

# Distributed Virtual Learning Environment: a Web-based Approach

Christos Bouras

*Computer Technology Institute- CTI  
Department of Computer Engineering and  
Informatics, University of Patras  
e-mail: bouras@cti.gr*

Alexandros Philopoulos

*Department of Computer Engineering and  
Informatics, University of Patras  
e-mail: filo@ceid.upatras.gr*

## Abstract

*In this paper, we present a web-based Distributed Virtual Learning Environment (DVLE) that could be used as a sophisticated tool for distance education and collaboration between teachers and students. We design and develop this application and the corresponding Graphical User Interface (GUI) as a proposal that aims at providing an integrated communicational environment which supplies the communicative needs of an educational community, using the latest telecommunication and multimedia information processing technology. The environment that we propose can be used for virtual lectures, virtual conferences, and exchange of useful material and experiences among teachers and students. It is a web-based application that is being developed as client/server architecture using standard protocols and technologies such as HTTP, HTML, Java and VRML.*

## 1. Introduction

Virtual environments (VE's) provide a way to combine the best features of real-world information navigation - memory of places and visual cues - with the best features of online navigation - fast searches and sorting and quick cross-referencing. By being able to manipulate data the same way we manipulate objects and ideas in real life, we can spend more time creating and understanding things and less time figuring out how to operate them. A well-designed virtual environment can help us experience things in a larger context and pick out useful information from complex systems. Environments that mirror the physical world can show us exactly what is going on and keep us informed.

Furthermore, distributed virtual environments (DVE's) provide gathering grounds for new communities and types of interactions, and they give people a voice like they have never had before. We can share experiences and

visions and learn to understand the other person's point of view. We can attend concerts, act in plays, and attend classes with an international audience. As long as people have something to say to somebody else, they can say it online.

One of the longer-term, but possibly most exciting, opportunity lies in the educational use of the 3D shared spaces for distance teaching and project collaboration. Carefully designed multi-user VE's could be used as collaboration and training tools. One can integrate interactive learning objects into a DVE and have people learn by doing in that environment and communicating with others. The communicative worlds similar to those described above would allow for students and staff to engage in distributed meetings, seminars and tutorials. Such technologies may mitigate some of the problems of isolation that distance learning brings. The students not only access course materials, but also meet in social spaces to discuss them with staff and fellow students. Furthermore, it is possible to provide educational material in many different media formats through a virtual world, giving to the users a unique experience.

The emergence of the Virtual Reality Modeling Language (VRML) [1, 7, 10] as a standard method of modeling virtual reality objects and worlds enables users to download interactive 3D models over the World-Wide-Web. The Living Worlds standard [11] adds multi-user access capabilities to these VRML models and other proposed extensions to VRML [6, 8, 12] enable the development of collaborative DVEs. It is obvious that the needed infrastructure for developing advanced multi-user virtual reality interface systems is already existing. The key issue is to develop innovative applications based on these technologies and to explore all the factors that have to do with the usage of them for certain purposes (educational, business, entertainment, etc.).

This paper describes the concept and implementation of a "Distributed Virtual Learning Environment (DVLE)" as a site in the Internet. In particular, we first provide an overview of the proposed system and the main features of

it. Then, we present the main characteristics and functionalities of the DVLE. The next section gives a detailed information about the technical issues regarding the implementation of the system. Finally, we provide some concluding remarks and the current status our work.

## 2. Overview of the DVLE

The proposed system aims at offering innovative opportunities for the educational use of shared spaces for distance teaching and communication. It provides gathering grounds for interaction and offer the participants experiences through which knowledge could be more easily and pleasantly acquired. The initialization of certain events when a user enters specific areas, the integration of sounds, interacting by clicking on certain objects, the animation of objects in the virtual worlds and the interaction of the worlds with other applications enhances the user's sense of realism. The DVLE provides to the users a shared sense of space, as all participants can be presented with the illusion of being located in the same place, such as in the same room, building, or terrain. It also supports a shared sense of time, meaning that participants are able to see each other's behavior as it occurs.

