Integrating Adaptive Games in Student-Centered 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: *The increasing adoption of e-Learning technology is facing new challenges such as how to produce student-centered systems that can be adapted to the needs of each student. In this context, educational video games are proposed as an ideal medium to facilitate adaptation and tracking of the students' performance for assessment purposes, but integrating the games in the educational flow presents technical and practical challenges. Moreover, their eventual integration should be accomplished according to the current standardization trends in e-Learning in order to simplify general adoption. There are still barriers between the gaming and e-Learning worlds preventing their mutual interaction. In this work we present a middle-ware to bridge this gap, integrating adaptive educational video games in e-Learning environments with a special focus on the ongoing standardization efforts.*

Keywords: Educational video games; Virtual Learning Environments; Adaptive Learning; Assessment; SCORM.

INTRODUCTION

Nowadays, the use of e-Learning is increasing as both industry and educational institutions embrace blended learning models in which traditional education is complemented with web-based e-Learning environments. At the same time, e-Learning systems have evolved from the original repositories of static content into richer Virtual Learning Environments (VLE), such as *Moodle*™ (Dougiamas & Taylor, 2003) or *Sakai*™ *(Farmer & Dolphin, 2005),* that comply with different standards and specifications to assure the interoperability of the materials (e.g. SCORM (ADL, 2006)). The new student-centered features being adopted (such as VLE-driven adaptation of the learning experience or user tracking and assessment) demand further standardization efforts and raise new technical challenges.

Besides, there is an emergent trend in Technology-Enhanced Learning advocating for the use of educational video games and game-like simulations (Tang, 2007; Torrente, Lavín-Mera, Moreno-Ger, & Fernández-Manjón, 2008). Educational video games present advantages such as their suitability to convey concepts (Hamid, 2001) or to increase students' motivation (J. P. Gee, 2003). Games also provide short feedback cycles that foster constructivist learning approaches in which students implicitly formulate and test hypothesis, receiving immediate feedback from the system (Pivec & Dziabenko, 2004).

However, games are not only interesting for those reasons. Another key feature of educational games is that their high level of interactivity can provide very fine-grained user adaptation and performance-tracking mechanisms (Moreno-Ger, Burgos, Sierra,

& Fernández-Manjón, 2008). This aspect can open new possibilities for student-centered VLEs. On the one hand, this allows collecting detailed data about the activities of the students. On the other hand, it allows providing learning experiences adapted to the needs of each student.

Nevertheless, in order to leverage this fine-grained assessment and adaptation potential, it is necessary to allow the games to exchange information with the VLE. While the exchange of information between active content and the VLE is addressed in standards such as the SCORM framework, we still need to deal with the current diversity of VLE and with a lack of specific standardization support for the peculiarities of game-based learning.

This article presents a general architecture to integrate games in VLE with special emphasis on supporting adaptation and assessment. This architecture is designed to provide an abstraction middle-ware that allows game designers to create adaptive educational games that are not compatible with a single VLE and are not committed to a specific educational standard. This offers the possibility of reusing the games in different VLEs and contexts, even if they support different families of standards (or even no standards at all).

This article is thus organized as follows. First we analyze the current state of the e-Learning field focusing on assessment, adaptation and standards; second, we discuss how video games can contribute to assessment and adaptation in e-Learning and the challenges behind this approach. Then we describe the proposed architecture and its implementation in the <e-Adventure> platform as a case study and finally, we present some conclusions and outline future lines of work.

VLEs: ASSESSMENT, ADAPTATION AND STANDARDS

VLEs are rapidly evolving, giving the instructor more support and advanced tools to create complex online learning experiences. However, the increasing complexity of the content, including highly interactive materials such as educational video games, requires further support for the instructors. The new VLEs need to facilitate tasks such as tracking the progress and the skills acquired by the student within the games, as well as to adapt the learning experiences to the specific needs of each student.

Another important issue in e-Learning is the interoperability of contents. E-Learning standards try to deal with the different aspects of e-Learning processes. At the present time, compliance to e-Learning standards is a crucial factor when selecting a new VLE implantation within a corporate or educational environment. This allows the reutilization of existing contents and protects the investment in developing new contents against future platform migrations.

E-Learning specification and standardization initiatives are numerous and diverse, involving different organizations and consortiums such as the IEEE or IMS Global Learning Consortium. Most of these contributions target the concept of allowing the creation of courses as aggregations of simple units of content, an approach usually referred to as the Learning Objects Model (Balatsoukas, Morris, & O'Brien, 2008). To achieve this kind of aggregation, it is important to package the contents in a standardized format (using for example IMS Content Packaging (IMS Global Consortium, 2004)) and to annotate the contents to facilitate their management (using for example the IEEE Learning Object Metadata standard (IEEE, 2002)).

Even though the use of the aforementioned standards to package and distribute the content is well established in many available VLEs, these standards do not simplify the widespread adoption of student-centered approaches with adaptation and performance tracking (Graf, Lin, & Kinshuk, 2005). For this reason, some initiatives such as the IMS Learning Design specification (IMS Global Consortium, 2003) are aiming to provide a standardized representation of the full learning process, taking into account pedagogical values.

With a lower degree of expressivity but a much wider adoption, there is also the SCORM framework (Shareable Content Object Reference Model), proposed by ADL (Advanced Distributed Learning). This framework, an initiative of the U.S. administration to improve e-Learning, is probably the most extended solution nowadays, merging and extending some of the specifications proposed by the aforementioned organizations in a single application model. That application model offers a combination of standards and specifications and gives extra recommendations covering a wide range of aspects in the creation of contents and how the VLE should manage and deliver them. In addition, SCORM defines a communication model that allows the interchange of information between the (potentially interactive) content and the VLE in a standardized way. The data exchanged between the content and the VLE is defined by the CMI data model.

