

A Web Services Broker for E-learning

Judith Rodríguez, Luis Anido, Manuel J. Fernández, Fernando Mikic, and
Luis M. Álvarez

Departamento de Ingeniería Telemática
Universidad de Vigo
Campus Universitario s/n, E36200 Vigo (Spain)
{jestevez,lanido,manolo,mikic,lmsabu}@det.uvigo.es

Abstract. Brokerage enriches the traditional searching services to include resource acquisition, distribution and billing and perfectly adapts to a scenario like e-learning where multiple, independent content providers and service providers exist. Brokerage offers a suitable framework where customers, mainly learners, can get in touch with providers. On the one hand, customers can locate, select and access the resources and services offered by providers in an efficient way. On the other hand, providers can benefit from services as marketing, customizable publicizing, or even accounting and billing. This paper proposes a brokerage architecture for Web-based E-learning services. It facilitates the integration of individual software components providing specific services to compose a complete E-learning environment.

1 Introduction

The application of Internet technologies to the learning field is one of the most prolific research areas during the last years. There are many educational resources and software platforms available through the Web. In this situation, heterogeneous proprietary data models and platforms have coexisted for years. As a consequence an interoperability and integration problem came up. To overcome this situation, a standardization process for E-learning technologies started several years ago. This process has produced promising results and several proposals has already become *de-facto* standards among e-learning developers community [1]. Most of these standards offer suitable data models to exchange and transfer information among heterogeneous systems. Nowadays, the trend among e-learning standardization community is to offer well-defined interfaces to access common services. For this, Web services are being widely used as the underlying support.

The Web services concept has strongly emerged as a new distributed computing paradigm that attempts to make use of existing Internet technology overcoming the drawbacks of other distributed technologies such as DCOM [2] or CORBA [3]. The E-learning field is also benefiting from this situation and several e-learning institutions and companies are planning to deliver their services using Web services technologies [4].

As the offer of available Web services increases it will become more and more difficult to find or even use them. An approach to overcome this situation is to apply the electronic brokerage concept to E-learning Web services. For this, we take advantage both of the e-learning standardization results and the advances in the Web services technology. This paper proposes a brokerage platform that facilitates service discovery, invocation and composition. This paper is organized as follows. Section 2 explains the concept of Web services and present the technologies involved. Section 3 is devoted to the presentation of our proposal. Finally, Section 4 presents the conclusions of our work and outlines the future trends.

2 Web Services

Web services constitute a distributed computer architecture made up of many different computers trying to communicate over the network to form one system. This area has been, and continues to be, one of the most prolific research fields, as Web services represent the latest in the evolution of modular and distributed computing.

The W3C [5] defines a Web Service as *“a software system identified by a URI, whose public interfaces and bindings are defined and described using XML. Its definition can be discovered by other software systems. These systems may then interact with the Web service in a manner prescribed by its definition, using XML based messages conveyed by Internet protocols”*. However, a more accurate and general definition of what Web Services are is that proposed by the Stencil Group [6]:

[Web Services are] Loosely coupled, reusable software components that semantically encapsulate discrete functionality and are distributed and programmatically accessible over standard Internet protocols.

Ongoing work in this field comprises the identification and definition of a set of standards and protocols (see section 2.1) for Web service management. The purpose of this work is twofold. On the one side, the new standards will allow to semantically encode the properties, capabilities, interfaces and effects of Web Services in an unambiguous and machine-understandable way in order to automate tasks as service discovery, execution and composition. On the other side, the agreement on the protocols and data models will promote interoperation of services.

E-business in general, and E-learning in particular, is an application domain where Web service technologies will suppose a step ahead. The work discussed in this paper aims to define an open brokerage architecture to integrate educational-related services. Next, we introduce the main building blocks of the semantic Web services, namely ontologies and the standards involved. Then, we discuss our proposal for a brokerage platform for educational services based on ongoing research on Web service technology. This paper concludes with a summary of the main contributions and a description of our present and future work.

