A Game-Based Adaptive Unit of Learning with IMS Learning Design and <e-Adventure>

Pablo Moreno-Ger[1], Daniel Burgos[2], José Luis Sierra[1],
and Baltasar Fernández-Manjón[1]

[1] Dpto. Ingeniería del Software e Inteligencia Artificial. Fac. Informática. Universidad Complutense. 28040, Madrid. Spain
{pablom,jlsierra,balta}@fdi.ucm.es
[2] Educational Technology Expertise Centre (OTEC). The Open University of the Netherlands. Heerlen. The Netherlands
daniel.burgos@ou.nl

Abstract. In this paper we illustrate how to conceive, implement and play adaptive Units of Learning (UoLs) that embed educational videogames. For this purpose we describe *The Art & Craft of chocolate* UoL, with the game *Paniel and the chocolate-based sauce adventure* as a key feature. The UoL includes a pre-test whose outcome is used to adapt the game. The UoL also assesses the learning process using an in-game exam. This UoL has been modeled using IMS Learning Design (LD), and the embedded game has been developed using the <e-Adventure> educational game engine. This UoL may be deployed in any LD-compliant environment, although some of the features like the adaptation of the game or automatic assessment require special plug-ins that enable the communication between the environment and the <e-Adventure> engine. These plug-ins have been developed as an open-source modification of the SLeD player.

Keywords: edutainment, adaptive e-learning, <e-Adventure>, IMS Learning Design.

1 Introduction

There is a growing interest for the introduction of computer and videogames in educational environments. Games have become one of the biggest entertainment industries, rivalling cinema and surpassing literature [8], mostly because modern games are attractive, engaging and immersive. Additionally, the research about the nature of fun and motivation in videogames highlights a number of elements such as short feedback cycles, high interactivity, or embodiment, which can have a significant impact in educational environments [9,17,18]. The pedagogical benefits of game-based approaches, as well as some of their shortcomings, have been thoroughly studied in the literature [2,10,13,20,29]. Typical problems include social rejection, excessively high development costs, and poor results when the resulting products include very precise and detailed content but fail completely when it comes to

E. Duval, R. Klamma, and M. Wolpers (Eds.): EC-TEL 2007, LNCS 4753, pp. 247–261, 2007.

providing entertainment (thus missing the appeal of videogames and its associated pedagogical benefits) [27,28].

Remarkably, educational videogames are complex software artefacts that are executed on the student's computer. This fact makes them very interesting from the perspective of adaptive learning because the videogame can behave differently every time it is run. Indeed, the possibility of choosing different levels of difficulty has been present in videogames since the very beginning. Most games become increasingly difficult as the user progresses (i.e. each level is more challenging than the previous one) and, additionally, it is usually possible to select a base level of difficulty so that the experience is neither too challenging nor too easy. The objective is to keep the player in *the zone* where he or she is forced to perform at the limit of his/her competence but without exceeding it.

The key idea is that videogames and adaptation are synergic fields and we should leverage this when creating adaptive contents and courses.

On the other hand, a field that could benefit most from adaptive learning (and that invests a lot of effort and research in the matter) is online learning. The so-called Learning Management Systems (LMS) facilitate and monitor the learning experiences of large groups of students. Even though these environments are sometimes targeted at a very specific group of users, it is also common to find systems targeted at broad audiences that have different learning styles, differences in their previous background and different learning objectives.

For this reason, there is a lot of research into providing adaptive learning experiences [4,24] in which the adaptation optimizes the focus of the content (by fitting different levels of previous knowledge or different objectives) and the overall learning experience (by fitting different learning styles). In this arena, the IMS Learning Design specification is one of the key elements because it facilitates a formal modelling of the intricacies of adaptive learning paths.

In this paper, we analyze the potential synergies between adaptive games and learning environments based on the IMS Learning Design specification [11]. For this purpose, we have conceived an adaptive Unit of Learning (UoL) built around an educational game. This UoL has been modelled with IMS Learning Design. For the implementation of the adaptive videogame we have used the <e-Adventure> educational game engine [21], leveraging its built-in adaptation and assessment features. The <e-Adventure> engine can be deployed to the student's computer from an LMS and, should the LMS support it, establish a bidirectional communication with the server that can be used to alter the behaviour of the game and to inform the LMS of the activities of the student within the game [19]. The result is an adaptive process with a complete feedback loop in which previous knowledge about the student and his/her performance is used to modify the game and in which the activity of the student within game is used to improve that knowledge and adapt the rest of the learning experience.

The structure of this work is as follows: section 2 describes the adaptive UoL. Section 3 introduces the supporting technologies used in its implementation: IMS Learning Design and <e-Adventure>. In section 4 we outline the technical details regarding the execution environment. Finally, in section 5 we present the conclusions and some lines of future work.

2 The *Art and Craft of Chocolate* UoL

Learning Management Systems in online education are often targeted at broad audiences with varying demographics. The students have different backgrounds, different levels of initial knowledge, different ambitions in terms of learning objectives and even different learning styles.

