This document is published in:

Pedro J. Muñoz-Merino, Carlos Delgado Kloos, "Enabling Interoperability for LMS Educational Services", Computer Standards & Interfaces (2009), 31 (2), 484-498.

DOI: http://dx.doi.org/10.1016/j.csi.2008.06.009

Enabling Interoperability for LMS Educational Services

Pedro J. Muñoz Merino*, Carlos Delgado Kloos, Jesús Fernández Naranjo

Department of Telematics Engineering

Universidad Carlos III de Madrid

Av. Universidad, 30. E-28911 Leganés (Madrid) Spain

pedmume@it.uc3m.es, **cdk@it.uc3m.es**, **jfernandez@inv.it.uc3m.es**

Abstract

Nowadays, e-learning is undergoing a standardization process. In this paper, an overview of e-learning standardization state of the art is provided and the relationship between Learning Management System (LMS) functionalities and current e-learning standards is presented. Some lacks are found and the importance of defining new standards to cover several LMS aspects is justified. This work describes new e-service specifications for LMS final user application functionalities in order to cover such lacks. In addition, a web-based authoring tool has been implemented according to these new specifications, generating XML files. In this way, interoperability between different new LMS aspects is enabled.

Keywords: *Authoring tool; e-services; e-learning; interoperability; standards*

* Corresponding author: e-mail: pedmume@it.uc3m.es Telephone: (+34) 91-624-8801 Fax: (+34) 91-624-8749

1. Introduction

Nowadays, e-learning is undergoing a standardization process. This is crucial in order to enable interoperability and reusability between different distance educational systems. As e-learning is a relatively new emerging and changing science, at present there are only six defined standards by IEEE LTSC [15] (Learning Technology Standards Committee). On the other hand, there exist multiple specifications (usually also called standards by many authors. In this paper, the criterion is to refer as specification when no standardization organization has approved it at this moment) created by different organizations like the IMS Consortium [12], ADL [1] or the OKI project [21]. There are some e-learning aspects for which different specifications overlap [6], and other important aspects for which there is not any specifications or it is not specific to e-learning.

The main computer systems that are used for distance education through Internet are called Learning Management Systems (LMSs). LMSs provide a set of functionalities or educational services. They can be open source, like .LRN (http://www.dotlrn.org/) , Moodle (http://moodle.org/) or Dokeos (http://www.dokeos.com/); or commercial like WebCT/BlackBoard (http://www.webct.com/) . Ideally, these systems should be according to most e-learning standards and specifications.

LMSs' future tendency is to be service oriented. In [4], it is called as the next LMS generation. In this scenario, LMSs are based on modular components and they can support different services that are not stick to a specific platform. This is also according to the IMS Abstract Framework [8] philosophy (that lists different Applications, Services, and Components) and the OKI [21] architecture. OKI provides a based-layer architecture that is shown in Fig. 1. The *infrastructure* represents the final resources of an institution, such as file systems or data bases. The *common services* are services that are used by several educational applications, such as

authorization or authentication. The *educational services* are specific educational modules like assessment or Course Management. Finally, the *educational applications* are the applications a user directly interacts with and these educational applications can use the implemented educational and common services.

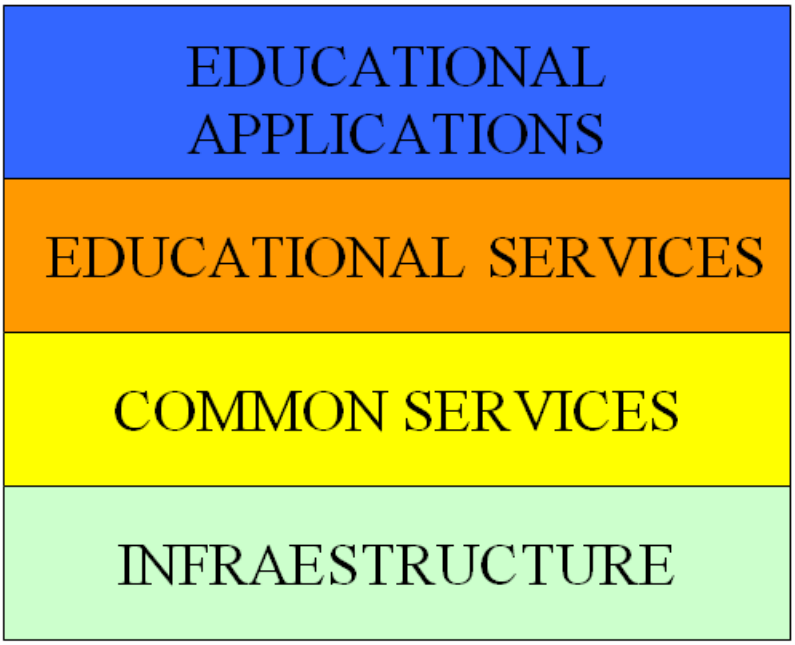

Figure 1: OKI architecture based on layers

The IMS Abstract Framework architecture is very similar to the defined by OKI, and a perfect relationship among layers of both architectures can be established. Both architectures capture the strong importance of LMS services.

In this context, we envision that the ideal scenario is one in which all the different educational services can be interoperable among different LMSs; and in which the entire design of different LMS courses can be done off-line (outside of LMSs) in an easy way for teachers without high technological knowledge using proper authoring tools; and next these complete courses can be imported within the different LMSs. To achieve this ideal approach, several issues should be addressed that are not solved at this moment. This work contributes in some of them.

1.1 Contribution of our work

In this work, we contribute to the ideal approach regarding the following issues:

1) We provide a new vision of entire reuse within LMSs. We propose a way of using the IMS Content Package (IMS-CP) specification to enable the global organization of all the elements within an LMS course. A new vision of complete reuse within LMSs is provided, in which everything of an LMS can be reused as a part or as a whole. It allows the design of courses off-line with the help of a proper authoring tool. Several use cases of this new vision are presented in the paper. In this line, we show a new way of using the IMS-CP specification in order to define general configuration information of an LMS course, how to set the layout, or how to describe the structure of the different resources, and services of an LMS course.

2) It sets the relationship between main LMS services and present e-learning specifications, determining some lacks. For each LMS service, its relationship to present standards is provided. In addition, some lacks of educational standards for some LMS services are determined, explaining and illustrating the advantages of such standardization based on the presented case studies.

3) We present new e-learning specifications for several educational services. We propose new data models and XML bindings that describe the different features of the selected educational services for which there are lacks of specifications. These new specifications are based on our analysis over different LMSs (mainly Moodle, .LRN, and BlackBoard/WebCT), other different tools, and our own ideas. For each educational service, the common features that are present among most of the different LMSs should be determined. These are candidates to be mandatory fields in the specification. In a similar way, important features that are present in some LMSs but not in others are candidate to be optional fields in the specification. Finally, some marginal features of some LMSs are candidates not to be part of the new specifications.

4) We present a new authoring tool for the new LMS service specifications. The authoring tool that we have created integrates all our new defined specifications for LMS educational

services and can generate XML files according to such new specifications. As far as we know, it is the first authoring tool for a wide spectrum of LMS services according to defined XML specifications. In addition, it shows that these LMS services can be put in terms easy to use by teachers without high technical knowledge.

1.2 Structure of the Paper

The remainder of this paper is organized as follows. Section 2 describes the main present e-learning standards and specifications. Section 3 explains the motivation of this work based on some case studies. In Section 4, we analyze the relationship between different LMS services and present e-learning specifications. Section 5 shows how to use IMS-CP for describing the overall organization of a complete LMS course; while Section 6 explains the semantics of the new different specification models and provides some details of the XML binding. In Section 7, the implemented authoring tool is presented. In Section 8, there is a discussion about related work in relationship to our work. Finally, Section 9 is devoted to our conclusions.

2. State of the art of E-learning standards and specifications

Different studies show the importance of interoperability and standards in e-learning, such as in [25]. There are multiple consortiums and organizations that create e-learning standards and specifications. Some of the most important are: IMS Global Learning Consortium [12], ADL (Advanced Distributed Learning Initiative) [1], AICC (Aviation Industry CBT Committee) [2], OKI project [21], IEEE LTSC (Learning Technology Standards Committee) [15], and ISO/IEC Joint Technology Committee Subcommittee on Standards for Learning, Education and Technology [13].

There are different classifications of standards depending on the categorization criterion. We have divided the existing standards in the following groups:

- **Architectural standards.** They try to define architectures involving learning components. The IEEE Standard for Learning Technology Systems Architecture (LTSA) is one example.
- **Data Standards.** They try to define data models for enabling the interchange of information between different learning systems. They have a data model and an XML binding. Most of IMS specifications are of this type.
- **Behavioural Standards.** They try to define programming interfaces for building educational modules in order to enable communication calls between different learning systems. The OSIDs of OKI are examples of this type. Note that each educational service can have a data standard and also a behavioural one. For example, there is an OKI OSID for assessment modules and there is also the IMS-QTI specification for describing the assessment data.

In the following subsections, there is an overview of current e-learning standards and specifications. Several works explain most important ones in a deeper way such as [8] or [9].

2.1 IMS Global Learning Consortium

IMS is a non profit organization that integrates different actors. IMS has defined a wide set of specifications related to different aspects of e-learning. Table 1 shows all the IMS specifications, with a brief description about the topic that covers. All the commented specifications are accessible through the IMS Global Consortium web page [12].

Table 1: An overview of IMS specifications

IMS Specification	Acronym	Brief Description
AccessForAll Meta-data		It describes and identifies learning resources according to user preferences and needs
Reusable Definition of Competency or Educational Objective	IMS-RDCEO	It defines competencies for a course, student, etc.
Content Packaging Specification	IMS-CP	It structures and packages the different course resources
Digital Repositories Specification	IMS-DR	It provides a framework to establish connections between repositories
Enterprise Specification		It defines persons, groups and the group relationships
Enterprise Services Specification		It defines how to interchange the information defined in the IMS Enterprise Specification
ePortfolio Specification		It defines general different features of a subject
General Web Services	IMS-GWS	It provides interoperability between different Web Service based specification implementations that are in different systems
Learner Information Package	IMS-LIP	It defines a user modelling including preferences, details, objectives, competencies, activities, interests, etc.
Learning Design	IMS-LD	It models different aspects of a Unit Of Learning of a course. Such aspects are the different roles, activities or resources, the synchronization of different user actions, the activity or resource sequencing depending on conditions, etc.
Learning Resource Meta-data Specification	IMS-MD	It defines a set of meta-information about learning resources
Question & Test Interoperability Specification	IMS-QTI	It describes tests, assessments and questions
Resource List Interoperability	IMS-RLI	It describes how to interchange metadata resources between different systems. In this way, a list of resources can be done
Shareable State Persistence		It extends the SCORM specification in order to access state information
Simple Sequencing Specification	IMS-SS	It describes how to sequence activities within a course
Tools Interoperability Guidelines	IMS-TI	It explains how to integrate LMSs with external tools
Vocabulary Definition Exchange	IMS-VDEX	It defines a grammar for the simple exchange of terms

2.2 ADL: Advanced Distributed Learning Initiative

ADL is a part of the Defense Department of the United States of America. The function of ADL is to document, validate and promote the use of specifications and standards developed by other sources. Their more successful e-learning specification is SCORM [1] (Shareable Content Object Reference Model). SCORM can be seen as a combination of three specifications: IMS MD, IMS CP and IMS SS. It also includes a Run-Time Environment in order to launch contents and track the use in LMSs.

