A VRML-based Environment Used to Learn Virtual Reality Concepts

Sabin Buraga, Ştefan Tanasă, Mihaela Brut
Faculty of Computer Science, "A.I.Cuza" University of Iaşi, Romania
{busaco, stanasa, mihaela}@infoiasi.ro

Abstract. *There are some difficulties to learn subtle aspects of modeling virtual scenes and different programming techniques for creating dynamic virtual worlds. This paper presents a VRML-based environment used for learning virtual reality concepts. The environment uses a complex library of VRML (Virtual Reality Modeling Language) objects and an intranet-style Web site. In section 2 of the paper we describe different types of virtual environments and we present the communication protocols used by VE and the actual standards used to generate virtual worlds. The third section describes the architecture of our environment and the library of over 50 VRML objects. To proper explain the VRML source-code, an annotation Web tool is implemented by using XML mark-ups added to each object. The available objects are used by the students to develop different virtual worlds.*

1. Introduction

In order to learn different aspects of virtual reality at a *Web Technology* course, diverse approaches are used. To give students the possibility to proper understand the virtual worlds and the distributed virtual environments we propose a collaborative learning virtual environment based on VRML. This environment was conceived by the members of a young research group on Web Technologies called *WebGroup*. The group functions inside the Faculty of Computer Science – "A.I.Cuza" University of Iaşi, Romania.

The environment consists of a complex library of VRML (Virtual Reality Modeling Language) objects. The library contains over 50 VRML objects that model primary 2D and 3D objects (e.g. circle, square, cone, pyramid etc.). These objects are augmented by a more complex collection of objects that models common real-world components (i.e. armchair, desk, window, flower, tree etc.).

To properly explain the source-code of each VRML construct, an annotation Web tool is implemented by using XML mark-ups added to each object. The available objects are used by the students to develop different virtual worlds.

2. Faces of Virtual Environments

2.1 Definitions

Virtual reality [1, 2] can be viewed as a software paradigm that offers to one or more users to explore and interact with a computer generated environment. Different types of devices allows to users to perceive and manipulate the visual objects as in the real world. The natural manner of interaction makes the participant to feel embedded in the environment. The virtual worlds are given by mathematics models and software programs.

A **virtual environment** (VE) [1] is the computer generated space, including all objects, static or dynamic ones. An object can have a concrete form or not, can have an attached

sound, a texture, or a movie. (An object that generates a sound can not have a physical representation.)

2.2 Devices for Perception of the VEs

Virtual reality differs from other computer simulations by the necessary use of the interfaces for special devices for image, sound and sensation transmission from environment towards users.

Usually, for viewing the environment the user wears a *head-mounted display* (HMD) or a *retinal display*, with two screens for each eye. HMD contains a locator for user position, direction and orientation. With this information, the computer computes the images from virtual environment and displays them into HMD (It is generated different images for each eye in order to give spatial sensation). The image refresh rate is less than 10 times per second. The scenes from virtual worlds have to be almost simple to allow the computer to bring faster images up to date. For this reason and for the insufficiency of the display devices, the participants at the virtual environment can distinguish the simulation by the real world.

To receive the sounds we have to use head-phones (usually are included in HMD). Depending on the user position and orientation, and objects position that raise sounds, the user percept the audio signals. For example, as we are getting closer to a bird, we hear louder its song; or if we turn around than we percept in a different way the audio signals.

For sensorial perception there is a special glove (*sensorial glove*) which determines the hand position, records the fingers movement and sends it to the computer. The user can touch objects, but he can not feel them. It's very difficult to generate sensations for a person when he touch a hard surface, lift an object or move a finger along a wavy surface. Another device used for sensorial perceptions is the *data suit*.

2.3 Classification of VE

There are several types of virtual environments.