Based on the above, it is obvious that the DVLE comprises a media over which educational procedures and interpersonal communication could be performed in a manner closer to the end user's need and perception. By simulating well-known everyday life procedures and actions, the DVLE aims at presenting sophisticated services to the inexperienced user. Moreover, the environments could serve as a meeting point among the members of a learning community (e.g. university teachers and students). They provide the means for exchanging ideas, accessing amounts of information and collaborate on learning activities.

There are three basic scenarios of use that the DVLE would support:

- Communication among students and teachers. This scenario anticipates for virtual chat rooms where subject-specific conversations among students would take place and for appropriately designed virtual places, where students can reach and communicate with their teachers.
- Educational procedures. This scenario concerns mainly the lectures given throughout the virtual environments, anticipating for virtual classrooms where appropriate teaching material could be displayed and presented by a teacher to the students.
- Information allowance. Announcement boards, slides' presentations, and links to WWW pages would connecting the DVLE with the amount of information circulated in the learning community and could give

access to educational material and other external services.

## 3. Model and Architecture Description

In order to achieve the above goals our design is based on the following concepts:

- Scalability. The ability to support a maximum number of simultaneous users which could vary according to the specific settings of each virtual world.
- Persistence. Persistence is realized by distributing and synchronizing user input as well as user independent behavior in order to achieve the impression of a single shared world.
- Extensibility. It should be possible to customize an existing VE, before presenting it to the users.
- Openness. It should be possible to interface a VE to external applications.
- Various contents' support. Several forms of data should be supported and embedded in the DVLE.
- Communication. The users in a DVLE should be able to communicate.
- A web-based application implemented with international accepted standards and technologies (HTTP, HTML, VRML, and JAVA).

Actually the DVLE system (figure 1) is a set of different virtual worlds that aim to offer to the teachers and the students the ability to navigate and interact in a 3D shared space. Based on the basic scenarios of use there are two main categories, into which all of the virtual worlds created can be classified:

- Lectures' virtual rooms.
- Subject-specific chat virtual rooms.

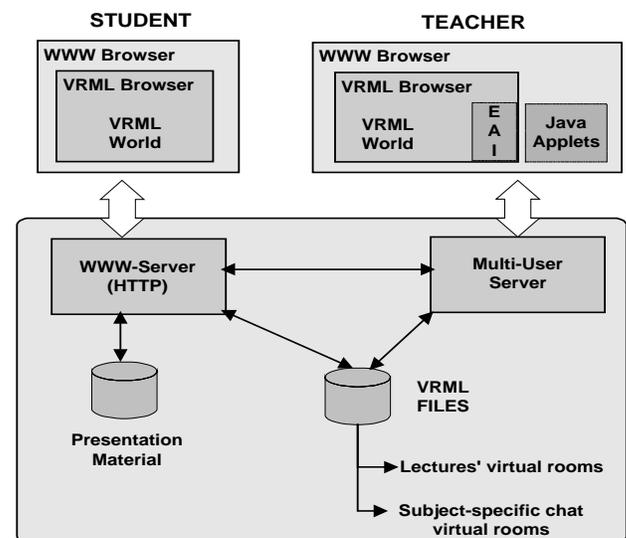


Figure 1. DVLE Architecture

There are also three main entities participating actively or passively in the DVLE system:

- Teachers' entities, having the greatest amount of access and authorities.
- Students' entities, having full access but limited authorities.
- Presentation material entities manipulated by the previous two active entities.

The proposed usage scenario of the DVLE application is the following. The teachers would interact with the whole system through the appropriate user interface, which consists of a typical web browser, a VRML browser and a set of Java applets. This interface could give the teachers the ability to upload learning material to the available WWW server and to initiate and control learning sessions within the appropriate virtual worlds. On the other hand, students will participate in the above mentioned learning sessions through a different user interface with less capabilities. Both user interfaces are easy to use and need no specific knowledge from the point of the user.