However, currently two versions of SCORM coexist: SCORM1.2 and SCORM 2004. Even though the newer version is more complete and adaptable, the full adoption of SCORM 2004 by mainstream VLEs (e.g. *Moodle*™, *Sakai*™ or *WebCT-Blackboard*™ *(Goldberg & Salari, 1997)*) is still an ongoing process. In addition, some environments such as those based on the IMS Learning Design specification (IMS Global Consortium, 2003) or supported by LAMS (Dalziel, 2003) are also valid alternatives for student-centered processes, but not directly compatible with SCORM.

Therefore, the adoption of this type of advanced VLEs demands dealing with a diversity of standards that may put the investment at risk. Given that developing interactive and adaptive content requires a significant budget, this can potentially become a major issue.

GAME-BASED LEARNING AND E-LEARNING

As it has been widely discussed in the literature during the last years, the use of video games can enhance the learning processes in many aspects (J. P. Gee, 2003). As we previously mentioned, the most frequently cited benefits of game-based learning are the increase in the motivation of the students (Lepper & Cordova, 1992; Malone, 1981), the relation between video games and constructivists theories (J. Gee, 2007) or their support for collaborative/competitive learning (Squire, 2003). However, the full potential of video games in adaptable student-centered online learning is almost undiscovered and requires further research.

Videogames, adaptation and assessment

The adaptation of educational content to suit different target audiences with different levels of initial knowledge is a common feature in student-centered learning, although it is difficult to achieve. Meanwhile, personalization and challenge adjustment are pervasive features in video game products. Game developers and publishers include mechanisms in their video games to adapt the game experience to suit the requirements of the widest possible range of users. The most obvious type of adaptation in video games is the inclusion of different levels of difficulty, trying to adjust the challenge to different levels of skill.

However, the potential is even bigger thanks to the high interactivity of games, which can be used to implement much more fine-grained adaptation mechanisms. Some advanced games can even carry out this adaptation transparently to the user. For example, the *MaxPayne*™ video game incorporates Dynamic Difficulty Adjustment techniques (Robin, 2005) that alter the game execution depending on the actual performance of the user.

On the other hand, getting to know the student in a virtual learning setting is also a significant challenge (Ahmad, 2004). Typical approaches collect data by asking the user directly, although there are research initiatives that try to infer information about the students by observing their interaction with the system (Charles, et al., 2005). The fine-grained interactivity provided by games can produce more detailed information about the interaction of the students that any other kind of non-interactive content. Gathering and processing this information can open up new opportunities in terms of automatic assessment and student profiling.

Current challenges integrating game-based learning in VLEs

From the previous discussion, we derive that educational games can be an ideal medium to deliver student-centered content in VLEs. However, some issues should be addressed to successfully exploit the potential synergies between adaptive game-based learning and e-Learning.

One concern is the flexibility and maintainability of the content, a key issue in e-Learning but which is rarely tackled in video games. While typical educational content such as PDF, PPT or multimedia files can be easily edited, video games are usually sold as closed products which cannot be modified (i.e. black boxes). Other aspect is that games must behave more openly in order to become a more useful tool in student-centered VLEs, allowing the instructor to know what happens during the game sessions and to modify the behavior of the game as desired. This requires the definition of specific models that allow the instructor to interact with the game experience remotely. This can be done using the currently existing standards mentioned in the previous section, but this approach presents two main issues.

On the one hand, a game developer who wants to integrate a game into a VLE must identify which standard/specification will be used in the VLE to store the data and how the games will exchange information with the VLE. Given the current situation, with diverse (and evolving) standards available, this does not guarantee the full interoperability of the contents, leaving the investment unprotected (Kanendran, 2004

). Besides, educational game developers must implement in each game the selected set of standards from scratch, which requires great efforts due to the inherent complexity of these standards. This is especially relevant if we want to move towards educational experiences that contain diverse types of games that communicate with each other (Torrente, et al., 2008).

On the other hand, when developing and adaptive and assessable educational video game, it is necessary to maintain a model of each student persistently, and to define how to adapt the game experience according to that user model. If these behaviours are programmed ad-hoc in the game, the investment could become useless if instructors need to modify the adaptive and assessable behaviour of the game (for instance, if the educational video game is to be used in a new educational context). This could be solved if instructors could directly set up the adaptation and assessment configuration of the game and connect the video game with a VLE to solve the problem of the student model persistence.

The integration of video games or 3D immersive virtual worlds is not new, as several initiatives have combined VLEs and interactive content to get the best of both worlds (Chen, Wang, Chang, Chao, & Shih, 2009)(Rey-López, et al., 2008). For instance, *SLOODLE™(Kemp, Livingstone, & Bloomfield, 2009) and NUCLEO™(Sancho, Fuentes, Gómez-Martín, & Fernández-Manjón, 2009)* use *Moodle™* as a backend for a 3D Virtual Environment which is used as a central server. Other example is Delta3D™ (McDowell, Darken, Sullivan, & Johnson, 2005), a 3D game engine that implements SCORM to enable the communication between the games and a SCORM-compliant VLE. However, all these approaches use ad-hoc implementations of the communication between content and VLE, which limits the impact to a concrete platform and hinders the general adoption in educational settings. Additionally, while the problem of connecting interactive content (such as games and interactive simulations) with a VLE in standard-compliant ways has been partially addressed (A. de Antonio Jiménez, 2008), there is still a need of research about how to use this connection automatically for adaptation and assessment purposes, and how to assure that the developed games will be resilient to future changes in the current standards. Therefore, to facilitate the inclusion of educational games into the current student-centered VLE we need to achieve a greater independence between the implementation of the games and the standards used to connect them with the VLEs for adaptation and assessment.

