Abstract – Learning management systems (LMS) can be defined as “a suite of services designed to deliver, track, report on and administer learning content, student progress and student interactions”. Current LMS are based on courseware tools which mainly use Internet and the Web as learning platforms. However, they use proprietary/closed technologies and formats such as HTML that are not adequate for developing and managing sophisticated educational applications. Moreover, they also lack adaptive features in order to allow an individualised learning. The current work proposes a LMS which is based on standard and open Web technologies such as XML, Java and Javascript. It also includes features to adapt the delivery of educational contents depending on specific user profiles. An implementation called XEDU is being developed which runs on Apache Web Server and MySQL database.

I. INTRODUCTION

Learning management systems (LMS) can be defined as “a suite of services designed to deliver, track, report on and administer learning content, student progress and student interactions” [1]. The importance of these systems is growing as far as terms such as “Internet-based Educational Systems”, “Educational Service Providers”, “Distance-learning “ or “On-demand Education” are becoming popular. Information and communication technologies (ICT) such as Internet and the Web provide an ideal framework for current LMS. They are mostly based on courseware tools either commercial or coming from university research groups.

Some of them use proprietary multimedia authoring tools like Authorware™ or Toolbook™ that bound the reuse and exchange of educational contents [2]. In other cases they are based on formats such as HTML that seem adequate for presenting and accessing information in a open environment like the Web but lack a data structuring capability. Moreover, they also lack adaptive features such as navigation support or presentation control mechanisms.

The current work proposes a LMS which is based on standard and open Web technologies such as XML, Java and Javascript. XML (eXtensible Markup Language) [3] is a language which describes a class of objects called XML documents. These documents have been used to represent and structure information from WebCT courses in IMS [14] and it can interface directly with any course that has been written to AICC standards to provide AICC tracking and management functionality. WebCT provides tools such as IMS Content Migration Utility which allows administrators to bring in or export content and assessment information from WebCT courses in IMS [14].

Commercial products such as TopClass [9], WebCT [10], or Learning Space [11] permit the use of open and non-proprietary formats of the Web like HTML and they are involved in the introduction of metadata standards [12]. Current Learning Space version is AICC compliant [13] and it can interface directly with any course that has been written to AICC standards to provide AICC tracking and management functionality. WebCT provides tools such as IMS Content Migration Utility which allows administrators to bring in or export content and assessment information from WebCT courses in IMS [14].

II. RELATED WORKS

There are several proposals of “open and flexible” LMS in the sense they have features such as the use of non-proprietary formats, the distributed access and choice of learning time/place or the personalization of teaching material for individual users. We are particularly interested into the organization of the teaching material (not only courses) and the facilities to manage it irrespective of the delivery medium.

Some of these LMS requirements is called IMS Meta-data and ADL SCORM Content-level.
However, these proposals mainly aim at using these metadata standards as mechanisms to improve the interoperability with their courseware products and they bound the information to be managed in these open formats. Moreover, most of the commercial proposals rely on HTML links (or sequences of Web pages) to organize the course material and this hardens the way to manage such information.

Other options have developed more advanced “open” features and not bounded to courseware issues. For instance, Microcosm [16] provides a link service to connect multiple distributed learning applications, or Gentle [17] (now ELS) that is based on the Hyperwave server which permits the organization of documents into clusters and collections, and the management of links as independent entities. ADL [1] proposes a Shareable Content Object Reference Model (SCORM) which defines a Web-based learning "Content Aggregation Model" and "Run-time Environment" for learning objects. It combines elements of IEEE, AICC and IMS specifications and expands them to include additional course structure capabilities. The Open University of the Netherlands proposes the use of EML [18] (Education Modeling Language) to codify units of study (e.g. courses, course components and study programs) in an integral fashion. It has an XML binding and an Edubox-player has been built that interprets EML files and creates a concrete learning environment. Our work is close to this proposal in the sense that we rely on a XML-based notation to represent every kind of learning information but we intend to cope with a wider range of learning structures [20] rather than strict EML "units of study".

III. LMS MODEL

The XEDU LMS is based on a model whose main services are described as follows:

Resource Authoring

This service addresses all the aspects related to the authoring of educational resources (see Figure 1). Such resources are defined as every information item that is used for education purposes. They are divided in two types: contents and structures. Contents represent the own data associated to the educational resources (based on multimedia formats, meta-data information's or binary programs) while structures concern their organization (in form of lists, kind or part taxonomies, or algorithms) [20]. It is important to remark that courses are only a type of organization and they are managed such as any other structure.

