

Use of 3-D virtual environments to support the learning of programming

Ricardo Antunes^{*1}, Benjamim Fonseca², Paulo Martins² and Leonel Morgado²

¹ Escola Superior de Tecnologia e Gestão de Leiria, Morro do Lena-Alto do Vieiro, Apartado 4163, 2411-901 Leiria, Portugal

² Departamento de Engenharias, Universidade de Trás-os-Montes e Alto Douro, Apartado 1013, 5001-801 Vila Real, Portugal

The research and development of approaches to help students overcome the difficulties of acquiring good programming skills is a concern common to many institutions and researchers worldwide. One such approach is the use of virtual learning environments that allow students to freely explore several contextualized learning activities provided to them. We propose the use of scriptable 3D collaborative virtual environments in particular, as a way to provide support and context for collaborative programming activities. One of the main problems regarding the use of such virtual environments in a formal educational environment is that of connection between real-world goals stemming from collaboration, and their execution in virtual environments. The coordination of efforts typically follows methods of communication amongst participants that are supplied by traditional systems, external to the environments where interactions are taking place. Here we analyze this problem and a pathway to its study.

Keywords collaborative virtual environments; learning of programming; collaborative programming.

1. Introduction

The concept of inserting a person in a computer-generated world or environment was introduced by Ivan Sutherland in 1965 [1], and implemented in 1968 using a head-mounted stereoscopic display [2]. Since then, various systems providing virtual worlds/environments have been developed, leveraging progress in computing power and graphics technology, and extending the concept to simultaneous hosting of several different users.

However, there are many interpretations for the expression “virtual environment”. Some authors are very restrictive, defining virtual environments as “*three-dimensional, multi-sensorial, immersive, real time, and interactive simulations of a space that can be experienced by users via three-dimensional input and output devices*” [3], while others define them simply as “*a computer representation of a space*” [*ibid.*]. Thus we need to clarify what we mean by “virtual environment”: we are considering a 3D virtual world, with users being represented as avatars (3D virtual embodiments) able to interact with virtual objects.

When virtual environments are designed to support synchronous multi-user access to it, *i.e.*, the users share the same virtual space and virtual objects, we refer to them as “collaborative virtual environments” (CVE) [16]. In CVEs, any modifications made to the virtual space by one user are automatically reflected in the world view of others, allowing collaborative activities among groups of users.

If one looks at the specific area of learning activities, however, the most common use of the concept “virtual environment” is found at the opposite end of the spectrum from this 3D perspective: today’s educational institutions are using systems like Moodle or WebCT, commonly referred to as “virtual learning environments” (VLE), to support teaching and learning. They commonly support features like discussion forums, synchronous chat rooms, multiple-choice questionnaires, and document repositories, provided not by a 3D virtual world interface, but in the style of a Web site. In order to provide a general description of such systems, we can say, in an abstract way, that a VLE is a collection of integrated tools

* Corresponding author: Ricardo Antunes, e-mail: antunes@estg.ipleiria.pt, Phone: +351 962655828

enabling the management of online learning, providing a delivery mechanism, student tracking assessment and access to learning resources.

In this paper we are concerned about how one can take advantage of both systems, exploiting both the management capabilities of a VLE and the rich environment of 3D virtual worlds (CVEs) in activities to support the learning of programming.

2. Learning activities inside 3D collaborative virtual environments

The growing popularity [4] of 3D CVEs in which people interact in real time inside virtual worlds makes them an interesting area for research and development. But we discern a major divide amongst currently-available systems (Fig. 1). Some assume the constant presence of a specific purpose and inbuilt narrative or meta-narrative for the users, such as World of Warcraft [5], Lineage [6] or EverQuest [7]; these are commonly referred to as Massively Multiplayer Online Role Playing Games (MMORPG). Others, such as Second Life [8], Active Worlds [9], or Open Croquet [10] allow the creation of interactive content by the users without necessarily imposing a narrative for their actions. This feature is of particular interest for development towards the educational domain, since it allows educators to develop freely their own materials, providing their own content and narratives if necessary.



