An X3D Framework for Developing Adaptive Virtual Environments

LUIZ FERNANDO BRAGA LOPES
JOSÉ BARBOSA DIAS JUNIOR
EDUARDO FILGUEIRAS DAMASCENO
ALEXANDRE CARDOSO
EDGARD AFONSO LAMOUNIER JUNIOR

UFU - Universidade Federal de Uberlândia
(lfbraga1978,juniorddos)@gmail.com
edamasceno@cefetrv.edu.br
(alexandre,lamounier)@ufu.br

Abstract. This work describes the concepts related to Adaptive Interfaces and presents a framework model for the management of Interaction Interfaces in Virtual Reality Environments. To that aim, characteristics of Adaptive Virtual Environments, as well as the technologies needed for the development of Virtual Reality Interfaces were identified. In addition to that, an Artificial Intelligence model was adopted for the reorganization of information based on user interaction in a specific digital teaching context. The combination of these three elements forms the basis of this adaptive framework, which provides an environment favorable to the development of specific educational approaches trough the generation of individualized contexts.

Keywords: Adaptive Interface, Virtual Reality, Artificial Intelligence, Digital Teaching.

(Received February 21, 2010 / Accepted July 02, 2010)

1 Introduction

Every user in front a computer, especially those in an educational environment, possesses different levels of knowledge, varied needs, abilities and preferences. In this context, the need to develop computational systems that can be adaptable to the diversity of characteristics of each individual user, or groups of users, has emerged from the prospect of individualized education, or processes, that would allow the student to progress according to his or her own interests and objectives, in his or her own pace. This need finds in the use of Adaptive Environments (AE) an important alternative. In an attempt to open the way for user's initial interactive process, an interface design possesses basic instruments which are very similar to those used in 'real-world' architecture, such as: esthetic (graphic elements) or plastic (geometric forms) modeling; consideration and classification of aspects related to technological investigation and innovation; building viability; rationalization of elements and resources; functionality; satisfaction of users' basic needs; and the different means available for execution. Although digital interfaces have become increasingly more complex due to successive additions of resources, most users only use a fraction of the functionalities provided by programs. In most programs, all commands are available to the user at the same time, which significantly decreases the screen's available working area, leading to visual pollution. This quickly becomes a problem when users need more room for their work, like the edition of an image or writing a document [20]. The aim of this work was to contribute to the development of an intelligent architecture that can adapt to the individual needs of each user, at the same time that it provides a "cleaner" interface, one in which only what is essential for managing the activity is available. Thus, some relevant reference work on the different characteristics of Virtual Environments was investigated, and three-dimensional modeling techniques based on XML (Extensible Markup Language) were applied in order to demonstrate the capability of the architecture developed here. This architecture, applied to the study of physics, provides intelligent services in virtual environments that can become adherent to the user through real-time generation of customized profiles, enhancing usability in this application domain. This set of factors boosts interactivity between user and machine, especially in educational, games and entertainment contexts.

1.1 Justification

The use of Virtual Reality (RV) is increasingly becoming more accessible and popular, especially in the teaching area, allowing users to explore more freely the objects displayed on several different presentation devices. This real-time interaction between users and a Virtual Environment (VE), and the possibility of watching scenes being changed in response to commands, as it occurs in video-games nowadays, make the interaction richer and more natural, generating more engagement and efficiency [2]. This adaptivity of objects in a VE can take place based on users' preferences, cognitive ability and navigation styles [6]. The employment of VR technologies, software as well as hardware, in addition to Artificial Intelligence (AI) heuristics, and the adoption of Interaction Methodologies empower the development of interfaces that are more persuasive and adherent to users. An example of this integration is illustrated in Figure 1, which applied to an educational environment can provide very rich interactivity between the interface and the student.

