A General Architecture for the Integration of Educational Videogames in Standards-compliant Virtual Learning Environments

Ángel del Blanco[1], Javier Torrente[1], Pablo Moreno-Ger[1], Baltasar Fernández-Manjón[1]
[1]Universidad Complutense de Madrid. Facultad de Informática.
C/ Profesor José García Santesmases s/n. 28040 Madrid, Spain
{angel.dba, jtorrente, pablom, balta}@fdi.ucm.es

Abstract

Although Virtual Learning Environments have become popular educational tools, they remain a very active research topic. Two important aspects being discussed for next-generation VLEs are how to track the performance of the students for assessment and how to provide personalized learning experiences (i.e. adaptive learning). In this line, educational videogames could contribute thanks to their highly interactive nature. However the combination of VLEs and educational games requires solving diverse conceptual and practical challenges. In this paper we present a general architecture to integrate games in VLEs with special emphasis on how to abstract the communication between the videogames and the VLEs for assessment and adaptation purposes.

1. Introduction

Virtual Learning Environments (VLE), such as Moodle, Sakai, .LRN or WebCT-Blackboard have become a powerful tool in education, providing instructors and students with multiple tools that support the learning process. In addition, VLEs are able to track the performance of the students not only through on-line exams, but also by monitoring the interaction between the student and the learning materials. VLEs can use this information to produce and maintain a persistent record of the activity of the student. Moreover, current research in VLEs explores how to use these student profiles to produce a user model of the student and personalize the learning experience according to that model. However, the effectiveness of VLEs for tracking the interaction is hindered by the passive nature of the web-based content, as it is very difficult to determine if html/hypermedia documents are really being attended or skimmed.

Another emerging trend is the use of videogames as educational materials due to their great educational potential, which has been thoroughly discussed in the literature [1-3]. Among these benefits, one is especially relevant for this work: games are highly interactive, establishing a very short action-reaction feedback cycle with the player. This interactivity means that videogames can track students' actions to determine if the learning goals are being achieved [4]. In addition videogames can use the information gathered from user interaction to drive the on-line adaptation of the educational game experience [5]. However, videogames cannot keep a persistent record of the overall progress of the students in the course (which may include other forms of content and other games).

Therefore a synergy between VLEs and educational videogames could result in mutual benefit. On the one hand videogames can provide effective mechanisms to track the performance of the students. On the other hand VLEs can use that information to keep an updated profile of the student and use it for adaptation. Moreover, adaptation can be split in two stages: a first general adaptation phase directed by the VLE and a second, fine-grained in-game adaptation phase.

Nonetheless this integration poses significant challenges. Usually VLEs are ruled by standards and specifications that define the communication protocol between VLE and learning materials (e.g. SCORM [6]). If videogames are to be integrated in VLEs they must follow such rules, which is an extra technical burden for games authors, as many specifications coexist with no de-jure standard communication protocol. Furthermore, once the communication channel between VLE and videogame is established, the problem of how to use it for assessment and adaptation purposes arises, as these communication protocols were not designed to connect VLEs with so interactive content as videogames are [7].

In this paper we present a general architecture that aims to tackle both issues by abstracting the communication between the game and the VLE for adaptation and assessment purposes.

2. The architecture

The architecture behaves as a two-layer middleware that connects the VLEs with games hiding the technical details of the communication protocol. In addition the architecture provides an abstract adaptation and assessment data model (AADM) that aims to exploit the communication channel between VLE and game. The model is used to define the adaptive and assessable behavior of the game.

