

This document is published in:

Delgado Kloos, C. et al. (eds.) (2011) *Towards Ubiquitous Learning: 6th European Conference of Technology Enhanced Learning, EC-TEL 2011, Palermo, Italy, September 20-23, 2011. Proceedings.* (Lecture Notes in Computer Science, 6964). Springer, 455-460.
DOI: http://dx.doi.org/10.1007/978-3-642-23985-4_38

© 2011 Springer-Verlag Berlin Heidelberg

Orchestrating Learning Activities in 3D Virtual Worlds: IMS-LD in Open Wonderland

David Maroto, Derick Leony, Carlos Delgado Kloos,
María Blanca Ibáñez, and José Jesús García Rueda

Telematics Engineering Department, Universidad Carlos III de Madrid,
28911 Leganés, Madrid, Spain
{dmaroto,dleony,cdk,mbibanez,rueda}@it.uc3m.es

Abstract. Immersive environments such as virtual worlds and virtual reality platforms are being increasingly used for educational purposes. Possibilities of these environments are huge, but also are the technical challenges that have to be overcome in order for these platforms to become more usable. One of the main problems that educators have to face when designing an educational experience for virtual worlds has to do with the orchestration of the learning sequence. In this paper, we present a novel approach to support the deployment and execution of Units of Learning (UoLs), described with an Educational Modeling Language such as IMS-LD [1], into a virtual world, with the aim to facilitate the creation of well defined learning sequences for these platforms.

Keywords: 3D virtual worlds, Education, Learning design, Orchestration, Open Wonderland, CopperCore, IMS-LD.

1 Introduction

Immersive environments such as virtual worlds and virtual reality platforms are being increasingly used for educational purposes [2], [3]. These platforms present a very rich interface, where users can find many objects and resources that can be accessed at any time during the learning experience. But this behavior is often not desired, since the creator of the learning experience usually wants learners to perform the tasks and to use the resources in a particular order.

Educational Modeling Languages (EMLs) [4] have emerged as a very useful tool in order for the teachers to carry out this orchestration of the learning activities. In our case, we have chosen IMS-LD because it is the most wide spread EML, becoming the *de facto* standard adopted by the educational community.

This orchestration of the learning sequence inside a virtual world offers two main benefits:

1. Teachers and course creators can gain control over the content that is accessible inside the virtual world at every moment.

2. In turn, learners can see only the resources needed to perform the activity in which they are involved each time, which helps to reduce their sense of confusion when entering an environment in which they can move anywhere and access too many resources.

In this paper, we present the work we have carried out in order to integrate an IMS-LD engine into a virtual world based on Open Wonderland [5].

The paper is organized as follows: in Section 2 we introduce the main characteristics of IMS-LD. Open Wonderland and its main features are described in Section 3. Section 4 explains the three core elements that support our work. We present the integration of an IMS-LD engine with Open Wonderland in Section 5. Finally, we discuss the conclusions and future work in Section 6.

2 IMS Learning Design

IMS-LD is a specification that enables to describe an educational experience, whatever the pedagogy used, based on the metaphor of a play. IMS-LD provides a generic and flexible XML language that introduces the concepts of Acts, Activities, Environments, Roles and Resources to describe the learning sequence in a way that can be later understood by a runtime system.

Due to its complexity, IMS-LD specifies three levels of implementation and compliance (A, B, C), where each level adds new functionalities over the previous one. In our work, we offer support for UoLs designed using only level A.

All the elements used by IMS-LD (i.e. activities, environments, etc.) are written down in a manifest file called *imsmanifest.xml*. Later, this manifest file is packaged together with the resources. This package, known as a Unit of Learning (UoL), follows the IMS Content Packaging specification [6] on delivering reusable learning content. This specification establishes a life cycle for a UoL consisting on three phases: authoring, deployment and execution.

Fortunately, we do not need to write the manifest file and create the package on our own during the authoring phase. There are tools that help us in the process of creating a UoL compliant with the IMS Content Package specification, such as ReLoad¹ editor.

Once we have the package representing the UoL, it can be delivered and deployed into a runtime system such as CopperCore².

3 Open Wonderland

Open Wonderland is, in the words of its creators, “a 100% Java open source toolkit for creating collaborative 3D virtual worlds”. This toolkit is completely extensible by means of modules that extend the functionality to create entire new worlds, new features in existing worlds, or new behaviors for objects and avatars.

¹ <http://www.reload.ac.uk/editor.html>

² <http://coppercore.sourceforge.net>

There have been many learning experiences developed over Open Wonderland, but all of them lack the same characteristic: there is no way to orchestrate a sequence of activities. The virtual world remains the same no matter the evolution of the learners. This is the main behavior we want to change with our developments. We will explain the tools and technologies used to carry out our work in the next section.