#### **4. Functional Specifications of the Lectures' Virtual Rooms**

Certain virtual worlds can act as centres for presenting real time lectures, which are defined explicitly by authenticated teachers. The main purpose of these rooms is to act as a medium for passing knowledge and information from the teacher to the students about the specific subject area that the teacher chooses to present. Therefore these virtual worlds could provide both teachers and students with the facilities that are vital for such an activity. The following are the main characteristics that this facility has:

- The teachers who are interested to use such a room for giving a lecture should first declare it to the administrators of the DVLE in order to define which virtual room is going to be used and give the author the appropriate privileges over that world. After declaring his intention to use such a room, the teacher is provided with a password that he could use to gain control over the world and configure it the way he likes. There is a mechanism for verifying that only the authenticated teacher is able to configure, use and control the learning session. After the successful authentication of the teacher, he is able to use the appropriate interface that gives him the ability to control the presentation of the learning session. Furthermore, the teacher should also declare through appropriate mechanisms (posted messages, e-mail, etc.) to the student community of the system the day and time of his lecture in order interested students to participate.

- Before the students can enter such a world, the teacher who is going to give the lecture should first configure it with the material that will be presented. The authenticated teacher should define the on-line learning session that constitutes from different types of media (images, text, sound, etc.). There is a mechanism for uploading the content that is going to be used to the appropriate web server. Through this mechanism the teacher who is responsible for controlling the learning session defines which media files should be transferred to the web server in order to be viewable from all other members of the learning session. This media files are constructed by the teacher and should meet a general format that will be given to the teachers by the constructors of the world in order to be able to embed them into the world and to project them to the suitable objects.
- The learning session is fully controllable by the teacher and the different media types are presented at the suitable objects within the virtual world. As the teacher changes the content that is presented all the users that participate in the certain on-line session within the world can view the changes. The teacher is able to comment his presentation so that students can understand every time what they see.
- There is the ability for interaction among the teacher and the participating students through the text chat system. The students could ask questions and the teacher would reply to them. The teacher is provided with an additional chat functionality that gives him the ability to control and moderate the conversation between him and the students in the lecture room. The teacher can filter the incoming chat and post selected messages to all members of the group. This functionality gives the teacher the ability to guide the conversation and permit only the messages that he thinks suitable for the specific conversation. This functionality simulates the classroom situation where the teacher is granting or permitting speech to the students.

#### **5. Functional Specifications of the Chat Virtual Rooms**

For the on-line communication of the users over the DVLE platform, there are virtual chat rooms, where only a maximum number of users are allowed to participate. Users are able to enter such rooms using a unique name, navigate inside it, see other students' avatars and enter the on-line text-based conversation. The text chat system has the following functionalities:

- Members of a virtual chat room can anytime be engaged to real-time multilingual chat conversations with each other, which are supported in various ways.

Public chat allows talking or listening to everybody at the same time. Group chat helps to structure the conversations of bigger crowds. Private chats can be initiated anytime among any members. In this way, certain combinations among the users could be achieved (e.g. student-student, teacher-students, teacher-teacher) providing a better and more powerful communication model.

- Announcements. Simple text and image based presentation spaces where all sorts of announcements concerning the conversation topics, meetings and activities can be placed.
- Slides' Presentations. Except for teaching purposes, presentations using slides' projectors can also be used for announcements and notifications. Here, users can place short presentations on subjects related to the discussion themes.
- Connection with different sites. Symbolic representations in each virtual chat room leading the users to certain web sites, with topics relevant to the on-going conversation (e.g. a 3D bookshelf object can serve as a link to a theme-specific library's web site).

All of the above functionalities need some input from the virtual chat room's creator, in order to function properly. This input comprises the presentation material attribute of the virtual room and is properly inserted and embedded into the virtual chat room during its creation procedure.

## 6. Implementation Issues

As already stated the DVLE would be accessed by web browsers through Internet and is being implemented with the use of international accepted standards and technologies, which are presented, in the following sections.

