**Monograph of next issue (July 2011)**
**"Business Intelligence"**

(The full schedule of **CEPIS UP**GRADE is available at our website)

# UPGRADE
The European Journal for the Informatics Professional
http://cepis.org/upgrade
Vol. XII, issue No. 2, April 2011

## Monograph
## Engineering e-Learning Systems
**(published jointly with Novática*)**
Guest Editors: *Ignacio Aedo-Cuevas, Kinshuk, Demetrios G. Sampson, Antonio Sarasa-Cabezuelo, and José-Luis Sierra-Rodríguez*

## UPENET (UPGRADE European NETwork)

## CEPIS NEWS

* This monograph will be also published in Spanish (full version printed; summary, abstracts, and some articles online) by **Novática**, journal of the Spanish CEPIS society ATI (*Asociación de Técnicos de Informática*) at <http://www.ati.es/novatica/>.

# e-Learning Services in Moodle 2.0

*Miguel-Ángel Conde-González, Alberto del Pozo-de Dios, and Francisco-José García-Peñalvo*

*The unstoppable advance of new technologies has made evident the need for updating learning platforms. This update is based on the incorporation of new functionalities in order to satisfy the changing users' needs. One of the ways to carry this out lies in the evolution of learning environments into a Service-Oriented Architecture (SOA). The implementation of these architectures will allow the creation of clients and external tools that can work with the resources contained in the platform, giving users a freedom of movement that they did not have before. Moodle 2.0 is an example of this evolution and this article presents some of the new possible applications.*

**Keywords:** e-Learning, LMS, Moodle 2.0, SOA, Web Services.

## 1 Motivation

The big scientific and technological progress that has occurred in society in recent years has caused new educational trends to appear [1][2]. e-Learning has brought in a different approach to learning processes, which introduces new requirements that learning platforms will support. Only a few years ago, using a computer as a tool for supporting the teaching and learning process was considered a technological step forward, but now it is seen as an everyday activity. It is sought to further enrich this process with the ability to access all the information wanted, when it is wanted, that is to say that knowledge can be available all time. This objective could be considered as the main target of learning platforms (LMS, Learning Management Systems). However they are not always able to satisfy these requirements.

This is due to LMS being too generic, poorly adapted to specialized circumstances, and hardly scalable [3], which hinders their growth and sustainability.

Taken this into account, it must also be considered that in most cases, the power of the LMS is not fully used. For example, many of the features of learning platforms are ignored, making them become in many cases mere resource containers[4][5][6].

Because of this situation, there are a number of initiatives that are trying to provide a degree of independence to these e-Learning platforms, so that they are no longer monolithic and closed platforms. They aim to provide development capacity and technology-independent growth. This independence is achieved by implementing service-oriented architectures, which facilitate interaction to and from the platform, and therefore the incorporation of new functionalities and the possibility to work with new contexts such as mobile devices.

This article will introduce this architecture and some of the applications developed to work with it. In the next sec-

tion we talk about SOA, Service-Oriented Architecture, and its possibilities, also presenting the relationship between

**Authors**

**Miguel-Ángel Conde-González** studied Computer Science in the *Universidad de Salamanca*, Spain, and now is a PhD candidate. From 2002 to 2004 he was working in an educational environment teaching several courses related to computers. In 2004 he decided to begin working in a software development environment, working in GPM. In 2005 he began working in the Clay Formación Internacional R&D Department, where he was involved in various e-Learning projects. Now he is researching at the University of Salamanca as a member of GRIAL (Group for Research on Interaction and e-Learning – http://grial.usal.es) and is also working there as a teacher. <mconde@usal.es>

**Alberto del Pozo-de Dios** is a Technical Engineer in Computer Systems whose work in the past two years has focused on developing (and backporting) the Web services of the new version of Moodle (2.0), working with the research group GRIAL, Clay International Training Group, DFWikiLABS developers of UPC and Moodle. <delpozo1988@usal.es>