Fig. 1 Virtual environments: a) Open Croquet; b) World of Warcraft. (source: http://www.mmhell.com/gallery/world_of_warcraft/WoWScrnShot2.jpg)

Second Life in particular already has a rich and lively educational community. For instance, Harvard Law School is presenting a class called “CyberOne: Law in the Court of Public Opinion” [11] taught partly in the real, physical world, partly in the virtual world of Second Life. Students of the Harvard Extension Law School will be able to experience portions of the class through videos, discussions, lectures, and office hours, all inside the virtual world. So far, however, educational activities developed within Second Life are, mostly, “*presentations and events highlighting the newness of this phenomenon but not substantially improved from what they could achieve on their own campuses in real buildings*” [12]. The same could be said about educational activities in other CVEs. (A notable exception is the Sloodle project [13], which we discuss in the next section.)

Although “traditional” activities like the one presented above may be suitable for some fields of study, there are others, such as programming, where a closer tracking of student’s progress path is important for the teacher to be able to provide adequate feedback. Learning how to program is not a straightforward task. Rather, it requires not only the development of problem interpretation and analysis skills as well as being able to conceive and implement syntactically and semantically correct solutions. This learning process generally requires an iterative sequence, progressing and receding before a final result is reached.

3. Linking VLEs and CVEs

To better understand how VLEs and CVEs may be used to provide contextualized learning activities let's consider a simple example. The idea is to create a textured cube inside an existing virtual world. Using a VLE the teacher can take advantage of a VLE's management features and schedule the task, provide information about it and provide the texture file to be applied to the cube. The CVE provides a rich 3D environment where the task takes place. The student develops a cube inside the CVE and once this task is complete he needs to pick the texture file that is available on the VLE and apply it to the cube, inside the CVE.

One problem that can be observed from the above example is the lack of communication between these two systems. The student needs to jump from the CVE where he/she is developing his/her activity to pick required resources and take them inside the CVE. Another gap between these two systems is the lack of communication when a task is complete. It is necessary to inform the VLE the task performed inside the CVE is done by means external to the CVE. This issue is even more relevant when the task is performed collaboratively. In this example one student could be responsible for creating the cube and another student by applying the texture to it.

From the pedagogical point of view, it is desirable to keep track of activities as they are being performed by students, in order to better understand student difficulties and help students overcome them. In the example above it is hard to know about what actions a student performed inside the CVE before the final goal was achieved, or what were the points where the student was struggling, if any.

Our proposal is that the management and tracking capabilities provided by VLEs should be extended towards tracking of in-world activities in CVEs. Sloodle is an ongoing project striving to integrate Moodle into Second Life. This project aims to make Moodle's features available in the Second Life virtual world so that students can have access to Moodle course resources using a 3D representation of this VLE inside Second Life. Instead, we intend to explore the benefits of extending existing VLE features to manage and keep track of progress in students' learning activities performed inside a CVE.

4. Using CVEs and VLEs for learning computer programming

The task mentioned in the previous section was a simple manipulation of 3D objects. However, it could also be done by use of programming techniques, employing the scripting capabilities of the environment itself. Second Life supports scripting: its scripting language is called LSL, "Linden Scripting Language" [14]. Such scripting capabilities are present in many other CVEs: for instance, Open Croquet supports development in Squeak, an implementation of the Smalltalk programming language [15].

This feature makes CVEs a potentially powerful medium for learning of programming concepts and techniques. For instance, the previous example could be used to learn object-oriented programming concepts: to create a visual representation of a cube, students need to know how to create it inside the virtual world and how to display it. In Open Croquet, this can be done using the `TCube` class which is a subclass of class `TFrame` which in turn is the base class of all 3D graphic objects in Croquet [15]. Additionally other concepts may be introduced to extend the cube's functionality such as properties, methods and messages sent to other objects. Thus the concepts of class, hierarchy, object/instance, properties, methods and messages can be applied and learnt by students, using VCEs' scripting capabilities. Similarly, other concepts like an object's response to user events or the use of supported libraries, like the OpenGL graphics library of Open Croquet, can be exploited as in-world learning activities to extend object's functionalities.