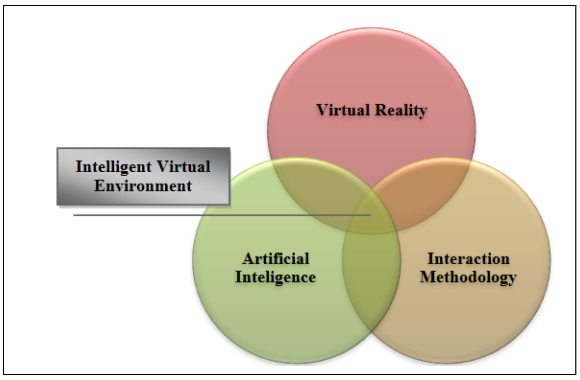

Figure 1: Integrated technologies providing a high degree of interactivity

Therefore, the justification for this work is based on cases where the target audience is very wide and much diversified, as in classrooms, and where there are many different users' models in the resolution of a digital activity.

1.2 Article Overview

Besides the introduction, the present work is organized into seven more Sections, described as follows. In Section 2, Adaptive Virtual Environments, their characteristics and how interfaces can be directed to the user are discussed. In Section 3, some relevant previous work on this matter is presented. Section 4, illustrates the architecture proposed, while Section 5 presents the modules that compose it, the relationship between them and their characteristics. At the end of Section 5, a diagram illustrating the architecture's working is presented. Section 6 illustrates the application of this model in a case study. The conclusions followed by some proposals for future work are presented in Section 7. In Section 8, the references that provided the theoretical foundations used in this work are listed.

2 Adaptive Virtual Environments

Adaptivity in VE, for study and research in the most diverse areas of science, aims at presenting information adapted to the users' knowledge, needs and preferences. A program is said to be adaptive, if it is capable of automatically change its behavior according to the context [8]. Statically designed digital interfaces offer a rigid structure that may not see to the requirements of, or even motivate and guide, users. In contrast, adaptive interfaces can gather information on users during the interface's definition process, at the same time as it is presented to them. The action of gathering information aims at automatically identifying the personal interests of each user, and forming an organized database of interests. The overall objective of these systems is to provide users with update content, subjectively interesting, with pertinent information, size and depth adequate to the context, and in direct correspondence to their individual profile. This profile may be understood as a source of knowledge that contains information, explicitly or implicitly acquired, on all aspects relevant to the user, with the purpose of being used in a software application adaptation process. Adaptivity has been studied as a possible solution to overcome usability problems and for the customization of different characteristics of 3D domains, such as the disposition of geometries and interaction controls [4]. During the development of this wok, some experiments were carried out. The following prototype (Figure 2) illustrates a real-time re-modeling process of a VE based on XML tags.

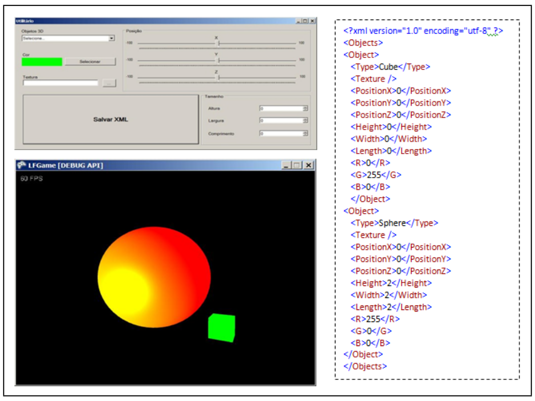

Figure 2: Alteration of objects properties in a virtual environment

An activity displayed on a digital interface has to adapt to users' profiles in terms of necessities or interests, and not the other way round. According to Gardner's Multiple Intelligences theory [7], each individual has different sets of abilities and, consequently, not all of them learn in the same pace. Therefore, it is the educator's role to discover alternative ways to promote the development of the various student abilities, restructuring the relationship between student the educator.

