

Web-based Virtual Environments for Teaching

Miguel Chover Óscar Belmonte Inmaculada Remolar Ricardo Quirós José Ribelles
Departamento de Lenguajes y Sistemas Informáticos
Universidad Jaume I
Campus del Riu Sec, 12071 Castellón, Spain
{chover,belfern,remolar,quiros,ribelles}@lsi.uji.es

Abstract

The application of virtual worlds through Internet for teaching allows that the student can submerge himself in friendly environments that make learning more pleasant. This article presents three educational applications developed in these worlds. The first, for vocabulary learning, shows a set of virtual worlds designed in VRML where the student interacts with different objects. When an object is selected, its written name can be read and the student can listen to its pronunciation. In the following application, a virtual classroom, the created world can be shared simultaneously by several users. There are diverse materials available as in the traditional classrooms, for example a blackboard, laser pointer and a slide projector. Finally, the latest advances in the creation of virtual environments have been applied in the third application using a game engine. This has made possible the design of a more realistic world where the student can interact with other users and share his experiences.

Keywords

Educational applications, virtual worlds, VRML, multi-user environments, game engine applied to education.

1. INTRODUCTION

The origins of virtual worlds in WWW go back to the beginning of the decade of 1990, with the development of VRML (Virtual Reality Modelling Language). The first version of the standard was VRML 1.0. In 1996, VRML 2.0 (Moving worlds) was introduced at the Siggraph conference. This is the approved international standard hereafter called VRML 97 (ISO/IEC 14772-1:1997) [Carey97].

The use of this standard against other available technologies, such as Java3D, provides as advantage the rapid creation of contents for the Web (Figure 1). The main characteristics of VRML are: the definition of 3D objects in remote locations, the definition of animations, the inclusion of connections to other worlds or Web sites, the modification of the scene by means of events, etc [Lea96] [Roeh97].

Even before VRML 2.0 was presented formally, a large number of working groups appeared to take new lines of design ahead. One of these groups, under the name of *Living Worlds* (LW), implemented a first version of a multi-user environment in which it was possible to insert avatars in scenes defined with VRML 1.0 [Walsh00].

More recently, VRML has evolved towards a new standard known as X3D. The X3D main characteristic is its scalability. This standard will be supported by current Internet browsers and 3D modelling tools [Web3D02].



Figure 1. VRML model of the Jaume I University.

No important contributions have been carried out in connection with the specification of virtual multi-user worlds, from the development of VRML 97 to the X3D standard which appeared in February 2002. In fact, the LW group has disappeared. This has caused the companies intending to commercialise this kind of system, to develop their own software, sometimes outside the VRML standard. Some of them are the following ones:

- *Active Worlds* (<http://www.activeworlds.com>). The first 3D environment in Internet from 1995. Also the home world Alpha World.
- *Blaxxun interactive* (<http://www.blaxxun.com>). One of the most important companies that uses worlds and avatars defined in the VRML standard.
- *DeepMatrix* (<http://www.geometrek.com>). The system is split into a server program running on the web

server where the VRML worlds and supporting HTML files are stored. The client is realised as a Java applet.

- *WorldsServer* (<http://www.worlds.com>). Some of its most important characteristics are: voice- to-voice communication, audio and video streaming, multi-lingual text chat, etc.
- *VRSpace* (<http://vrspace.sourceforge.net>). A free, cross-platform modular 3D community software.

The common deficiency of these systems is the lack of visual realism. Game engines currently offer new advances in real time computer graphics science [Eberly00] [DeLoura00] [Watt00] [Möller99]. Current games can be also played through a computer network, including Internet. These are some of the reasons why they can be a considered good alternative to systems based on VRML. The use of available game engines allows us to have an open modular software and multi-platform that can incorporate the new technologies developed in the field of real time modelling, visibility algorithms and illumination models.

This paper presents three applications related to virtual worlds using different technologies.