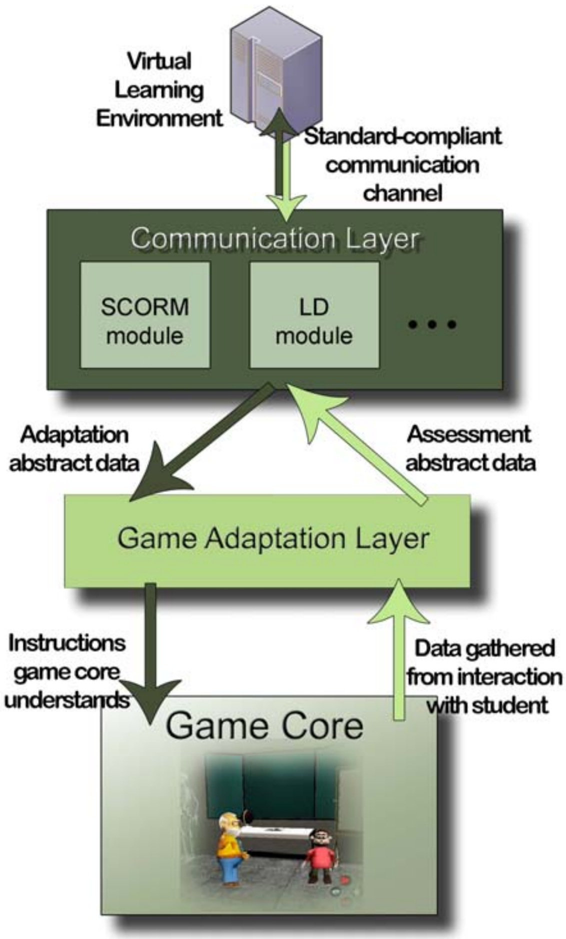

Fig 1. General structure of the architecture

The first layer is the Communication Layer (CL), which is responsible of establishing and managing the communication channel between the VLE and the game in a standards-compliant way. Thus the CL realizes actions such as *establish communication*, *get/set data from/to VLE* or *disconnect*. The CL decides the communication standard to be used (e.g. SCORM or an API provided directly by the VLE) and as a result these actions are dependent on the standards and specifications supported by the VLE. The CL needs to "understand" diverse specifications, which in this architecture is achieved through pluggable modules that implement a common API.

The CL uses the communication channel to provide services in terms of the AADM. Therefore the CL provides an API with methods such as *set student grade*, *send objectives results*, *get student name*, *get student skills level*, etc.

The second layer, which is called *Game Adaptation Layer (GAL)*, monitors the student-game interaction through an API that the game core must implement. The GAL uses the interaction data collected to maintain a record of the activity of the student. Besides the GAL uses the services provided by the CL to gather information from the VLE about the student, the course, etc. The activity record, along with the information provided by the VLE, can be used to personalize the game experience, even transparently to the student. For instance, the GAL could detect special situations such as if the student is lost in the game, if the difficulty level is too challenging/boring, etc., and perform some modifications in the game as defined by the instructor (e.g. *increment difficulty level*).

In addition, the record of the student can be used to evaluate the performance of the student. Instructors can identify situations that are relevant for student assessment (e.g. the game has not been completed successfully, a game task has been accomplished, etc.) and generate an assessment report (e.g. assign grade "A", penalize with a 10% reduction of the grade, etc.). Then the GAL uses the CL to transmit the assessment report to the VLE, which processes the information and keeps it persistently in the student's profile.

The behavior of the GAL is defined using the AADM. Thus the model supports the definition of both the situations that will trigger adaptation or assessment (*e.g. the student score is greater than 60%*) and the actions that must be performed in those situations (*e.g. set difficulty level "advanced")*.

3. Discussion of the architecture.

The architecture presented simplifies the integration of videogames into a VLE with numerous benefits. Educational game designers can focus on the design of the adaptive and assessable behavior of the games as they do not need to know the technical details of the communication. This adds not only persistence (the output of a game experience can be used as input for the next step in the course thanks to the VLE), but also a natural way to introduce games in complex courses just as any other learning resource.

However, some issues are still open. Probably the most relevant issue is that we must determine how the middleware is configured. This is complex due to the diversity of the standards that the CL must "understand". Some of these standards define their own communication protocol and/or communication data model (e.g. SCORM), but others do not (e.g. IMS Learning Design [8]). Hence the CL needs as input a *Communication Settings Profile* (CSP) which indicates the standard to use and, in some cases a definition of the communication protocol and/or data model and how to map it to the AADM.