AN ARCHITECTURE TO INTEGRATE GAMES IN STUDENT-CENTERED VLEs

In this section, we describe a general architecture that facilitates the integration of educational games in student-centered VLEs. The architecture uses a general model for assessment and adaptation concepts that hides from the instructors and content developers the technical difficulties derived from this process, and tries to alleviate the potential issues described in the previous section in terms of standards compatibility, adaptation and assessment.

Overview of the architecture

The architecture is divided in two modules. Each module deals with a different issue hindering interoperability. These issues are located on the game-side and on the

VLE-side respectively. On the one hand it is necessary to solve the problem of the communication between game and VLE. The development of games must be as independent as it is possible from the standards being used for enabling the communication or the specific VLE that the content is going to communicate with. On the other hand, it is necessary that the in-game adaptation and assessment tasks could be executed with independence of the game. Moreover, it seems interesting to abstract the content creators from the knowledge of the standards in terms of adaptation and assessment thus facilitating this process.

The first module, called *Communication Module* (CL), is the responsible for establishing and managing the communication channel between the VLE and the game in a standardized way. This module will typically execute actions such as "start / end" communication and "send / receive" data. The results of these actions depend on the current standard being used, being the CL the responsible of identifying it (or managing the agreement of the communication protocol to be used between game and VLE). The CL has a set of modules (one for each standard that it implements) that offer the necessary information to allow the actual communication with the VLE. All these modules implement a common API facilitating a plug-and-play architecture where different modules can be added or removed.

The second module is the *Adaptation and Evaluation module* (AEM). It has two missions. On the one hand, it decides the changes that should be done in the game in terms of adaptation and eventually execute those changes (*Adaptation*). On the other hand, this module monitors the student interaction in the game in order to extract information about the progress of the student; later on the information is processed and submitted to the VLE for assessment purposes through the *CM (Evaluation)*.

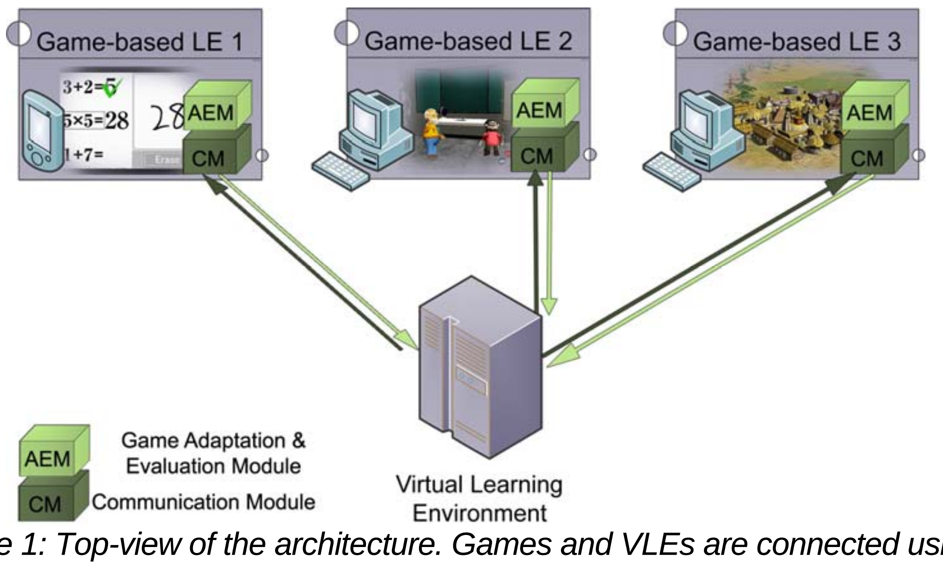

Figure 1: Top-view of the architecture. Games and VLEs are connected using a communication protocol.

The middle-ware is built on the *General Data Model for Adaptation and Evaluation* (GDMAE). This model is not part of the middle-ware; nonetheless the architecture uses the terms defined in that model to communicate throughout all the components involved in the process. The fact that the middle-ware uses abstract adaptation and assessment constructs means that both the game and the VLE can be developed independently. Also these general terms allow people without specific knowledge about the design and implementation of the game and/or the standards that will be used to define assessment and adaptation for game-based learning scenarios. The API communication functions provided by the CM are expressed in terms of that data model. For example “receive the name of the student” or “store the activity assessment”. The model defines situations involving adaptation and/or assessment tasks. The AEM will use these data along with the information gathered from the game and the VLE to decide whether to perform adaptation or not and what will be the concrete in-game adjustment to be performed.

The role of the games in terms of assessment and adaptation is to provide valuable information about the performance of the student. This information is used by the middle-ware to drive a fine-grained online adaptation of the learning experience and to produce assessment data as attribute-value pairs or in the form of a report that can be attached to the VLE’s student profile. That information about the user activity can be also utilized to update the student profile, therefore supporting a full adaptation cycle.