2.1 Web Services Technology

Standardization work in the field of Web Services produced a set of standards that enable developers to implement distributed applications, using different tools provided by many different vendors. These applications use a combination of software modules from systems in other departments or even other companies. The most outstanding standards are organized as a layered structure, and are briefly described next. From the lower to the highest layer we have:

- XML (*eXtensible Markup Language*) [7] is the ubiquitous language for the creation of different information formats and the exchange of structured information among software applications.
- SOAP (*Simple Object Access Protocol*) [8] is a protocol for message-based communications among applications. It is based on XML and uses mainstream Internet transport protocols (*e.g.* HTTP) to carry its data.
- WSDL (*Web Services Description Language*) [9] developed by the W3C. It is an XML-based format to abstractly describe the functionality (*i.e.* the interface) offered by a Web Service. WSDL separates the description of the service itself from the concrete details of *how* and *where* its functionality is offered.
- UDDI (*Universal Description Discovery and Integration*) [10] defines a protocol and a registry for describing available Web service components to allow software to discover what Web services are available and how to access them. The information provided by UDDI registries consists of three conceptual components: (1) *white pages* of service provider contact information; (2) *yellow pages* that categorize services by standard taxonomies; and (3) *green pages* that document the technical information about the services.
- WSFL (*Web Services Flow Language*) [11], from IBM, provides a framework to describe the business logic required to assemble various services into and end-to-end business process.

Although there are a lot of ongoing proposals in the Web services field (*i.e.* SOAP extensions, WXSLS) the ones enumerated above are becoming *de-facto* standards among researchers/developers in this field.

3 Brokerage Platform for Educational Services

Our work aims at the identification of a set of common services for a Web services brokerage platform in the e-learning domain. Figure 1 graphically outlines a description of the environment of our work.

The purpose of the brokerage platform is to provide a common place where customers, mainly learners, can benefit from the large offer of educational services without having a prior knowledge about where these services are running or how to *use* them.

Next, we introduce the different *actors* involved in the educational Web services brokerage platform and we clarify their interactions through the explanation of several use cases.

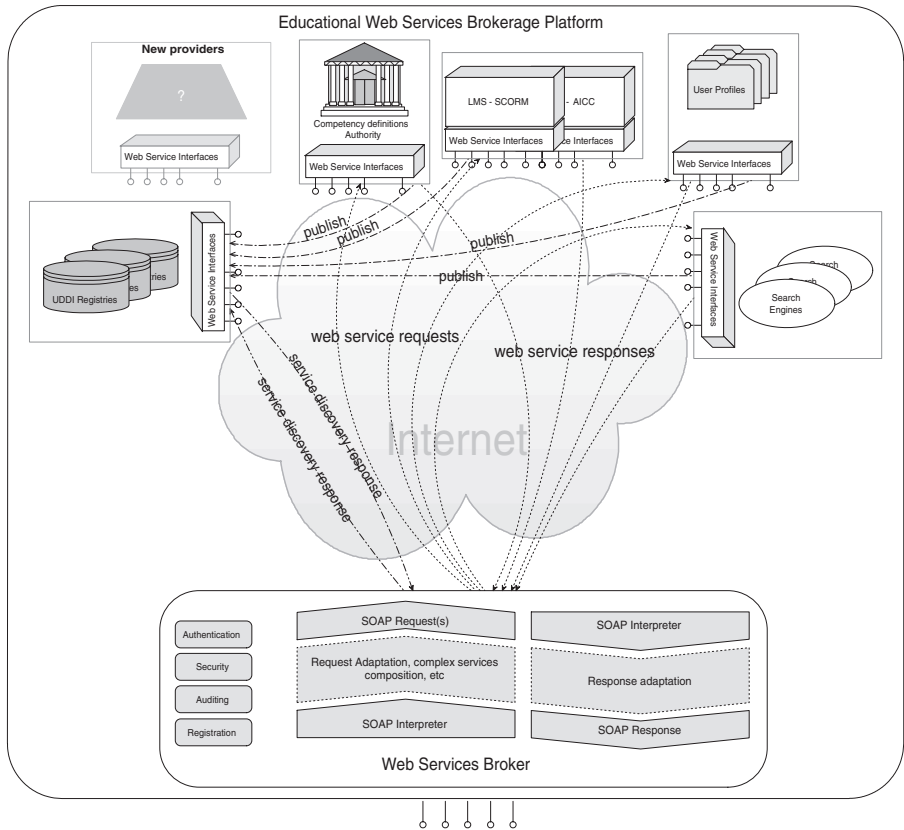


Fig. 1. Educational Brokerage Proposal

3.1 Service Providers

Service providers are those institutions or individuals that made public their services through Internet. Service providers can be categorized into two different groups. On the one hand, there are providers that are strictly linked to our educational environment:

- *Learning Management Systems (LMS)*. They represent the proprietary educational content delivery platforms of different institutions. Typically, they are compliant with well-know recommendations in this field like SCORM [12] or CMI [13].
- *Competency Definitions Authorities*. They are institutions responsible for defining and certifying the skills, knowledge, tasks, learning outcomes and curriculum to achieve a specific expertise. Usually, the information managed by these institutions follows standardized specifications like the *IMS Reusable Definition of Competency or Educational Objective* [14].

