The Use of Web Services as a Strategy for Integration of LORs to LMSs

Alessandra Pereira Rodrigues[1], Liane Margarida Rockenbach Tarouco[2], Marcelo Augusto Rauh Schmitt[3]

[1]Instituto Federal Sul-rio-grandense, Pelotas, RS, Brasil
ale@pelotas.ifsul.edu.br
[2]Pós-Graduação em Informática na Educação - UFRGS, Porto Alegre - RS, Brasil
liane@penta.ufrgs.br
[3]Instituto Federal do Rio Grande do Sul, Porto Alegre, RS, Brasil
marcelo.schmitt@poa.ifrs.edu.br

Abstract. Web Services is a solution used in the integration of systems and have some communication between different applications. We describe the use of Web Services as a strategy of integration between a Learning Objects Repository and a Learning Management System. To demonstrate this integration are presented the first results obtained in implementing the integration of Learning Object Repository (LOR) CESTA2 and Learning Management System (LMS) MOODLE. This integration raises important contributions to educational institutions which uses the DSpace as Institutional LOR and MOODLE as Institutional LMS, because they can integrate them obtaining all the benefits that this integration can offer.

Keywords: *Web Services*, Learning Object Repository, Learning Management System, Integration.

1 Introduction

The integration of information systems is a challenge often confronted by professionals in the field of Information Technology. The systems aren't usually designed to be integrated, they are developed in a customized manner to find a specific problem. Therefore, the systems use languages, technologies and platforms, according to the need and ease of applicability in each situation. Thus, integration emerges from the need to share data across heterogeneous systems. The diversity of languages and technologies used in the implementation of information systems complicate the communication between them. With the advent of the Internet and the consequent evolution of technology, it started to appear the possibility of communication is established directly between systems hosted on servers geographically separated to allow the provision of an integrated service. This communication not mediated by humans presents greater difficulties, but it began to emerge some solutions for the integration among them, that this article presents, which is based on Web Services [1].

The aim of integration is to combine several systems giving the user the illusion of interacting with one. In this case, users are provided with a uniform logical view of data that are physically distributed across heterogeneous data sources [2]. There are now technologies and solutions created specifically to support and encourage integration, such as XML and object-oriented programming. This is especially important in areas like education, where there is demand to integrate the systems of Learning Management Systems and Learning Objects Repositories available. With this type of integration is possible to have the growth of the use of Learning Objects (LOs) in the teacher's planning and thereby facilitate their daily work.

In this article we discuss the problems that make it necessary the integration of the systems, followed by approaches that enable this integration, the presentation of Web Services as a strategy for the integration of the LOs Repositories to Learning Management Systems and the first results of implementing a plugin for integrating a LOR CESTA2 to LMS MOODLE.

2 Approaches/Strategies for Systems Integration

The problem of systems integration is known for a long time [3]. Many solutions have been used in attempts to accomplish this process. The architectural perspective of an information system provides an overview from different ways of approaching the integration problem. The classification presented in Figure 1 distinguishes the approaches of integration according to the level of abstraction, where integration is performed.