3 Cases of Virtual Environments Design - Domain Analysis

Information collected in the form of preliminary questionnaires is used by adaptivity techniques to customize users' experience. Such techniques are commonly used by Internet Portals in which, by the use of personal identification and a password, pages are shown in an individualized way. A good example is the virtual libraries, which can automatically display subjects previously researched by the user [4]. Hence, a computerized system is capable, to a great extent, of self-management, as control parameters and tolerance limits are established, which provide the conditions to enhance the result intended. Thus, the relevant adaptive systems that contributed to the development of this research are presented below.

In a recent article, a framework for the creation and management of adaptive virtual environments that can be used in real-time was presented. Special attention is given to client-server communication, which makes such an adaptation possible. This framework can support various applications such as environments for education or simulation, where the environment can evolve while the user is performing tasks and acquiring more content. This work's main contribution is the specification and implementation of a client-server interface that is capable of managing virtual environments on the client's machine, and to adapt them in real-time. That case study focus on an environment designed to teach physics to high-school students. In this context, students can interact with experiments, access information on contents and take tests, which are specific to each user knowledge level. The student is guided to interact with each experiment, performing a series of tasks that are proposed by the system [1]. In another article, the intention was to present a software development process model, with corresponding methodology, that is specifically orientated to the development of a small application for a multimedia systems classroom, the so-called virtual laboratories. In this project, named VirtLab, the multimedia reproduction of a real laboratory into a computer system is presented as a virtual laboratory (Figure 3). The virtual components correspond to those of a real laboratory and are, for example, devices, accessories and substances normally found in this task model [16].

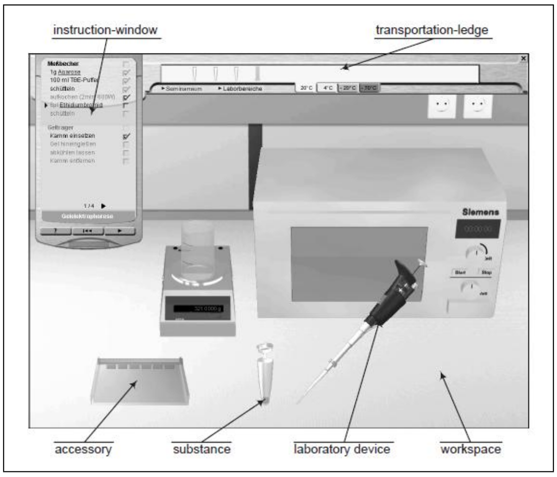

Figure 3: Experiments related to a virtual laboratory workspace

Examples of those factors may be displayed on the interface as wizards, guidance, error management, standard help or audiovisual alerts [9]. Applied to a pedagogic domain, a VE must be sufficiently flexible and attractive to attend all students' needs, function in all levels of ability and academic performance, and engage a variety of intellectual possibilities, so that students may gather all their talents for the resolution of their tasks [21].

4 The X3D-Based Framework

For an interaction design to be successful, the need to involve various technologies leads to the understanding about the importance of how the users act and react to situations, and how they communicate and interact. One of the main benefits of these technologies, such as VR, is to offer alternative ways to represent and interact with information, which is not possible with traditional interaction technologies. Thus, VR has the potential to offer apprentices the possibility to explore ideas and concepts in different ways [14]. The environment developed in this work is characterized by its high interactivity, being capable of altering its behavior in a dynamic way in response to the variations detected when users are carrying out tasks (Figure 4). Therefore, having a pedagogical context in mind, the specific objectives of this work involved: (i) collect information on the handling of 3D interfaces, identifying the characteristics that can contribute to the management of the architecture; (ii) provide an environment composed by objects that offer interactive details relevant to the interface; (iii) apply a computational intelligence model that can supply responses in adequate time, not negatively affecting the feeling of involvement of users; (iv) adapt the interface to users, and not users to the interface, minimizing information overload; and (v) introduce this adaptive model in a wide and diversified area (classroom).