**Francisco-José García-Peñalvo** holds a PhD in Computer Science (2000, *Universidad de Salamanca*, Spain). He works as a Professor in the Computer Science Department of the University of Salamanca. He is Director of the Research Group in Interaction and e-Learning (GRIAL – http://grial.usal.es). His main research interests are e-Learning, computers and education, adaptive systems, web engineering, semantic web and software reuse. He was a member of the Technological and Academic Committee of the White Book for Digital University 2010. He was Technology and Innovation Pro-Chancellor of the University of Salamanca from March 2007 to December 2009 and the President of the Technical Committee of the SCOPEO Observatory (an observatory for e-learning activity, innovation and trends), from its inception until December 2009. He has published more than 100 papers in international journals and conferences. <fgarcia@usal.es>

> **" The unstoppable advance of new technologies has made evident the need for updating learning platforms "**

SOA and Web services. Then the inclusion of web services within Moodle 2.0 will be analyzed, and how these allow integration with different utilities. We will also explain an example of a service and its implementation.

## 2 Service-Oriented Architecture (SOA)

To be able to consider the latest advances that LMS can provide it is necessary to define what the service-oriented architecture is and what its potential applications are in the context of e-Learning. SOA is a concept in software architecture that is based on creating a set of services between business processes and applications of varying granularity [7]. This architecture has as its main objectives: 1) Model the business logic as services; 2) Provide access to functionality without knowing the underlying technology; 3) Minimize technological dependencies between the business layer and the application layer, so being able to change both independently.

The architecture and service concepts will be defined to facilitate the understanding of SOA architectures. According to IEEE-STD-1471-2000 ("Recommended Practice for Architectural Description for Software-Intensive Systems") an architecture is "*the fundamental organization of a system embodied in its components, their relationships to each other and to the environment and the principles guiding its design and evolution*" [8]. The architecture could be seen as the structural organization of the components of a system.

Once the term architecture has been explained, it must also be known what a service is. There are different approaches to what services are. They can be considered from a business point of view as "*a functionality built as a reusable component for being used in a business process*" [9] or from a technical point of view as "self-described elements independent of the platform that support rapid, cheap and distributed composition of applications" [10]. In any case, applications or service providers will provide services in order to furnish functionality without showing their implementation. That is, we know what the service does but not how that action is done internally.

The main benefit in implementation of this architecture is the structuring of the component elements and the establishment of communication elements, so that if two systems want to communicate each other they do not need to see how the other works, only using this layer as an intermediary which does know how these systems work. If at any time it is necessary to replace or make a change in one of

the two systems, the change is then independent of the other, as they have been developed to be independent of the other system, depending only on the data returned[11]. This way of linking components provides the following advantages: 1) it allows the replacement of individual components without affecting other components; 2) all systems are connected to the exchange layer in the same way, making the system more homogeneous; 3) ease of operation and maintenance; 4) a simple, robust and scalable architecture.

### 2.1 SOA and Web Services

It is important to understand that SOA is not synonymous with Web services. While SOA is a development paradigm (and a strategy of Information Technology, IT), Web services are one of the technologies that can be used to implement SOA. It is however to be noted that the implementation of SOA is becoming more rapid due to Web services and these are becoming the de-facto standard for the implementation of these architectures.

Once the difference between SOA and Web services have been established, it is necessary to explain what Web services are. There are several definitions of what Web services are, involving complexity in giving an adequate explanation of all they are and mean. They could be considered as a set of technologies or applications that exchange data with each other in order to offer services. Providers offer their services as remote procedures and users request a service by calling these procedures through the Web [12]. These services provide standard communication mechanisms between different applications, which interact with each other to present dynamic information to the user. The use of Web services provides a range of advantages (many of them derived from the benefits of implementing a Service Oriented Architecture) as:

■ Promoting interoperability, since the interaction between a supplier and a service requester is designed to be completely independent of platform and language.

