Integration in Generic Tool Learning Design to Support Complex Learning Methodologies

Luis de la Fuente Valentín

Department of Telematics Engineering
Carlos III University of Madrid
Av. Universidad, 30 E-28911 Leganés, Spain
http://gradient.it.uc3m.es

Abstract: Current learning modelling languages do not allow formalization of scripts where generic tools are required. This limitation is especially relevant on remote courses when using collaborative models. Integrating information from external services into IMS LD will enhance its expressiveness by reducing the required administrative tasks at deploy time. This document analyzes external service requirements on learning model, and discusses a solution consisting on an extension to be used in conjunction with the current IMS LD specification.

Keywords: IMS LD, service integration, external tools, distance learning, standards, e-learning

Introduction

Availability of high quality learning objects does not guarantee the best results in a learning experience. Content must be used together with an appropriate pedagogical model. Learning modeling languages are a step towards reusability of learning materials. They allow course designers to create scripts where pedagogical models are formalized.

The IMS LD specification provides an activity-based modelling language with which a wide range of methodologies can be expressed. Content is delivered following a given path, satisfying requisites of the used pedagogy. Also, there are mechanisms to identify different user roles in the course, which allow working with collaborative learning models. Conditional behaviour of sequence of activities is also possible, by defining a property based course state and conditions related to the defined state.

Although the expressiveness of the LD specification is enough to formalize collaborative models, there are some features that are not still covered. For instance, the use of this kind of models on distance learning where communication tools are not enough and collaborative software is required. The problem with using external tools and services is the difficulty to integrate the information they provide and the formalized learning model. This document discusses the problem of generic services integration with the IMS LD specification.

This paper is organized as follows: First, an overview of the problem is given. Next section compiles current ideas or initiatives which address the problem. Last section summarizes the work done so far in the discussed area and presents the ideas to tackle the problem.

The problem of external tools usage on IMS LD

Learning Objects were born with the idea of encapsulating concepts atomically. Each Learning Object must be self-contained in order to guarantee reusability and portability.

However, a collection of Learning Objects is not enough when the aim is to provide high quality learning courses. The order in which content is delivered, the time spent in activities, and the contents that students require depending on their user profile are relevant in pedagogical models.

The IMS LD specification joins both learning methodologies and content, and packages them in a standard[1] format. The language has been designed to support a wide range of pedagogies in online learning, and it can be adapted to be used in other scenarios such as blended learning.

[11] The IMS LD specification has not been recognised as a standard, but it is the most used language to package courses.

Units of Learning

In the context of IMS LD, when course designers finish the authoring stage, the result is a single compressed file (usually ZIP file) containing all the content and the instructions to deliver them at run-time. This file is called a Unit of Learning (henceforth UoL) and is the package that will be distributed to teachers.

To deploy the packaged course, a compatible engine is required. This is a software application able to read the file and deliver content in the expected sequence. Proper UoL creation guarantees the learning flow to be the same in different instances of the same course, with the condition of having a IMS LD compatible engine.

IMS LD uses of IMS Content Packaging [1] to store the entire course related files, and extends this info with a pedagogic model. UoL behaviour can be described using the theatre metaphor [3]. A course is a theatrical play divided in one or more acts in which the actors (students and/or teachers) play different roles. The acts occur in sequence, but inside acts parallel situations may take place. During the course, a set of previously defined properties may change, allowing the content to be dynamically adjusted. The change of property values is done by events that are defined and packaged together with the course.

External requirements on collaborative schemes

The IMS LD specification allows using content, which is defined at design time, and services. That is, tools that support interactions with students. In IMS LD, only conference and send-mail services require the use of external tools. However, there are some learning schemes in which additional tools are required.

Collaborative schemes require high level of interaction between users. This means communication, resource exchange, creation and modification. As a result, this learning flow – when applied in distance learning – requires the use of specific tools, which are not integrated in the LD specification. A simple use case is an activity in which the learners must create a conceptual map, and they are required to do it in a shared board. The deployment of this activity requires the use of a tool which provides the shared board.

It is common in a UOL to have an activity that is open until the score obtained in a given test reaches a threshold. Working in groups may imply that this threshold is not only based in a personal score, but in the mean value of the entire group. As a result, the script language used to package a course might include a way to compile group and social network information.