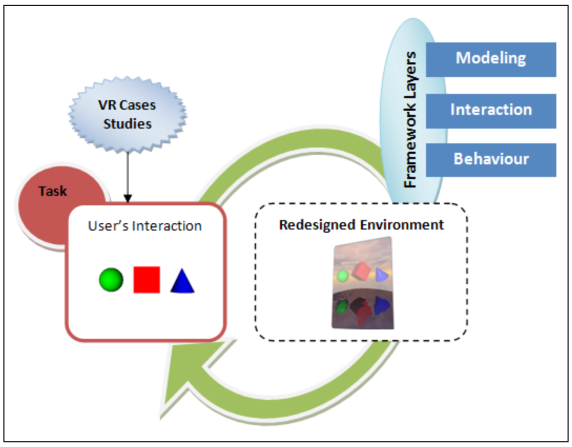

Figure 4: Cycle proposed by the architecture

4.1 The Framework's Technologies

According to the objectives described initially, the framework proposed intends to improve users' didactical and methodological experience. The motivational element for that was the identification of users' adequate profile and problematic interactivities, with the objective of proposing more attractive (alternative) forms of interactions by using different interactive mechanisms. On this point, adaptivity is a particularly important functionality. It can be implemented in many different ways in educational systems in order to recognize the importance of an individual's communication with the information area, and help them to overcome navigation, interaction and learning difficulties. In the following sections, the technologies employed in the framework layers are presented.

4.1.1 The SAI Interface

SAI (Scene Access Interface) is an Application Programming Interface (API) used to establish communication between X3D standard and Java language, as illustrated in Figure 5. Its objective is to enable access to an X3D scene through an external application. As a result, it is possible to insert and remove objects, notify events and perform changes that affect the scene and, consequently, the elements related and external to it. In other words, the scene can be totally controlled in an indirect way through a program written in programming languages or script. Currently, the only SAI specification implementation is the Xj3D browser, an integration technology of continuous development, made available by the Web3D Consortium. It is a toolkit found in a Java Package that can be used to initialize an application browser, enabling the creation and manipulation of content in the scene graph. The interfaces used by SAI are documented in its API and in the X3D specifications [15, 22, 19].

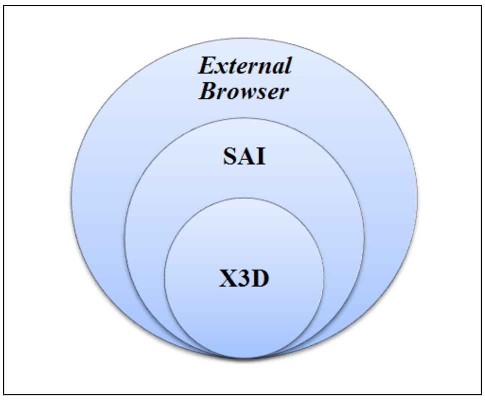

Figure 5: Presentation of the interactive interface organization model

4.1.2 Welcome to Virtual Worlds, the X3D language

The development of real-time 3D data presentation in network and local applications evolved from its origins, with Virtual Reality Modeling Language (VRML), to the considerably more mature and refined X3D standard. X3D is a language that describes 3D scenes in an open standard, capable of representing and communicating objects, whose development was based on the XML syntax [19]. This syntax was chosen because it permits an improved interoperation with the web as well as allowing for the incorporation of new technologies in a standardized way. So that the visualization of X3D files is possible in the browser, it is necessary to install a specific plug-in. X3D technology provides different resources for 3D graphs, such as geometry transformations, illumination, materials, textures, mapping, pixels, vertices and hardware acceleration. In addition to that, it also allows animation with timers and interpolators of continuous conduction, as well as the interaction with non-conventional devices, such as the mouse and keyboard. Navigation in the environment takes place through the use of vision points (scene graph cameras), with characteristics like collision, proximity, visibility, detection and various types of lighting [3].