- *Vocabulary learning*. This application has been developed using the VRML standard. The user can walk through a virtual world where can find some objects of common use. He can select these objects and, after that, the application presents its name and pronunciation.
- *Virtual classroom*. This application defines a multi-user world using the standard VRML that represents a classroom. This classroom is provided with diverse material like a slide projector, a laser pointer and a blackboard, where the users can write or paint.
- *3D Chat*. This application was developed using the game engine Crystal Space [Tyberghein02]. Each user in this virtual world is represented as a 3D object called avatar and these can interact with each other.

In following sections each of these applications is described. This paper ends showing conclusions and guidelines for future work.

2. VOCABULARY LEARNING

The first application has focused on vocabulary learning. One of the most important challenges for teachers at primary school is the incorporation of a great number of foreign children who do not speak the language of the country. A virtual world has been built up to improve the language learning skills of these children. The user can walk through this world selecting the objects in the scene (Figure 2). After selecting an object, its name is shown and its pronunciation can be heard. In this way, children obtain progressive learning based on the comprehension and oral expression.



Figure 2. Some of the available objects in the world.

The application has been developed using a data base of words that identify basic objects (eg: table, armchair) and actions (eg: run, read). Each record in the data base has the following fields:

- Name of the object.
- Pronunciation.
- URL of the 3D representation of the object.

The system can be easily extended for teaching other languages by simply adding the spelling and pronunciation of the words in the desired language.

Once the data base has been defined, three different worlds with three levels of difficulty for each have been built up. These virtual worlds are called: my home, my city and my world. They include the most common objects in these environments.

3. MULTI-USER VIRTUAL CLASSROOM

The world built up in this application simulates a multi-user classroom. The virtual classroom was designed as accurately as possible to emulate a real one, and the users are represented as avatars. Each avatar can interact with the others and share its experiences (Figure 3).



Figure 3. Snapshot of the virtual classroom.

Specific objects have been added to the classroom, with which the user can interact: a blackboard, a laser pointer and a slide projector.

- The *blackboard*: It is possible to write and draw on it with some basic primitives such as: lines, circles, rectangles and ellipses. The colour of these primitives can be chosen by the user.
- The *slide projector*: It works on a list of predefined images that the user can visualise in a sequential way.

- The *laser pointer*: It allows the users to point either in the slides or in the blackboard. In this way, it is possible to focus the other users on this point of the classroom.

This application has been developed using VRML 2.0 and the node extensions and the multi-user functionality provided by the DeepMatrix system (<http://www.geometrek.com>).

3.1 DeepMatrix system

DeepMatrix is a multi-user application based on the standard VRML. This product is freely available for education or non-commercial use. It allows several users to share a virtual world and to interact with it as they would do in a mono-user world. The system has two parts clearly differentiated:

1. A server program that runs on a machine that acts as a Web server. The VRML and HTML files defining the world reside in this server. A chat service is also included.
2. The client software is a JAVA applet which uses EAI (External Authoring Interface) to communicate with the VRML browser plugin that the client uses (Explorer, Netscape, etc).

Figure 4 shows the way these two components interact.

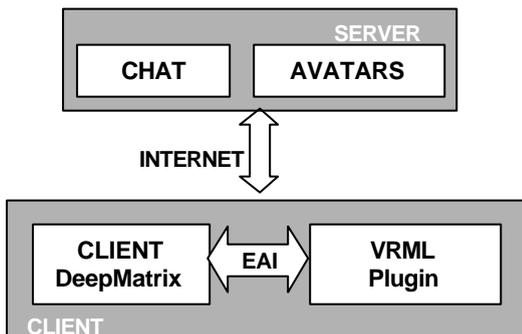


Figure 4. System architecture.

The multi-user virtual classroom is executed within the applet client as it is indicated in Figure 5.