The inclusion of adaptation techniques to fit the needs of different students is thus pedagogically and commercially sound. In particular, we are mainly interested in studying adaptation in the context of educational games and their use in complex UoLs.

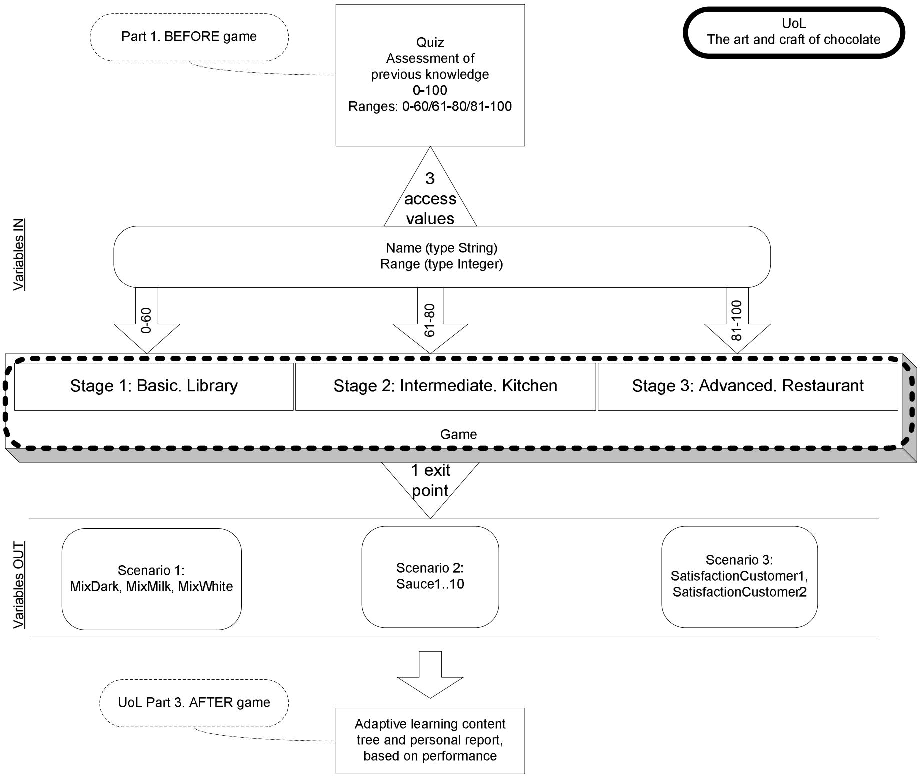


Fig. 1. Basic outline of the Unit of Learning

In order to show how an adaptive UoL involving educational games can be effectively achieved, we have conceived a sample UoL that includes a sample adaptive game. The UoL is entitled *The Art and Craft of Chocolate* and deals with advanced uses of cocoa and chocolate in cooking. The educational goal is to let learners learn how to prepare chocolate sauces by mixing different chocolate types and how to use these sauces to prepare a variety of sophisticated dishes.

The technical goal, however, is to provide a proof-of-concept implementation of the different mechanisms required in order to achieve adaptive UoLs embedding educational videogames. In particular, we focus on two different issues: (i) letting the UoL modify the game's state and, (ii) letting the game modify the UoL's state. For this purpose, we structure the UoL as follows (see Fig. 1):

- The first step of the instructional design includes some traditional content and some basic tests to capture the student's initial level of knowledge.
- Then the game is launched and adapted according to that information.
- The game itself includes an in-game exam. The results of that exam are used to grade the student. Additionally, this assessment can also be used to decide the flow of the rest of the learning experience.

The following subsections describe these phases in greater detail.