One of the most relevant aspects related to VR is that users' intuitive knowledge concerning the physical world can be directly transferred to the VE. This enables users to manipulate the information through experiences close to real, i. e., in VE it is possible to create for users the illusion of worlds that exist in reality through 3D representations. As a result, VR has the potential to provide learning through search, discovery, observation and the construction of a new vision of knowledge, offering the apprentice the opportunity to better understand the object of study. Like most computational systems, an important factor related to the success of systems with VR is the guarantee of users' satisfaction concerning these systems usability [18]. This technology, therefore, has the aptitude to cooperate to the apprentice's cognitive process, providing not only theory, but also the practical experimentation of the content in question. Finally, VR systems must present real-time performance, as to guarantee acceptable levels of immersion, interaction and navigation [17]. The Scene Graph that composes the X3D will dynamically evolve according to user input and other dynamic events. Event nodes, usually stored in a branched event structure, when triggered, can be routed into other nodes, such as animation interpolators. Browsers may also support ECMA (European Computer Manufacturers Association) Script or Java, that can interact with both the event models in particular (collision nodes), and with the entire scene graph, creating and manipulating nodes to produce a rich interactive experience for the user [10].

4.1.3 Interaction between the Browser and the Virtual Scene

Users' interaction with the interface is related to the computer's capacity to detect and react to users' actions, promoting alterations in the application. Despite the benefits provided by computational technology, interfaces' sophistication, during many decades, forced users to adjust themselves to the machines. Fortunately, since the beginning of the computer age, researchers have been searching for ways to make machines to adapt to people. This is being achieved with the evolution of hardware, software and telecommunication technologies [12]. The next step illustrates the reception of the 3D scene events with the use of SAI, which allows "listening"to changes in any field of the scene. In order to do so, it is necessary to implement an X3DfieldEventListener interface (Figure 6), included into a Java class. Open source browsers, such as Xj3D [22], chosen for this architecture, provide an attractive and low cost option for the development of applications.

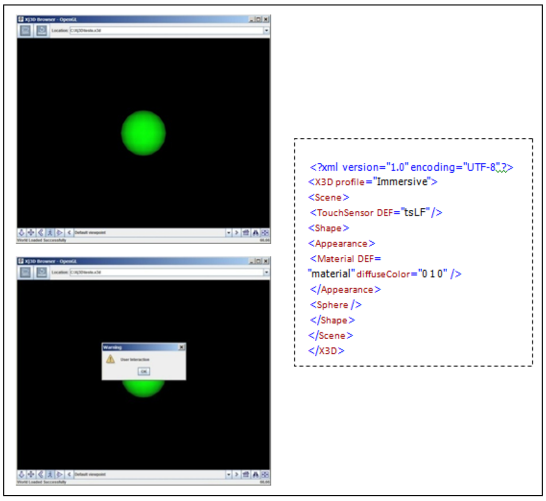

Figure 6: Interaction besides an Xj3D native browser

4.1.4 Dealing with uncertainty, Bayes' theorem

The main objective of the model presented in this work is to form adaptive VEs trough user's interaction with the system. The adaptation strategy employed is responsible for establishing a link between the interfaces and users' profiles, which are used to provide those interfaces' ordination. To that aim, researches in the AI field have been employing several resources for the resolution of information search problems and to recommend the best options. To facilitate this process, these resources apply a variety of search engines, using customizable interfaces, recommendation systems, and intelligent assistants [11]. By just using user interaction, an adaptation strategy would be restricted to modify the interface to the next user access, not permitting changes in execution time. Hence, with the objective of providing a certain level of "intelligence" to the applicative, an AI technique named Bayesian Networks (BN) was used. Triggered by users' actions, the system has the ability to be ahead of users' expectations, offering more or less components related to the activity at hand, or the next one to be displayed, in execution time. The technique, based on the theory of probabilities, permits that the updating of beliefs in response to the evidences that become available, even in situations where there are uncertainties, is satisfactorily dealt with by probabilistic methods. As a result, from a user interaction (e.g., the selection of a link), the adaptation strategy would have the information necessary to define if the distribution and organization of the next links should or not be modified. The adoption of this AI perspective in an architecture design may influence a VE adaptation, in the sense of creating an environment that directs a particular adapted content to the user [5].