### 6.1. The Implementation of the Virtual Worlds

Virtual Reality Modeling Language (VRML) [1, 7] is used for the creation of the virtual worlds. As known, VRML97 (ISO/IEC 14772-1) [10] has already established itself as a standard for the exchange of 3D descriptions on the Internet. A VRML file (extension wrl) is an ASCII file, which is interpreted by a VRML-compliant browser and converted into a 3D display of the described world. VRML files describe 3D objects and worlds using a hierarchical scene graph. Entities in the scene graph are called nodes. VRML97 defines 54 different node types, including geometry primitives, appearance properties, sound and sound properties, lighting models and various types of grouping nodes. Nodes store their data in fields, and VRML 2.0 defines 20 different types of fields that can be used to store everything from a single number to an array of 3D rotations.

Furthermore, VRML 2.0 defines an event or message-passing mechanism by which nodes in the scene graph can communicate with each other. In particular, most node types have at least one eventIn definition and thus can receive events. Incoming events are data messages sent by other nodes to change some state within the receiving node. Some nodes also have eventOut definitions. These are used to send data messages to destination nodes that some state has changed within the source node. The connection between the node generating the event and the node receiving the event is called route. The ROUTE statement is a construct for establishing event paths between nodes.

Another important feature is script nodes, which can receive events from other nodes, process them, and send events to other nodes. Event processing is performed by a program or script contained in (or referenced by) the Script node's URL field. This program or script may be written in any programming language that the browser supports. The VRML 2.0 specification defines Script node bindings for the Java and JavaScript languages [9, 10]. The Java script interacts with the VRML world through the Java Script Authoring Interface (JSAI). The JSAI allows Java to send events to nodes in the VRML file, create new scene components, and query for information about the scene. The Java script receives events from nodes, which stimulates it into execution, where it can perform algorithms, or utilize other Java packages. The script can then send results of this execution to the nodes as events. This allows Java to provide complex behaviors to the objects in a scene.

Java can also control a VRML world externally, using the External Authoring Interface (EAI). The EAI allows one or more VRML worlds to be added to a Java applet or application. The EAI allows much of the same functionality as the JSAI. But rather than being stimulated by an event coming in to a Script node, the EAI allows a currently running Java applet to control a VRML world, just like it would control any other media. For example, pressing an AWT widget under control of the Java applet could cause an event to be sent to the VRML world, which could change the state of the world.

### 6.2. Adding Multi-user Interaction

For making the virtual worlds multi-user shared spaces an appropriate multi-user technology should be used. In our case, we use blaxxun server [2] who is a multi-user server that supports the operation, administration and usage tracking of virtual worlds, which can be inhabited by many users simultaneously. It is an open system that supports all relevant standards: HTML, VRML, Java, vCard, ActiveX, OCX, Direct3D and OpenGL to enable 3D multi-user interaction and it works together with the blaxxun client (web browser plug-in). The architecture of

blaxxun server offers many basic services (user management, text chat, bots, shared objects, etc.) and a number of optional services (e.g. authorization, and chat moderation).

In addition to this, blaxxun client provides a way of routing VRML events [4] to all visitors of a multi-user place. In this way, all users in the virtual world can view animations and interactions that are triggered by one user. In order to be able to share events, someone should declare in the world the certain mechanism that is provided from blaxxun technology for passing the VRML events that are to be shared first to the server and then distributing them to the other clients.

Finally, Blaxxun client supports a VRML 2.0 Script EAI interface [5], which allows JavaScript or VBScript to control the VRML 2.0 plug-in, running on the same HTML page. In particular, this scripting interface allows reading and writing any fields values of named nodes. Additionally VBScript (in Internet Explorer) and JavaScript (in Netscape) methods can be called from internal VRMLScript script nodes. Furthermore, Blaxxun client supports the proposed VRML 2.0 Java EAI interface, that allows a Java applet to control the VRML 2.0 plug-in. The applet can interact with the VRML world in the following ways:

- Accessing the functionality of the browser interface (i.e. to create new geometry).
- Sending events to eventIns of nodes inside the scene (i.e. to change positions or colors of objects).
- Reading the last value sent from eventOuts of nodes inside the scene (i.e. to find the last position of an object).
- Getting notified when events are sent from eventOuts of nodes inside the scene (i.e. when an object is being clicked).