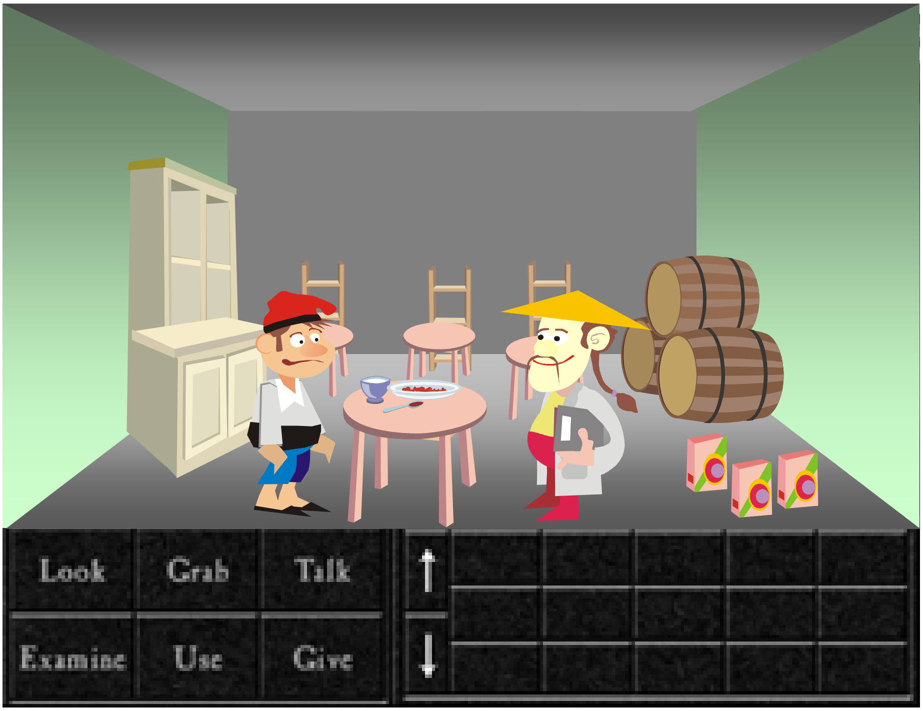


Fig. 2. Screenshot of the *Paniel and the Chocolate-based Sauce Adventure* game

2.1 The Game

The main part of the UoL is an educational *point and click* adventure game entitled *Paniel and the Chocolate-based Sauce Adventure* (Fig. 2). The game covers the two chapters of the syllabus (creation of basic chocolate masses and their use to create sauces for different recipes) and includes a final in-game exam in which the student is required to apply the knowledge acquired by finding out about their tastes and preparing the right sauces to accompany the right dishes.

2.2 The script

A key aspect when writing games is to achieve the right balance between the elements that make games engaging (a good story, interesting situations, self-guided exploration, etc.) and the content itself [25]. A simple presentation of the content in a linear fashion with some graphics will not attract the student as much as a more elaborate story.

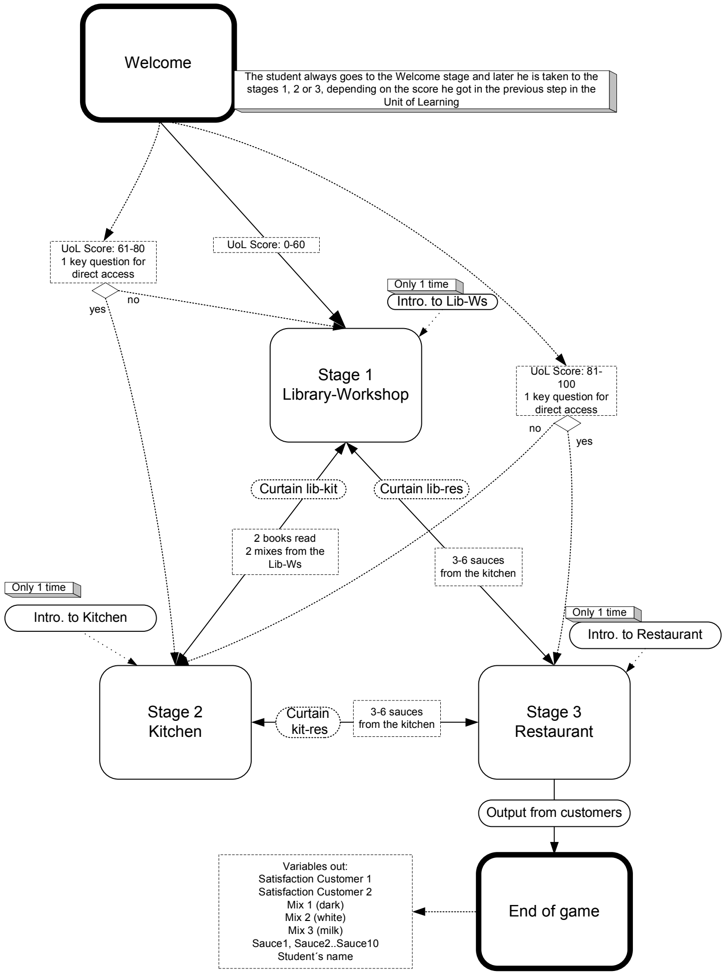


Fig. 3. High-level design of the *Paniel and the Chocolate-based Sauce Adventure* videogame

In this game, Paniel is a young cook enrolled in a course about chocolate. After arriving at the site and engaging in conversation with the beautiful secretary, Paniel is instructed to go to the workshop where the Master Cook is waiting with his first instructions: to browse through the library and prepare some basic chocolate masses with the ingredients available there (black chocolate, milk chocolate, white chocolate, etc). It is up to the player to find the right recipes by consulting the books or by experimentation.

In the second level, the player is asked to prepare a number of chocolate-based sauces in preparation for the final exam. This part of the game is very open as far as the recipes for the sauces are concerned, and information on how to marry sauces with dishes is scattered. Some information can be found in books at the library, the Master Cook provides many recipes and useful information, and some of the recipes can only be obtained by talking the secretary into giving the player some of her personal secret recipes.

Finally, when the player considers that the number of sauces and the knowledge about how to use them is enough, he or she can start the exam. In the exam, the player finds Paniel in the school's restaurant with the tasks of getting information about the customers' tastes and preparing the dishes that will fit those tastes with the available sauces and without the possibility of crafting more sauces or going back to the library for more information.