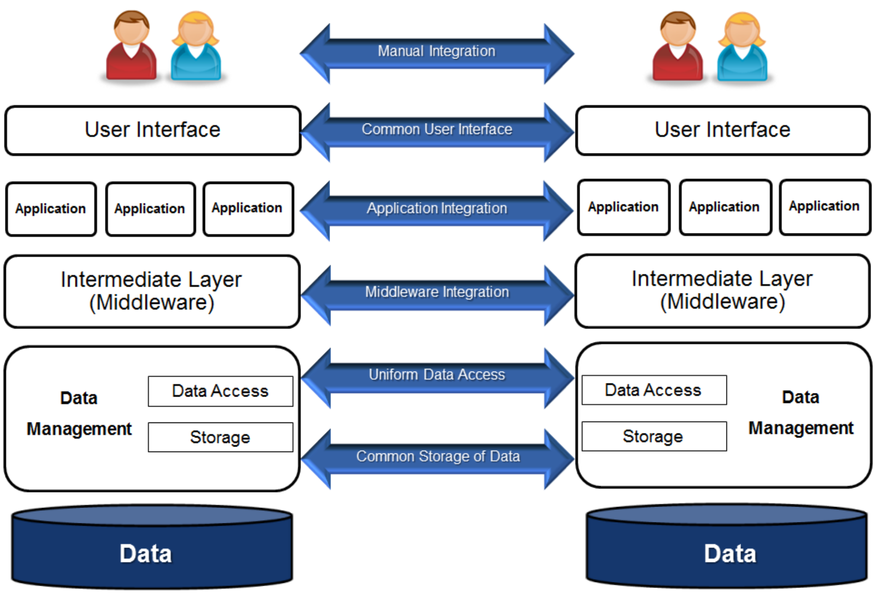

Fig. 1. General Approaches of Integration on Different Levels of Architecture. Adapted version of figure is available in [2].

The approaches presented in Figure 1 can be used alone or combined according to the need for integration. The following describes each of the approaches listed in Figure 1 [3]:

Manual Integration. In this approach users interact directly with all relevant information systems and selected data are manually integrated. That is, users have to deal with different interfaces and query languages. In addition, users need to have detailed knowledge about the location, the logical representation and semantics of the data.

Common User Interface. In this case, the user is provided with a common user interface (e.g. a web browser) that provides a uniform look and sense. Relevant information data systems are still presented separately, the homogenization and integration data has still to be made by the users.

Application Integration. This approach uses applications of integration that access different data sources and return results integrated to the user. This solution is practical for a small number of systems. However, the applications grow even more as the number of system interfaces and data formats to mix and integrate.

Middleware Integration. Middleware provides reusable functionality that is commonly used to resolve issues dedicated to the integration problem, for example, as is done by SQL middleware. While applications are exempted to provide common functionality for integration, integration efforts are still required. In addition, different middleware tools usually have to be combined to build integrated systems.

Uniform Data Access. In this case, a logical data integration is performed at the level of data access. Global applications are provided with an unified global view of physically distributed data, although only the virtual data available at this level. However, the overall supply of physically integrated data may be delayed because the data access, integration and homogenization have to be done at the runtime. Portals and Systems Database Federated are examples of applications that use this approach to integration.

Common Storage of Data. In this approach, the integration of physical data is accomplished by the transfer of data to a new data storage, local sources can be reformed or they remain operating. In general, physical data integration provides quick access to data. However, if the local data sources are retired, the applications that access them should be migrated to new storage. In the case of local data sources remain operating, the periodic updating of the common data store should be considered. Data warehouses are examples of this approach.

Over time, several solutions have been adopted in an attempt to perform the systems integration (Fig. 2).

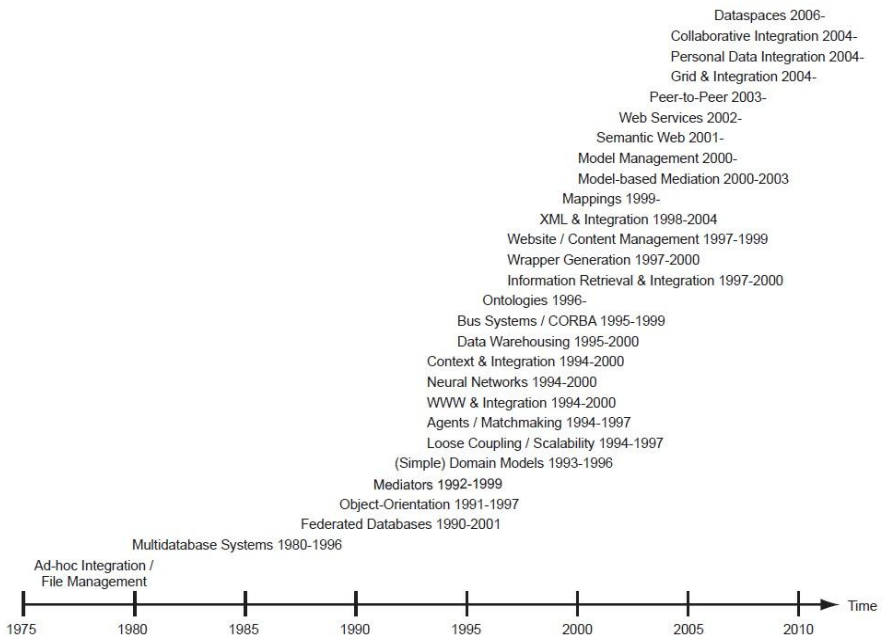

Fig. 2. Data Integration Trends over time. Figure available in [2].