2.3 AICC: Aviation Industry CBT Committee

Aviation Industry CBT Committee [2] was funded in 1988 for the standardization of the aviation industry products. One of their defined specifications is AICC/CMI (Computer Managed Instruction) that can be applied to e-learning. The aspect covered by AICC/CMI is very close to SCORM. At present, SCORM is more used than AICC/CMI, but some of the concepts defined in AICC were used by SCORM.

2.4 IEEE Learning Technology Standards

The IEEE Learning Technology Standards group is devoted to create standards and guidelines for learning technology. There are four Working and Study Groups: Digital Rights Expression Languages, Computer Managed Instruction, Learning Object Metadata, and Competency Data Standards. All the standards are accessible through the IEEE Learning Technology Standards web page [15]. At present, there are only six defined standards:

- **IEEE Standard for Learning Technology-Learning Technology Systems Architecture (LTSA)**. It provides high level information about an e-learning system and its components.

- **IEEE Standard for Learning Technology-Data Model for Content to Learning Management System Communication** . It describes the data interaction between a content object and an LMS Run Time Environment.
- **IEEE Standard for Learning Technology-Extensible Markup Language (XML) Schema Binding for Data Model for Content Object Communication**. This is an XML binding of the data model defined in the previous standard.
- **IEEE Standard for Learning Technology-ECMAScript Application Programming Interface for Content to Runtime Services Communication**. It defines an API (Application Programming Interface) describing services using the ECMAScript language.
- **IEEE Standard for Learning Object Metadata** . It defines a set of metadata for educational resources.
- **IEEE Standard for Learning Technology-Extensible Markup Language (XML) Schema Definition Language Binding for Learning Object Metadata**. This is an XML binding of the data model defined in the previous standard.

2.5 The OKI project

OKI [21] (Open Knowledge Initiative) defines APIs (Application Programming Interfaces) for a set of e-learning services. Each API for a service is called OSID (OKI service Interface Definition). Each OSID defines a set of classes with their attributes and methods (not necessarily in Java although the first API and service implementation was done in this programming language). In this way, different services of an LMS can be implemented following the OSIDs.

While IMS specifications usually define interchanged data models, OKI specifications defines a set of components and the way they can interact, defining behavioural models. Therefore, OKI

and IMS specifications usually define different aspects, but sometimes they can overlap because when defining behavioural programming interfaces, some data model considerations must be taken into account for some services.

3. Motivation and Intuition

3.1 Case Studies

In this subsection we explain some use cases that motivate our work. For explaining such cases, we use a fictitious character that is called Professor Smith. We will go through some aspects of Prof. Smith's life related to her teaching work, and we will see how our work helps Prof. Smith's needs in several contexts.

Prof. Smith teaches at University, and she also teaches some courses at companies. In addition, she has some contacts with editors and collaborates with them for writing books. Prof. Smith must make their courses available within the Moodle platform for her university courses, while she must use different platforms when she teaches at companies, but especially .LRN.

i) Case 1: Design of a Course from scratch for an editorial

Prof. Smith received a call from an editor to write a book about *Java programming*. The course is going to be used by many institutions. Each institution has its own different LMS. Indeed, not even the editorial knows a priori all the different LMS that will be used, but there is an initial estimation of 25 different LMSs. However, the editorial desires to provide support for the Java programming course through each specific LMS. The editorial requests Prof. Smith to be the author of the course including the design of the course within different LMSs.

Prof. Smith accepts the invitation and she think about doing the following things regarding the LMS course:

- Structure the course into five topics in this order: Introduction, classes and objects, conditionals, advanced concepts of object-oriented programming, and Input/Output.
- Add different resources related to each topic: web content, files, SCORM compliant lessons, etc.
- Add two assessments at the end of the course (complaints with IMS QTI).
- Enable one general forum for each topic, and one specific forum for commenting on a personal work. For the general forums all the students can post new threads and responses without limits, while each student can only open a new thread in the specific forum commenting on his/her work. Each specific forum has its own features such as if default notifications are enabled or not, if the posts can be rated and evaluated, etc.
- Enable three different chats with different configuration parameters.
- Enable three different assignments that students must submit, each one with a different weight in the final students' scoring.

The desires of Prof. Smith for doing this task are the following:

1) She does not want to do the configuration for each specific LMS from scratch. Doing the configuration for each specific LMS would imply to repeat a set of tasks as many times as different LMSs. For each LMS, she should write all the different names for the forums, set which forums are allowed with more than one thread, which forums can be rated, which services requires notifications by e-mail to students, the different weights of the assignments, the different topics, the layout of the LMS course, the order and type of the different resources, etc. The list of possible tasks is very high, and it includes all the configuration parameters of all the services, resources, and general information of the course.

2) Instead of repeating a lot of tasks for each specific LMS and having to learn the own features of each specific LMS, Prof. Smith wants to do all the required configuration actions only once, in an independent manner from any LMSs

3) Prof. Smith would like to do all the authoring of the courses off-line from an authoring tool, without having to know of LMS specific editors. She would like to have an authoring tool that allows the configuration of all the possible services and resources of an LMS, being complete with respect to the possible features of each LMS service. In addition, she would like to have a very user-friendly authoring tool, in terms easy to understand by teachers without high technological knowledge.

4) She would like to generate some LMS course package (with all the possible information of an LMS course) that is independent from any LMS, but it can be imported into whichever LMS.

ii) Case 2: Improvement of an existing course about Data Bases

Prof. Smith has a running university course within the Moodle platform that she designed for the last year course edition. This is a course about *Data Bases*. For this year edition, she wants to improve her course. The topics covered last year include the following: Data Bases Fundamentals, Relational Data Bases, SQL language, Oracle, and JDBC. This year she will also include mySQL as a new topic within the course. Within the Moodle instance of the course, she enabled several forums, chats, news, assignments, assessments, etc.

In particular, the changes that she wants to perform in the new edition of the course are the followings:

- She has seen a forum through Internet about the Oracle topic, that has very good posts and it is very instructive regarding the Oracle topic. So, she would like to include it as a new forum within the Moodle platform including the information of the different posts, but removing some inappropriate messages, and adding some new messages of her own creation.

- She would like to substitute current assignments by other different ones, because she desires to change the evaluation exercises. She would like to replace them by a set of data base assignments that she has found in a course within BlackBoard platform, but changing the weight of each assignment to the final grade, and removing the assignment about PostgreSQL as this is a topic not covered in her course.
- She also wants to include different assessments that are in IMS-QTI format that she has found in a teacher repository through Internet.
- She wants to include a SCORM package about the new mySQL topic that she has found from a Web page.

The desires of Prof. Smith for doing this task are the following:

1) She desires an authoring tool that can import all the different educational services that she wants to include in the new edition of the course. This authoring tool should be able to import the forum about Oracle; the new student assignments; the IMS-QTI, and SCORM files.

2) She desires an authoring tool that allows the edition of the different imported resources and services. In this case, she wants to edit the initial forum posts, in order to remove inappropriate messages, and to include some new ones. In a similar way, she wants to edit the assignments, to remove one, and to change the different weights in the final grade.

3) She wants that Moodle can export the old entire LMS course as a package, and an authoring tool that can import such generated package including all the defined LMS services from the old course. She also wants an authoring tool that can edit all the information of the old course. In this way, she can for example remove the old assignments or correct some errors she detected in the names of the chats.

4) She wants an authoring tool that can compose a new course as the union of several pieces, based on the information of the initial old course, and the different services and resources she retrieved. In this way, the initial course evolves to the new desired edition. The authoring tool should create a new course package that describe the entire course including all its services, and

the course might be imported and interpreted in whichever LMS, in this case in Moodle, that is Prof. Smith's university platform.

iii) Case 3: Reuse of a course from the university to a company

Prof. Smith wants to reuse a course about *Security* that she teaches at university. She wants to use the same course for teaching it in a company. She should adapt the course from the Moodle platform to .LRN. The course in Moodle looks like in Fig. 2:

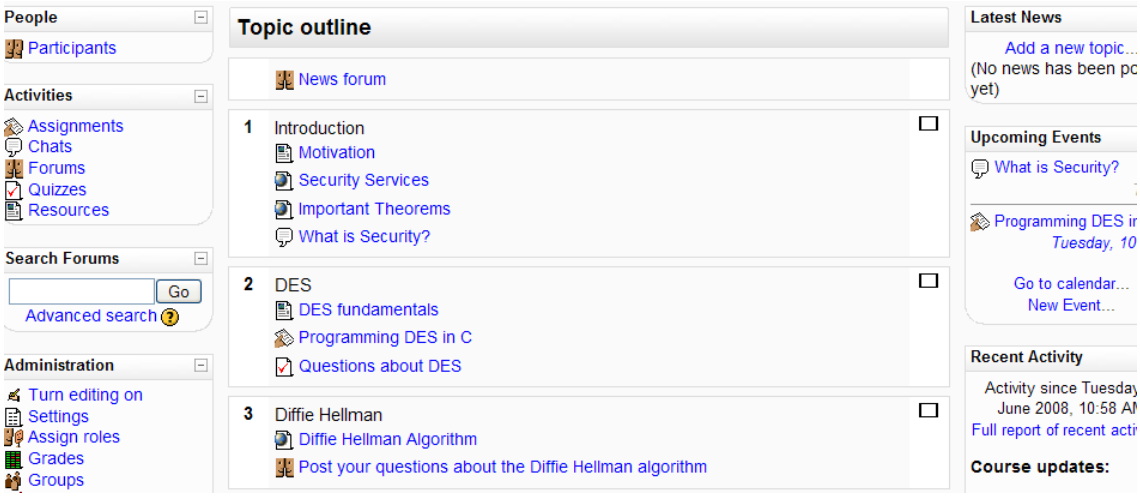

Figure 2: Course in Moodle about "Security"