5 Implementation - Process Route

From a pedagogical point of view, it has been attempted to develop an architecture that would enhance learning through users' relationship with the environment's graphic interface. Thus, users feel confident in exploring those virtual spaces made available to them. Hence, the Interaction Layer consists of a group of virtual objects, in which each object is associated to an event, or group of events, to assist users in their tasks. The main responsibilities of the objects in this layer are:

a) Present an initial interface - in case students do not possess a user model yet - or a customized interface, based on the current users' models;

b) Receive requests (inputs) from users;

c) Collect information about users, such as features and behavior during the use of the computer or system by monitoring computer use (register programs concomitantly with the beginning of the activity, and monitor the activity's relevant processes), or by requesting this information directly from users;

d) Control scene navigation through, for example, restrictions to the movements of the view point (camera);

e) Pause and re-start the activity, allowing flexibility during the activities' development and administration;

f) Register the activities' running time; and

g) Re-present information to users in a customized fashion (individualized).

The Behavior Layer consists of a set of pieces of information that promotes interface adaptation to users. The elements that compose this layer are the following:

a) Build, represent, and keep adaptation models, according to users' models;

b) Update users' models according to adaptation rules, adopting a retro-feeding system and analyzing the results for possible system effectiveness improvements;

c) Process the information provided by users through an interface object, updating or, in case they do not exist yet, creating users' models;

d) Establish targets or infer suppositions about users, or group of users, based on users' models; and

e) Reformulate users' consultations, tasks and targets according to users' models, and make them available for the corresponding adaptation object.

The Modeling Layer, on the other hand, can be defined as a representation of some visual characteristics of 3D objects included in the scene, as well as the evolution of the interaction interface itself. This layer builds the environment through the observation of users' behaviors when using the system, collecting information on their characteristics and interests. As a result, the system becomes capable of inferring facts (environment re-organization) for users based on their actions. The services provided by this layer are listed below:

a) Encase characteristics, such as the colors and sizes of objects included in the scene;

b) Remodel the visualization environment (scene objects), as well as the interaction environment (navigation controls, interface arrangement);

c) Convert (update) files from VRML to X3D format; and

d) Include sonorous and visual alerts, as well as non-conventional help menu (Figure 7).

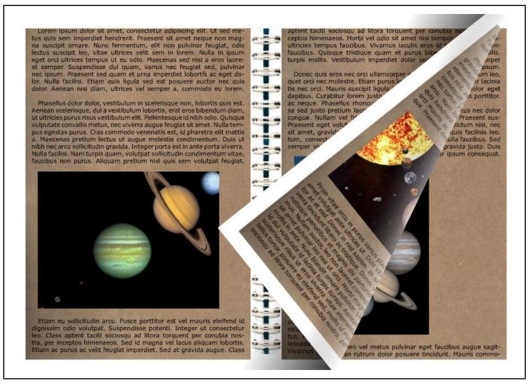

Figure 7: Interface that provides the behavior of a reference manual

6 Case Study - Assessing the Approach

The analysis and modeling work, discussed in the previous Sections, have been validated with a prototypical implementation of the Solar System (Figure 8), loaded in a commercial browser plug-in [13]. The action of uploading an X3D file directly results in the installation of some kind of plug-in, in which the interaction controls are fixedly displayed and in different positions, which can be changed by updating introduced by manufactures.