There are several possibilities to represent contents and structures, for instance using “learning objects” and “packaging structures” [14], “units of study” [18] or “elements” and “templates” [19]. The current LMS functionality is independent from the particular resource implementations. At this moment, we are using LMML contents [21] and own structures [22]. These elements are developed outside the system and the LMS only provides procedures to make them accessible.

Resource Publishing

The publishing of educational resources depends on the selected structure (e.g. a course), the system configuration and the user characteristics (see Figure 2). Each structure has assigned a specific presentation schema and a set of navigational tools. These structure schema are tailored using a system configuration in which display parameters are set up (e.g. text fonts, screen background or frames assigned to each structure component).

The presentation aspects can also be dynamically adapted depending on the user preferences, conditions or status. The navigational aspects such as the selection of links or their direction can be also adapted.

Figure 1.- Resource Authoring services.

Figure 2.- Resource Publishing services.
User management

In this section, the services for managing the information about the user are described (see Figure 3). This information comes from documents that contain personal data [23] or the user knowledge about a given topic. Users can also introduce their preferences.

Course/data administration

These services are addressed to organize the aspects around a course, from the selection of the structures and contents to be taught to the tracking of the user accesses (see Figure 4). They also involve the enrollment of users in a given course, the checking of prerequisites, its temporal scheduling or the setup of evaluation procedures. Finally, course certifications are produced and analysis of system and user performance can be made.

IV. LMS IMPLEMENTATION

The implementation of the previous services is performed in the context of the XEDU framework [24]. XEDU is being developed and its general structure is shown in Figure 5.

The main system is a Web application that runs on an Apache Web Server and it uses a MySQL database. This application is formed by several Java servlets which process the XEDU information elements and a "Control" applet which calls for them. The "DBResource" servlet is addressed to manage the information related to educational infrastructure and stored in the database either for teachers or students. It is invoked when the required data are in a database format and they are converted to a DOM model which is transferred to the calling applet. There is another servlet ("XMLResource") that reads the information coming from XML documents (e.g. administrative procedures) and obtains the equivalent DOM model. The Cocoon servlet [6] is also applied to perform resource management since it allows the teacher to build its own educational structure for a given topic. There is available a starting empty XML template and he can introduce, by means of XSL scripts, a component hierarchy for this topic (see Figure 6). Such hierarchy is used like the content repository and its component can be organized in different ways according to the student requirements.
The "ForumTool" servlet deals with the management of the different communication forums in which a specific user is involved. They are implemented as XML documents and their management is similar to the other educational resources. The "SessionTrace" servlet is addressed to store the session information that a user tracks. Figure 7 shows an XML document which gathers the session date, the interval of time and the accesses to the resource items.

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE log SYSTEM "log.dtd" >
<log>
  <session date="18/11/2001">
    <time start="9:42:14" end="9:42:57"/>
    <trace step="AC2D:8"/>
    <trace step="AC2D:1"/>
  </session>
  <session date="20/11/2001">
    <time start="9:56:14" end="9:57:11"/>
    <trace step="AC2D:1"/>
  </session>
</log>

Figure 7.- Trace information example.

The resource publishing is based on the "Control" applet which configures the Web pages accessed by the users. They are divided into several areas or frames as Figure 8 shows. There is an upper frame that gathers the title and a set of general options such as Presentation, Classroom, Administration and Communication. The lower frame stores utilities such as Search, E-mail or Help. When the user clicks on an option like Classroom, the left frame is assigned with an index map attached to the material structure. The main frame is used to show the content information (e.g. the content of a practice activity about XML).

V. CONCLUSIONS

This paper shows the design and implementation of an Open and Flexible LMS called XEDU. It takes advantage of standard Web technologies such as XML, Java and Javascript. The first one permits the strict separation between resource elaboration and its publishing in different formats. The use of Java and Javascript programming languages improves the XEDU portability. XEDU versions for Windows and Linux environments have been developed. Currently, we are checking its application in several computing subjects at the Informatics School (Polytechnic University of Valencia). Teachers who are responsible for introducing learning material are focused on its elaboration and they are not concerned about the presentation details. This feature eases their task. On the other hand, students have the possibility to deliver their assignments in a format that can be easily processed by the teacher.

VI. REFERENCES


[18] EML (Educational Modelling Language), http://eml.ou.nl/introduction/