Prof. Smith would like to press an "Export course" button within Moodle, download a course package file, and import this course package within .LRN with an "Import course" button. She would like to have within .LRN the closest course features, structure, etc. that she had within Moodle. But these "import" and "export" functionalities of entire courses within LMSs are not available at this moment. In case they would be available, then the course of Fig. 3 would look like in Fig. 3:

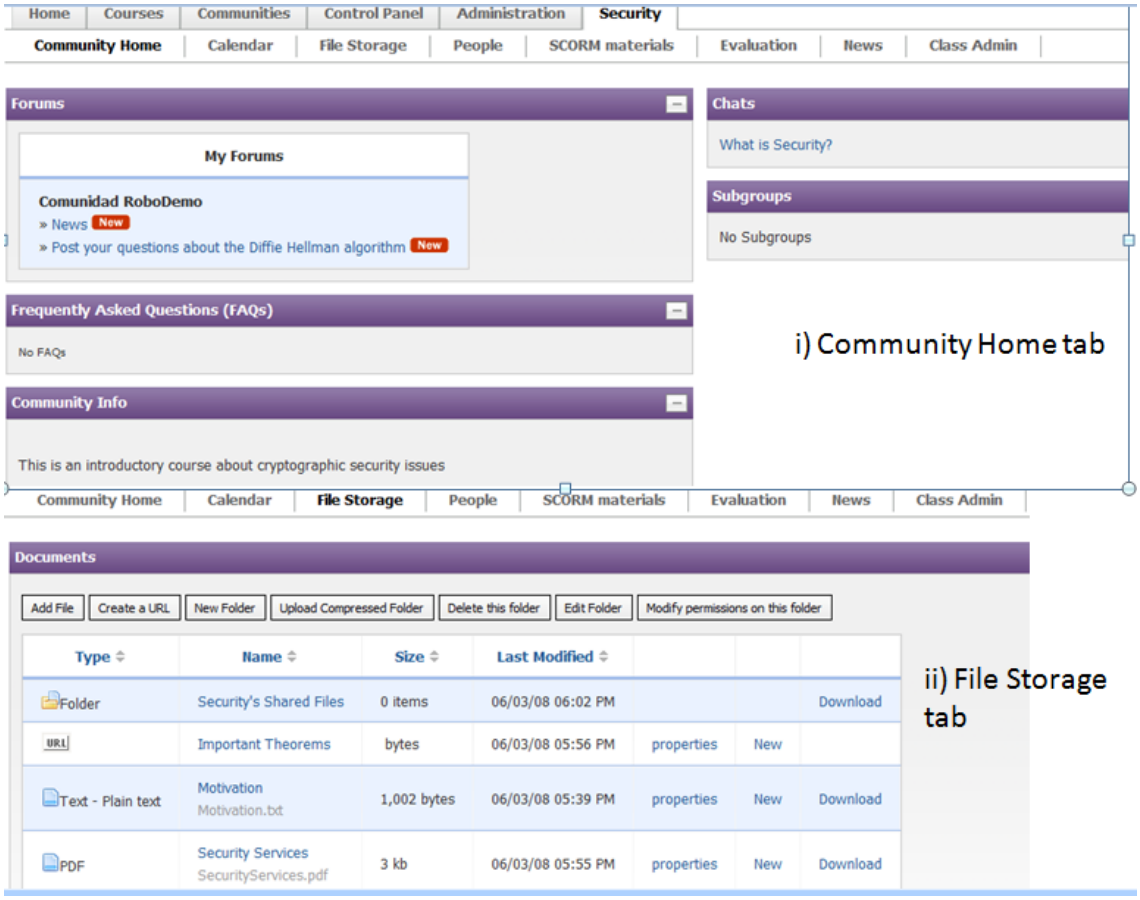

Figure 3: Course in .LRN about "Security"

As each LMS has its own features, then the "look and feel" of the course will be different depending on the platform that contains the course, and transformation criterions should be set for each aspect. It is important to note that this is only one mapping possibility of the course from Moodle into .LRN, but there are other different possible mappings. For example the "Evaluation" tag might be removed, and instead the "Evaluation" module might be within the "Community Home" tab. In this case, we can see several examples of specific mappings:

1) The course is organized in topics within Moodle. However, the course is organized by functionalities within .LRN (Fig. 3i shows the "Community Home" tab that is grouped by forums, chats, etc. but not into topics).

2) The reading resources are distributed in the main page within Moodle, but in the "My Files" and SCORM tabs within .LRN. We can see in Fig. 3ii how the URL (Important Theorems), a

plain text (Motivation), and a file (Security Services) that are resources within Moodle, are located within the "File Storage" in .LRN, while the "DES fundamentals" SCORM resource is in the "SCORM materials" tab within .LRN.

3) In the same way, the Calendar is located at a different position. Not in the main page, but in a specific tab. The same can be said to the news.

In addition, if we focus on the forums configuration, we can note that there are forum functionalities that are quite similar in Moodle and .LRN, but sometimes with different names (for example the "Standard Forum for General Use" of Moodle is equivalent to the "flat" and "open" forum of .LRN, being able to create new threads; or the "subscribe notifications" in .LRN is similar to the "Yes, Initially" option of the "Force everyone to be subscribed" option of Moodle). But there are other functionalities that have not a direct mapping because they are not exactly the same. Therefore, in such cases, a translation criterion is required. For example, the "moderated" option of .LRN for forums, set that a post is only available when a moderator approves it. Such option is not at present in Moodle.

Similar arguments can be said for other functionalities different from forums, such as the assignments, or the chats. And in general, the same can be said from any LMS functionalities. Therefore, some common vocabulary should be set between functionalities of different LMSs, but the final transformation (including presentation) of each specific LMS can be different according to the specific LMS features.

iv) Case 4: Collaboration among different teachers for the design of a course

Prof. Smith needs to collaborate in the creation of a new course for a company. The course will be delivered in the BlackBoard/WebCT platform. She needs to collaborate with other four teachers, but none of them knows the BlackBoard/WebCT platform because each one only knows the specific platform that is used in his/her university.

Instead of learning a new platform, they prefer to use a unified authoring tool in which they can create all the necessary for a course, and finally import the packaged course within the BlackBoard/WebCT platform.

In addition, they are planning to split the work, so that each one will be in charge of a set of services. They want to interchange their works and review the other materials. They want to have an approach that makes easier all this collaboration among different teachers.

v) Case 5: Adding course adaptation and personalization depending on certain services parameters

Prof. Smith wants to add an IMS-LD package within the data base course. In this IMS-LD package she wants to add some adaptation and personalization conditions related to different LMS services. In particular she wants to add:

- In case that three or more news are added related to some of the topics, then an automatic forum should be created with some specific configuration parameters such as being threaded or not, being moderated or not, number of threads per user, etc.
- In case a student performance in an assignment is less than 5 points out of 10, then he/she should repeat the reading of some resource.

3.2 Requirements for the ideal approach

For the ideal vision that fulfils Prof. Smith's desires, it is needed the following:

1) Data specifications for each one of the LMS services. Some specifications exist but new ones should be created for the LMS services with a lack of specification.

2) Integration of the different LMS service specifications.

3) Specification that models the high level of an LMS course, including general information, the layout, or the enumeration of services.

4) An authoring tool that is easy to use by teachers without high technological knowledge, and that includes the creation of all the LMS services, LMS resources, and the high level of an LMS course. All of this according to defined specifications, being able to import XML instances and generating XML code for each one of the aspects.

5) LMSs that implement players for each service, being able to import and export the different services, and also the high level information of an LMS course.

In this work, we contribute to points 1, 3 and 5, while points 2 and 4 are out of scope of this research. Regarding points 1 and 4 we determine important services of LMSs for which there are not defined specifications at present or they have a lack. For such aspects we define new specifications including a data model and an XML binding, and we have also created a new authoring tool that is conformant with such new specifications. Regarding point 3, we present an extension of the IMS-CP specification that models the high level information of an LMS course.

3.3 Importance of the new specifications

From the previous use cases of Prof. Smith, we can resume a set of reasons that justify the importance of the new defined specifications about the commented LMS educational services:

1) **To enable interoperability for each aspect between different LMSs.** Different materials related to each aspect can be interchanged between different LMS systems, without any extra effort.

2) **To enable reusability between different created materials.** An author can create materials related to a specific aspect with independence of the LMS where it is going to be executed, and he/she can reuse existing materials in order to compose new ones.
3) **A complete course can be moved from one LMS to another.** If we suppose all LMS functionalities instances can be expressed in some common specifications that all LMS are conformant, then an active course in an LMS will be able to be imported without any effort into another LMS, interchanging the XML information instances of the specifications.
4) **Adding more power to existing e-learning specifications.** Some existing specifications need to refer to activities or services in some moments. Some activities or services can be LMS functionalities instances. If we enable new specifications related to LMS functionalities, then these functionalities will be able to be used as specific activities or services into the existing specifications. Moreover, with the integration of specifications, existing specifications can use parameters of the data model of the new specifications. Specific examples of this can be found within case 5 of Prof. Smith.

4 Relationships between LMS services and present specifications

On one hand, we need to do a list with the most important LMS functionalities or services. From [20], we have extracted a list of the most important LMS functionalities. In addition, we have incorporated some a few extra LMS functionalities that were not used in that experience. Table 2 shows the different LMS functionalities. On the other hand, we have studied carefully all the e-learning specifications related to data descriptions and also some non specific e-learning specifications found in [17] and [26].

Next, we have established the relationships between the LMS services and the present specifications. We illustrate these relationships in Table 2. We have divided the different LMS

services into five groups depending on its relationship to present specifications. We devote a subsection for each one of the five groups, where we explain in more detail Table 2.