- First, a VE can be **static** or **dynamic**. The static ones contain just fixed objects, that don't have movement elements (e.g. a building). If there is a object that changes its position as the time pass than the environment becomes dynamic (e.g. a pigeon that is flying).
- The dynamic virtual worlds can be **interactive** or **non-interactive**. The user can interact with objects from environment and change their state. These objects can be considered as the part of the environment and can have artificial intelligence features or not. A special class of virtual environments is those who permit to modify the virtual scene, by creating or destroying objects.
- The interactive virtual environments can be **single user** or **multi user**. In the single user case, each user can explore the environment, but he/she can not interact with other users from same place. In fact, there is an instance of virtual world on the user's computer. It is like viewing a HTML document: each user can view independently a web page.

In multi user environments the participants can take different forms, human like or not, called *avatars*. So, each user has a body in the virtual world. The communication between users can be written (text), verbal (speech) or non-verbal (e.g. gestures, mimicry). Those environments that supports non-verbal communication are more complex because is very hard to acquire exactly the gestures from real world and there are too many gestures and minor differences implies major distinctions in interpretation. Each avatar must have a set of default gestures or the capability of making new gestures. Also, there are the more complex virtual

environments that permit direct interactions between connected users (e.g. actions of hand-shaking).

The **collaborative** environments have a special importance. In this case, the characters (avatars) can collaborate to achieve a common scope. For instance, if two participants wants to move a table, then one lift the front side, and the another one the back side, after that they shift on same direction and put down the table. Two or many characters need to cooperate, especially in the context of a learning environment.

2.4 Architecture of Multi-User VE

Usually, the architecture of the multi-user environments is given by the classical client/server model used on Internet. There is a central node that manages the whole activity of the virtual environment. This node serves as a common server in a way very similar with a Web or FTP server. There are more complex architectures in which the clients have in charge administrative activities. So, there are no simple clients, they put in use their software and hardware resources and the hole application forms a dynamic cluster. A cluster is a set of computers that cooperate for simultaneous and distributed execution of an application. The resources and execution are distributed and such virtual reality generated worlds are called **virtual distributed environments**.

When we have a cluster, that depends on the available computational power, the load of the machine, and the quality of the Internet connections, a node gets some objects to manage (e.g. these that are closer). If the number of characters that are connected into the virtual environment increases, then the computational power increases too.

We observe that the cluster-oriented virtual environments can be a good answer to many problems that arise in developing a complex system such as a distributed learning environment. The generated virtual worlds are various, may have many components (objects) in order to increase the grade of realism. The real world can be render better and it will be useful in learning and in different scientific experiments, with less effort and lower costs.

2.5 Communication Protocols

On Internet there are several Web sites that offer three-dimensional interaction between users. The actual virtual environments are used for entertainment, educational or military purposes. There are virtual environments for simulations that are not possible or are too expensive in the real world (e.g. plane pilot training, surgeons training).

The protocols used to interchange the information between the connected users and the environment or between different components (nodes) of the virtual distributed environment are **HTTP** (Hyper Text Transfer Protocol) and **RTP** (Real-Time Protocol) [1, 2].

HTTP is the classical transfer protocol of hypertext information and is used mainly for web content transmissions. RTP is designed for real-time transmissions.

Also, there are different distributed environments that use special and non-standard protocols in order to eliminate some inconveniences from the standard protocols. Notable examples are **VRTP** (Virtual Reality Transfer Protocol) and **VSCP** (Virtual Society Client Protocol) protocols [1].

2.6 Standards for Generating Virtual Worlds

To create a virtual training and learning space, we can use **VRML** (Virtual Reality Modeling Language) [2, 3, 4], which is standard of the Web 3D Consortium and in the same time it is the international standard ISO/IEC 14772-1:1997. ISO (International Organization

for Standardization) and IEC (International Electro-technical Commission) are worked together to elaborate this standard.

The VRML worlds can be viewed by using stand-alone applications, different plug-ins for a Web browser (e.g. Cortona or WorldView), or a special VRML browser. The VRML-based virtual environments usually need different special client software programs (i.e. plug-ins or stand-alone applications) to proper interact with other avatars and objects.