Under the technological aspect of systems integration some solutions are outlined in Zavalik [1], which also points out to the use of Web services as a way to fully integrate heterogeneous environments. The Web Services technology is a possibility of increasing the interoperability of systems using open standards.

2.1 *Web Services*

Web services represent a model that aims the integrating different computer systems. According to the W3C (World Wide Web Consortium), a Web Service is defined as "a software system designed to support interoperability between machines on the network." Through it is possible to establish communication between two applications by exchanging XML messages. This model has proven highly viable for extraction and data integration, enabling interoperability on a large scale [4] [5] [6]. With the adoption of standards administered by the W3C and OASIS (Organization for the Advancement of Structured Information Standards), it is possible that applications developed at different times, with different technologies, interact in a variety of platforms.

Basically, Web Services are based at standards as HTTP (Hypertext Transfer Protocol), SOAP (Simple Object Access Protocol), XML (Extensible Markup Language), WSDL (Web Services Description Language) and UDDI (Universal Description, Discovery and Integration). The HTTP protocol is responsible for transporting data over the network. You can use also the HTTPS protocol to establish safe connection. A Web Service is always identified by a URI (Uniform Resource Identifier). Data is transferred in XML format, encapsulated by SOAP protocol. The SOAP standard is based on the concept of RPC (Remote Call Procedure). It performs, therefore, the invocation of a method, with its name and its arguments. The procedure call is formatted in XML and sent over HTTP. This allows you to overcome any firewall restrictions. WSDL is a pattern for describing a Web Service and how it can be accessed. It corresponds to a document written in XML that contains data types, messages, operations and communications protocols used in a Web Service. UDDI is a pattern developed for organizing and registering Web services. It corresponds to an enterprise being developed within the industrial consortium UDDI, originally sponsored by IBM, Microsoft and Arriba, in order to accelerate the interoperability and use of Web Services, by the proposal of a service registering names of organizations and service description. UDDI uses WSDL descriptions from a lot of public services.

Figure 3 summarizes the technologies used in the structure of a Web Service. For the representation and structuring of data in messages received/sent the XML is used. Calls to operations, including input/output parameters, are encoded in SOAP protocol. The available services are described using WSDL language and are advertised through the UDDI.

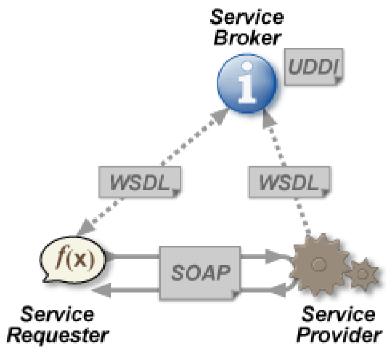

Fig. 3. Web Services Technologies. Created by H. Voorman[1].

3 Integration of LORs to LMSs

Learning Objects (LOs) are suitable digital resources to support learning as they were exploiting the multimedia content, and they encourage interactivity and be built in small modules, allowing students to build knowledge in small portions. Generally, such resources are stored in repositories by developers, called LOR - Learning Objects Repository, which allow you query and retrieval of resources through inspecting the metadata associated with them. However, the environment most used to

[1] Source: http://en.wikipedia.org/wiki/File:Webservices.png, Autor: H. Voormann
License: Creative Commons Attribution 3.0 Unported

provide the LO within a pedagogical context is the Learning Management System (LMS). It's a necessity, therefore, that the LOR be integrated with LMS in order to facilitate the teachers work in the integration of LOs in their plan of teaching and learning [7]. In a preliminary case study [8] showed that the integration of systems for the teacher, especially the LMS to LOR, facilitates its work in a way that he has available in a single environment the main tools to build its teaching and learning plan as well as provide resources for students involved in the learning unit [9].