Table 2: Relationship between LMS functionalities and current specifications

Functionality	**Has sense a new specification for it within an LMS course?**	**Are there present data specifications related to this aspect for Learning Management Systems?**	**Level of Coverage of the present specifications**
Forum	YES	NO	None
File download	YES	YES. Non e-learning specifications	Full
Calendar	YES	YES. Non e-learning specifications like iCalendar, hCal, vCal	High
News about the course	YES	YES. Non e-learning specifications like XMLNews	High
E-mail	YES	YES. Non e-learning specifications like MessageML	Full
FAQ	YES	NO	None
Lists of students (for all courses in which is enrolled)	YES	YES. IMS Enterprise	Full
On-line polls	YES	YES. IMS QTI	Full
Additional information about the teaching staff (other courses, location, tutoring, research…)	YES	YES. IMS Enterprise	Medium
Submission management system	YES	NO	None
Assessments	YES	YES. IMS QTI, IMS ePortfolio, IMS LIP, IEEE PAPI, etc.	Full
Help about using the tool	NO	--	--
Visualization of academicals record	YES	YES. IMS Enterprise	Full
Visualization of Structured Content	YES	YES. IMS CP, SCORM, AICC	Full
Sequence of Content Personalized to Students	YES	YES. IMS LD, SCORM, IMS SS, IMS LIP, IEEE PAPI, etc.	High
Search within the portal	NO	YES. IMS MD, IEEE LOM	High
Weblog	YES	YES. Non e-learning specifications	Full
Receiving relevant information about the course by SMS	YES	NO	None
Videoconference	YES	NO	None
Contents in audio	NO	--	--
On-line interactive classes: real-time slides, e-blackboard, sound, etc.	NO	--	--
Creating your own course web page	NO	--	--
Power Point via web	NO	--	--
Competition among students	YES	NO	None
Job offers service	NO	--	--
Application access log	NO	--	--
Knowledge of your own learning evolution and status at any time	YES	YES. IMS ePortfolio, IMS LIP, IMS RDCEO	Full
Knowledge of your own level for each topic	YES	YES. IMS ePortfolio, IMS LIP, IMS RDCEO	Full
Photo album	YES	YES. Non e-learning specifications (e.g. XML Photo Album)	Full
On-line tutoring	YES	NO	None
Games	YES	NO	None
Incorporate web search engines (Google, Altavista, etc.)	NO	--	--
Chat	YES	NO	None
Work in group	YES	Yes. IMS Enterprise	Full
Subgroups	YES	NO	None

4.1 LMS services for which it has not sense to define a new data specification within an LMS course

All the services that have a "NO" in the second column of Table 2, are the ones for which it has not sense to define new data specifications within an LMS course. We have found one of these two reasons for it: i) They are not specific services for each course but general to an LMS. Therefore, each course has not its own instances of these services, but there is only one global instance for all the LMS. These services may have configuration parameters and data to model, but we are not interested in them in this research because we are oriented to LMS courses. ii) The technology provides the LMS functionality and no extra information is needed to characterize the functionality.

Among the services that are not specific for each course but general to LMSs are the following: help about the tool (the help is the same for all the LMS courses. It could use the IMS-CP specification, or being a set of files in a file storage), application access log, job office service, and search within the portal (in this case IMS MD or IEEE LOM might be used for defining the vocabulary set of possible searching terms, but some new specification would be required to say which are the fields of the module to do the searching).

Among the services that do not need extra data to be modelled are: contents in audio, on-line interactive classes, power point via web, and your own course web page. The latter can be seen as a course module possibility that allows a student to put a reference to his/her web page about the course. But this possibility has not much different to configure: only the link to the web page. When the students put their links, then it can be seen as a set of resources (web pages) to the course. The three former features (audio, interactive classes, power point) can be seen as resources, and they can be within a file storage, or being launched by some tool. But they do not need much more to be modelled, so there is no need of new data specifications for them.

4.2 LMS services that are covered completely by the union of several specifications

It comprises services that need several existing specifications to be covered. From Table 2 we can see the followings:

- *Sequence of content personalized to students*. The sequence of contents can be done with SCORM, IMS-LD or IMS-SS. Nevertheless, none of these specifications cover completely all the adaptation possibilities (for example there is a problem for modelling the repetition of the same activity for a user in IMS-LD). Depending on the case, one is more suitable than others, and sometimes they need extensions to cover more possibilities, like in [23] for SCORM. In addition, whichever of these specifications can use others to take into account adaptation parameters (e.g. IMS-LIP, or IEEE-PAPI for adaptation depending on parameters of the user modelling).

- *Knowledge of your own learning evolution and status at any time, and Knowledge of your own level for each topic*. These two LMS services can be modelled with the help of several specifications. The main one is IMS ePortfolio that allows modelling the learning progress of a student. Indeed, IMS ePortfolio has a lot of information, so part of this information is used for one LMS module and some other part for the other module. In addition, IMS-LIP and IMS-RDCEO are used to refer to students' parameters and competencies.

- *Assessment*. This is modelled mainly by IMS-QTI. In addition, IMS-LIP or IEEE-PAPI can be used to adapt the assessment to certain user parameters.

4.3 LMS services that are covered completely by only one specification

This group comprises LMS services that can be described with only one specification. Two cases are possible:

1) *LMS services described by e-learning specifications.*

- “On line polls”. It can be seen as a specific case of assessments in which the results are not evaluated, as it is a survey. Therefore, it can be modelled with a subset of IMS-QTI.
- “Visualization of academicals records”. This can be done filling in the properly IMS Enterprise parameters, within the “membership data objects” of the specification.
- “List of Students”. It can be described completely with the IMS Enterprise Specification, as it allows the description of the members of a group, providing several information about each person. The group would be the class in this case.
- “Work in Group”. It can be also described by the IMS Enterprise. An example is when subgroups of students should be done for submitting assignments.

2) *LMS services described by non e-learning specifications*. There are some LMS course functionalities that can be described completely with non e-learning specifications. This is the case of the e-mail messages (with MessageML), or the Photo Album (e.g. with the XML Photo Album). In the case of the Photo Album, it has sense to have a specification because an LMS course can have several Photo album instances (e.g. one for each topic of the course).

4.4 LMS services that are covered partially by one or more specifications

Among this group, we can see the calendar, news, and teacher’s information. For modelling the calendar, there are the v-cal, i-cal or h-cal specifications. For the modelling of news we have the XMLNews specification. Although a lot of the required functionalities for the calendar and the news are covered with the existing non e-learning specifications, however there are some specific aspects that can be included within the calendar and news by the fact of being within an LMS (e.g. like event notifications). In addition, only some of the functionalities of existing non e-learning standards are used with LMS services. For these reasons, we have defined new specifications for the news and the calendar. Details of the differences with existing specifications can be seen in Section 6 where our calendar and news specifications are explained briefly.

Finally, for the modelling of the teachers' information, we could use the IMS Enterprise specification as a starting point, because this specification includes some features for describing persons (e.g. information like name, surname, address, etc. is valid for a teacher). Nevertheless, there is a lot of information that is required for a teacher that is not included with IMS Enterprise (e.g. tutoring hours, Master Thesis offers, research articles, etc.). For this reason, we have defined a new specification for this aspect.

4.5 LMS services that are not covered by present specifications

There are several LMS services for which there are not concrete specifications. For the general vision of complete LMS interoperability, new specifications for those LMS services are required. We have defined new specifications for some of these LMS services. Those services are forums, faqs, submission management system, chats, and subgroups. A special note is required for subgroups in this moment. As we introduced before, the IMS Enterprise defines a group as a collection of persons and relationships between them, but this is different to define the LMS configuration within a subgroup (e.g. for morning and afternoon subgroups of a course). For this reason the Subgroup aspect in this context (that is the one for which we have developed a new specification) has not a relationship with respect to the IMS Enterprise.

There are other LMS functionalities of the list for which we have not developed any new specifications yet, but they have not any related specifications at present. These aspects are: Video Conference, Competition among students, On-line Tutoring, and Games. The reason why we have excluded these LMS functionalities at this moment is that there is not a strong commonality of their configuration in different LMSs. This is because it is very difficult to have the same configuration parameters because there are a lot of possibilities of games, or competitions. But concrete specifications could be done for each specific game or competition.

Finally, regarding the “receiving SMS” service, instead of doing a new specification for it, this functionality could be added to several of the existing specifications, as we have done with the e-mail notifications in our new specifications. Therefore, for example we could add this functionality for a forum, so that each post message might be received by SMS. It could be done by adding some new tags to the XML description of the forum, in a similar way as it is done for the e-mail notifications.

5. Use of IMS-CP for enabling reuse of a complete LMS course

In this Section we show a method to use IMS-CP to describe an entire LMS course. In this way, an LMS could import an IMS-CP and deploy an entire LMS course. We provide general indications about how to describe a generic LMS course with IMS-CP. We illustrate it with a course example, which is the same course that we used within Moodle (Fig. 2) and .LRN (Fig. 3). Fig. 4 represents the visualization of an IMS-CP with the RELOAD editor, describing the course of Fig. 2 and Fig. 3.

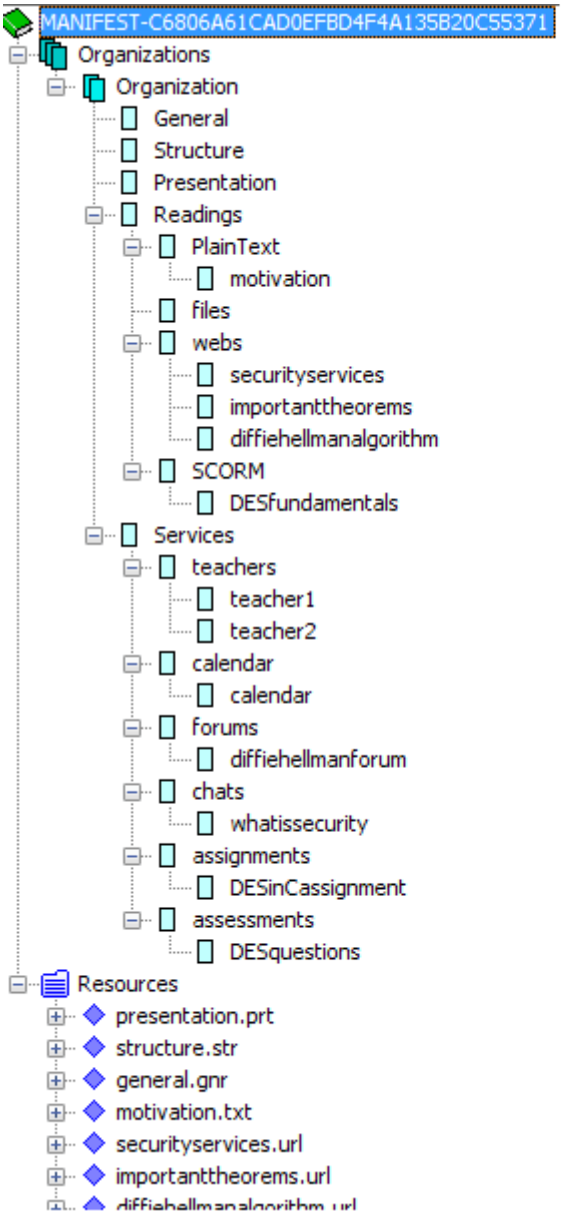

Figure 4: IMS-CP for describing an LMS course using the RELOAD editor

Our proposal of using IMS-CP for the description of a complete LMS course forces to use the following items within the organization of an IMS-CP package:

- *General*. It represents general information of the course such as name, description, or enrolment policy (open, closed, on demand, etc.). It is related to an XML resource with the *gnr* extension.