2.3 Different Paths Through the Game

As sketched in Fig. 3, the high-level design of the game structure is not conceived to be played entirely by every player every time it is run. Indeed:

- Students with previous experience in confectionery will probably already be familiar with the recipes of plain chocolate masses and should be able to skip the first part of the educational game.
- On the other hand, some other students may already have a deep knowledge of the subject. In this case, they can skip the entire learning section in the game and proceed directly to the in-game exam.

Notice that the student is always forced to play the game because it includes the in-game exam that is used as the main grading tool.

2.4 Profiling the Student

In order to exploit the game's adaptive capabilities, the first part of the UoL focuses on capturing information about the students. In full implementations of the system, this step would not need to be as detailed since some of the information would be available in the student's profile.

Apart from basic questions querying the user about his/her name and other preferences, the UoL includes a questionnaire in which we assess the student's initial level in order to skip those parts of the course with which he or she is already familiar. Thus, the questionnaire reflects this information by assigning a score to the student. Then the game is executed and, depending on the initial score, some of its levels are skipped. More precisely, as depicted in Fig. 3:

- If the result is below 60% the game will start from the beginning.
- If the result is between 61% and 80% the first level will be skipped.
- Should the result be above 80%, the game would proceed directly to the final exam with a default set of recipes initially available.

2.5 Built-in Assessment and Reporting

The final in-game exam is used to assess the student's proficiency when it comes to marrying the sauces with their corresponding dishes in order to suit the tastes of the two customers.

After the customers are served, each of them will be satisfied to a certain degree. Their respective satisfaction are the numbers reported back to the UoL, which, in this instance, simply calculates their average and uses it as the new value for the student's grade.

In this simple case, it would be perfectly possible to have the game report simply a final grade. However, the example illustrates the point that turning game data into grades is essentially part of the pedagogical model behind the course and, therefore, the overall instructional design is the most suitable place to tackle these dependencies and calculations.

Depending on the grade obtained in the exam, the learning flow may be altered. A very low mark would result in the learner having to play the game again, overwriting the results of the initial questionnaire so that the game is adapted again, taking into account this new and more refined information. In some cases this may mean playing the game with a lower ranking (the result of the initial questionnaire was not precise enough) or higher (the previous run of the game improved the level of the student).

On the other hand, with an average or high mark the learning process moves forward, displaying two respective pieces of content that would represent different learning paths in a longer UoL.

3 Implementing the *Art and Craft of Chocolate* UoL

The design of rich and complex learning experiences incurs high development costs, which can be alleviated by modelling mechanisms that enable the interoperability of the educational designs. This is precisely one of the objectives of the IMS Learning Design specification [16]: to provide a standardized formalization and modelling tool that enables the interoperability of complex UoLs.

In addition, the inclusion of educational games in these UoLs demands an affordable and cost-effective approach to producing such games. We can meet these requirements by using the <e-Adventure> engine, with the additional advantage of being able to produce games that can be delivered as adaptable and assessable learning objects [19]. Therefore, we have implemented the *Art & Craft of Chocolate* UoL using these technologies, as described in the following sections. The marriage of IMS LD and <e-Adventure> is also addressed in [6].

3.1 IMS Learning Design

The IMS Learning Design specification can be used to model complex instructional designs (or UoLs) in a standardized way, which allows the interchange of those designs and their processing across different learning environments.

Instead of supporting a specific set of pedagogical approaches, specification provides basic syntactic constructs in order to define learning flows consisting of plays, acts, activities, activity structures and environments.

According to [15] a "learning design" specifies the teaching-learning process; that is to say, under which conditions, what activities have to be performed by learners and teachers in order to attain the desired learning objectives.

As stated in the specification, the design of personalization in IMS LD is supported through a mechanism of conditions, properties and global elements. Personal characteristics and information about the state of the learning experience are stored in "properties". Conditions can be defined to adapt the learning design to learner characteristics in runtime.

On the other hand, the specification does not cover (as a design decision, not as a limitation) what a learning activity truly is. The approach taken in [32] envisions service-oriented architectures in which the LD environment uses the Unit of Learning as a map to guide the learning experience, requesting the different services from the available service providers when demanded by the UoL.

Given the relevance of the IMS LD specification, we used it to model our proof-of-concept UoL. The preliminary test, the adaptive game and the alternative learning paths after the game are all *Activities* in the UoL, and we define *Properties* to store the information that should be sent to the game as well as the information that the game should transmit to the UoL.