The General Data Model for Adaptation and Evaluation

This model defines two set of terms: 1) a set of general terms to express the situations that will imply adaptation or evaluation and 2) the terms to configure the CM. “Adaptation” is defined as the action of adjusting the game experience according to the profile of the student or the current in-game situation. “Evaluation” (which is also a synonymous for “assessment”) is defined as the identification of an in-game situation that is relevant for evaluating the performance of the student and committing to the VLE the assessment actions that must be performed (e.g. set a student grade).

To express what adaptation actions should be performed when certain situations are detected in the VLE we use the concept of *adaptation rule*. The structure of adaptation rules uses the well-know concept of rule, which includes a set of actions to execute when a set of conditions are achieved, and could be summarized as follows:

If (VLEstate) then (changeGame)

Where "*VLEstate*" is a general adaptation term (as defined in the model) that declares a finite set of conditions that must be met in the VLE side and where "*changeGame*" is a general term that identifies the set of actions that must be performed in the game.

Similarly, we define the concept of assessment rule to reflect the actions that must be carried out in the data model at VLE-side when certain circumstances in the game resulting from the interaction of the learner are detected. Assessment rules are defined as follows:

If (gameState) then (changeVLE)

Where "*gameState*" is a general assessment term that identifies a particular state in the game that, when reached, will involve the triggering of the “*changeVLE*” action block. "*changeVLE*" is a general term characterized according to the assessment data that should be generated and sent to the VLE for processing.

This model includes the definition of a language for specifying the general terms, which are closed to natural language and define how to adapt and assess the game-based learning experience, distancing in his manner the author of adaptation and assessment from the technical details.

The Communication Module (CM)

The communication module is responsible for requesting and transmitting information in both directions to the VLE and to the AEM module. The CM abstracts the communication between the game and the server, but internally depends on the standard used. It requests the data of the student profile, receiving such information in the specific format that the standard or the specification declares. Moreover it converts the standard-specific data to the general language that the corresponding module understands. In turn, it receives information from the AEM using evaluation general terms that need to be transformed and sent to the VLE, which will store them following the specification in use (if this specifies how to do so). By using the CM it is possible to receive and store adaptation and assessment information in the VLE regardless of the actual standard that the server uses.

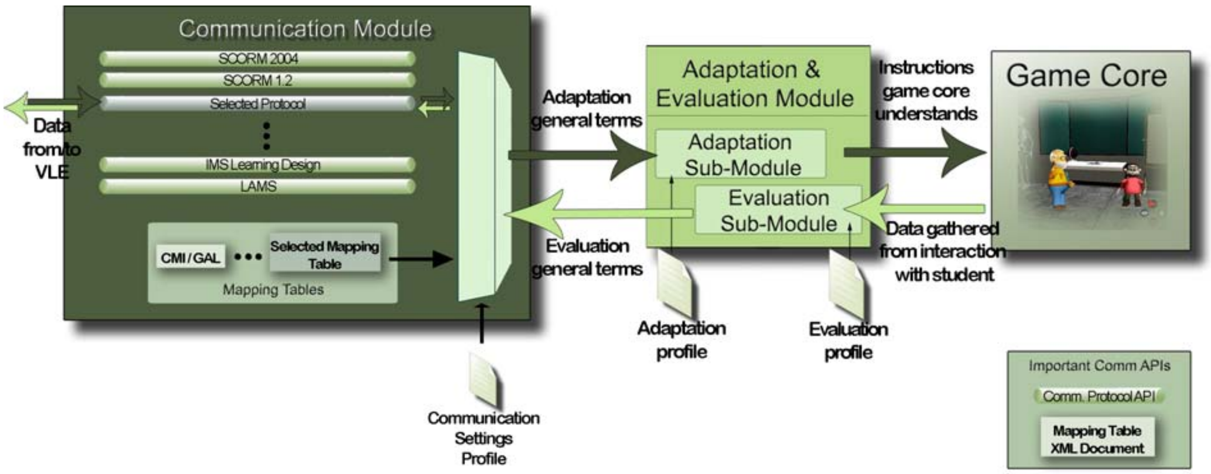

Figure 2: Communication sequence between the two middle-ware modules and the game core.

The CM module includes out-of-the-box implementations for some common standards in student-centered environments (based on specifications such as SCORM1.2 or SCORM2004). The CM also gives support to specific VLEs which do not contemplate a standardized communication protocol. This is the case of the LAMS Learning Authoring and Management Tool (Dalziel, 2003). When the game is run, the CM consults a configuration file (communication settings profile) to select the standard used to establish the communication with the VLE.

For each standard or specific VLE, the CM includes two relevant elements: an implementation of the communication protocol defined or supported by the VLE, and a profile (mapping table) specifying how to map the information in the cases where the standard already provides a pre-set data model (e.g. CMI data model for

SCORM). The abstract adaptation and evaluation data model must be connected to the language defined by the communication protocol. When the selected standard does not define the communication data model (e.g. IMS Learning Design), the communication settings profile can include a mapping table defining an application-specific data model. The CM must use this table in the communication process to store and retrieve the required information. In these cases, the information contained in the profile will depend on the specific course (although it will be independent from the specific game).

The components of the CM have clear and formally defined interfaces to allow flexible and systematic extensions and modifications, which guarantees interoperability and a longer life cycle for the middle-ware. New standards can be easily plugged in into the CM by carrying out two tasks: First, the communication protocol must be implemented following a specific API. In some cases this API will be provided by the standard (e.g. SCORM). In some other cases, it will be dependant on the VLE. Secondly the mapping table must be written, defining the translations between the abstract constructs and the VLE-dependant constructs.