The required communication between LOR and LMS can be provided through the integration by Web Services, which performs the integration through software components (services by the Internet) that support machine-machine interaction over a network through messages based XML protocols that are transmitted by the Internet. Depending on the functionality of integration offered, web services represent a uniform approach to data access or a common interface to access data for later integration manual or application-based integration. In this case, the approaches used were the standardization of data access integration followed by an application-based integration. The integration architecture designed and implemented is presented in Figure 4.

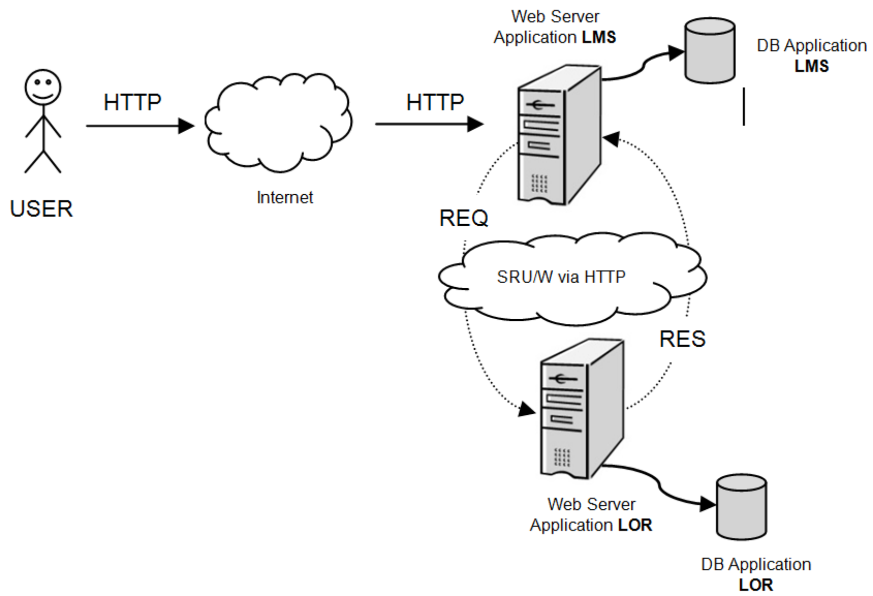

Fig. 4. Integration Architecture using Web Services Strategy

User access is done from a plugin. The request is made to the web server where it's located the LMS. The LOR can also individually be accessed by following the same procedures mentioned. The communication between systems is facilitated by the protocols SRU/W.

About the accomplishment of architecture functionality tests were used Learning Objects Repository CESTA2 and Learning Management System MODDLE. Collection of Entities Support the use of Technology in Learning (CESTA) is a project developed by the Interdisciplinary Center of New Technologies in Education (CINTED) at Federal University of Rio Grande do Sul - Brazil, which in its first version was classified as Referatory, which stores only metadata and the addresses of where the files are each object. In order to expand the possibilities of this Referatory and prevent the loss of access to resources caused by possible changes in local storage facilities, the CESTA had to store the learning objects also become a repository and it's the called CESTA2 [7]. The implementation of the repository CESTA2 used as base the Content Management System DSpace[2].

The communication process between client and server is mediated by the transport protocol HTTP. The client through an application sends a query expressed in CQL (Contextual Query Language) with default parameters to the server. The server processes the parameters and sends the query to the database, which returns the found records to the server. The protocol SRU/W formats the output in XML and displays the response to the client. The interaction of the conceptual model from the protocol SRU/W can be understood by the sequence of steps illustrated in Figure 5 [10]:

1. Client Application makes the request to the SRU/W.
2. SRU/W makes the request and is transported over HTTP.
3. HTTP is a request to the SRU/W server.
4. SRU/W server interacts with the search engine to formulate the answer.
5. The response is returned by using HTTP.
6. HTTP is the answer to the SRU/W client.
7. SRU/W client presents the result for the Client Application.