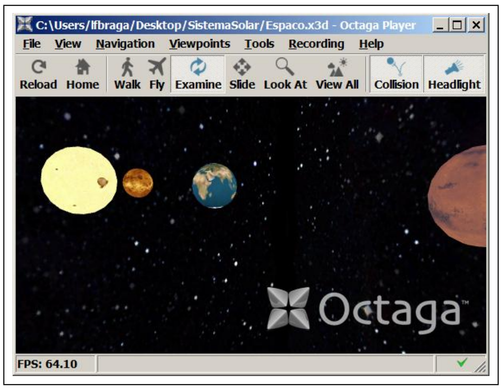

Figure 8: X3D Solar System showed in a commercial plug-in

The process initially takes place with students going through a kind of log in to have access to the system; in case it is their first access, they are directed to register into the system. From then on, the collection of information on users starts with the definition of the file that will contain the data on the interaction. The next interface presents students with various educational contexts.

Taking into consideration that students had already had access to the system, they will be able to proceed to the next task from the beginning, or to continue with some task already started. After choosing the activity, a generic interface is presented along an X3D file that contains the virtual scene. From this point on, as the users interact with the environment, the concepts related to their profiles can be changed, in case an evolution in their knowledge state is identified (Figure 9). According to this state, the system determines if new objects should be inserted or updated in the VE. Thus, the system can adapt itself to users during their interaction with the environment.

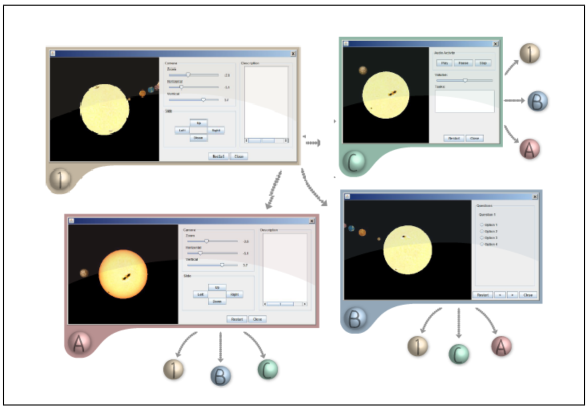

Figure 9: Variations interfaces seeking accommodation to User Profile

Based on the results obtained from the interactions taking place along the process, the environment may lead to the adoption of more adequate strategies to lead students' learning of a particular subject. The objects composed by events and their respective characteristics may be reassessed and altered at any moment. In the end, all the system's processes result in this interface which will have to be unique for each user, containing easy access to all resources that, in thesis, will better contribute for the evolution of their learning.

7 Conclusions and Proposals for Further Research

A question of relevance for the community that researches VEs concerns the difficulty a system has to create and update users' profiles, since, in the great majority of these systems, users tend to have a static profile, i.e., one that do not progress with time, leading to loss of interest. In this work, an architecture that tries to assist in preserving users' interest was presented. Based on dynamic user interaction, it is possible to predict new interests, and surprise users with objects displayed in more attractive areas. This results in a more adequate learning environment to users' individual profile, as well as supplying the conditions for developing different forms of competence, and thus empowering the learning process. Many digital interfaces that require the manipulation of information by users for the performance of tasks were not necessarily designed having users in mind. Actually, they were only typically designed as systems to carry out certain functions. Although they may work effectively, from developers' perspective, generally users end up by being penalized. The objective of an interactive project consists in redirecting this concern. Essentially, it means to develop interactive interfaces that are easy, pleasant and effective to use, always from users' perspective. Therefore, with the advance of digital interaction technologies, it is necessary that systems such as VR are developed in such a way that they may attract and keep the users' attention, as well as present information in an adequate context. With the architecture presented here, it is attempted to contribute to the study of adaptive interfaces for the creation of an information-rich environment, which will increase the level of satisfaction of users when carrying out tasks pedagogical environments.

Based on the results obtained from the interactions taking place along the process, the environment may lead to the adoption of more adequate strategies to lead students' learning of a particular subject. The objects composed by events and their respective characteristics may be reassessed and altered at any moment. In the end, all the system's processes result in this interface which will have to be unique for each user, containing easy access to all resources that, in thesis, will better contribute for the evolution of their learning.