■ Promote standards and protocols based on text, making easier the access to their content and the understanding of their operation.

■ Since Web services are based on HTTP, they can take advantage of firewall security systems without changing the filtering rules.

■ Reduce complexity through encapsulation, because service requesters and service providers deal with their own interfaces they need to interact. As a result, a service requester does not know how the provider implemented the

> **" e-Learning has brought in a different approach to learning processes, which introduces new requirements that learning platforms will support "**

❝ A *service* is a functionality built as a reusable component
for being used in a business process ❞

service, and at the same time, the provider does not know how the customer uses the service.

■ Allow interoperability between platforms from different vendors by using open standard protocols, since specifications are managed by an open organization, the W3C.

### 2.2 SOA in e-Learning

In order to incorporate interoperability in learning platforms and make them flexible and scalable, it is necessary to define a new generation of learning platforms. These kinds of platforms are going to be based on Service Oriented Architectures. This type of solution will provide a separation between the service interface and its underlying implementation. It will no longer be important whether an application that wants to connect to a platform is implemented in a different technology from the core of the LMS. SOA provides independence in the evolution of software, which allows adding new functionalities whatever the underlying LMS version.

Among the possible applications of SOA various approaches should be considered:

■ Use SOA to provide information to external contexts. For example the use of service-oriented architectures for semantic searches of information in a learning platform, as illustrated by the LUISA project [13].

■ Small adaptation of learning platforms to other applications, such as authentication services and backoffice and administrative communication tools [14].

■ Linkages between platforms and applications in which the integration is completely transparent to users, allowing two-way communication and providing a mode of presentation fully adapted to the LMS. Several specifications are proposed for this, such as IMS LTI (Learning Tools for Interoperability) for transparent integration of applications into platforms, or OSIDs (Open Service Interface Definitions) from the OKI project (Open Knowledge Initiative) [15], which describe ways to communicate with the platform using other tools.

■ Mixed adaptations, where applications require communication with the LMS to extract and incorporate information into it, but without having themselves to be incorporated therein, such as mobile clients for learning plat-
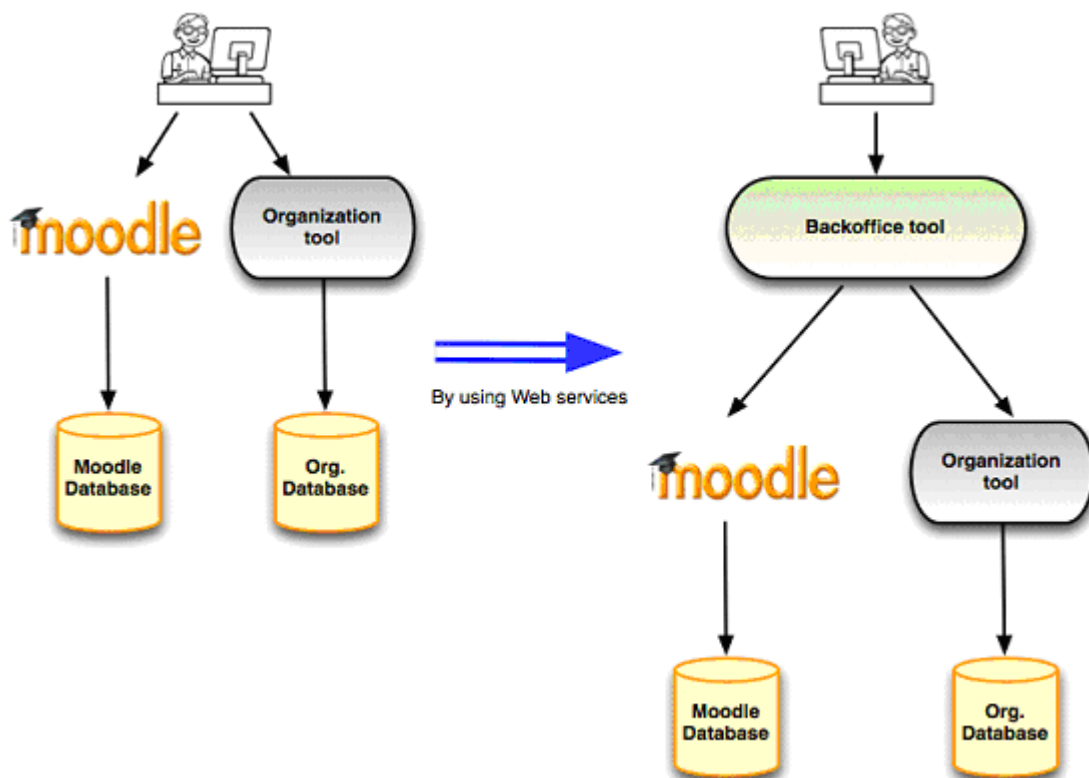