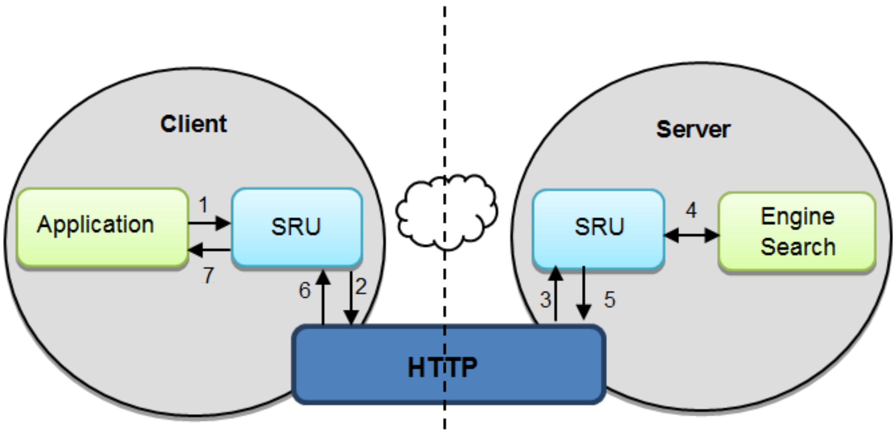

Fig. 5. Architecture Communication Protocol SRU/W. Adapted from [10].

[2] http://www.dspace.org/

An initial implementation of integration was performed with version 1.9 of MOODLE using MrCute module version 2, which already integrates MOODLE to external repository Jorum[3] and enables the creation of an internal repository in the local installation of MOODLE, allowing a stored OA this is displayed for all courses maintained in this installation. At the end of 2010, when released version 2 of MOODLE, there was another option to add the LOs in LMS MOODLE, from repositories integrated into it. This integration is performed from specific plugins for each repository. Currently, repositories such as Merlot, Facebook, YouTube, Picasa, GoogleDocs, Alfresco, Box.net, Legacy Course Files, Amazon S3, URL Downloader, WebDAV and Wikimedia are already integrated in the standard version of MOODLE 2.

Tests were performed with version 2.0 of MOODLE, which showed that the repository MrCute requires changes to continue operating in this new version of the environment. Before we continue with the implementation of the MOODLE version 2 to CESTA2, an analysis to determine the best strategy for achieving such integration is: to continue the integration from the module MrCute or build a new plugin. The analysis allowed us to observe that the main features of the module MrCute, as the ability to store LOs at IMS format into MOODLE local repository and retrieval of LOs contained in an external repository, already built-in structural applications in the default implementation of version 2 of MOODLE, without the need to install any additional module.

Additionally, the second version of MOODLE makes it easy to integrate with other external repositories through an Application Program Interface (API), which provides classes and methods for the construction of new plugins. For these reasons, it was realized that the most interesting was to invest in the implementation of a plugin for MOODLE version 2, which allows the connection to repositories for which was already available a Web service interface, such as DSpace, which is a repository of commonly used today. This would facilitate an integration that can be used by any repository not only implemented but also with the DSpace with other types of repositories, provided they use the protocol SRU (Search/Retrieve via URL) and the SRW (Search/Retrieve Web Services) as protocols communication, with a few adjustments in the MOODLE standard version.

The first steps in implementing this integration solution have already been taken. The results show that the integration of two systems, CESTA2 and MOODLE, using technology based on Web services and communication protocols, especially the SRU/W, allowed them to communicate. The evidence drawn from the capture of network traffic (Figure 6) show that communication is carried out successfully. The arrest was made at the time a query was made to the repository CESTA2 through the plugin implemented this solution integrated into the LMS Moodle.

[3] http://www.jorum.ac.uk/

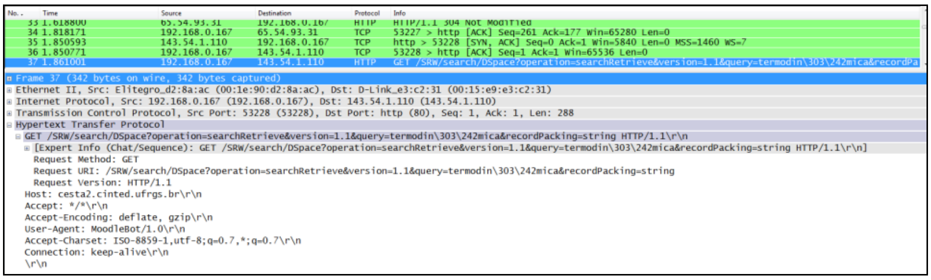

Fig. 6. Package details captured at query time to the repository CESTA2.