With this structure the underlying communication standards are completely transparent. Authors only need to modify the middle-ware configuration file to change the standards and specifications used in the communication.

The Adaptation and Evaluation Module (AEM)

The AEM is responsible for carrying out the adaptation tasks in the game according to the adaptation rules and using data provided by the VLE. It is also responsible for monitoring the activity in the game to detect situations that require evaluation. As both adaptation and assessment are well-defined tasks with both common and specific features, this module delegates both responsibilities in two sub-modules respectively (thus both are addressed individually): the *Adaptation Sub-Module* (ASM) and the *Evaluation Sub-Module* (ESM).

The ASM receives input data about student's information stored in VLE (e.g. progress in the educational process, social and cultural factors, etc.) for adaptation decision-making process (figure 2). These are "get actions" expressed in "*VLEstate"* terms. The ESM also uses the CM to submit information about the achievements of the student in the game (e.g. the level of completion of the activity is 60% or the completion of learning goals is 75%) and sends it through a standard-compliant channel to the VLE. These are "set actions" expressed in "*changeVLE*" *terms*. As aforementioned, the CM is the responsible of managing these general terms, translating them to the specific low-level communication protocol if necessary. This is how the CM adds independency from the specific standard supported by the VLE both for adaptation and assessment communication. In addition, the AEM communicates with the game core in order to receive information about user's in-game interaction (using "*gameState" terms*) and to make changes as a result of adaptation decisions (using "*changeGame"* terms).

The in-game situations that should produce either some kind of adaptation and/or evaluation are defined by the author in the adaptation and evaluation profiles. These profiles contain a set of rules defined as it was explained in General Data Model for

Adaptation and Evaluation. Both sub-modules must "understand" the meaning of each term to concrete them in actions in the game or VLE.

CASE STUDY: INTEGRATION IN <E-ADVENTURE>

To test this approach we have implemented the architecture in the <e-Adventure> platform. <e-Adventure> (Moreno-Ger, et al., 2008) is a platform created in order to facilitate the introduction of video games and game-like simulations in the educational process, trying to overcome some barriers that hinder the generalization of educational games. <e-Adventure> provides a game editor to create the adventures and a game engine to execute the created games. Both components play different roles in respect to our architecture. On the one hand, the architecture has been integrated in the game engine to enable the communication with VLEs. On the other hand the game editor allows the configuration of both modules, generating the settings, adaptation and evaluation profiles (which are called assessment profiles in <e-Adventure>) that the architecture needs. This process is performed transparently to the author. When the final version (runnable version) of the game is produced using the editor, those profiles are packaged within the game, following the specifications in terms of content packaging that the standard being used defines (if any). When a game is going to be exported, the platform allows selecting which will be the exportation type between the different VLE and standards that supports (exportation profiles).

Implementation of the Communication Module

To include the CM in the platform we have realized the following tasks: to implement a module for every standard, to give support to the metadata for the new included standards and to add new types of exportation profiles for the <e-Adventure> games. At the moment, we have given support for SCORM v1.2 and 2004, LAMS, and an *ad hoc* protocol defined to communicate with servers running IMS Learning Design (the server side of this protocol has been implemented as a plug-in for *CopperCore* and .LRN).

The CM is made up of a set of modules with the particularities of each standard or specific VLE. Every module implements the communication protocols that the standard uses. In case the standard has a specific data model, it must be also included in the specification of the module. When there is not a data model defined, it will be set in the adaptation and assessment profile via the mapping table, as exposed previously.

There exist different types of game exports as Learning Objects in reference to the different types of standards and VLE that the current architecture implementation supports. Choosing one of the exportations, the communication sub-module that will be in use is selected.

In addition an editor of metadata has been added to include this type of information in every exportation type that should need it.

Implementation of the Adaptation and Evaluation Module

The <e-Adventure> platform includes devices that not only track the student's activity for assessment purposes but also allows making changes in the game for adaptation tasks. This platform has already implemented the parts of the AEM that are responsible for these actions which are specific for each game engine. It also allows adding assessment and adaptation profiles where the sets of rules of adaptation and assessment are defined.

The profiles have been modified taking into account the peculiarities of the architecture. In these profiles the user will be able to choose a standard in order to establish both communications with the VLE and the data model. The user can either define the data model in the profile or use a default model provided by the selected standard. The user can determine conditions and execute actions which modify the state of the VLE data model. The mechanism of variables and flags that <e-Adventure> provides is used to introduce changes and to define conditions on the game state (Moreno-Ger, Sierra, Martínez-Ortiz, & Fernández-Manjón, 2007). The AEM will be configured by filling in these profiles with adaptation and assessment rules and selecting a standard.