**Figure 1:** Management of Moodle and a Database of an Organization before and after the Implementation of the Web Services, by using a Backoffice Tool.
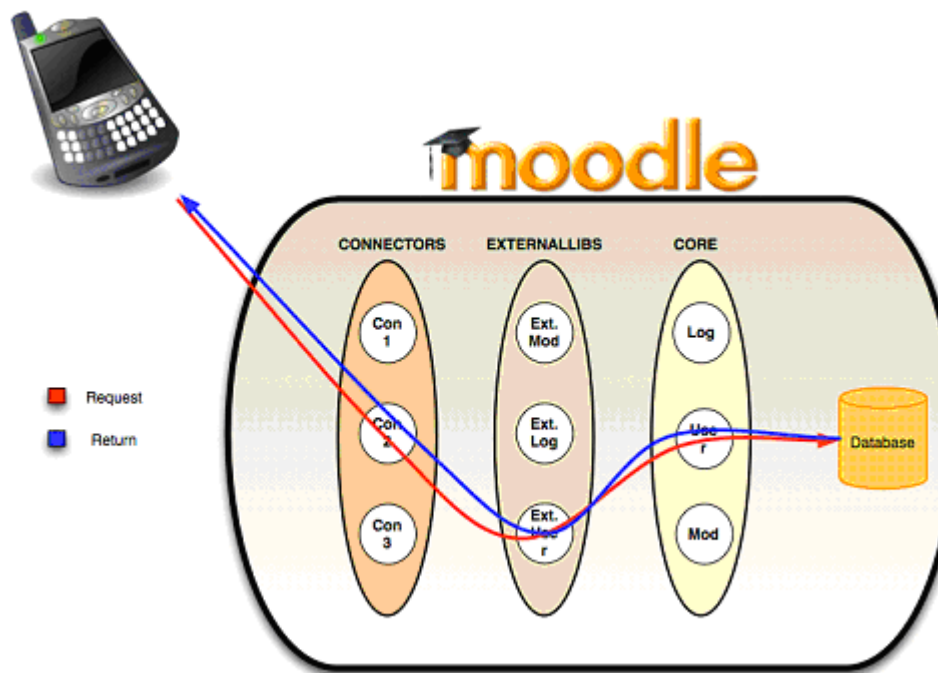
**Figure 2:** Web Services Layer in Moodle 2.0.

forms. Moodbile [16] is a notable example of such integration.

In any case the aim is to provide new functionalities to learning platforms, allowing them to evolve to an evolutionary, scalable and flexible model from a monolithic model, which tends to become obsolete.