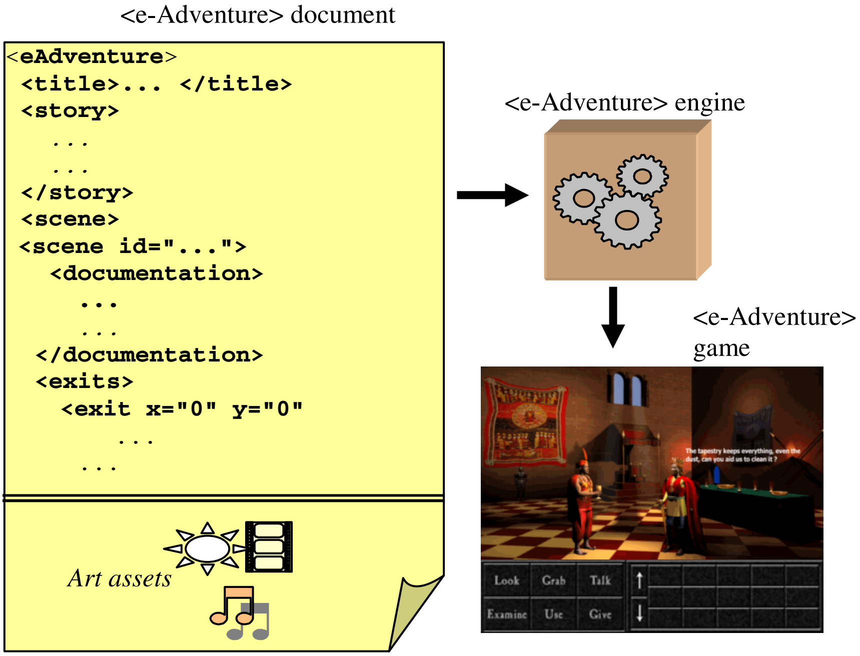
In our instructional design, the previous test sets the *initial-knowledge* property according to the requirements described in section 2.2. When the game is launched, it should have access to the value of that property in order to modify the behaviour of the game accordingly.

Similarly, the levels of satisfaction of the virtual customers in the in-game test should be stored in the *satisfaction-customer-1* and *satisfaction-customer-2* properties. As mentioned below, deciding what to do to obtain the final grade from the values reported by the games is an issue that belongs to the definition of the instructional design. In our case, the definition of the UoL simply calculates the average of these values and stores it in the *final-grade* property.

It is important to note that the concepts "having access to the value of some properties from the game" and "letting the game store some values in properties" mentioned here pose a significant technological challenge not covered by IMS LD. The rest of this section deals with this issue.

3.2 <e-Adventure>

<e-Adventure> (formerly known as <e-Game>) is a game engine designed to facilitate the creation of educational games, focusing on pedagogical aspects such as adaptation and assessment.



```
<eAdventure>
 <title>... </title>
 <story>
   ...
   ...
 </story>
 <scene>
 <scene id="...">
   <documentation>
     ...
     ...
  </documentation>
  <exits>
    <exit x="0" y="0"
        ...
     ...
```

Fig. 4. The <e-Adventure> engine produces executable videogames from their descriptions as XML documents

One of the main shortcomings in the development of educational games is the excessive development cost. Additionally, game development is very demanding in terms of programming skills and game developers are not usually experts in education. Thus, one of the main objectives of <e-Adventure> is the simplification of this process to the point of allowing instructors or educational institutions to develop educational games without requiring a broad team of expert game developers.

To achieve this purpose, <e-Adventure> extends our previous work in a *document-oriented* approach to the production and maintenance of content-intensive applications [26]. For this purpose, <e-Adventure> draws from the existing expertise in the field of Domain-Specific Languages [30] and Descriptive Markup Languages [7], and it provides an XML syntax [3] for the definition of the games. With the objective of keeping the language simple, it only supports a very specific game genre: *point and click* adventure games. Previous research experience suggests that this genre is especially appropriate for education, given its bias for content rather than action [1,14].

When using <e-Adventure>, game *writers* describe the games using the XML syntax and package them along with the necessary art assets (graphics, animations, music, etc.). The engine can read these packages and execute the games (Fig. 4).

The script for the *Paniel and the Chocolate-based Sauce Adventure* game was written as a *point and click* adventure game following the conventions and structure proposed by <e-Adventure> [22]. Then the <e-Adventure> markup was added and the art assets were gathered.

Nevertheless, in addition to facilitating the development process, <e-Adventure> is focused on the field of online education. Even though the <e-Adventure> engine can run on its own, it was designed to be deployed from an LMS and to establish a communication link that would enhance the value of its built-in pedagogical features such as adaptation and assessment, as described in the rest of this section.

In its current version, <e-Adventure> can communicate with our test environment as described in section 4.

3.3 Adaptation of the Games

Educational games developed for <e-Adventure> can be designed with adaptation in mind. When deployed from a compliant Learning Management System, the implementation of the engine can query the LMS for a set of properties which are used to adapt the game. The games are defined so that the different values of those properties will change the initial state of the game. Since in <e-Adventure> every action can be conditioned to that initial state, the values of those properties can force the game to skip some levels, to include new parts of increased complexity or to steer the student to alternative paths more suited to his/her learning style.

In this case, the initial state of the game will be different depending on the result of the initial questionnaire. As we said earlier, if the grade is below 60%, the game runs from its default initial state. However, if the grade is between 61% and 80%, the game will be set to an initial state in which the first level has already been completed. Finally, a grade above 80% sets the game in the state in which the student is ready to face the exam.

3.4 Assessment and Feedback

The <e-Adventure> engine also includes built-in assessment and feedback mechanisms. While the game is being run, the engine monitors the student's activity. The definition of the games includes information about which game states are relevant from a pedagogical perspective and, whenever the game enters one of these states, the engine notifies it in order to let the learning environment update its state.