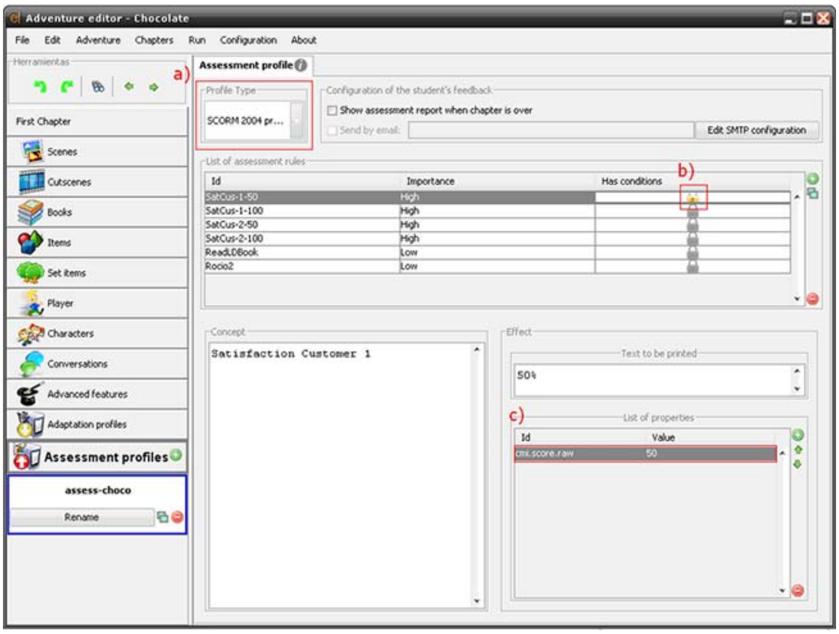

Figure 3: The <e-Adventure> editor. The screenshot shows an assessment profile with some assessment rules. a) The area that allows selecting a standard or VLE. b) The area for defining conditions (“gameStates”). c) Area for defining what will be set in VLE (“changeVLE”).

Example of communication between VLE and game

The architecture presented can be exemplified with the “Paniel and the Chocolate-based Sauce Adventure” game, which can be integrated into different student-centered VLEs. The game was originally developed to be integrated only with the *CopperCore* IMS Learning Design module (Burgos, 2008). The goal of the game is to introduce chocolate-based cooking techniques from a practical perspective, and is divided in three stages of different levels of difficulty. The initial level teaches how to make chocolate, the second level teaches how to make chocolate-based sauces, and the third level teaches how to marry them with dishes (the most challenging level). First, we analyze the theoretical behaviour, and afterwards we describe the real behaviour in our case study.

The AEM is set up with an adaptation profile that modifies the game depending on the prior knowledge of the student (figure 4a). When the game is executed, the AEM requests information from the VLE asking the overall grade of the student (figure 4c-step1). Then the CM, which has previously set up the communication channel with the VLE accordingly to the communication settings profile, codifies the request using the appropriate model, and sends it to the VLE. After the CM receives the VLE response, it translates that response into adaptation terms and sends them to AEM. The AEM then interprets the abstract response and uses the adaptation profile to decide to which level the student should go, skipping the first levels if appropriate. In this example, the adaptation profile determines that if the student overall grade is greater than 50% and less than 75% the first level is skipped. If the overall grade is greater than 75% the second level is skipped. Finally, if due to any cause the overall grade could not be retrieved from the VLE, or if the overall grade is less than 50%, no levels are skipped (fig 4c-step 5).

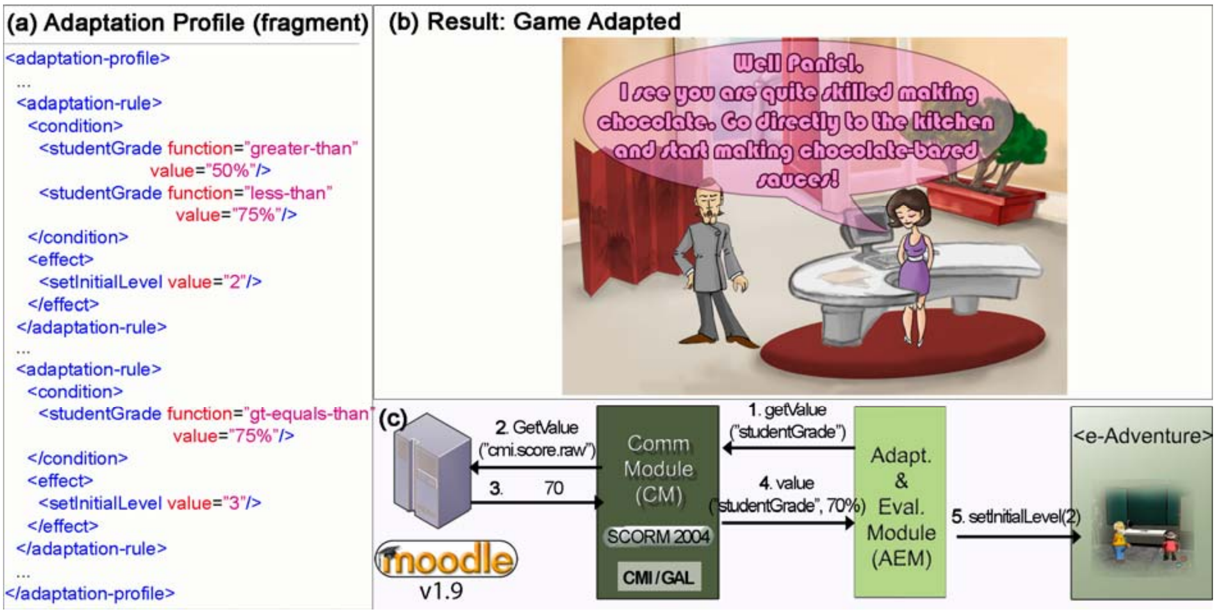

Figure 4: (a): Fragment of the adaptation profile used in the example. (b): Screenshot of the game adapted. (c): Communication sequence between the two modules (CM and AEM), the VLE and the game core. While (a) and (b) are independent of the actual platforms being used, (c) depicts the communication through the platforms and standards that were actually used.

Besides, the game includes an in-game test that produces a final grade. This abstract grade is submitted to the VLE through the middle-ware, following the same steps (translation of the game concepts into general terms and then translation into VLE-specific constructs). This grade can then be used in future executions of the game to make the initial adaptation decision.