### 3 Web Services in Moodle 2.0

Moodle is an open-source learning platform, meaning it is a free application that educators can use to create effective learning sites online. The new version of Moodle was released in November 2010. This new version was seen as the opportunity to do things differently, to give a radical change to the platform and adapt it to the technologies that are flooding the telecommunications market. Many of these changes include support for external repositories (Picasa, YouTube, Flickr, Wikimedia, etc.), new modules and blocks, changes in the core, etc., and one of the most important changes is the support for Web services. These Web services will greatly expand the possibilities of Moodle, from being a monolithic platform to become a scalable application that can interoperate with users. These services could provide solutions to emerging needs of the users, such as the following:

■ The emergence of mobile devices with Internet access, with various interfaces and features. Many of these devices allow navigation through the application, but, due to their physical limitations the navigation can be very complex. Therefore, knowing that the number of these devices is growing, it would be advisable that Moodle facilitates the creation of alternative interfaces adapted to these platforms.

■ The number of organizations that rely on Moodle as their e-Learning platform has increased in recent years, leading to changes in the system requirements (scalability) and diversification, as there are continually emerging new needs, such as new functionalities (without corrupting the pedagogical principles of Moodle). It is still necessary to adapt Moodle to the information systems of organizations where it is implemented.

■ Backoffice integration. The possibility is provided of creating an application that can interact with multiple systems at once, executing the same action in several places, making it all work together, avoiding having to do the operations several times. For example, you may need to register a user in the platform and in the database of the organization (left side of Figure 1).

❝ It is important to understand that SOA (Service-Oriented Architecture) is not synonymous with Web services ❞

> **❝** SOA will provide
> a separation between
> the service interface and its
> underlying implementation **❞**

By using Web services, the action would have to be done just once using a backoffice tool, which is responsible for carrying out the actions both in the database of the organization and in Moodle, using Web services, without modifying or accessing the code of the platform (right side of Figure 1).

### 3.1 Architectural Approach

The first step in Web service development was to define an architecture that would ensure interoperability. Therefore, these Web services had to fulfil certain requirements established by the Moodle development team [17]:

■ Web services must be accessible from any connection system, now and in future, and they should be able to be invoked regardless of the language used in the request (interoperability).

■ The Web services structure must be developed so that even if there are changes in the Moodle core, it is not necessary to make many (or any) modifications to the set of feature they provide, i.e. the API (Application Programming Interface).

■ The functions in the API should be extensible to encourage contributions.

■ The Web service must be adapted to the Moodle system privileges (capabilities) to ensure safety.

According to these requirements, Moodle 2.0 Web services are divided into three basic layers [18], shown in Figure 3. External customers connect with the platform via the connectors that receive the requests, interpret them and call the corresponding functions of the Web services. These are defined within externallib.php files distributed over all the directories in Moodle. These functions are responsible for the access to the database or the invocation of the Moodle functions in order to carry out such tasks.

*1. Connectors*. So far, the platform can be connected through 5 protocols. Each of these protocols has its own connector, which is responsible for receiving the request from outside, checking if the wanted function exists and checking the permissions of the invoking user. If the user is authorised to use the function, connectors will analyze the data (parsing) and will call the appropriate function. Connectors support plug-ins, so it is easy to add new connectors for external systems to be able to connect with Moodle using protocols different to the ones that come as standard with the platform.