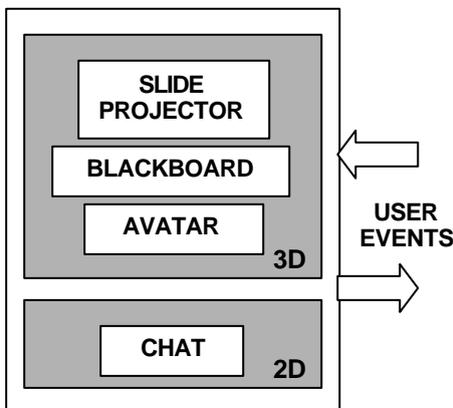


Figure 5. Applet client scheme.

The user generates two types of events through the interaction with the blackboard or the slide projector: *local* events, these only concern to our local world, and *remote*

events, that are transmitted through the network to the DeepMatrix server which distributes them to all the clients. These remote events offer information on the actions that other users are making in the classroom.

4. 3D CHAT

The last developed applications create a 3D virtual world where a user group can communicate by a chat service. In that environment, each user has an own 3D representation or avatar. The application allows the user to walk through the world and to communicate with the different avatars using a chat service.

In order to construct the application, a game engine has been used. This type of software incorporates the recent advances developed in the field of real-time rendering [Moller99]. In this way, it is possible to design more realistic environments with a great number of visual effects.

Crystal Space [Tyberhein02] has been chosen from among the available game engines for the following reasons:

- It is a public domain engine (GNU Library General Public License).
- Open source.
- Multi-platform: Mac OS Classic, Mac OS X, UNIX, Windows, Linux, Irix, FreeBSD, NetBSD, and Solaris among others.
- Good rendering quality either in software or in hardware mode.
- OpenGL and Direct X support.
- In continuous development and maintenance.
- Auxiliary tools (map converters, avatars, etc.)

The world used in our application is one of the example worlds in the Crystal Space. The avatars have been imported from the game "Quake". The environment is shown in Figure 6.



Figure 6. Appearance of the virtual world.

It is not necessary to have graphic acceleration in the computer to run this application, although it is recommended.

4.1 System architecture.

The system has a client-server architecture. The user executes a *Client application* that is fundamentally in charge of the visualisation and the communication with two server applications. One of these servers holds the positions of all the avatars (*Avatar Location Server*), and the other manages the chat service (*Chat server*). The system architecture is in Figure 7.

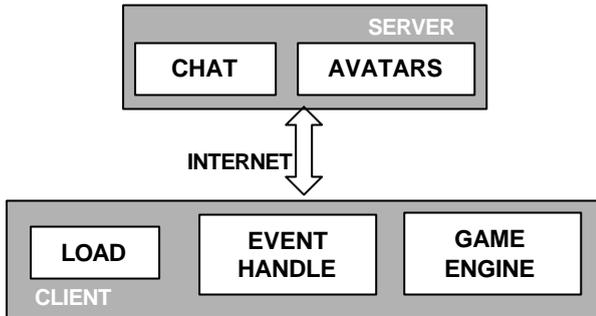


Figure 7. System architecture.

- *Chat Server.* We use a telnet chat service.
- *Avatar Location Server.* The server monitors the state of each client (nickname, position, orientation, time) and communicates this information to the other clients. The pseudocode of this server is shown in Figure 8.

```

for each connection
  if no client assigned to connection
    assign client to connection
  else
    if client sent location update
      update world state
      return world state to client
    end if
  end if
end if
  
```

Figure 8. Avatar Server Algorithm.

- *Client application.* The client is in charge fundamentally of the world visualisation. It manages the communication with the servers in order to show the success of the chat. It draws the avatars in their positions. Figure 9 shows the algorithm executed by the client application.

The client always maintains two states for each avatar: the state at the moment t_0 and the new state t_1 . When the avatars are drawn in the world, they are placed in a position that interpolates both states in order to obtain a continuous visualisation.

Communication management is handled without using blocking communication calls. In this way, the application is simplified because it is not necessary to use threads.

```

For each frame
  Update the camera
  if time_since_last_update > T
    send client status to the server
    draw interpolated world
  end if
  if server sent update
    save the new world state
    interpolate world state
  end if
  
```

Figure 9: Client application Algorithm.