- *Presentation*. It represents presentation aspects of a course such as the colour, size of the letter, etc. It is related to an XML resource with the *prt* extension.
- *Structure*. It represents the structure of the course, such as how many tabs the course has, the services and resources that are located within each tab, and the order o the elements. It is related to an XML resource with the *str* extension. As each LMS only allows a set of structure possibilities, then each specific LMS would have its own structure file (for example .LRN allows the presentation grouped by tabs, but Moodle does not; or Moodle allows the presentation by topics, but .LRN does not). This would be a difficulty for the complete LMS interoperability. In the future, LMSs may evolve to support a standard set of structure possibilities, so that reusability in this aspect may be possible. But up to this date, there are several ways to overcome this problem: 1) Several files within the structure organization field can be written, one for each LMS. 2) Each LMS can set a transformation criterion for a standard structure file, for example not taking into account features that it does not implement.
- *Readings*. This is the list of activities within the LMS course that require a reading by the students but not an active or social interaction. There is a branch in the organization for each type of resource. For example, for the course described in Fig. 4, we have branches for plain text, files, web links, and SCORM resources. Under each branch we have the correspondent specific resources.
- *Services*. This branch of the organization contains all the services that are related to a specific course. The different types of services are bound to a specific sub-branch. Under each sub-branch there are all the specific service instances. For the example course we have teachers (with two instances as there are two different teachers in the course), calendar (only one for the course), forums (with the forum about Diffie Hellman), etc.

Regarding the resources section of the IMS-CP, we will have all the different files that are referred in the organization. This includes the files for the presentation, the structure and the general information, but also all the reading and service files. Each different reading resource

file will have a different extension depending on the type. In the same way, each different service resource file will have a different extension depending on the type (e.g. *frm* for forums, *qti* for assessments, or *assgn* for assignments). Therefore, the resource type field would be not only *web-content* and *files*, but also the different types of readings and services.

In this way, with the organization and resource fields, LMS courses can be defined completely and packaged within an IMS-CP.

6. Data models of the new e-services specifications and XML binding

The planning for doing the new e-services specifications was divided in the following steps:

1) Study of the commented functionalities over different LMS platforms (specially .LRN, Moodle and WebCT/BlackBoard) and other non LMS systems.
2) Obtain a list with the parameters and data information that model each LMS functionality. This compounds the data model for that functionality. This model is based on the previous study of the different LMS platforms, non LMS systems and our own ideas.
3) Build an XML binding for the previous data models. An XML Schema is created for each LMS functionality. Each functionality instance can be expressed in XML format.

Next, the new models are presented. The Forums functionality is described in detail as a prototype example, while the rest are explained briefly.

6.1 Forums

The specification let define different possible configuration parameters for a forum in an LMS. The aspects that compound the data model are the following:

- Forum type. It can be news forums (only teachers can post messages to students), teacher forums (only for communication between teachers) or general forum (teachers and students can participate).
- Title. It is the forum name.
- Author. It is the person who created the forum.
- Moderator. It is the person that can do moderator actions such as removing non convenient messages.
- Creation Date. It is the date of the forum creation.
- Presentation type. It can be nested (different threads of messages are allowed) or flat (all the messages are at the same level, so there is no message hierarchy).
- Publication policy. It can be open (messages are public instantly), moderated (the moderator accepts or rejects the publication of messages) or closed (new messages cannot be published).
- Permission for creating new discussions. It can be free (users can post new messages without restrictions) or controlled (users can only post messages as response to messages created by a moderator).
- Reediting permission. It sets if a user can modify his/her own messages once messages have been published.
- Maximum number of messages per user.
- Evaluation type. It sets if messages can be evaluated by other users, which roles have permissions to evaluate, the maximum number of evaluations per user, who can visualize the evaluation or the evaluation deadline.
- Changes notification. The forum can have a changes notification procedure. For example, when a user posts a new message, all forum users can receive a notification message in their e-mail. The changes notification can be set to mandatory, by default or voluntary.

- Discussions information. In general, a forum is composed of a set of discussions. Each discussion is composed of a set of published messages. In a general case, each forum message is characterized by:
    - Topic. This is the subject of the message.
    - Author. This is the person who writes the message.
    - Text. This is the message body.
    - Attached File. This is the name of a file that is attached to that message
    - Evaluation. This represents the evaluation mark of that message by the authorized users.
    - Number of Evaluations. This is the number of users that evaluated the message until now.

An XML binding for this data model has been created, so any forum configuration instance can be written in XML. Fig. 5 shows a graph representing the forums XML Schema. With this XML Schema, any combination of the previous data model can be written in XML.

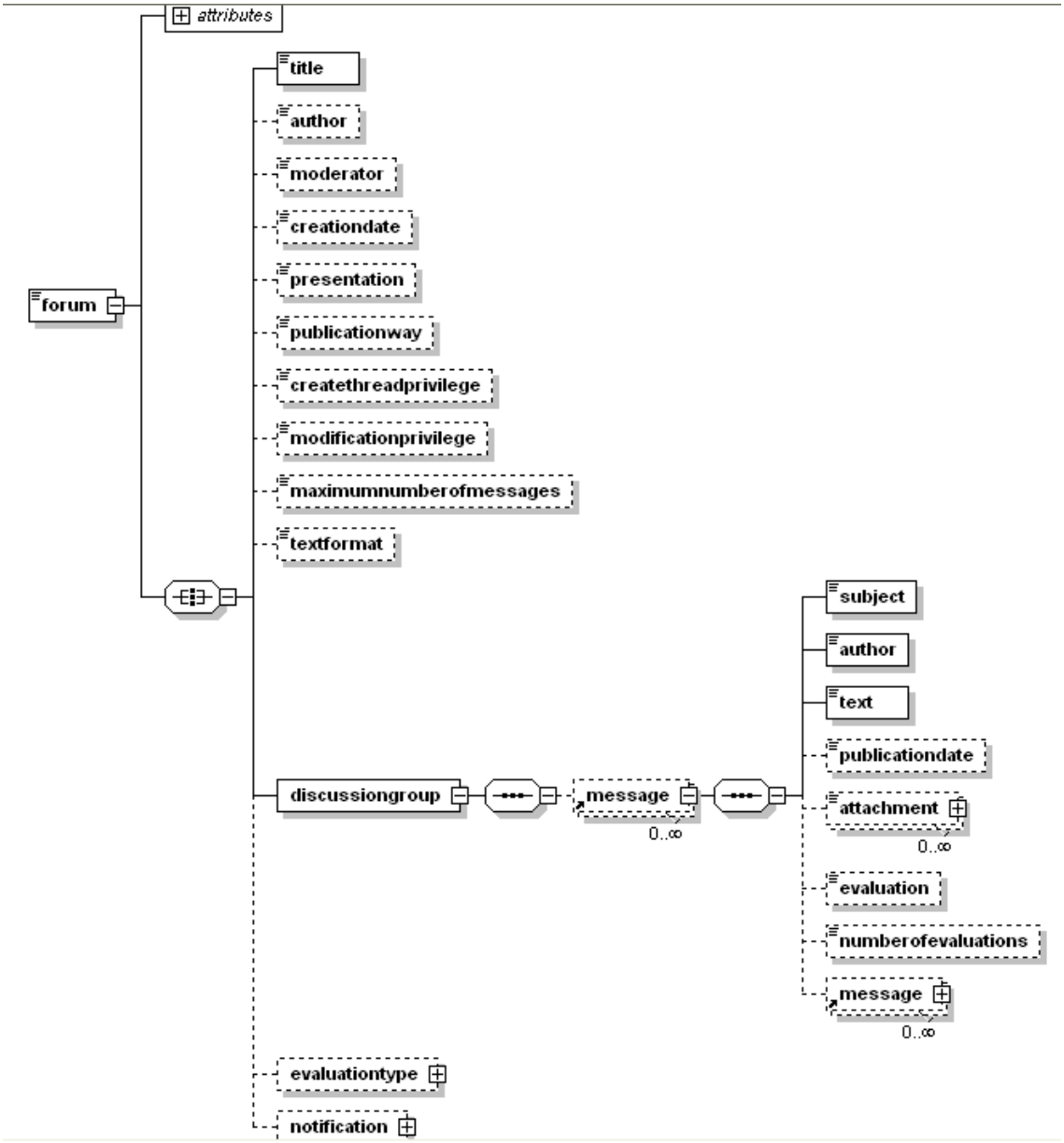

Figure 5: XML Schema graph for description of forums

6.2 Calendar

Our model includes general information that is present in hCal or i-Calendar microformats like title, description, start date and end date. Nevertheless, our model does not include some information that is not so important in LMSs like the location of the event (because it is distance learning). But, our model includes other relevant information that is not present in existing Calendar standards but it is LMS specific:

- Event Profile. It can be global (all the platform users can view it), course (only students that belong to a course can view it), group (only students that belong to a specific group of a course can view it) and personal (only the own person can view it).
- Frequency. This is the repetition frequency of the appointment. It could be diary, weekly, monthly or yearly.
- Changes Notification. It is the same concept as in forums and it is the same for the rest of presented models.

6.3 News

The News specification that we have created has a set of elements that are covered by the existing XMLNews specification such as: title, author, body, link, attachment, publishing date or expiring date. Furthermore, there exist a lot of features of XMLNews that are not usually used in LMSs. On the other hand, there is a set of features of our News specification that are not covered by XMLNews, such as:

- News type. It sets the scope of the news. It can be global, subject or group.
- State. It can be published or archived.
- Evaluation. It is the same concept as in forums and it is the same for the rest of presented models.
- Changes Notification.

6.4 FAQs

The FAQ data model contains the following elements:

- General information about the FAQ block: Title, author, creation date, description, presentation (if questions and answers can be seen in the same page or not).

- Evaluation.
- Changes Notification.
- Group of questions. A FAQ block is composed by a set of questions and answers. Each one has a question, an answer, an evaluation, and a number of evaluations.

6.5 Teaching Staff Information

The Teaching Staff Information has the following data information: name, surname, role (it could be Professor, Teaching Assistant, Tutor, etc.), office, telephone, fax, e-mail, tutorial hours, list of subjects (for each subject, the model describes the information of that subject: name, description, credits, timetable, link, degree, year, course and semester), research areas (each research area has its name, description and link), publications (each teacher paper has its correspondent data), Master Thesis offers, observations, evaluation and changes notification.