> **❝** Moodle is an open-source
> learning platform, meaning it is
> a free application that educators
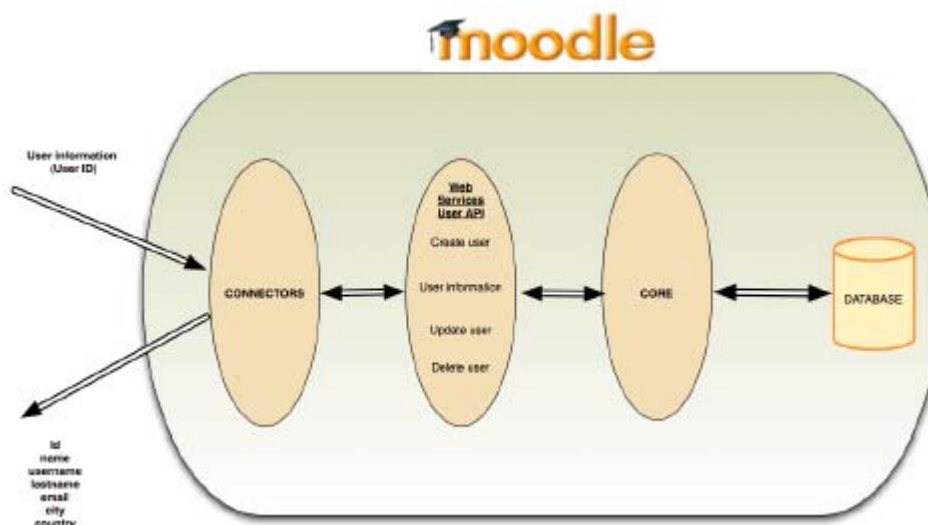> can use to create effective
> learning sites online **❞**



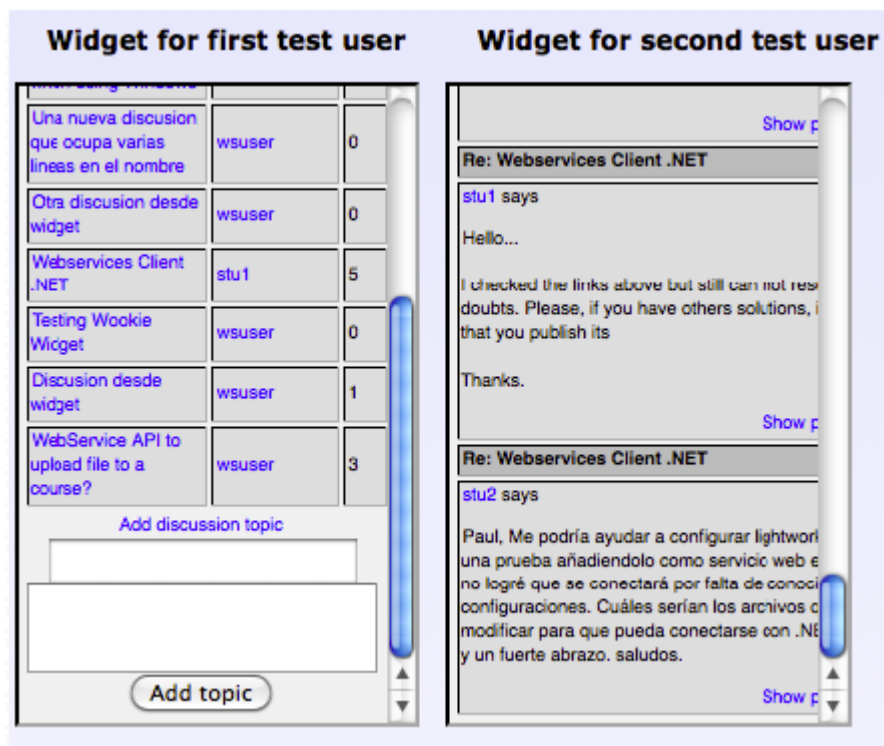**Figure 3:** Recovering Information from a Particular User in Moodle 2.0.

**Figure 4: E**xample of Forum Widget created using Wookie.

*2. Externallib*. This layer consists of a set of files, named externallib.php, that are spread throughout the directory tree of Moodle. These files are called from the connectors and contain all the features offered by the Web services API. That is, they comprise all the Moodle features to deliver them to the outside, trying to reuse as much code as possible from the platform. Naturally, before beginning to perform any of the actions, user permissions are checked with respect to the action to take, and the parameters received or returned are also checked. The parameter check is achieved by having a number of methods that indicate the parameters that the functions must receive and that they should return.