In the present, the Web 3D Consortium is working at a new standard, called **X3D** [4], based on the current VRML 2.0 standards. To assure platform independence and flexible processing of involved data, the X3D documents will be XML compatible and will offer a very good support for programming languages. The specifications will support complex interactions between users.

3. The VRML-based Environment

To facilitate the development of a distributed learning virtual environment, our team is developing a complex library of VRML objects. These objects can be used to build different virtual worlds offered to connected users by the software of the VE.

3.1 Types of Objects

The library models primary 2D and 3D objects (e.g. circle, square, cone, pyramid etc.). These objects are enriched by a more complex collection of objects that models common real-world components (i.e. armchair, desk, window, flower, tree etc.).

In the present, the library has 50 defined objects. To increase the transfer time over the Internet, each virtual world – VRML source-code – is compressed. The compression format uses the standard GZIP algorithm and most VRML players are able to process it.

The students are able to study and to experiment the source-code of each VRML object. Additionally, they can combine different object in order to build diverse 3D scenes that simulate reality. Also, they have the possibility to create other primary objects to be added later to VRML library.

3.2 Modeling Object Properties and Relations

To attach different comments to each written object of the VRML library, an XML-based annotation tool was developed. These annotations can contain metadata documents to express other important information: author, version, name group of the object, copyright and so on. The annotations are public – they can be accessed by everyone – or private – only the author of an annotation is able to view it.

In fact, these comments are XML documents that are easily processed by open-source XML parsing libraries (e.g. Expat or JAXP) [2]. This annotation tool is available on Web as a standard CGI (Common Gateway Interface) program written in Perl language [5].

The annotation tool is planned to be used to search for different objects. Further version of the application will use an XML-based query language to easily locate a certain object of the VRML library.

The VRML objects are organized into tree hierarchies and the relations between them are modeled by XML mark-ups, too. The object trees can be accessed on Web via another CGI program.

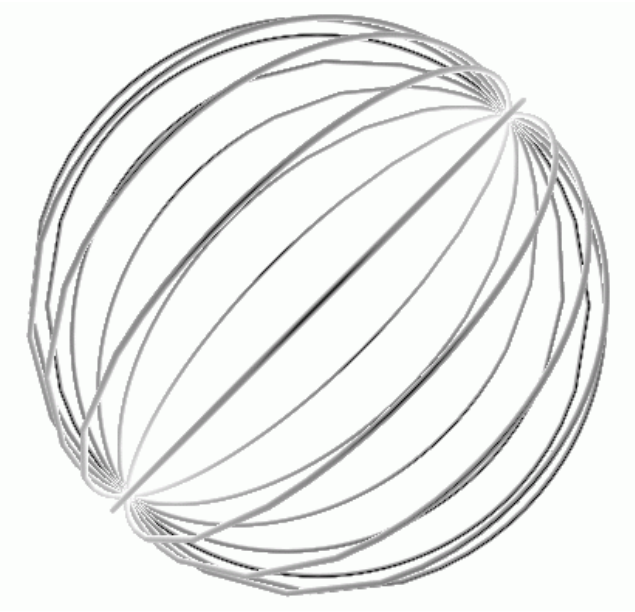

Figure 1 *Example of a Complex Object of the Library*

3.3 Examples of Virtual Worlds

Using the actual VRML objects of the library, a number of ten full available virtual worlds are developed. These worlds model real-world scenes (such as beautiful sceneries or an office place), or simulations (i.e. the Solar system). The conceived virtual worlds are mainly intended to be used in educational VR materials.

Most of them are animated or include audio content and require many computational resources. Each virtual world can be downloaded by the students in order to be studied or modified.

An example of a virtual world that includes several VRML objects is given in figure 2.

3.4 Distributed learning virtual environment

The distributed learning virtual environment consists of a Web site that can be used in the intranet of the Faculty of Computer Science. The Web site is built as a dynamic XHTML code that embeds VRML constructs. The students are able to list all available VRML objects of the library, to download the developed virtual worlds and to study the source-code of each of them by using the annotation tool. Also, an easy to use discussion forum – called *WebDis* – was made in PHP (PHP: Hypertext Preprocessor). This forum has multiple threads of topics and is managed by the course leader. A Web snapshot of the forum is given in figure 3.