### 6.3. Technical Description

The blaxxun server in combination with VRML technology gives us a powerful and flexible platform for developing the DVLE with the proposed functional specifications, which are mentioned in a previous section. The DVLE application uses extensively the functionality of shared events of the blaxxun server and the EAI of the blaxxun plug-in. These two techniques give the necessary programming mechanisms that allow us to develop the DVLE as a shared space with the appropriate functionalities.

The main problem in the creation of such a shared space is how to provide mechanisms that can allow to share events among the different users and to give control of these events to one authorised person. For example, the teacher should be the only one that could trigger events (e.g. click on an object and change the slides that are displayed during a lecture) and that these events are

distributed to all other users so as when the teacher make a change in the virtual world all the others will see that change.

The solution to the above problem utilizes the shared event mechanism of the blaxxun server. As it is described in previous section, this mechanism allows sharing events of different types. This means that the amount of information that is distributed to all clients depends on the type of the event that is to be shared. For example, when sharing an event of type SFBool a value of 0 or 1 is distributed, while when using an event of type SFString a whole string is distributed. Therefore a carefully design of the events that should be shared is very important, because a wrong decision could lead to performance problems.

The main idea is to describe in the VRML file and the external programs (Java applets) all the necessary behaviors that are needed for the described functionalities. However, instead of using the standard VRML event routing mechanism for passing the events, that trigger the behaviors, directly to the desired nodes of the world, the event sharing mechanism is used for passing the events first to the blaxxun server and by there to all the clients. In this way, when an event happens in a client's world, this event is distributed and triggers a specific behavior to all the other clients.

## 7. Conclusion

Some of the functional specifications require no time at all to implement as they are offered as basic services of the server (e.g. text chat system, moderated discussions, authentication-registration service, monitoring of the space availability of the virtual rooms, etc.). Furthermore, with the appropriate configuration the server could support as well some other specifications (defining a maximum number of participating users, restricting multi-user scenes, etc.).

Many of the functional specifications require additional implementation time after the virtual worlds are created. The development of the lectures' virtual worlds where certain events and material should be shared among the various users requires the editing of the VRML files in order to include the event sharing mechanism of the blaxxun server. Other features such as the control interface of the teachers, the uploading mechanism, the configuration and authorization mechanisms of the shared space require the development of the appropriate Java and JavaScript programs that can work in combination with the VRML files.

We are now in the phase of the design and implementation of the needed virtual worlds. Certain care is given to the design of the event sharing mechanism in order to avoid possible future problems. The next step is to develop the appropriate Java applets for the described

mechanisms and the corresponding user interface. The development of this application gives us the opportunity to deal with many interesting technical issues, concerning the creation and usage of virtual environments and shared spaces for educational purposes.

## References

- [1] Ames A. L., Nadeau D. R., Moreland J. L., VRML 2.0 Sourcebook, ISBN: 0471165077, 2<sup>nd</sup> Edition, John Wiley & Sons, 1996.
- [2] Blaxxun Interactive, blaxxun Community Server – Executive Overview, 1998.
- [3] Blaxxun Interactive, blaxxun Developer – Shared Events, 1998.
- [4] Blaxxun Interactive, blaxxun Cclient, 1998.
- [5] Broll W., Extending VRML to Support Collaborative Virtual Environments, German National Research Center For Information Technology, 1996.
- [6] Carey R. and Bell G., The Annotated VRML97 Reference Manual, Addison-Wesley, 1997.
- [7] Honda V., Matsuda K., Rekimoto J., Lea R., Virtual Society: Extending the WWW to support a Multi-user Interactive Shared 3D Environment, Sony Computer Science Laboratory Inc.
- [8] Morrison M., *et al.* Java 1.1 Unleashed, 3<sup>rd</sup> Edition, Samsnet.
- [9] VRML 97, International Specification ISO/IEC 14772-1:1997.
- [10] VRML Architecture Group. Living Worlds: Concepts and Context, February 1997.
- [11] Wray M., Hawkes R., Distributed Virtual Environments and VRML: an Event-based Architecture, In Proceedings of the Seventh International WWW Conference (WWW7), Brisbane, Australia, 1998.