*3. Moodle core*. The Moodle core layer consists of all the libraries that contain functions that may be used by the externallib layer functions, i.e. user-related functions, courses, groups, etc. This layer has been improved in Moodle 2.0 because many of these functions formerly printed error messages on the screen when there was a problem, so they have been rewritten to return exceptions when there is an error. (Until now, Moodle had no API, but due to these changes one is now being generated).

As an example of message exchange and Web services structure, the following picture (Figure 3) shows how to recover information from a particular user.

To be able to retrieve data from a user, first it is necessary to select a connector and pass parameters: the action to perform and the information required to carry it out, in this case the "User Information" and the ID of the user to retrieve. The connector will check the permissions and whether the function exists in the user Web service API. In that API it will check the validity of the parameters and will invoke the appropriate function within the Moodle core. Once the information is retrieved, the answer is built and returned to the client via the connector that was used for the connection.

**❝The authors of this article have taken part in the definition of the new services-based architecture and have performed several deployments to verify its operation ❞**

## 3.2 Moodle Web Services Applications

It is obvious that Web services will open a new world of possibilities related to the export of information from Moodle and the integration or interaction with other tools. The authors of this article have taken part in the definition of the new services-based architecture and have performed several deployments to verify its operation:

■ *Backoffice tool*. During the development of the external layer of the Web services, it was decided to implement a testing tool that would test if the API functions worked properly, besides their usability. It was necessary to cover as many functions as possible, so a small-scale backoffice tool was created, which allowed an administrator to perform the basic steps in a Moodle platform. Among these steps were those to be able to manage users, courses, roles, logs, etc., and also including elements in order to test each function of the Web services API and all the connectors.

■ *Visualization tool.* Adaptation of a logging visualization tool to by using the Moodle Web services [19]. This tool is designed to work with the information about the user activity on learning platforms (so far in Moodle), showing for example: information that allows analysis of the best times to do technical maintenance on the platform; make social studies (allowing visualization of the connections between people existing in the platform, word clouds to see the most used terms, etc.). Web services provided the tool with the logs from the platform without having to access the database directly. In this way the tools developed could be independent from the implementation technology and version of the underlying platform

■ *Portable learning components.* Another aspect to consider, for possible exploitation of the new services architecture, is the ability to export functionality of the platform to other contexts, such as informal learning environments, social networks, mobile devices, etc. Exporting this functionality requires a form of representation, which will be by widgets. Widgets are small, portable items that can work in any HTML context, providing interaction, content or the functionality of another web context [20]. There are different types of widgets, as well as engines to generate them. The authors of this paper have recently finished the development of a widget for the forum export. To do this, the Apache Wookie widget engine has been used [21]. An example of the current work with widgets is shown in Figure 4, showing a widget created to manage forums.

## 4 Conclusions

Learning processes are changing, not only due to the development of new technologies, but also sociological changes such as: the emergence of social trends favoured by 2.0 tools; the new training needs more and more student orientation; the new contexts and situations where students learn; the fact that the receivers of the information have been born in the digital age and therefore use the technology at a level not previously considered; new educational paradigms derived from proposals such as Bologna legislation which advocates for the recognition of both formal and non formal learning; etc.

All this shows the need for the development of tools used in learning processes such as LMS. To do that, SOA architectures are needed. These architectures are an opening to allow the increase in functionality of learning platforms, providing a way to avoid stagnation and paving the way towards new trends such as personal learning environments (PLEs) [22].

Moodle 2.0 and the tools defined for its testing are an example of how a platform can go further and can actually be scalable and flexible thanks to the foundation that Web services provide. By using service-oriented architectures on this well-known LMS, a way to enhance its capabilities, independently of the technology, has been achieved. This type of application will ensure the evolution of the "species", which are the ways students use to learn, such as learning platforms and contexts to which they can be applied.

### References

[1] F.J. García Peñalvo. *Estado Actual de los Sistemas E-Learning. Teoría de la Educación. Educación y Cultura en la Sociedad de la Información 6*, 2005.

