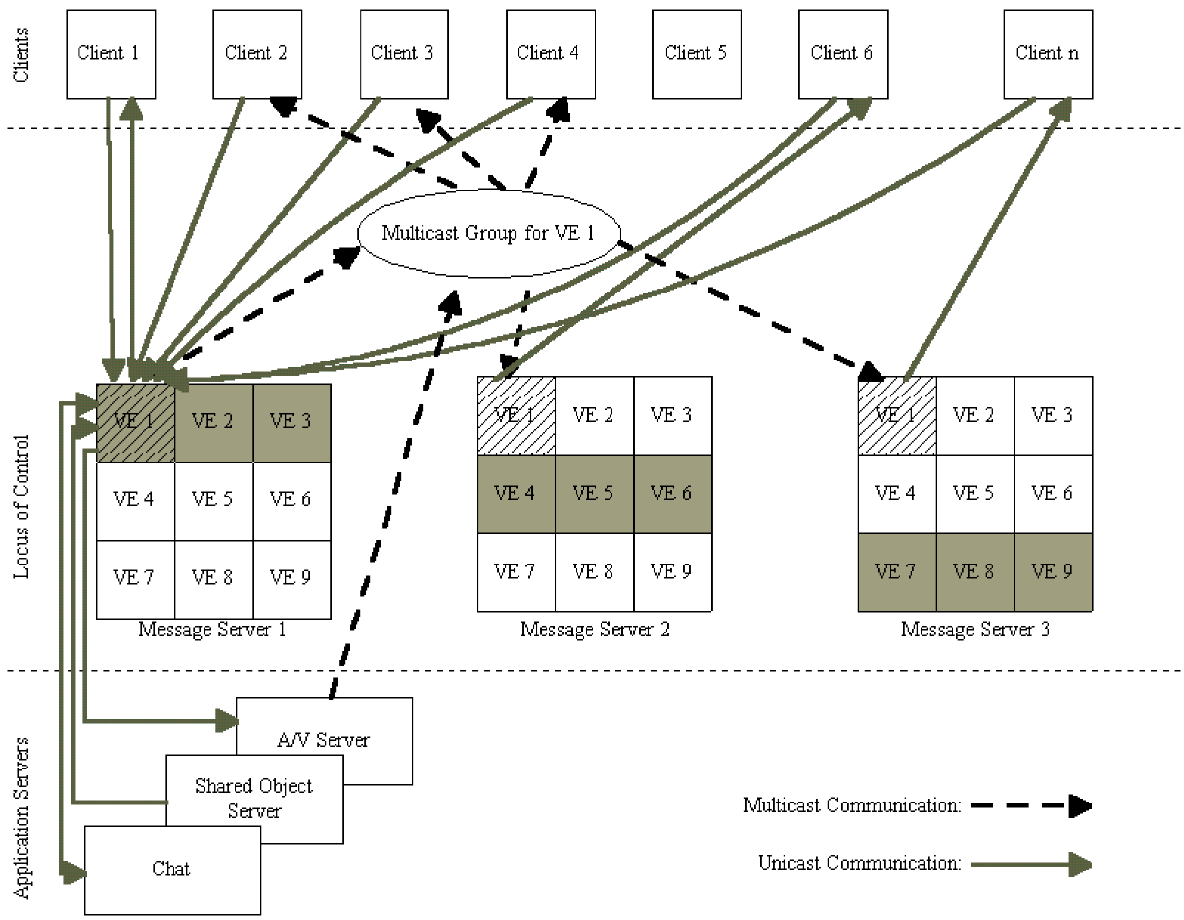

Figure 2: The interaction between the pLVE components

6. Conclusion - Future work

In this paper a learning virtual environment is defined, the functionality of these environments is described, a suitable system architecture is presented and a suitable communication protocol is introduced.

Our next step is the implementation of of a learning virtual environment, which will be supported by the pLVE. The development of this environment would give us the opportunity to deal with many interesting technical issues. In addition to the technical issues, the educational issues that may come up while using this application may be also interesting and help us use in a more efficient way the new ways of communication and interaction that distributed virtual reality technologies offer.

7.References

[1] BARRUS ,J. W., WATERS, R.C., ANDERSON, D.B., Locales and Beacons: Efficient and Precise Support for Large Multi-User Virtual Environments, Proceedings of the IEEE VRAIS'96, IEEE Computer Society Press, Las Alamitos, CA, March 1996.

[2] BLAU, B., HUGHES, C. E., MOSHELL, J. M., LISLE ,C., Networked virtual environments, Computer Graphics, Zeltzer, D. (Ed), 25(2): 1992, pp. 157-160.