Another important aspect of CVEs' scripting features is the possibility to connect with external resources. For instance, in Second Life objects can send and receive data to/from external Web-based systems using the common Hypertext Transfer Protocol (HTTP). Besides its potential for learning activities, this can be used to bridge the gap between CVEs and VLEs by providing a means to communicate between the two systems. In-world objects may be developed to gather and send information about student's achievements, thus allowing a better understanding about his/her way toward the final result.

5. Final remarks

Although CVEs are being used to provide rich contextual learning experiences, they lack mechanisms to provide close track of such in-world activities. On another hand, VLEs' features allow the management of learning activities but don't provide such a rich environment for activities to be performed. One possible way that deserves further research is to extend the functionality of VLEs in order to allow a closer management and understanding of students' programming difficulties and explore the potential offered by scriptable CVEs in order to implement in-world student tracking mechanisms and external communication capabilities in order to send this information to VLEs.

References

- [1] Tolga K. Çapın, Igor S. Pandzic, Nadia Magnenat-Thalmann and Daniel Thalmann, *Avatars in Networked Virtual Environments*, ISBN 0471988634, Chichester, UK: John Wiley & Sons Ltd. (1999)
- [2] Ivan E. Sutherland, A head-mounted three-dimensional display. In *Proceedings of the Fall Joint Computer Conference*, 1968, pp. 757-764 (1968).
- [3] Jesús Ibáñez Martínez, Carlos Delgado-Mata, and Ruth Aylett, *Virtual Environments: A Multi-disciplinary Field*, Upgrade, 7, 2 (2006).
- [4] Bruce Woodcock, *MMOGChat*. Online: "<http://www.mmogchat.com>" (retrieved 2006-10-28).
- [5] *World of Warcraft*. Online: "<http://www.worldofwarcraft.com>" (retrieved 2006-10-28)
- [6] *Lineage*. Online: "<http://www.lineage.com>" (retrieved 2006-10-28).
- [7] *Everquest*. Online: "<http://eqplayers.station.sony.com>" (retrieved 2006-10-28).
- [8] *Second Life*. Online: "<http://www.secondlife.com>" (retrieved 2006-10-28).
- [9] *Activeworlds*. Online: "<http://www.activeworlds.com>" (retrieved 2006-10-28).
- [10] *Open Croquet*. "<http://www.opencroquet.org>" (retrieved 2006-10-28).
- [11] Charles Nesson, Rebecca Nesson, and Gene Koo, "CyberOne: Law in the Court of Public Opinion". Online: "<http://blogs.law.harvard.edu/cyberone/>" (retrieved 2006-10-28).
- [12] D. Livingstone and J. Kemp. *Massively multilearner: recent advances in 3D social environments*, *Computing and Information Systems Journal*, 10, 2, Paisley, UK: University of Paisley (2006).
- [13] J. Kemp and D. Livingstone. Putting a Second Life "Metaverse" skin on learning management systems. In Daniel Livingstone and Jeremy Kemp (eds.), *Proceedings of the Second Life Education Workshop at the Second Life Community Convention*, San Francisco, 26 August 2006, ISBN 1-903978-32-7, Paisley, UK: The University of Paisley, pp.13-18 (2006).
- [14] *Linden Scripting Language*. Online: "http://history.secondserver.net/index.php/Linden_Scripting_Language" (retrieved 2006-10-28).
- [15] David A. Smith, Andreas Raab, David P. Reed, Alan Kay, *Croquet Programming – A Concise Guide*, Palo Alto, CA, USA: Qwaq (2006). Available online: "http://www.opencroquet.org/Site_PDFs/Croquet_Programming_1.0B.pdf" (retrieves 2006-10-30).
- [16] Dave Snowdon, Elizabeth F. Churchill and Alan J. Munro, *Collaborative Virtual Environments: Digital Spaces and Places for CSCW: An Introduction*. In Elizabeth F. Churchill, David N.Snowdon & Alan J. Munro (eds.), *Virtual Environments: Digital Places and Spaces for Interaction*, London, UK: Springer-Verlang London (2001).