In second place, the GAL must be configured as well, which includes a definition of how to carry out both adaptation and assessment. A valid solution could rely on *adaptation and assessment profiles* which would be basically a set of adaptation and assessment rules. Those rules could be compounded by a *condition* over the “state” of the game experience (which would include also information about the student profile that the VLE keeps) and a set of *effects*. Therefore when such conditions are satisfied the set of effects is triggered, which can both adapt the game experience or send data to the VLE to modify the student profile.

Here is where the strength of the architecture relies. The AADM should be enough flexible and abstract to allow instructors without a technical background to define how the game experience is to be assessed and adapted, which is a critical point from an instructional design perspective. Hence the AADM should help instructors to solve complex problems such as how to provide adaptation in terms of learning styles, prior knowledge, disabilities or even cultural background. Besides the adaptation and assessment model should include natural mechanisms to access the information stored in the profile of the student on the VLE side, supporting in this manner the two-stage adaptation process where VLE and game work together.

4. Conclusions and future work

In this paper we present a general architecture to simplify the integration of videogames in VLEs. The architecture provides two layers (communication, adaptation) and is very flexible because it can be configured to work with multiple VLEs and videogames. Support for different communication standards can be easily added by defining pluggable modules in the communication layer. Finally, the architecture supports the collaboration of multiple roles in the definition of game-based courses, as it isolates the tasks that game developers, VLE administrators and instructors must carry out.

This work is mainly a contribution from a Software Engineering perspective, and we have also implemented and tested several of the ideas discussed here. On the one hand, the implementation of the CL is complete, including communication modules for SCORM (version 1.2) and VLEs based on .LRN supporting IMS Learning Design. On the game side, we have tested the architecture with the <e-Adventure> [9] educational game platform. This platform supports the concepts of assessment, adaptation, and communication settings profiles, and provides utilities to edit them when designing an educational game.

Therefore, our next work will be focused on modelling the AADM, which is a great challenge, and release a development framework that implements the principles of the architecture.

Acknowledgments

The Spanish Committee of Science and Technology (projects FLEXO-TSI-020301-2008-19 and TIN2007-68125-C02-01) has partially supported this work, as well as the Complutense University of Madrid (research group 921340) and the EU Alfa project CID (II-0511-A).

References

[1] D. Michael and S. Chen, *Serious Games: Games that Educate, Train, and Inform*. Boston, MA: Thomson, 2006.

[2] J. P. Gee, *What video games have to teach us about learning and literacy*. New York ; Basingstoke: Palgrave Macmillan, 2003.

[3] C. Aldrich, *Learning by Doing: A Comprehensive Guide to Simulations, Computer Games, and Pedagogy in e-Learning and Other Educational Experiences*. San Francisco, CA: Pfeiffer, 2005.

[4] P. Moreno-Ger, D. Burgos, J. L. Sierra, and B. Fernández-Manjón, "Educational Game Design for Online Education," *Computers in Human Behavior*, vol. 24, pp. 2530-2540, 2008.

[5] P. Moreno-Ger, P. Sancho Thomas, I. Martínez-Ortiz, J. L. Sierra, and B. Fernández-Manjón, "Adaptive Units of Learning and Educational Videogames," *Journal of Interactive Media in Education*, vol. 2007,

[6] ADL, "Advanced Distributed Learning Sharable Content Object Reference Model (ADL-SCORM)." 2006, Retrieved 4th April 09 from www.adlnet.gov.

[7] R. R. A. de Antonio Jiménez, J., & Madrigal Alfaro, Z., "Architecture for Integration of Simulations with SCORM in E-learning Environments," in *Proceedings of ICALT 2008*, Santander, Spain, 2008, pp. 217-221.

[8] IMS Global Consortium, "IMS Learning Design Specification, Version 1.0 Final Specification," 2003. Retrieved 4th April 09 from: http://www.imsproject.org/learningdesign/index.html

[9] J. Torrente, P. Moreno-Ger, B. Fernández-Manjón, and J. L. Sierra, "Instructor-oriented Authoring Tools for Educational Videogames," in *Proceedings of ICALT 2008*, Santander, Spain, 2008, pp. 516-518.