In our case, the relevant states are those related to the completion of the exam. When the exam ends, the variables that define each customer's satisfaction are identified as relevant and the engine reports their value to the learning environment.

3.5 Integration of <e-Adventure> and IMS Learning Design

The integration of <e-Adventure> and IMS Learning Design addresses two different issues:

- On the one hand, <e-Adventure> should infer several adaptation properties from the UoL execution state. The best results can be achieved when the games are defined along with the UoL and these game properties are aligned with properties in the UoL. However, it is possible to develop these elements separately and then provide a set of rules that translate properties in the UoL into game states as depicted in Fig. 5.

- On the other hand, the assessment and feedback provided by <e-Adventure> should be communicated to the IMS LD's execution environment. Again, the easiest approach is to directly align some properties from the definition of the state of the game with properties present in the UoL and again it is possible to supply a document with rules that translate game states to properties in the UoL.

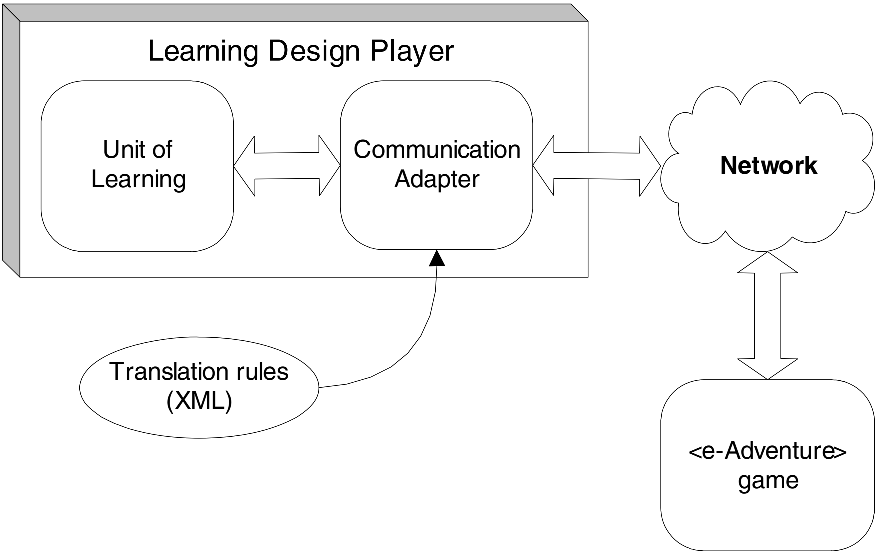


Fig. 5. Communication between the game and the UoL. The communication adapter may use an XML file with translation rules to match game states with IMS LD properties.

4 Executing the *Art and Craft of Chocolate* UoL

IMS Learning Design is simply the formalization mechanism that allows a computer to interpret the design of the UoL. To provide the learning experience modelled by the UoL we need an environment that can understand the IMS Learning Design specification and *play* the UoL. Playing a UoL implies managing the interactions of the users, providing the services required by the UoL, maintaining the values of the properties for different users in different runs of the UoL, and guiding the execution flow of the activities.

There are several systems capable of this, although in our case we have used a modified version of the SLeD environment [23]. Next sections give the details.

4.1 SLeD

SLeD (*Service-based Learning Design Player*) [23] is a front-end for a CopperCore Run-Time (CCRT) environment, which is the reference implementation of the IMS Learning Design specification [31].

One of the key elements in the CCRT is the CopperCore Service Integration (CCSI) layer [32], which enables the integration of different service providers in a CopperCore environment. Examples of services supported by CCSI include forums, web search mechanisms or assessment mechanisms based on the IMS Question & Test Interoperability specification [12].

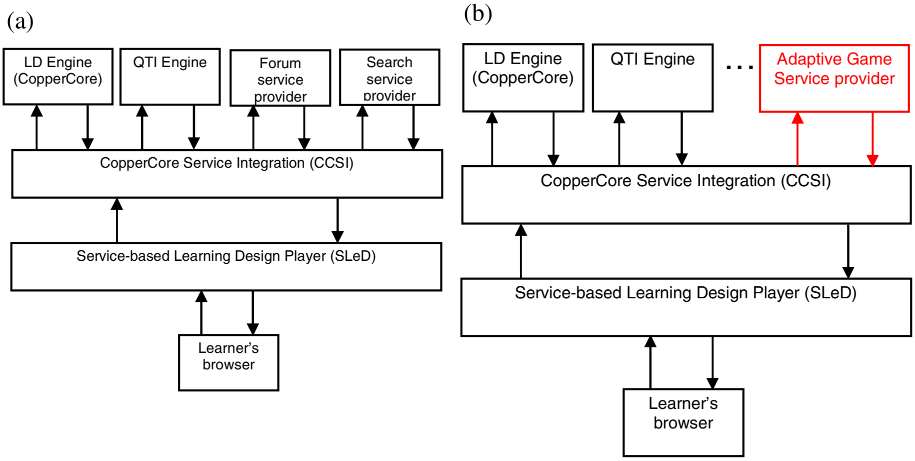

(a)
LD Engine (CopperCore)
QTI Engine
Forum service provider
Search service provider
CopperCore Service Integration (CCSI)
Service-based Learning Design Player (SLeD)
Learner's browser
(b)
LD Engine (CopperCore)
QTI Engine
...
Adaptive Game Service provider
CopperCore Service Integration (CCSI)
Service-based Learning Design Player (SLeD)
Learner's browser

Fig. 6. (a) Architecture of a SLeD environment. The student interacts with the SLeD layer which, in turn, uses the CCSI layer to communicate with the different service providers; (b) the test environment with the new service provider. It supports the execution of UoLs such as *The Art & Craft of Chocolate*.