While testing this example in an <e-Adventure> study case, we use the same profiles but in this case we set the architecture to work with SCORM 2004. to the game was integrated in *Moodle*™ v1.9 (the VLE) as a SCORM/AICC activity. As we have not implemented the GDMA yet in the case study, the terms related to the game state in the rules are expressed in terms of <e-Adventure> variables and flags. The terms related to the VLE state are expressed as SCORM v1.2 data model elements. The <e-Adventure> editor assists the process to fill this information in the profile.

For this example, the <e-Adventure> game developer must only know the game flow (a task that is well identified in the editor) and the SCORM 1.2 data model. Should the developer want to distribute the game through a different platform (for example, SCORM v1.2), the only required task is to modify the communication profile, using elements from the SCORM v1.2 data model. The game itself does not need to be changed.

CONCLUSIONS AND FUTURE WORK

In this paper we have discussed the benefits that game-based learning can bring to e-Learning in terms of adaptation and assessment. However, bringing both worlds together is a technically challenging task ought to the complexity of both fields. Especially relevant in this concern is the diversity of VLE communication and standards (SCORM, IMS LD, etc.) in e-Learning.

Our contribution is a general architecture for the integration of games in VLE, consisting of a two-module middle-ware which abstracts the existing standards. Using this architecture, the communication between a standards-compliant VLE and an adaptive educational video game is independent of the specific game or standard. This allows game developers to create games without needing to be concerned with the internal details of each possible implementation of the student-centered VLE, focusing in this manner on the design of pedagogically relevant aspects. For example, if a teacher is interested in using a game with assessment and adaptation characteristics that use the architecture to communicate via SCORM in a LAMS VLE, they will only have to modify the communication setting profile. In this way teachers can exploit these educational features. Even they are also able to modify certain tasks of adaptation and assessment by changing, adding or eliminating the existing rules in the adaptation or assessment profiles. However, if the game would implement SCORM ad hoc, it could only be used in a SCORM-compliant VLE for the specified purposes of adaptation and assessment unless these features are implemented again. This fact widens the range of teachers that can reuse the educational games as this approach is scalable to other platforms, games and educational settings (as opposed to other initiatives that integrate educational games in VLEs). The interoperability, maintainability and reuse of the contents are addressed as the architecture is flexible enough to support new standards and revisions thanks to the notion of pluggable adapters.

We have tested this middle-ware in the <e-Adventure> educational game platform, which provides an authoring environment for educational games with special emphasis on the integration with VLEs, adding support for the APIs provided by the architecture. We use the rules model implemented in <e-Adventure> to implement the concepts exposed in the General Data Model. The general terms of the model have not been implemented in the platform because, at this time, we are still studying the possibilities that they offer and how to properly take advantage of them for education. The preliminary results are promising, but also indicate some issues that will require further research. On the one hand, adaptation is a very complex issue. To exploit all the potential of adaptive game-based learning the general adaptation model must be extended and refined far beyond its current state. The discussion of how to adapt the content and in what circumstance it should be adapted is still an open research question. Moreover, the automatic detection of in-game situations which require adaptation deserves its own line of research.

On the other hand, the middle-ware must be expanded to include more modules for additional communication standards, including VLE-specific plugins for those environments that do not provide a standardized method for content-to-VLE communication. On the game side, we are also working on the implementation of the architecture for different game engines in different platforms.

Acknowledgments. The Spanish Committee of Science and Technology (TIN2007-68125-C02-01) and the Ministry of Industry (grants TSI-020301-2009-9, TSI-020110-2009-170, TSI-020312-2009-27) have partially supported this work, as well as the Complutense University of Madrid (research group 921340) and the PROACT1 EU LLP project (505469-2009-LLP-ES-KA3-KA3MP). Special thanks to Ernie Ghiglione from the Macquarie E-Learning Centre of Excellence for providing us with all the information about LAMS.

REFERENCES

A. de Antonio Jiménez, R. R., J., & Madrigal Alfaro, Z. (2008). *Architecture for Integration of Simulations with SCORM in E-learning Environments.* Paper presented at the Eighth IEEE International Conference on Advanced Learning Technologies (ICALT 2008), Santander, Spain.

ADL. (2006). Advanced Distributed Learning Sharable Content Object Reference Model (ADL-SCORM) Retrieved 20th January, 2006, from http://www.adlnet.org/

Ahmad, A., Basir, O., & Hassanein, K. (2004). Adaptive user interfaces for intelligent e-Learning: issues and trends. . *Paper presented at the The Fourth International Conference on Electronic Business, ICEB2004.*

Balatsoukas, P., Morris, A., & O'Brien, A. (2008). Learning Objects Update: Review and Critical Approach to Content Aggregation. *Educational Technology & Society, 11*(2), 119-130.

Burgos, D., P. Moreno-Ger, J.L. Sierra, B. Fernández-Manjón, M. Specht, & R. Koper. (2008). Building Adaptive Game-Based Learning Resources: The Marriage of IMS Learning Design and <e-Adventure>. *Simulation & Gaming, 39*, 414-431.

Charles, D., Kerr, A., McNeill, M., McAlister, M., Black, M., Kcklich, J., et al. (2005). *Player-centred game design: Player modelling and adaptive digital games*. Paper presented at the Digital Games Research Conference, Vancouver, Canada.