4 Elements Supporting our Developments

We have developed a module for Open Wonderland which is responsible for the second and third phases of the life cycle of a UoL: deployment and execution. The module we have developed introduces modifications both in the clients and server of Open Wonderland. We have also designed a new web interface in the web administration page of Open Wonderland, which allows managing all the tasks related to the deployment, deletion and configuration of UoLs.

But, before we explain the details of the implementation, it is important to know the three elements that support the rest of our developments. These elements are: the snapshots system of Open Wonderland, the security module of Open Wonderland and the CopperCore runtime engine.

Open Wonderland Snapshots System. The snapshot of an Open Wonderland virtual world is a collection of files, representing every single object that was present inside the virtual world just when the snapshot was created. These objects are represented as XML files (one file for each object). These files contain all the information needed for the platform to render each of the 3D virtual objects. We will use these files as the resources for the learning activities in which the course creator will divide the learning sequence.

Open Wonderland Security Module. 3D objects of the Open Wonderland platform can have some capabilities associated to them. Among these capabilities, there is one that can make the object visible or invisible for the users, depending on the group the user belongs to. This capability is expressed through the security module.

CopperCore Runtime Engine for IMS-LD. IMS Learning Design is a complex, semantically rich specification, so it is not trivial to provide full support for it. All the checking, synchronizing and personalizing of the learning process is called the business logic of Learning Design, and this is exactly what CopperCore handles for us. CopperCore is a J2EE runtime engine for IMS-LD which exposes a SOAP API [7] to communicate with it and to make use of its functionality. This engine is the basis for our integration of IMS-LD into an Open Wonderland virtual world.

5 Integration of Coppercore with Open Wonderland

The architecture of the system that we have developed is shown in Fig 1.

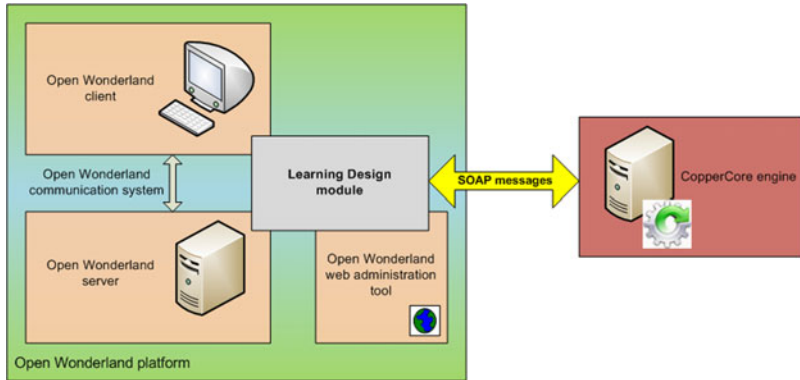


Fig. 1. System architecture

The best way to explain our work is to follow the sequence of steps that a teacher should take in order to have a learning experience, based on IMS-LD specification, deployed and running into Open Wonderland. We can divide these steps into *authoring*, *deployment* and *execution*.

5.1 Creating the UoL: Authoring Phase

This phase starts with an empty world in Open Wonderland. In this stage, the course creator must populate the virtual world with all the 3D objects needed to perform each of the activities, no matter the sequence, and later create the UoL. So in fact, this phase can be divided into two subphases: Creation of a snapshot with all the 3D objects needed to perform the whole learning experience, and creation of the learning sequence using the objects from the snapshot and an IMS-LD editor (e.g. ReLoad).

The result of this phase is a UoL containing the *imsmanifest.xml* which describes the sequence of learning activities for each role, and the resources associated to each activity.

5.2 Deployment of the UoL

Once we have the learning experience defined in the UoL, we need to deploy it into a runtime system to later execute it. For this purpose, the teacher can use our new Open Wonderland web interface to upload the UoL into CopperCore. When the teacher clicks the upload button, it triggers a series of functions in order for the platform to be configured for the next phase. These functions are:

- Validation and publication of the UoL into CopperCore.
- Creation of a new Open Wonderland snapshot. This snapshot contains exactly the same objects that the snapshot created during the authoring phase, but modified through the inclusion of the security module.

- Creation of new groups into Open Wonderland. Our module creates a new group for each of the activities, environments and roles defined in the UoL. We refer to groups for activities and environments as “control groups”, since they will be used internally by our module to orchestrate the learning sequence.