6.6 Assignments

This functionality is devoted to model the assignments a teacher assign to a student. In the model, the assignments can be divided into groups. Each group has a name, description, a weight in the final score, change notifications and a list with all the correspondent specific assignments.

A specific assignment has title, description, link, attachment file, estimated time, deadline, minimum and maximum of students per group, submission procedure, maximum number of submissions, if it influences the final mark, etc.

Finally, there is a modelling of the response a student submits. It contains the student name, the assignment identification (to know which assignment the student submitted), state (only

submitted or evaluated by the teacher), submission date, response file, response text, teacher mark, feedback comments by the teacher, teacher name that did the evaluation, etc.

6.7 Chats

The data information related to a chat in our specification is the following: title, description, frequency (this is the frequency the chat session will be repeated. It can be: always open, only once time, diary or weekly), date and time of the chat session, archive, archive time, permissions (it can be enabled to save the chat session and the time the conversation will be available as well as the permissions to access it), communication type (it can be teacher to students or any communication), evaluation, changes notifications, and conversation (it describes all the interchanged messages. For each message there are: author, text, attachment, evaluation and number of evaluations).

6.8 Subgroups

A course can be composed of various subgroups. The information to model a subgroup is the following: name, description, subscription (it can be open, closed or need approval), visibility (it can be visible in case other groups can see the information of this group or isolated in case other groups has no access to the information), notifications changes and list of functionalities, resources and services that are available for such group.

7. Implementation of the Authoring tool

One of the key aspects for the successful adoption of e-learning specifications is the existence of properly authoring tools that create instances according to the specifications and LMS players that are capable of uploading, interpreting and running these created instances.

The absence of properly authoring tools that cover certain defined e-learning specifications is one of the main problems that make a specification not to be successful. First, we define what we consider an ideal authoring tool. We believe the ideal e-learning authoring tool has the following features:

- **It is very easy to use for teachers and course designers.** They do not need strong technical knowledge in order to use it. The authoring tool can be used easily by teachers and course designers from all the different knowledge areas, even if they have not previous e-learning experience.
- **It integrates the most number of specifications together in the same tool (ideally all)**. In this way, the task of generating all the material of a course is reduced. This is due to several factors: first, it is not necessary to open several tools for the different specifications and integrate all the different generated files taking into account the different relationships between them; furthermore, the teacher or course developer has to learn only one tool environment instead of several.
- **It can generate all the semantic possibilities of the specification**, creating the different XML instances that can be exported to whichever system.
- **It can import XML files that are according to the specification** and that can be generated from other authoring tools.
- **It is open source.**

At present, there are a lot of e-learning specifications without any authoring tool. Moreover, there are other specifications that have authoring tools but lack some of the points that describe an ideal tool (specially, the first and second points).

We believe the absence of proper authoring tools is one of the key aspects that make a specification not to progress. In this work, we have developed a properly authoring tool that is conformant with the previous ideal authoring tool points:

- It is easy to use. Teachers and course developers should only fill in a set of forms for each specification. These forms have easy terms that teachers can understand without previous e-learning experience.
- It integrates the new eight defined specifications. The authoring tool incorporates the creation of instances of the following new specifications: Forums, Calendar, News, FAQs, Teaching Staff Information, Assignments (Submission Management System), Chats and Subgroups
- It allows all the expressivity of the previous specifications allowing all the combinations of possibilities.
- It can import XML instances according to the eight defined specifications.

Fig. 6 shows the initial page of the authoring tool. In the middle, there is a list of elements including the eight related to the new specifications. The user can select one of them depending on the XML e-learning specification instance he/she wants to create. In the upper side, we can access to the 'Components Repository' where all the previous saved XML instances of whichever specification are available and they can be loaded and re-edited again. In the bottom side, there is the 'Import Specification' option that let import an XML instance of whichever specification. This XML instance could be made from other authoring tools. Once the instance is imported, it can be re-edited, doing the properly modifications and a new XML instance can be generated according to the changes.

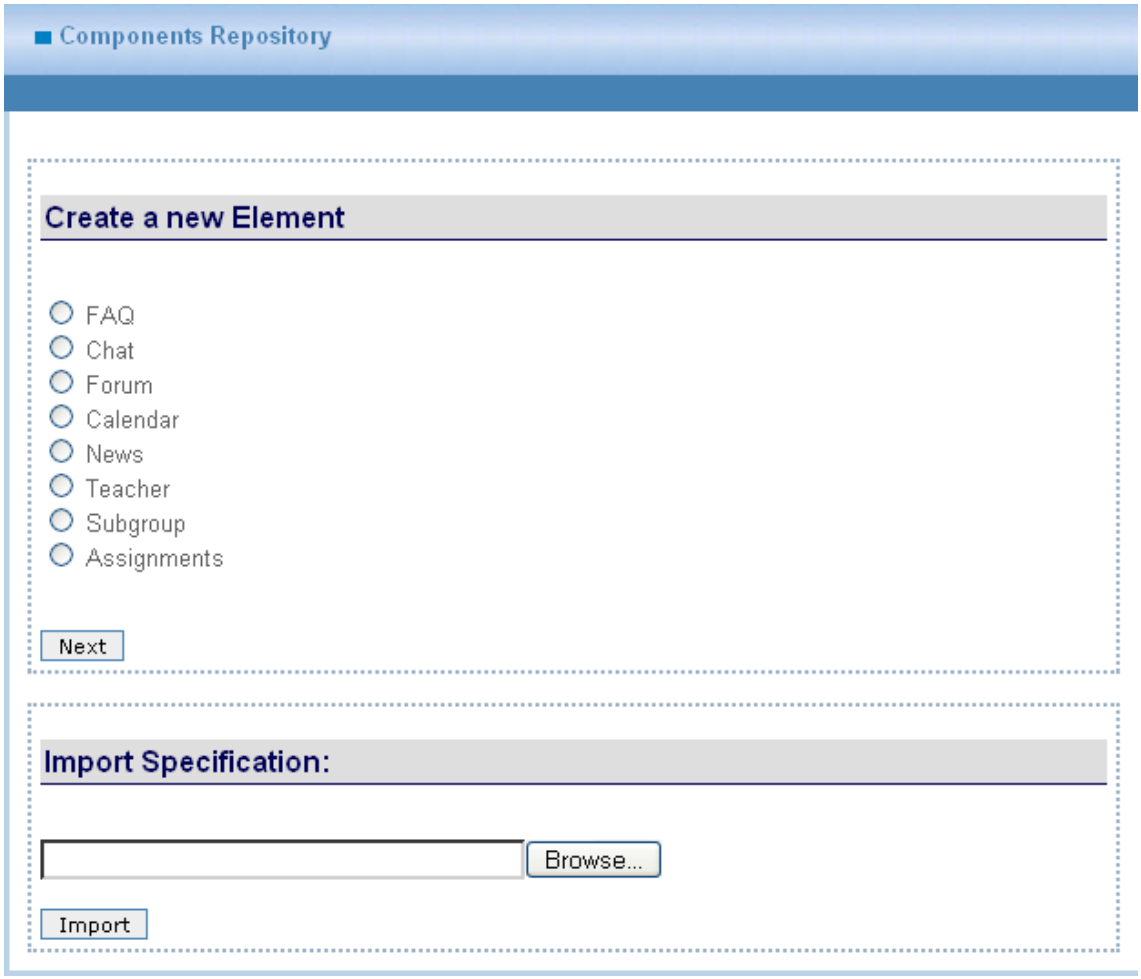

Figure 6: Initial page of the Authoring Tool

Each one of the presented functionalities has a set of forms through the authoring tool to describe it according to the specification. As an example, Fig. 7 shows the first form for a FAQ functionality. The different parameters that model a FAQ block are set through a set of forms for which the presented in Fig. 7 is the first one.

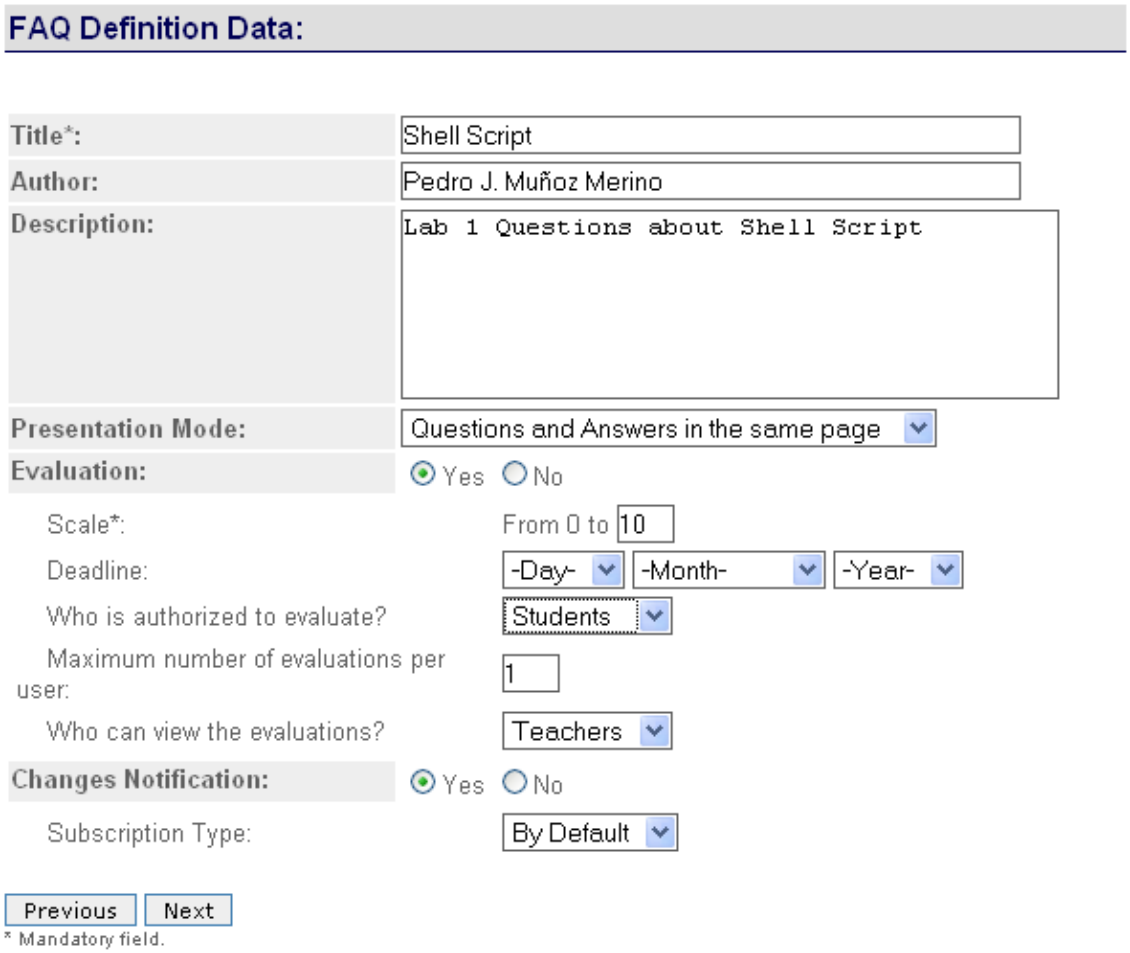

Figure 7: First form for defining a FAQ block

Finally, the authoring tool let view, store or download the generated XML instance of each specification. As an example, Fig. 8 shows a part of the XML view of the created FAQ block.