References

[1] AQUINO, M. S., SOUZA, F. F., FRERY, A. C., SOUZA, D. A. C. M., FUJIOKA, R. C., and Vieira, M. M. S. An infrastructure to generate adaptive virtual environments with the management of client-server communication in real time. *Symposium on Virtual and Augmented Reality*, 10:61–69, 2008.

[2] BOWMAN, D., KRUIFF, C., J., L. V. J., and I, P. *3D User Interfaces: Theory and Practice*. Addison-Wesley, 2005.

[3] BRUTZMAN, D. and DALY, L. *3D: Extensible 3D Graphics for Web Authors*. Morgan Kaufmann, 2007.

[4] CELENTANO, A. and PITTARELLO, F. Observing and adapting user behavior in navigational 3d interfaces. In Press, A., editor, *Working Conference on Advanced Visual interfaces*, pages 275–282, 2004.

[5] CHENG, J. and GREINER, R. Learning bayesian belief network classifiers: Algorithms and system. In *14th Biennial Conference of the Canadian Society for Computational Studies of Intelligence,*, 2001.

[6] FRERY, A. C., KELNER, J., MOREIRA, J. R., and TEICHRIEB, V. User satisfaction through empathy and orientation in three-dimensional worlds. *Cyber Psychology and Behavior Journal*, 5:451 – 459, 2002.

[7] GARDNER, H. *Frames of Mind: The Theory of Multiple Intelligences*. Basic Books, 10 edition, 1993.

[8] HENRICKSEN, K. and INDULSKA, J. Adapting the web interface: An adaptive web browser. In *Second Australasian User Interface Conference (AUIC.01)*, 2001.

[9] JURISTO, N., LOPEZ, M., MORENO, A., and SÁNCHEZ, M. Improving software usability through architectural patterns. In *International Conference on Software Engineering (ICSE)*, pages 12–19, 2003.

[10] KURMOS, L., John, N. W., and ROBERTS, J. C. Integration of haptics with web3d using the sai. In ACM, editor, *Web3D*, pages 25–32, 2009.

[11] MONTANER, M., LÓPEZ, B., and de la ROSA, J. L. A taxonomy of recommender agents on the internet. In *Artificial Intelligence Review*, number 4, pages 285–330. Springer Netherlands, 2003.

[12] PINHO, M. S., KOPPER, R., NI, T., and BOWMAN, D. Design and evaluation of navigation techniques for multiscale virtual environments. *IEEE VIRTUAL REALITY*, pages 24–31, 2006.

[13] Player, O. P. Octaga and joint collaboration, 2010.

[14] PREECE, J., ROGERS, Y., and SHARP, H. *Interaction Design: Beyond Human-computer Interaction*. John Wiley & Sons, 2 edition, Jan 2007.

[15] SAI. Extensible 3d (x3d) - iso/iec 19775-2:2004. http://www.web3d.org/x3d/specifications/ISO-IEC-19775-2-X3D-SceneAccessInterface/, 2010.

[16] SCHERP, A. Software development process model and methodology for virtual laboratories. In *IASTED International Multi-Conference 'Applied Informatics' (AI 2002)*, pages 18–21, 2002.

[17] SEO, J. and KIM, G. J. Design for presence: A structured approach to virtual reality system design. In *Teleoperators and Virtual Environments 11*, volume 11, pages 378–403, 2002.

[18] SMITH, S. P. and HARRISON, M. D. User centred design and implementation of virtual environments. *Human-Computer Studies*, pages 109–114, 2001.

[19] Standards, X. I. S., 2010.

[20] STUERZLINGER, W., CHAPUIS, O., PHILLIPS, D., and ROUSSEL, N. User interface façades: towards fully adaptable user interfaces. In Press, A., editor, *UIST '06*, 2006.

[21] WISKE, M. S. *Teaching for Understanding: Linking Research with Practice*. Jossey Bass Education Series, October 1997.

[22] Xj3D –Project of the Web3D Consortium. http://www.web3d.org/x3d/xj3d/, 2010.