5.3 Execution of the UoL

The first thing that the teacher has to do before the experience starts, is to place each user in his corresponding group (role) inside Open Wonderland. Our module is responsible for passing all the users registered in Open Wonderland to CopperCore, with the aim to also register them into the IMS-LD engine. This way CopperCore can track the itinerary of every single participant and react in consequence.

Once CopperCore has registered every user, our module asks it for the activity tree of each one of them. These activity trees are parsed in order to obtain the activity that every user has to perform first, with the aim to introduce them into the control group corresponding to that activity. This way, when a user accesses to the virtual world, he is able to see the objects needed to perform the first activity planned in the UoL for him, and only that objects.

We have also designed an enhanced client interface for Open Wonderland clients that allows users to manage their activities. Through this interface, users can obtain the description of the current activity or terminate it and move to the next. Every time a user pushes the button to terminate an activity, a message is sent to CopperCore in order to calculate the new activity tree for every user and introduce them into the corresponding control group for the activity that they have to perform next.

As we can note, all the tasks related to the execution of the UoL are handled by CopperCore. Our module is responsible, on the one hand, for communicating to CopperCore the events that take place into the virtual world which can produce a change in the state of the learning sequence (i.e. the termination of an activity). On the other hand, our module is also responsible for obtaining the new status of the UoL from CopperCore and move the users to their new corresponding control group into Open Wonderland.

6 Conclusions and Future Work

We have presented a novel work to integrate an IMS-LD engine into a virtual world, with the aim to orchestrate learning sequences in these kinds of platforms. We have only found a single reference about a similar work [8], but the approach followed in our developments is somehow different. We have designed a functionality that allows the complete reconfiguration of the virtual world during the runtime, as participants in the learning experience evolve, rather than designing some new virtual world objects that communicate with an IMS-LD engine to obtain the resources of the activities, as done in [8]. In our approach, 3D virtual objects are in fact the resources of the activities.

In the field of future developments, we must point out that although CopperCore offers full support for IMS Learning Design including levels A, B and C, our module currently supports only level A. Development of new functionalities to support levels B and C is in the roadmap. We would also like to test this work with real teachers and course creators, as well as students, with the aim to evaluate its usability and global performance.

Acknowledgments. This research has been partially supported by the following projects: The Spanish CDTI project “España Virtual” funded by the Ingenio 2010 programme, subcontracted by Deimos Space; The Spanish project “Learn3: Towards Learning of the Third Kind” (TIN2008-05163/TSI) funded by the Spanish “Plan Nacional de I+D+i” of the Ministry of Research and Innovation; The project “eMadrid: Investigación y desarrollo de tecnologías para el e-learning en la Comunidad de Madrid” (S2009/TIC-1650) funded by the Government of the Region of Madrid.

The authors want to thank Francisco José Sánchez for his valuable help during the implementation phase.

References

1. IMS Global Learning Consortium, Learning Design Specification (2003), <http://www.imsglobal.org/learningdesign> (accessed: June 2011)
2. European Community, EU project for language education in 2nd Life, <http://www.avalonlearning.eu> (accessed: June 2011)
3. Ibáñez, M.B., García, J.J., Galán, S., Maroto, D., Morillo, D., Delgado Kloos, C.: Multi-User 3D Virtual Environment for Spanish Learning: A Wonderland Experience. In: Proc. 10th IEEE International Conference on Advanced Learning Technologies (ICALT 2010), Sousse, Tunisia, pp. 455–457. IEEE Computer Society Conference Publishing Services, Los Alamitos (2010), doi:10.1109/ICALT.2010.132
4. Koper, R., Manderveld, J.: Educational Modelling Language: Modelling Reusable, Interoperable, Rich and Personalised Units of Learning. *British Journal of Educational Technology* 35(5), 537–551 (2004), doi:10.1111/j.0007-1013.2004.00412.x
5. Open Wonderland, Open Wonderland 3D Virtual Worlds (2010), <http://openwonderland.org> (accessed: June 2011)
6. IMS Global Learning Consortium, Content Packaging Specification (2009), <http://www.imsglobal.org/content/packaging> (accessed: June 2011)
7. CopperCore, SOAP API Javadoc (2008), <http://coppercore.sourceforge.net/documentation/javadoc/soap/index.html> (accessed: June 2011)
8. Fernández-Gallego, B., Lama, M., Vidal, J.C., Sánchez, E., Bugarín, A.: OPENET4VE: A Platform for the Execution of IMS LD Units of Learning in Virtual Environments. In: Proc. 10th IEEE International Conference on Advanced Learning Technologies (ICALT 2010), Sousse, Tunisia, pp. 472–474. IEEE Computer Society Conference Publishing Services, Los Alamitos (2010), doi:10.1109/ICALT.2010.137