[3] BOURAS, CH., PHILOPOULOS, A., TSIATSOS, TH., Using Multi-user Distributed Virtual Environments in Education. World Conference of the WWW, Internet and Intranet - WebNet 2000, San Antonio, Texas, USA, October 30 - Noveber 4, 2000 (to appear).

[4] BROLL, W., Bringing People Together - An Infrastructure for Shared Virtual Worlds on the Internet. Proceedings of the IEEE WE-TICE '97 (Sixth International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises) Cambridge, Massachusetts, USA, IEEE Computer Society Press, Las Alamitos, June 18th - 20th 1997.

[5] BROLL, W., DWTP - An Internet Protocol For Shared Virtual Environments. Proceedings International Symposium on the Virtual Reality Modeling Language 1998 (VRML'98), ACM, ACM SIGGRAPH, pp. 49/56.

[6] BRUTZMAN, D., ZYDA, M., WATSEN, K, MACEDONIA, M., virtual reality transfer protocol (vrtp) Design Rationale. Presented at Workshops on Enabling Technology: Infrastructure for Collaborative Enterprises (WET ICE): Sharing a Distributed Virtual Reality, Massachusetts Institute of Technology, Cambridge Massachusetts, June 18-20 1997, http://www.stl.nps.navy.mil/~brutzman/vrtp/vrtp_design.ps

[7] HAGSAND, O., Interactive Multiuser VEs in the DIVE System, IEEE Multimedia Magazine, Vol 3, Number 1, 1996.

[8] HONDA, Y., MATSUDA, K., REKIMOTO, J. AND LEA, R., Virtual society: extending the WWW to support a multi-user interactive shared 3D environment. Procs. of VRML'95, San Diego, CA. Aug 1995. Also as SCSL-TR-95-035., http://www.csl.sony.co.jp/project/VS/VRML95.ps.Z

[9] LOEKKEMYHR, A., Requirements of Multi-Site Applications of Virtual Reality: Application Areas, State-of-the-Art, Current Practices and Future Trends. European Commission DG JRC ISPRA ISIS/STA, Italy, May 1998.

[10] LOCKE, J., An Introduction to the Internet Networking Environment and SIMNET/DIS", http://www-nps-net.cs.nps.navy.mil/npsnet/publications/DISIntro.ps.Z.

[11] MACEDONIA, M. R, ZYDA, M. J., PRATT, D. R., Exploiting Reality with Multicast Groups: A Network Architecture for Large-Scale Virtual Environments, Proceedings of the IEEE VRAIS'95, IEEE Computer Society Press, Las Alamitos, CA, March 1995, pp. 2-10.

[12] NORMAND, V., BABSKI, C., BENFORD, S., BULLOCK, A., CARION, S., FARCET, N., FRECON, E., HARVEY, J., KUIJPERS, N., MAGNENAT-THALMANN, N., RAUPP-MUSSE, S., RODDEN, T., SLATER, M., SMITH, G., STEED, A., THALMANN, D., TROMP, J., USOH, M., VAN LIEMPD, G., KLADIAS, N. The COVEN project: exploring applicative, technical and usage dimensions of collaborative virtual environments, Presence: teleoperators and virtual environments, MIT Press, Vol.8, No2, 1999, pp.218-236.

[13] SINGHAL S., ZYDA M., Networked Virtual Environments: Design and Implementation. ISBN 0-201-32557-8, ACM Press, 1999.

[14] WATERS, R.C., ANDERSON, D.B., SCHWENKE, D.L. Design of the Interactive Sharing Transfer Protocol. Proceedings of Sixth IEEE Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (June 18-20, 1997, MIT, Cambridge, Massachusetts), pages 140-147. Los Alamitos, California: IEEE Computer Society Press, 1997.

[15] WANG, Q., GREEN, M., SHAW, C., EM - An Environment Manager For Building Networked Virtual Environments, Proceedings of the IEEE VRAIS'95 Conference, IEEE Computer Society Press, Las Alamitos, CA, March 1995, pp. 11-18.

[16] WHITE, S., VNet:Multiuser VRML. http://www.csclub.uwaterloo.ca/~sfwhite/vnet/

[17] WHITE, S., VRML Interchange Protocol - Specification, 1998, http://www.csclub.uwaterloo.ca/~sfwhite/vnet/VIP.html

[18] YOUNGBLUT, CH., Educational Uses of Virtual Reality technology. Institute for Defense Analysis, IDA Document D-2128, January 1998.