[2] F.J. García Peñalvo. Advances in E-Learning: Experiences and Methodologies. Information Science Reference, Hershey, PA, USA, 2008. ISBN-10: 15990 4756X.

[3] J. Fontenla, M. Caeiro, M. Llamas. *Una Arquitectura SOA para sistemas de e-Learning a través de la integración de Web Services. Congreso Iberoamericano de Telemática*, CITA 2009, pp. 22-29.

[4] University of North Carolina. Sakai Pilot Evaluation Final Report. <http://www.unc.edu/sakaipilot/evaluation/FinalRept-Oct15-09-sm.pdf>, 2009.

[5] L. Cuban. Oversold and underused: Computers in the classroom. Harvard University Press, Cambridge, MA, 2001.

[6] C. Milligan. The Road to the Personal Learning Environment? <http://zope.cetis.ac.uk/members/ple/resources/colinmilligan.pdf>, 2006.

[7] The Server Labs. *¿Pero, qué es realmente SOA?* <http://www.theserverlabs.com/folletos/Folleto%20SOA.pdf>. [Last accessed on 31/05/2010]

[8] M.W. Maier, D. Emery, R. Hilliard. IEEE 1471 and systems engineering. Syst. Eng. 7, pp. 257-270, 2004.

[9] M. Keen, D. Adamski, I. Basu, P. Chilcott, M. Eames, M. Endrei, B. Fagalde, R. Raszka, S.D. Seabury. Implementing Technology to Support SOA Governance and Management. IBM Redbooks publication, 2007.

[10] M.P. Papazoglou. Service-Oriented Computing: Concepts, Characteristics and Directions. Fourth International Conference on Web Information Systems Engi-

neering, pp. 3. IEEE Computer Society, 2003.

[11] J. Alba. *¿Qué es SOA - Arquitectura Orientada al Servicio*. Bit, vol. 167, pp. 52-53, 2008.

[12] W3C. G*uía Breve de Servicios Web*. <http://www.w3c.es/divulgacion/guiasbreves/ServiciosWeb>, 2010.

[13] LUISA. Learning Content Management System Using Innovative Semantic Web Services Architecture. <http://luisa.atosorigin.es>, 2009.

[14] S.Pätzold, S. Rathmayer, S. Graf. Proposal for the Design and Implementation of a Modern System Architecture and integration infrastructure in context of e-learning and exchange of relevant data. In: E-Learning, E.I.F. (ed.) ILearning Forum 2008., pp. 82-90, 2008.

[15] OKI Project. <http://www.okiproject.org/>. [Last accessed on 31/05/2010]

[16] M. Alier, M.J. Casany. Moodbile: Extending Moodle to the Mobile on/offline Scenario. IADIS International Conference Mobile Learning, Algarbe, 2008.

[17] Ferrán Recio Calderó. *Arquitectura de los webservices de Moodle* <http://blogs.dfwikilabs.org/moodle_ws/2008/04/10/arquitectura-de-los-webservices-de-moodle/>. [Last accessed on 31/05/2010]

[18] Moodle Happens. Moodle 2.0 Web Services architecture <http://blogs.dfwikilabs.org/pigui/2009/04/27/moodle-20-web-services-architecture/>. [Last accessed on 31/05/2010]

[19] D.A. Gómez, R.T. Sánchez, F.J. García. *Semantic Spiral Timeline como apoyo al e-learning*. Journal of Universal Computer Science (j-jucs), 2008.

[20] W3C. Widgets 1.0 Packaging and Configuration. <http://www.w3.org/TR/widgets/>, 2008.

[21] Apache Incubator. Wookie Project, <http://incubator.apache.org/projects/wookie.html>.

[22] S.Wilson, O. Liber, M. Johnson, P. Beauvoir, P. Sharples, C. Milligan. Personal Learning Environments: Challenging the dominant design of educational systems. Journal of e-Learning and Knowledge Society 3, pp. 27-38, 2007.