The CCSI layer manages the communication between the service providers so that they can exchange information and the different services can be triggered when required by the current state of the UoL. Thus, when in a CopperCore environment the UoL requests an assessment or a search operation, the request is passed to the appropriate service through the CCSI layer as in Fig. 6a. This process is detailed in depth in [32].

4.2 Adding Support for Adaptive Games

Neither the IMS Learning Design specification nor the SLeD player were designed with adaptive games in mind. However, the Learning Design specification allows a certain degree of freedom when defining the services required by the different activities and the CCSI layer was specifically designed to facilitate the definition, implementation and connection of these services.

Furthermore, in order to set up our test environment for adaptive games integrated with IMS Learning Design, such as our *Art & Craft of Chocolate* case study, we modified the CCSI layer to include a new type of service called *Adaptive Game Service* (Fig. 6b). This service supports launching an adaptive game and it also establishes bidirectional communication with the game that can be used both to adapt the game and to allow whatever happens inside the game to adapt the rest of the UoL.

The other necessary step was to slightly modify the SLeD front-end so that whenever it identifies a game resource, the appropriate service will be invoked through the CCSI layer.

The result is an IMS Learning Design player enhanced to support adaptive games. When the games are launched the service sends them information that can be used to modify the game's behaviour. Additionally, the service expects the games to report back about what the learner does within the game. Besides, this protocol is understood by the <e-Adventure> educational game engine. The resulting environment allows for the execution of adaptive UoLs that, like our *Art & Craft of Chocolate*, integrate adaptive games compliant with <e-Adventure>.

5 Conclusions and Future work

While the IMS Learning Design specification is a powerful tool when it comes to modelling adaptive learning experiences, educational videogames are an ideal medium to deliver adaptive content.

Previous experiences regarding the combination of IMS Learning Design and educational games have been either based on creating UoLs that behave like games [5] or on embedding games into them and treating the games as immutable elements just like a PDF file would be treated.

The combination of these technologies is an interesting line of research. For this reason, the proof-of-concept UoL *The Art & Craft of Chocolate* was designed as presented in this work. Its design requires, at least at a basic level, the same key elements that a more complex adaptive learning experience would require.

Even if the UoL itself does not cover a lot of content or a broad and complex subject, its main objective is to eliminate the technical barriers to the joint use of IMS LD and educational games in an integrated fashion in order to provide a rich adaptive learning experience.

The implementation and the execution environment described here (using the <e-Adventure> engine and a modified version of the SLeD player and CopperCore) successfully proves this point and opens the gates to the development of more complex UoLs in which the overall learning experience and the events inside the educational games can influence each other.

Future lines of work include the design of more complex UoLs including complex games covering different subjects. Additionally, the communication mechanism and the *Adaptive Game Service Provider* developed for CopperCore were designed generically in order to allow them to work with different game implementations other than the <e-Adventure> engine.

Acknowledgements

This paper is partially supported by the European projects TENCompetence (IST-TEL/2004-2.4.10, www.tencompetence.org) and ProLearn (IST Contract number 507310, www.prolearn-project.org), and the research group <e-Ucm> (www.e-ucm.es). The Spanish Committee of Education and Science (Projects TIN2004-08367-C02-02 and TIN2005-08788-C04-01) and the Regional Government / Complutense University of Madrid (research group 910494 and grant 4155/2005) have partially supported this work.