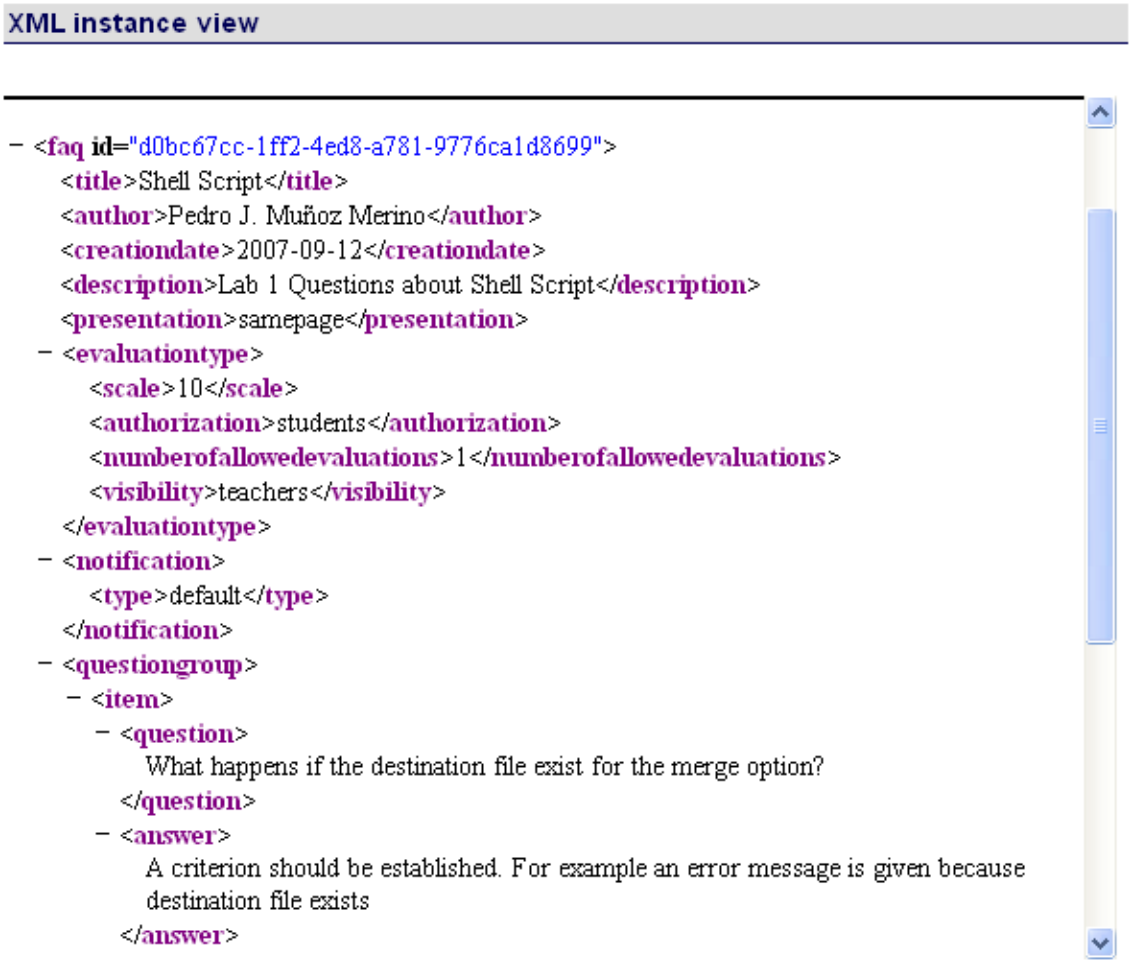

Figure 8: A part of an XML instance example according to the FAQ specification

Finally, we give some a few guidelines about the implementation technical details. The tool has been implemented as a Web application using the J2EE technology and the Apache Tomcat Web Server. It is according to the Model View Controller (MVC) design pattern using the Struts API. In addition, it uses the JDOM API. This is a Java library that let developers do processing tasks with XML files including reading and writing. This is essential because e-learning specifications manipulate XML files. The graphic interface of the application is composed by a set of JSP pages. The design of these pages was done taking into account the W3C standards XHTML 1.0 and CSS. The design was done to reach the objective of easiness and pleasant for most of the users. The different pages are divided in four parts: a header (with

the name of the application and other links, like the link to the main page), a left margin (with internal links), a center part (with the different data forms) and a foot.

8. Related Work

In this section, we relate our work with respect to other existing works. For each of the four contributions of Subsection 1.1 (*Contribution of our work*), we have a correspondent subsection in this Section 8, that shows how each of our four contributions relates with respect to other existing works and why our approach is new and innovative.

8.1 General vision for complete LMS interoperability

Interoperability for different e-learning aspects is not a new issue. Many authors have already exposed the importance of interoperability in an e-learning context, such as in [7] or [8]. However, this interoperability is usually referred in previous works for specific learning objects, contents, or assessments, but not as interoperability for a whole LMS course, as it is the case of our vision. This complete LMS interoperability approach requires, among other things, a high level description of the course, including general configuration parameters, the layout presentation structure, and the organization of the different services that compound it. For this high level description of a course, there are not any specifications at present. Therefore, LMSs have not a complete course import/export button to interoperate among different LMS platforms.

We propose a way to use IMS-CP to enable such high level description of a course. This is a different way of using IMS-CP with respect to the regular one. Several works have already used IMS-CP for assembling different contents of a course (e.g. [18]) which is its regular use. In this regular use of IMS-CP, a package compliant with the specification is imported into an LMS as a resource within a course. Most of LMSs allow to import these IMS-CP as course resources (therefore a course can have several IMS-CP resources), but none of the LMSs can import an IMS-CP package that describes an entire LMS course. We want to enable this possibility.

As far as we know, there are not other works that use IMS-CP for describing a complete LMS course. But there are other works that use IMS-CP for purposes different to the commented regular one. For example, IMS ePortfolio specification does it for describing what a person or organization has learned along the time. In a similar way, reference [28] does not provide a way to describe a complete LMS course as our proposal does (it does not describe all the high level LMS course information), but it allows the integration of SCORM with services which is a commonality with our approach. Reference [28] models it as a call to a web service using the concept of Sharable Content Services (SCSs), but in our case it is a reference in the IMS-CP organization to a file with a different extension depending on the type of service. In that approach [28] content is on demand, instead of being imported by LMSs, which is another difference with respect to our work.

8.2 Lacks of present specifications regarding LMS services

OKI [21] and IMS Abstract Framework [11] define an architecture based on layers (Fig. 1 shows the OKI architecture based on layers). Each layer has a set of possible elements. Any of these elements of such layers can be subject to standardization. Our work has focused on the analysis of the services of the two upper layers of the OKI architecture (or the correspondent three upper layers of IMS Abstract Framework). We do not focus on the other layers of the architectures because we are interested on the final educational services and applications of LMSs.

In addition, our work has analyzed the relationship between LMS services and current data specifications, but not with respect to behavioural specifications. The organization that has done a deeper effort in analyzing behavioural specifications for educational services is OKI with the definitions of OSIDs. In this way, OKI defines the behaviour of a service, but not the data. For example, there is an OSID of assessments that describes the behaviour of an assessment LMS component. But there is another specification for describing the data

of an assessment that is IMS-QTI. In our work, our new specifications will describe data of services but not behaviours. Therefore, our specifications are in the same line as IMS-QTI, but not as OSIDs.

IMS Abstract Framework lists a set of services. This list includes all the services for which IMS has already released a specification (like meta-data, assessment, sequencing, competency management, etc.), and some others that have not an IMS specification (like the calendar). Nevertheless, this list is not exhaustive (for example it does not include services like forums, chats, faqs, or assignments). In addition, IMS does not provide new information models for the services for which they have not released a specification. In a similar way, OKI defines a set of services different from IMS Abstract Model (some of them are similar), but this list is not exhaustive, so it does not include all the functionalities of an LMS. Moreover, the specifications are about behavioural models, but not data models. In our work, we try to show an exhaustive list of LMS services and we relate them with respect to existing data specifications.

The data interoperability between services can be achieved by defining a set of common terms and vocabulary between LMSs for each service. In our case, we have focused on LMS services for which there are not available data specifications at present neither in IMS nor in other e-learning standardization organizations. For services that share a lot of terms between LMSs (like forums, chats, assignments, etc.), we believe that it is better to define new specifications so that different systems can share the same vocabulary and understand common information; while for services that have a big diversity between different LMS platforms other approaches that are more flexible should be considered like the one commented in [4].

Previous works have already reported on some drawbacks of current educational standards, such as in [6]. Reference [6] comments the following drawbacks:

- *Some standards need extensions to cover some aspects*. Some examples of works that have extended present standards are [23] that extends SCORM to support adaptive courses, or [24] that extends IMS RDCEO for the competency model.
- *Some standards that overlap*. An example is the user modelling with the IEEE-PAPI and IMS-LIP specifications. Some terms of both specifications have a similar meaning. Reference [6] provides a solution based on ontologies to solve this gap about standards that overlap. Specifically, it provides an example using IEEE-PAPI and IMS-LIP.
- *Some aspects for which there are not specifications at present*. This is the point in which our contribution has focused. Reference [6] enumerates it as a drawback, but it does not analyze which aspects of LMSs have a lack of specifications. This is done in our work. The need of new specifications is also reported in other works, such as in [4] that classifies the different current e-learning specifications into groups depending on the LMS generation they belong to. In this way, reference [4] describes first generation LMSs linked to specifications as IMS-LOM; second generation LMSs linked to specifications as SCORM, IMS-LD, IMS-LIP, or OKI; and the next future service-oriented LMSs generation linked to specifications like IMS Abstract Framework or ePortfolio. In this context, we think that the new LMS service specifications of our work can be an important part to be located between the second and next generation of e-learning specifications.