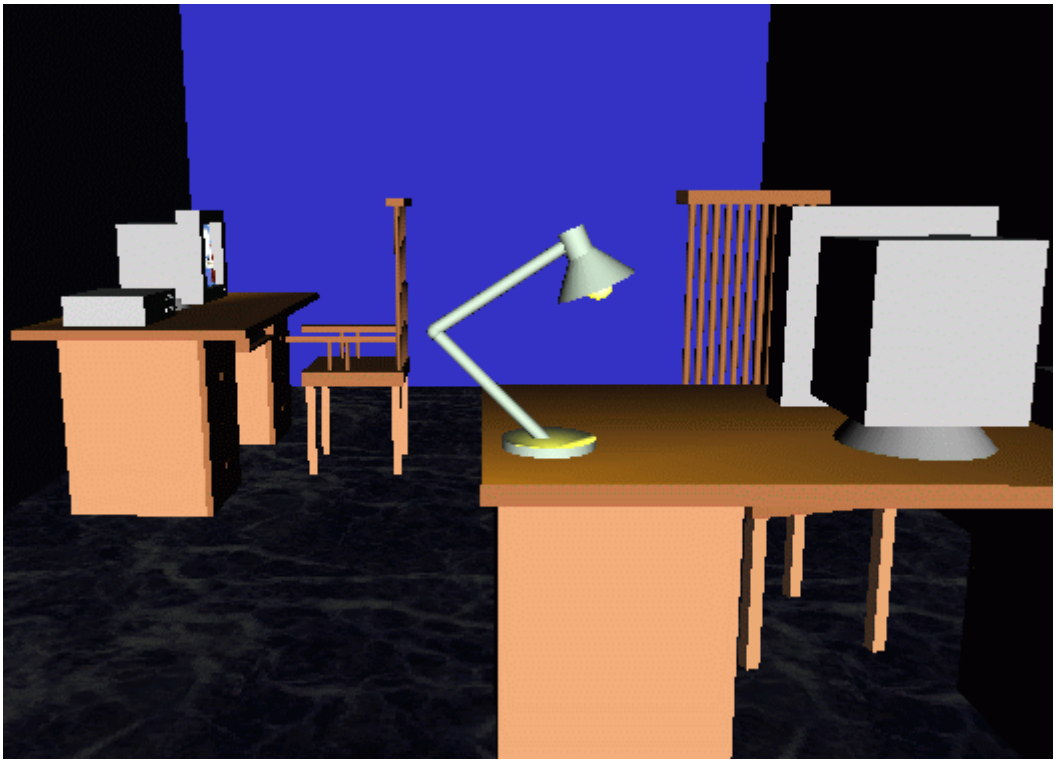

Figure 2 *A VRML-based Virtual Office*

4. Further work

The future plans of our team include:

- Development of other basic VRML objects
- Development of other complex VRML objects and worlds
- Due of severe requirements of hardware resources for rendering of complex animated VRML worlds, we want to experiment another model. The library of VRML objects will be distributed among several interconnected nodes, in a cluster of computers. To accomplish this goal, we need to design another XML format to store relations established between objects and conceived virtual worlds. A Web-oriented tool will be implemented, too.
- The annotation tool can function as a Web service, not only as a CGI script. This approach can give more flexibility.
- Development of a formal model to express interactions between objects and VE or between virtual worlds and users (students and professors). A Petri Nets model is in preparation.

5. Conclusion

The paper was presented a VRML-based environment used for learning different virtual reality concepts. The environment consists of a Web site that can be accessed by the students in the intranet of the Faculty of Computer Science. This environment uses a complex library of VRML (Virtual Reality Modeling Language) objects. The developed objects models simple or complex fragments of virtual scenes that can be used later in whole virtual worlds. To proper explain the VRML source-code, an annotation Web tool is implemented by using XML mark-ups added to each object. This annotation tool is written in Perl language and functions as a CGI script.