References

1. Amory, A., Naicker, K., Vincent, J., Adams, C.: The Use of Computer Games as an Educational Tool: Identification of Appropriate Game Types and Game Elements. British Journal of Educational Technology 30(4), 311–321 (1999)
2. Betz, J.A.: Computer Games: Increase Learning in an Interactive Multidisciplinary Environment. Journal of Educational Technology Systems 24(2), 195–205 (1996)
3. Birbeck, M., et al.: Professional XML, 2nd edn. Wrox Press (2001)
4. Brusilovsky, P.: Adaptive Educational Systems on the World-Wide-Web: A Review of Available Technologies. In: WWW-Based Tutoring Wprkshop at 4th International Conference on Intelligent Tutoring Systems, San Antonio, USA (1998)
5. Burgos, D., Tattersall, C., Koper, R.: Can IMS Learning Design be used to model computer-based educational games? Binaria, 5 (2006)
6. Burgos, D., Moreno-Ger, P., Sierra, J.L., Fernández-Manjón, B., Koper, R.: Authoring Game-Based Adaptive Units of Learning with IMS Learning Design and <e-Adventure>. International Journal of Learning Technology (Special Issue on Authoring Adaptive and Adaptable Hypermedia) (In Press)
7. Coombs, J.H., Renear, A.H., DeRose, S.J.: Markup Systems and the Future of Scholarly Text Processing. Communications of the ACM 30(11), 933–947 (1987)
8. ESA, E.S.A. Essential Facts about the Computer and Videogame Industry (cited April 8th, 2007) 2005 Available from: http://www.theesa.com/files/2005EssentialFacts.pdf
9. Garris, R., Ahlers, R., Driskell, J.E.: Games, Motivation and Learning: A Research and Practice Model. Simulation & Gaming 33(4), 441–467 (2002)
10. Gee, J.P.: What video games have to teach us about learning and literacy, p. 225. Palgrave Macmillan, New York, Basingstoke (2003)
11. IMS Global Consortium. IMS Learning Design Specification (2005) (cited April 8th, 2007) Available from: http://www.imsproject.org/learningdesign/index.html
12. IMS Global Consortium. IMS Question & Test Interoperablity Specification (cited April 8th, 2007) (2005) Available from: http://www.imsglobal.org/question/index.html
13. Jenkins, H., Klopfer, E., Squire, K., Tan, P.: Entering the Education Arcade. ACM Computers in Entertainment 1(1) (2003)
14. Ju, E., Wagner, C.: Personal computer adventure games: Their structure, principles and applicability for training. The Database for Advances in Information Systems 28(2), 78–92 (1997)
15. Koper, R., Olivier, B.: Representing the Learning Design of Units of Learning. Educational Technology & Society 7(3), 97–111 (2004)
16. Koper, R., Tattersall, C.: Learning Design - A Handbook on Modelling and Delivering Networked Education and Training. Springer, Heidelberg (2005)
17. Malone, T.: What makes computer games fun? SIGSOC Bulletin 13(2-3), 143 (1982)
18. Malone, T.W., Lepper, M.R.: Making learning fun: A taxonomy of intrinsic motivations for learning. In: Snow, R.E., Farr, M.J. (eds.) Aptitude, learning and instruction III: Cognitive and affective process analysis, pp. 223–253. Lawrence Erlbaum, Hillsdale, NJ (1987)

19. Martinez-Ortiz, I., Moreno-Ger, P., Sierra, J.L., Fernández-Manjón, B.: Production and Deployment of Educational Videogames as Assessable Learning Objects. In: Nejdl, W., Tochtermann, K. (eds.) EC-TEL 2006. LNCS, vol. 4227, Springer, Heidelberg (2006)
20. Mitchell, A., Savill-Smith, C.: The Use of Computer and Videogames for Learning: A Review of the Literature. m-Learning, Trowbridge, Wiltshire: Learning and Skills Development Agency (2004)
21. Moreno-Ger, P., Martínez-Ortiz, I., Sierra, J.L., Fernández-Manjón, B.: Language-Driven Development of Videogames: The <e-Game> Experience. In: Harper, R., Rauterberg, M., Combetto, M. (eds.) ICEC 2006. LNCS, vol. 4161, Springer, Heidelberg (2006)
22. Moreno-Ger, P., Sierra, J.L., Martínez-Ortiz, I., Fernández-Manjón, B.: A Documental Approach to Adventure Game Development. Science of Computer Programming 67, 3–31 (2007)
23. OUUK. Sled player (2005) (cited April 8th, 2007) Available from: http://sled.open.ac.uk
24. Paramythis, A., Loidl-Reisinger, S.: Adaptive Learning Environments and eLearning Standards. Electronic Journal of eLearning 2(1), 181–194 (2004)
25. Prensky, M.: Digital Game Based Learning. McGraw-Hill, New York (2001)
26. Sierra, J.L., Fernández-Valmayor, A., Fernández-Manjón, B.: A document-oriented paradigm for the construction of content-intensive applications. The Computer Journal 49(5), 562–584 (2006)
27. Sim, G., MacFarlane, S., Read, J.: All work and no play: Measuring fun, usability, and learning in software for children. Computers & Education 46(3), 235–248 (2006)
28. Squire, K.: Game-Based Learning: An X-Learn Perspective Paper. MASIE center: e-Learning Consortium (2005)
29. Squire, K.: Video games in education. International Journal of Intelligent Simulations and Gaming 2(1), 49–62 (2003)
30. Van Deursen, A., Klint, P., Visser, J.: Domain-Specific Languages: An Annotated Bibliography. ACM SIGPLAN Notices 35(6), 26–36 (2000)
31. Vogten, H.,, Martens, H.: CopperCore 3.0. 2005 (cited April 8th 2007) Available from, http://www.coppercore.org
32. Vogten, H., Nadolski, M.H.R., Tattersall, C.: CopperCore Service Integration, Integrating IMS Learning Design and IMS Question and Test Interoperability. In: 6th IEEE International Conference on Advanced Learning Technologies, IEEE Computer Society Press, Kerkrade, The Netherlands (2006)