8.3 Relationships of our specifications with other specifications

Section 2 describes specifications and standards that were specifically designed for e-learning. Nevertheless, there are other XML specifications with their correspondent data models that were created with other purposes different from e-learning, but can be used within LMSs. A lot of these XML specifications can be found for example in [17] and [26]. Each of these specifications tries to model different aspects of the world. Examples are XMLNews (for news), iCalendar, vCalendar, hCalendar (for Calendars), BiblioML (for bibliographic records), ParlML

(for parliamentary language), MessageML (for e-mail messages) or MathML (for mathematical notation).

On one hand, we have defined specifications that describe LMS services, for which there already exist a non e-learning specification (for the calendar, and news) or an e-learning one (for teachers). In such cases, we have created new specifications because there are some specific LMS aspects of the service that are not covered and others that are not usually used within LMSs. Nevertheless, in the three cases we might have taken as a starting point the existing specification instead of creating a new one. More details of the relationships between our calendar, news, and teachers specifications with respect to the h-cal, v-cal; XMLNews; and IMS Enterprise specifications are in Section 6 where all our new specifications are presented.

On the other hand, the rest of our defined specifications describe LMS services, that as far as we know such LMS services have not associated specifications neither within e-learning nor outside. The relationships of these specifications with existing ones are presented in Section 4.

Some LMSs have already incorporated some of the non e-learning specifications. For example, Moodle has an import/export functionality for the calendar according to the v-cal specification. In this way, the calendar of a person can be interchanged between different applications such as LMSs, Outlook, or PDAs. One step further in the ideal approach is to have each service of an LMS with this type of import/export functionality.

8.4 Authoring Tools

There exist several authoring tools that are compliant with present e-learning specifications. Some examples are for SCORM [14], IMS-QTI [19], or IMS-LD [22]. First of all, the authoring tool that we present is new because it is compliant with new e-service specifications that we have defined for different LMS services for which there were some lack.

A challenge regarding authoring tools is to achieve a user-friendly interface that at the same time covers all the aspects of the specification. For example in IMS-LD, there are some tools that cover almost all the aspects of the specification but are not easy to use by teachers without high technological knowledge, this is the case of RELOAD [22] or COSMOS [16]; and there are other tools that only cover some parts of the specification but are easy to use by teachers without high technological expertise, this is the case of COLLAGE [10] or LAMS [5]. With the authoring tool that we have implemented we show that for describing the selected LMS services, it is possible at the same time to describe all the aspects of the specification, and also to be easy for teachers without high technological skills. Our authoring tool shows that these aspects can be described in easy terms for teachers using several forms. Note that for other existing e-learning specifications, such as IMS-LD (for the more complex nature of the specification), there are more difficulties for achieving this double objective.

Finally, it is important to note that our authoring tool covers different specifications, which is according to the philosophy of an ideal authoring tool that includes all the aspects of e-learning in one authoring tool. Most of the present authoring tools only cover one specification. Therefore, for the entire interoperability of all the aspects, a course designer should learn and use different tools. Another example of authoring tool that includes several specifications is RELOAD that includes SCORM, IMS-LD, and IMS QTI, and has solved the problem of integration between SCORM and IMS QTI [27]. Our authoring tool is a tool that covers different service specifications in a unique interface. As far as we know, our authoring tool represents the first implementation of a service authoring tool that includes many LMS services generating XML instances complaint with specifications.

9. Conclusions and Future Work

In this paper, we have presented an ideal vision of complete interoperability of LMS courses,

including the different services as the most important part. We have presented several use cases with the fictitious Prof. Smith, which show the advantages of such ideal approach.

In order to obtain such advantages, e-learning has a need of standardization of their different topics. There are some aspects for which different specifications overlap, while there are other aspects for which there are not defined specifications. In this paper we relate the different main LMS services with respect to present e-learning specifications. Some lacks of specifications are found and we have created eight new specifications for LMS services. Each one includes an XML binding and a data model. Moreover, a way to use IMS-CP is presented in order to include all the LMS course information, orchestrating all the different LMS course services.

In addition, proper authoring tools are necessary for the successful adoption of the specifications and to achieve the general ideal envision. In this work, a proper authoring tool has been implemented that is conformant with the new defined specifications, is easy to use by teachers without high technological expertise, and includes several services in the same tool.

In this way, with the definition of the new specifications, and the creation of the new authoring tool, a step is given to the direction of the general ideal envision, enabling interoperability between different LMS educational services.

As future work, several ideas can be inspected:

- The creation of different LMS players that can import and run the XML instances according to the defined specifications.
- An exhaustive analysis of the integration of the new service specifications with existing e-learning specifications. For example, with the way to use the IMS-LD with the different parameters of other specifications for adaptation purposes.
- New specifications can be done for different LMS services, including a data model and

an XML binding for each one. This could be done for example for games or competition applications.

- The authoring tool may be integrated within existing LMSs or within existing authoring tools.

Acknowledgements
Work partially funded by Programa Nacional de Tecnologías de la Información y de las Comunicaciones, MEC-CICYT project MOSAIC-LEARNING TSI2005-08225-C07-01 and 02

References

[1] Advanced Distributed Learning (2004). SCORM 2004. Available at: http://www.adlnet.gov/scorm/index.cfm

[2] AICC. Aviation Industry CBT Comittee. Available at: http://www.aicc.org/

[3] T. Bray, J. Paoli, C.M. Sperberg-McQueen, E. Maler and F. Yergeau (2006). Extensible Markup Language (XML) 1.0 (Fourth Edition). W3C, Available at http://www.w3.org/TR/REC-xml/

[4] D. Dagger, A. O'Connor, S. Lawless, E. Walsh and V.P. Wade, Service-Oriented E-Learning Platforms: From Monolithic Systems to Flexible Services, IEEE Internet Computing, 11 (3) (2007) 28-35

[5] J. R. Dalziel, Implementing Learning Design: The Learning Activity Management System (LAMS), 20th Annual Conference of the Australasian Society for Computers in Learning in Tertiary Education. Adelaide (2003)

[6] V. Devedžić, J. Jovanović and D. Gašević, The Pragmatics of Current E-Learning Standards, IEEE Internet Computing, 11 (3) (2007) 19-27

[7] E. Di Nitto , L. Mainetti, M. Monga, L. Sbattella, and R. Tedesco, Supporting Interoperability and Reusability of Learning Objects: The Virtual Campus Approach, Educational Technology & Society, 9 (2) (2006) 33-50.

[8] C. Fallon, E-learning Standards: a guide to purchasing, developing, and deploying standards-conformant e-learning, (St. Lucie Press, 2003)

[9] N. Friesen, Interoperability and Learning Objects: An Overview of E-Learning Standardization. Interdisciplinary Journal of Knowledge and Learning Objects (1) (2005) 23-31

[10] D. Hernández-Leo, E. D. Villasclaras-Fernández, J. I. Asensio-Pérez, Y. Dimitriadis, I. M. Jorrín-Abellán, I. Ruiz-Requies, and B. Rubia-Avi, COLLAGE: A collaborative Learning Design editor based on patterns, Educational Technology & Society, 9 (1) (2006) 58-71

[11] IMS Global Learning Consortium (2003). Abstract Framework. http://www.imsglobal.org/af/index.html

[12] IMS Global Learning Consortium. http://www.imsglobal.org/

[13] ISO/IEC Joint Technology Committee Subcommittee on Standards for Learning Education and Technology http://jtc1sc36.org/

[14] S. Jun-Ming, T. Shian-Shyong, W. Jui-Feng, C. Kuan-Ting, L. Yi-Lin, T. Yi-Ta, An object based authoring tool for creating SCORM compliant course, 19th

International Conference on Advanced Information Networking and Applications, 2005, 209-214

[15] Learning Technology Standards Committee: http://www.ieeeltsc.org/

[16] Y. Miao, CoSMoS: Facilitating learning designers to author units of learning using IMS LD, 13th International Conference on Computers in Education, 2005, 275-282

[17] Microformats. http://microformats.org/

[18] M. Magee, N. Alliance, Building Digital Books with Dublin Core and IMS Content Packaging, Int. Conf. on Dublin Core and Metadata for e-Communities, 2002, 91-96

[19] I. Martínez-Ortiz, P. Moreno-Ger, J. L. Sierra, B. Fernández-Manjón, <e-QTI>: A Reusable Assessment Engine, International Conference on Web based Learning, ICWL, 2006, 134-145

[20] P.J. Muñoz, C. Delgado, R. Seepold, R. García, Rating the Importance of Different LMS functionalities, IEEE Frontiers in Education Conference, San Diego, California, 2006.

[21] Open Knowledge Initiative. http://www.okiproject.org/

[22] RELOAD project: Learning Design Editor (2005), http://www.reload.ac.uk/ldeditor.html

[23] M. Rey-López, A. Fernández, R. P. Díaz, J.J. Pazos, J. Bermejo-Muñoz: Extending SCORM to Create Adaptive Courses. EC-TEL conference, 2006, 679-684

[24] D. Sampson, P. Karampiperis, D. Fytros, Developing a Common Metadata Model for Competencies Description, Interactive Learning Environments 15 (2) (2007) 137-150

[25] M.M. Shaker, A. M. Khidhir, Y.A. Al-Rizzo, Design and Implementation of a Virtual University Prototype Technology requirements and Evaluation, in: Proc. CSIT2006, v3, 560-568

[26] The Coverpages. http://xml.coverpages.org/

[27] H. Vogten, H. Martens, R. Nadolski, C. Tattersall, P. van Rosmalen, R. Koper, CopperCore Service Integration - Integrating IMS Learning Design and IMS Question and Test Interoperability, International Conference on Advanced Learning Technologies, ICALT, 2006, 378-382

[28] G. Vossen, P. Westerkamp, Towards the Next Generation of E-Learning Standards: SCORM for Service-Oriented Environments, IEEE International Conference on Advanced Learning Technologies, ICALT 2006, 1031-1035

[29] XML Photo Album, http://xmlalbum.sourceforge.net/

Figure legends

Figure 1: OKI architecture based on layers

Figure 2: Course in Moodle about “Security”

Figure 3: Course in .LRN about “Security”

Figure 4: IMS-CP for describing an LMS course using the RELOAD editor

Figure 5: XML Schema graph for description of forums

Figure 6: Initial page of the Authoring Tool

Figure 7: First form for defining a FAQ block

Figure 8: A part of an XML instance example according to the FAQ specification

Table legends

Table 1: An overview of IMS specifications

Table 2: Relationship between LMS functionalities and current specifications