4 Final Considerations

The use of an integration solution between a Learning Objects Repository and a Learning Management System using Web Services was a solution that has continued an investigation of different strategies for integrating systems and started with the design and implementation a federation of learning repositories [11] that collects and consolidates several metadata repositories in the country. In this federation, the collection of metadata is done through the use of OAI-PMH on a regular and automatic, but at present, the access to metadata only can be done by end users through a web-based interface.

The flexibility of this solution, which combines the use of an internal repository for MOODLE, with access to external repositories was very suitable, because addresses the issue of access to a generic collection, containing learning objects. This solution allows teachers-authors to work with courses by using only the interfaces they already know, that is, those that exist in MOODLE itself.

References

1. Zavalik, C. Integração de Sistemas de Informação através de Web Services. Dissertação de Mestrado. PPGC/UFRGS. Porto Alegre. 2004.
2. Ziegler, P., Dittrich, K. R.: Data Integration? Problems, Approaches, and Perspectives., Conceptual Modelling in Information Systems Engineering, Editor(s): John Krogstie In, Andreas Opdahl, Sjaak Brinkkemper; 2007, Springer. Proceedings… Available in: <http://www.ifi.uzh.ch/pax/uploads/pdf/publication/428/ZieglerSolvberg2007.pdf>. Access in: 15 apr. 2011.
3. Ziegler, P.; Dittrich, K. R. Three Decades of Data Integration - All Problems Solved?, 18th IFIP World Computer Congress (WCC 2004), Volume 12, Building the Information Society 2004, Kluwer Academic Publishers. Available in: <http://www.ifi.uzh.ch/pax/uploads/pdf/publication/869/ZieglerWCC2004.pdf>. Access in: 15 apr. 2011
4. Hansen, M., Madnick, S., and Siegel, M. Data integration using Web services. In Lecture Notes in Computer Science 2590. Springer, 2003,165–182.
5. Abiteboul, Serge, Benjelloun, Omar, and Milo, Tova (2002). Web Services and Data Integration. In Ling, Tok Wang, Dayal, Umeshwar, Bertino, Elisa, Ng, Wee Keong, and Goh, Angela, editors, *Third International Conference on Web Information Systems*

Engineering (WISE 2002), pages 3–7, Singapore, December 12–14. IEEE Computer Society.
6. Benbernou, Salima and Hacid, Mohand-Said. 2006. Dynamic Web Service Calls for Data Integration. Journal of Software, Vol 1, N.1 Page(s) 1-10 July 2006. Available in: <http://www.academypublisher.com/jsw/vol01/no01/jsw01010110.pdf>. Access in: 15 jul. 2011.
7. Tarouco, L. M. R.; Schmitt, M. A. R. Integração do Moodle com Repositórios Abertos. MoodleMoot, 2010.
8. Rodrigues, A. P.; Konrath, M. L. P.; Tarouco, L. M. R. Integração de Repositórios à Ambientes Virtuais de Aprendizagem. In: V Congresso Latino-Americano de Objetos de Aprendizagem, 2010, São Paulo. 2010.
9. Leal, J. P., & Queirós, R. (2010). Integration of repositories in eLearning systems. Paper presented at the ICEIS 2010 - Proceedings of the 12th International Conference on Enterprise Information Systems, 1 DISI 127-132.
10. SRU: Search/Retrieval via URL – SRU, CQL and ZeeRex (Standards, Library of Congress). Available in: <http://www.loc.gov/standards/sru/>. Access in: 25 abr. 2011.
11. FEB. Federação Educa Brasil. Available in: <http://feb.ufrgs.br>. Access in: 3 jun. 2010.