The avatars have two types of animations, the movement animation and the animation that happens when the avatar is quiet. The transition from one animation to another depends only on the speed of the avatar and is solely managed by the client.



Figure 10: Scene of the virtual world with the chat console.

In Figure 10 is shown a scene where two users appear while they are chatting.

5. CONCLUSIONS AND FUTURE WORK

Using virtual worlds for teaching through Internet allows the students to immerse himself in friendly environments that make the learning process more pleasant. The students can interact with the world in a realistic way. For that reason if the advantages of these worlds are added to the advantage to connect by Internet, this produce that the learning process is easier [Brodlie99].

In the virtual worlds, the students learn in an individual way interacting with the computer. However, the users can interact with each other in the multi-user virtual worlds. This allows that learning process to be more complete, because students can share experiences and have a teacher who guides them.

The worlds created with VRML have all these advan-

tages, but if a game engine is used to design these worlds, more realistic environments can be obtained. Some game engines are public domain open-code and they allow us to incorporate all the desired technologies. This is not possible using VRML alone.

The game engines incorporate the advantage of modularity and the applications developed can be executed in a great number of platforms. However, these engines are not properly documented and they still present difficulties at the time of generating contents for the environments.

One of the current lines of work we are working on is the improvement and extension of the educational applications presented in this paper. When the students start using them, their necessities will be monitored and the applications adjusted to meet these necessities.

For example, the next step in the virtual classroom application is to specialise it in Computer Graphics teaching. In order to make this possible, 3D objects will be included in the environment to teach some concepts of Computer Graphics.

The 3D chat application has been designed with an open system for the inclusion of the new technology in the field of the real-time rendering. This application will incorporate in the future a similar system to the virtual classroom with the last techniques of virtual realism [Fairuz01].

6. ACKNOWLEDGMENTS

This work has been partially supported by the collaboration project "Mundos virtuales para la enseñanza del vocabulario" of the State School Jaume I, by the research project "Mundos virtuales multiusuario en Internet" of the Caixa-Castelló Bancaixa Foundation (P1.1B2000-21) and by the project "Modelado Multirresolución en juegos por ordenador" of the Spanish Ministry of Science and Technology (TIC2001-2416-C03-02).

7. REFERENCES

- [Brodlie99] Ken Brodlie, Nuha El-Khalili, Ying Li, "Using Web-Based Computer Graphics to Teach Surgery", GVE99, Coimbra, Portugal, 1999.
- [Carey97] Rikk Carey, Gavin Bell. "The VRML 2.0 Reference Manual", published by Addison Wesley Longman, 1997.
- [DeLoura00] "Game Programming Gems", Ed. Mark DeLoura, 2000.
- [Eberly00] Dave Eberly, "Game Engine Design: A Practical Approach to Real-Time Computer Graphics", 2000.
- [Fairuz01] Mohd. Fairuz Shiratuddin, Walid Thabet, "Can Game Engines JUMP-START the Use of VR Technology in Construction?", AVR II&CONVR, October, 2001.
- [Lea96] Rodger Lea, Kouichi Matsuda, "Java for 3D and VRML Worlds", Ed. Sams.net, 1996.
- [Möller99] Tomas Möller, Eric Haines, "Real-Time Rendering", 1999.
- [Roehl97] Bernie Roehl, Justin Couch, Cindy Reed-Ballerich, Tim Rohaly, Geoff Brown, "Late Night VRML 2.0 with Java", 1997.
- [Tyberghein02] Jorrit Tyberghein, Jorrit Tyberghein, "Crystal Space site". (<http://crystal.sourceforge.net/>), 2002.
- [Walsh00] Aaron E. Walsh, Mikael Bourges-Sevenier, "Core Web 3D", Prentice Hall, 2000.
- [Watt00] Alan Watt, Fabio Policarpo, "3D Games Volume 1: Real-time Rendering and Software Technology", 2000.
- [Web3D02] Web3D consortium, <http://www.web3d.org>.