On the other hand, there are service providers that offer *general* services. Nevertheless, we will only focus on those managing data models relevant for our purposes.

- *User Profiles Servers*. They offer services over repositories that store personal and academic information about final customers, *i.e.* learners. Several information models may be used by this kind of providers, like LIP [15], PAPI [16] or vCard [17][18].
- *Search Engines*. They provide search services over Internet resources. They can be either general or educational-specific. In any case, we will only be interested in those that can “understand” educational metadata information models like LOM [19].
- *UDDI Registries*. As explained in section 2.1, they provide information about service providers and technical information about services offered by them. In addition, they offer discovery services over that information.

3.2 Web Services Broker

The *Broker* is the core component of the brokerage platform. It is responsible for receiving customers’ requests, fulfil them and retrieve the corresponding results to the customer. As it can be inferred from the discussion above (c.f. Sect.3.1), there are many information models of which customers should be aware in case the *Broker* did not exist. The *Broker* acts as the intermediary in all the transactions between customers and service providers, making transparent details like the data models managed, the location of service providers or the flow of exchanged messages to achieve the response of one customer request. Thus, the tasks for which the *Broker* is responsible include:

1. Receive and process customers’ request messages.
2. According to previous step, the *Broker* may proceed with one or more of the steps below:
 - Invoke UDDI registry services to discover the most suitable service or services according to the customer request.
 - Compose a flow of services over different providers to fulfil the request.
 - Make the required data model transformations (*i.e. mappings*) to invoke the suitable services.
 - Access information about users to adapt the request processing to customer’ preferences or capabilities.
 - Request customer interaction if necessary to complete the process.
3. Retrieve the results from the previous step, make the needed adaptations to return them to the customer, or inform the customer about possible errors resulting from request processing.

3.3 Customers

In our case, final customers are students searching for educational services and resources to achieve specific capabilities, skills or fulfil the curricula requirements. Nevertheless, the role of *Broker* customers will be played by the educational software used by the *Learners*. To make reading easier, from here on we will use the term *Learner* to refer to this *Learner* software.

3.4 Interaction among Actors

Figure 1 shows graphically the interactions between actors as dotted and dashed-dotted arrows. Arrows marked as *publish* reflect the interactions between the UDDI registry and the remainder service providers to make publicly available their services and to support their discovery. To carry out this task, service providers describe their services interfaces using WSDL and supply these descriptions to UDDI registries.

Those arrows marked with *service discovery* show the interactions between the *Broker* and the service providers in order to discover suitable Web services and how and where to invoke them.

Finally, the arrows marked with *Web services responses/requests* reflect the interactions among the *Broker* and the different service providers in order to fulfil customers' requests.

All these interactions are made through Internet and represent the exchange of SOAP messages over HTTP.

3.5 Usage Case Scenarios

In this section we describe two possible scenarios.

Usage scenario 1. In this scenario the *Learner* asks for suitable courses to obtain a specific competency (*e.g.* computational science algorithms). Figure 2 shows the hypothetical flow of messages among actors.

Firstly, the *Learner* invokes the appropriate *Broker* service (*searchForCertificatedExpertise*). The SOAP request message would carry information such as the *Learner* identifier, the subject of interest, etc. Once the *Broker* receives the request, it would process it. The requested service is composite, *i.e.* it is composed of a set of services whose invocation flow is specified using WSFL. In this case, to carry out the requested service the *Broker* must do the processing explained next¹:

1. Obtain the suitable competency definitions from the requested subject using the services offered by *Competency Definitions Authority*.