Using the environment, the students are able to list all available VRML objects of the library, to download the developed virtual worlds and to study the source-code of each of them by using the annotation tool. Also, an easy to use discussion forum can be accessed.

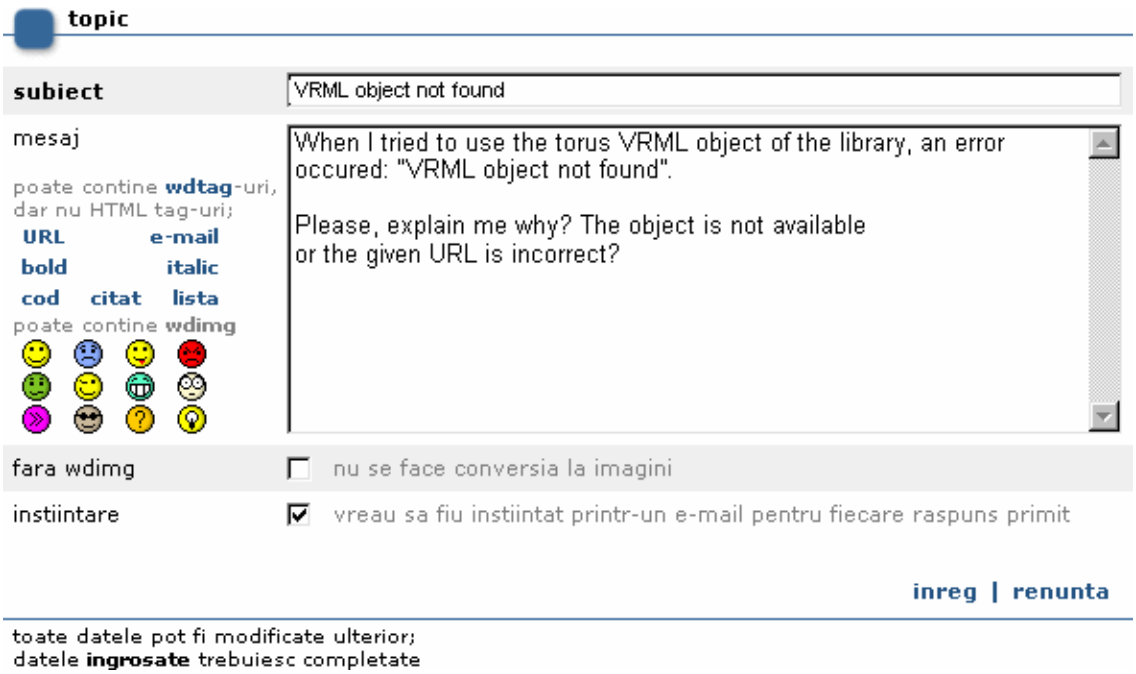

Figure 3. *A message from the WebDis – the Web-based discussion forum of the environment*

Also, the environment can be used for training and knowledge (self-)evaluation purposes.

Future developments will include additional basic and complex VRML objects, a Web service that can be used to annotate VRML source-code, a cluster-based architecture of distributed VRML worlds and a Petri nets theoretical model used to express interactions between objects and VE or between virtual worlds and users.

References

[1] S. Singhal, M. Zyda, *Networked Virtual Environments*. Addison-Wesley, New York, 1999
[2] S. Buraga. *Web Technologies*. Matrix Rom. Bucharest, 2001. In Romanian.
[3] A. Fernandez, *VRML Interactive Tutorial*. Universidade do Minho, Portugal, 1998: http://www-venus.cern.ch/vrmltut/
[4] * * *, *Web 3D Consortium*: http://www.web3d.com
[5] S. Buraga, V. Tarhon-Onu, S. Tanasa, *Web Programming in bash and Perl*. Polirom. Iasi, 2002. In Romanian.