The use of these tools is not granted by the specification, but the problem appeara when creating the course UOL. The use of external URLs to refer these services does not guarantee availability, and makes UOL not the self-contained.

In the way to integration: Existing initiatives

The current version of IMS LD (version 1.0) is recent (February 2003) and the existing tools for authoring and deploying UOLs require deep understanding of the specification. The specification is then at its “validation” stage, when actual problems are being identified. Despite of this, there are some initiatives providing certain amount of integration with other tools and/or specifications.

Low level integration

The use of external URLs integrates any kind of tool in a given course. This is a one way integration, meaning that the learning flow calls other resources to be used but these tools do not send back any information. The advantage of URLs is the simplicity of the solution. On the other hand, feedback from the external tool may be required from the learning flow in order to adjust property values and test conditions.

Besides, the availability of external resources cannot be guaranteed by this method. A learning flow must not depend on availability of non-packaged tools, and since there is no expected feedback there is no way to check for.

QTI and SCORM: Integration with other standards

Similar to LD, IMS Question and Test Interoperability [1] provides a language to create, package and deploy assessments in an easy and interoperable format. Since one of the most relevant parts of any course is grading students, the integration of assessment in a packaged course becomes a desired feature. By combining this, it is desirable a way to package QTI resources within a UoL, having the possibility of manage the student results inside the LD flow.

This possibility is discussed in one of the guides provided by the IMS Consortium, in which a way to integrate properties from LD and QTI is provided. This integration leads to information exchange between engines, because LD and QTI players may be different software packages from different vendors. IMS guidelines do not cover the software architecture details and they are only related to the joint behavior. The implementation details of QTI and LD integration can be found at [9].

Analogously, the SCORM model could be used together with IMS LD. This would imply that a lot of existing SCORM packages could be reused within LD, which has the advantage of having a more powerful sequencing scheme. As a result, existing content would be used in courses with very different learning methodologies, including collaborative schemes.

The SCORM usage in IMSLD has been discussed at [4], but there is no example of use, or implementation guideline.

Communication between parallel data flows

The learning flow described by Learning Design can be expressed as an isolated data flow where the sequence of events depends on some events that are packaged together with the UoL. In this sense, the use of external tools can be described as an external data flow, whose information is susceptible to be used as LD events, meaning an external modification of the learning flow.

The composite-based approach [5] proposal is to use Workflow Process Definition Interface (XPDL) to manage the data flow and IMS LD to manage the learning flow. The required coordination between them can be reached with the use of a Petri Net Markup Language.

Integration with eGames

The e-Adventure project assumes the benefits that electronic games produce in learning schemes by allowing the creation of new games to non-expert authors. The project has as its main goal the creation of a gaming environment where graphical adventure videogames can be described by documents that contain the game behavior. These documents are written in a markup language called e-Adventure.

The e-Adventure is based on XML. This fact facilitates the integration of both technologies: the path of the course and its adaptation options can be defined using LD, and the states of the course can be defined by e-Adventure based activities.

In [8], a mechanism to reach this integration is described in detail. It is based on the IMS LD and QTI integration, which has been developed using the capabilities of Coppercore and the Coppercore Service Integration layer.

Generic Services Integration

The stages of the research project are outlined as follows.

- Implement of an IMS LD engine. This provides an experience to help the understanding of the specification from the technical point of view [6].
- Design courses with different requisites to increase the understanding of the LD capabilities, since course designer’s requirements are rather different than the developers [2].

Current ideas

Depending on the nature of the external tool to be used by the UOL, the elements where the relevant information is stored may change significantly. For example, some tools store the level of participation, but others are focused just in the obtained results. This fact increases the complexity of the integration. The approach to solve the problem consists on external services integration in general terms, allowing particular tools to internally decide the behavior of specific details.

The idea is to increase the scope of the specification by proposing a new XML set of elements that enhances the functionality of the existing version of IMS LD. The use of the proposed set of elements is optional, so compatibility will be guaranteed. If the used tools and the IMS LD runtime environment support the new set of elements, it will be used and the course will be able to take advantage of this. On the other hand, if support is not available, the engine will follow the “simple” alternative, minimizing the impact over students.

One of the aims of the new element is to describe generic services similarly that current services do. Awareness box, shared board, file-storage are examples of services that are useful in collaborative learning and that can be described in the discussed work.