Chen, J.-H., Wang, T.-H., Chang, W.-C., Chao, L. R., & Shih, T. K. (2009). Developing an Interactive Video Game-Based Learning Environment. *JOURNAL OF SOFTWARE, 4*(2), 132-139.

Dalziel, J. (2003). *Implementing Learning Design: The Learning Activity Management System (LAMS).* Paper presented at the 20th Annual Conference of the Australasian Society for Computers in Learning in Tertiary Education., Adelaide.

Dougiamas, M., & Taylor, P. (2003). *Moodle: Using Learning Communities to Create an Open Source Course Management System.* Paper presented at the World Conference on Educational Multimedia, Hypermedia and Telecommunications 2003, Honolulu, Hawaii, USA.

Farmer, J., & Dolphin, I. (2005). *Sakai: eLearning and More.* Paper presented at the 11th European Univeristy Information Systems (EUNIS 2005), Manchester, UK.

Gee, J. (2007). *Good Video Games and Good Learning: Collected Essays on Video Games, Learning and Literacy (New Literacies and Digital Epistemologies)*: {Peter Lang Publishing}.

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

Goldberg, M. W., & Salari, S. (1997, June 12 -15, 1997). *An Update on WebCT (World-Wide-Web Course Tools) - a Tool for the Creation of Sophisticated Web-Based Learning Environments.*

Paper presented at the NAUWeb ’97 - Current Practices in Web-Based Course Development,, Flagstaff, Arizona (United States).
Graf, S., Lin, T., & Kinshuk. (2005). *Improving student modelling: the relationship between learning styles and cognitive traits.* Paper presented at the Cognition and Exploratory Learning Digital Age (CELDA2005), Porto, Portugal.
Hamid, A. A. (2001). e-Learning: Is it the "e" or the learning that matters? *The Internet and Higher Education, 4*(3-4), 311-316.
IEEE. (2002). IEEE Standard for Learning Object Metadata from http://ltsc.ieee.org/wg12/index.html
IMS Global Consortium. (2003). IMS Learning Design Specification, Version 1.0 Final Specification, from http://www.imsproject.org/learningdesign/index.html
IMS Global Consortium. (2004). IMS Content Packaging Specification, Version 1.1.4 Final Specification, from http://www.imsglobal.org/content/packaging/index.html
Kanendran, T. A., Savarimuthu,J. and Durga-Kum, BV. (2004 25-26 November). *e-Learning Standards.* Paper presented at the Proceedings of the 3rd European Conference on E-Learning, Paris, France.
Kemp, J., Livingstone, D., & Bloomfield, P. (2009). SLOODLE: Connecting VLE tools with Emergent Teaching Practice in Second Life. *British Journal of Educational Technology, 40*(3), 551-555. doi: http://dx.doi.org/10.1111/j.1467-8535.2009.00938.x
Lepper, M. R., & Cordova, D. I. (1992). A desire to be taught: Instructional Consequences of Intrinsic Motivation. *Motivation and Emotion, 16*, 187-208.
Malone, T. (1981). What makes computer games fun? *Byte, 6*(12), 258-276. doi: http://doi.acm.org/10.1145/1015579.810990
McDowell, P., Darken, R., Sullivan, J., & Johnson, E. (2005). *Delta3D: A Complete Open Source Game and Simulation Engine for Building Military Training Systems.* Paper presented at the The Interservice/Industry Training, Simulation & Education Conference (I/ITSEC).
Moreno-Ger, P., Burgos, D., Sierra, J. L., & Fernández-Manjón, B. (2008). Educational Game Design for Online Education. *Computers in Human Behavior, 24*(6), 2530-2540.
Moreno-Ger, P., Sierra, J. L., Martínez-Ortiz, I., & Fernández-Manjón, B. (2007). A Documental Approach to Adventure Game Development. *Science of Computer Programming, 67*(1), 3–31.
Pivec, M., & Dziabenko, O. (2004). Game-Based Learning in Univeristies and Lifelong Learning: "Unigame: Social Skills and Knowledge Training" Game Concept. *Journal of Universal Computer Science, 10*(1), 4-12.
Rey-López, M., Díaz-Redondo, R. P., Fernández-Vilas, A., Pazos-Arias, J. J., López-Nores, M., García-Duque, J., et al. (2008). T-MAESTRO and its authoring tool: using adaptation to integrate entertainment into personalized t-learning. *Multimed Tools Appl, 40*, 409-451. doi: 10.1007/s11042-008-0213-4
Robin, H. (2005). *The case for dynamic difficulty adjustment in games*. Paper presented at the Proceedings of the 2005 ACM SIGCHI International Conference on Advances in computer entertainment technology, Valencia, Spain.
Sancho, P., Fuentes, R., Gómez-Martín, P. P., & Fernández-Manjón, B. (2009). Applying multiplayer role based learning in engineering education: Three case studies to analyze the impact on students’ performance. *International Journal of Engineering Education, 25*(4).
Squire, K. (2003). Video games in education. *International Journal of Intelligent Simulations and Gaming, 2*(1), 49-62.
Tang, S. a. H., M. (2007, Nov 14-15). *Describing Games for Learning: Terms, Scope and Learning Approaches.* Paper presented at the The Fifth Annual International Conference in Computer Game Design and Technology, Liverpool, UK.
Torrente, J., Lavín-Mera, P., Moreno-Ger, P., & Fernández-Manjón, B. (2008). Coordinating Heterogeneous Game-based Learning Approaches in Online Learning Environments. *Paper presented at the the Sixth International Game Design and Technology Workshop and Conference (GDTW2008)*.