Moreover, the proposed XML application is able to capture external information and translate it on LD property values. This feature is similar to the integration of IMS LD with QTI, but including methods to define how to capture group information. To make possible this information exchange, an interface layer with a fixed API is required. Any tool willing to work with this scheme must provide support for this API.

The ideas in this document will be developed and deployed under .LRN [7], which is the platform where the LD engine has been created. Nevertheless, the XML application will not be focused on the capabilities of this LMS, and it will be generic enough to be deployed on any LD compliant software.

Differences and similarities with other solutions

Any proposed solution for the integration problem can be discussed in terms of simplicity of use, coverage of the problem and compatibility with current the version of specification.

- Simplicity of use: Since IMS LD is complex by itself; the aim of the generic services integration is not to increase this complexity. Of course, plain use of URL is the simplest solution, but it does not cover the whole problem. Communication between parallel data flows proposal implies the usage of at least three different languages, with the implicit increase of complexity on both design and deploy time.
- Coverage of the problem: The discussed approach will be developed to provide a realistic user interface for both designers and teachers, the coverage of new elements will be restricted to the most relevant issues. Despite the coverage is lower than in other solutions, it is not a disadvantage since all well known cases will be able to be used. This approach has been taken in CSCL by the use of patterns to ease course creation.
- Backward compatibility: Extending the specification does not take into account with compatibility issues. Due to this, all the proposed features will be complemented with an alternative in the learning flow in case the extra requirement is not supported. Backward compatibility issues during course design are also present in the composite-based approach, in which no changes in the specification are required.

Conclusions

Course packaging provides a way to easily create contents and design learning materials in and standard methodology. Main objective is reusability of courses, especially in those cases where complexity at runtime becomes a handicap for teachers.

Current modelling languages provides a way to design and package complex pedagogical models, without taking into account the problem of external services. In this sense, generic services integration has the idea of solving this problem, allowing the deployment of a wide range of learning methodologies also on distance learning.

To solve the difficulty with the integration of external tools in an IMS LD course, new *generic service integration* elements will complete the IMS LD specification, increasing its expressiveness and thus making it more versatile without having cost on usability.

References

[1] IMS Global Consortium: www.imsglobal.org, (On line).

[2] L. de la Fuente Valentín, A. Pardo, C. Delgado Kloos, *Using Learning Design to Deploy and Administer Engineering Courses* (accepted for publication). IEEE Frontiers in Education Conference, 2007.

[3] R. Koper, B. Oliver, and T. Anderson, *IMS Learning Design Information Model, version 1.0,* IMS Global Learning Consortium, Inc, 2003.

[4] C. Tattersall, D. Burgos, H. Vogten, H. Martens, and R. Koper. How to use ims learning design and scorm 2004 together. In International Conference on SCORM 2006. Tamkang University, Taipei, Taiwan, January 2006.

[5] Palomino-Ramírez Luis, Martínez-Monés Alejandra, Bote-Lorenzo Miguel L., Asensio-Pérez Juan I., Dimitriadis Yannis, *Data Flow between Tools: Towards a Composition-Based Solution for Learning Design*, Proceedings of the 6th IEEE International Conference on Advanced Learning Technologies*, ICALT 2007, Niigata, Japan*, 2007.

[6] J. Escobedo del Cid, L. de la Fuente Valentín, S. Gutiérrez, A. Pardo, C. Delgado Kloos, *Implementation of a Learning Design Run-Time Environment for the .LRN Learning Management System* (accepted for publication). Journal of Interactive Media in Education, 2007 [jime.open.ac.uk/].

[7] The .LRN platform. dotlrn.org, Dec. 2006. (On line).

[8] D. Burgos, P. Moreno-Ger, J.L Sierra, B. Fernández-Manjón, R. Koper. (2007). Authoring Game-Based Adaptive Units of Learning with IMS Learning Design and <e-Adventure>. *International Journal in Learning Technologies*.

[9] H. Vogten, H. Martens, R. Nadolski, C. Tattersall, P. Van Rosmalen, R. Koper. (2006). Integrating IMS Learning Design and IMS Question and Test Interoperability using CopperCore Service Integration. *Proceedings of Proceedings of the International Workshop in Learning Networks for Lifelong Competence Development at the TENCompetence Conference in March 30th-31st, 2006.*, Sofia, Bulgaria: TENCompetence