¹ To simplify the usage scenario explanation, it is supposed that the *Broker* has already discovered the necessary Web services in the *UDDI Registries*

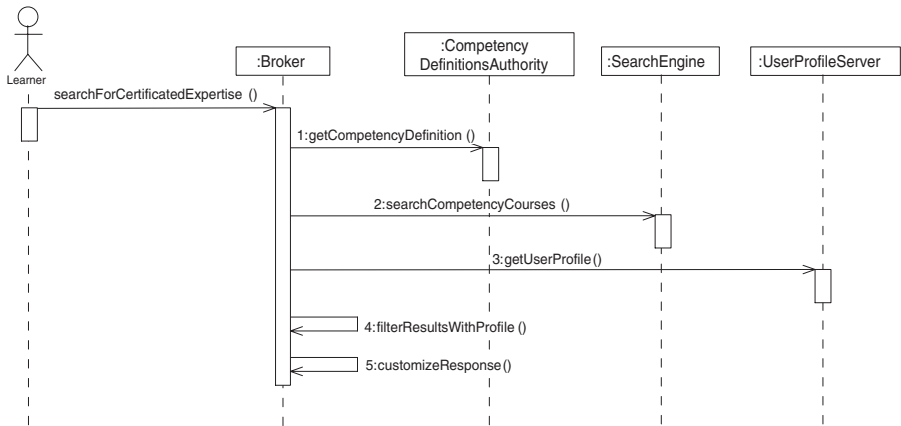


Fig. 2. Flow of messages for the usage scenario 1

2. Compose the suitable message request to invoke the *Search Engine* services to get a list of educational courses that fulfil the competency definitions obtained in the previous step. We assume that there exists a specialized *Search Engine* that is able to search suitable courses for a specific definition of competencies.
3. Access to academic and preferences *Learner* information using *User Profile* services. The user profile contains the degree level of the *Learner*, completed courses, preferences as preferred data models for information exchange, or personal information required to complete the enrolment.
4. Use the information gathered from the previous step to filter out the results from step 2. For example, the *Broker* can remove from the list those courses that are not suitable for the *Learner* either as they are too basic or too complex for the *Learner* degree level.
5. Customize the response message for the *Learner* based on presentation preferences established in the profile.

Usage scenario 2. This scenario describes a situation where the *Learner* has made the choice of both the course and the institution to follow that course. Then, he requests the *Broker* to perform the enrolment in the suitable LMS on his behalf (*enrolInCourse*). Figure 3 shows the hypothetical flow of messages caused by the processing of the request message in the *Broker*. As in the previous case, the definition of the requested service states that it implies the invocation of several services whose *choreography* is defined in WSFL. Needed steps are shown below:

1. Get the WSFL definition for the suitable service for the enrolment in the requested institution. For this task the *Broker* would use the services provided by the *UDDI registries*.

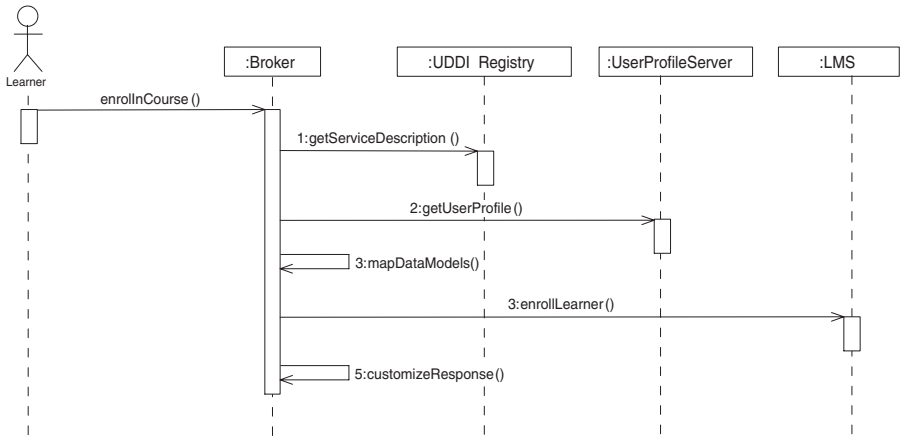


Fig. 3. Flow of messages for the usage scenario 2

2. Obtain the personal information of the Learner using the *User Profiles* services. This information allows to automatically providing the information necessary to make effectively the enrolment without Learner intervention.
3. In this case we suppose that the information model used by the *User Profile* (e.g. LIP) is different from the one used by the enrolment service of the LMS (e.g. PAPI). For that reason the *Broker* must map the two information models.
4. The *Broker* sends the enrolment request message to the LMS.
5. Once the previous steps are completed, the *Broker* customizes the message response from the execution of the enrolment service and sends it back to the Learner. This response message would contain for example the login, password and URL to access the course.

4 Conclusions and Future Trends

At present the Web services technology is one of the most active working areas. Therefore, many related proposals have been developed in parallel by the most outstanding institutions and organizations in the Web development area. E-learning is one of the domains that is benefiting from this technical development. Nevertheless, new problems appear as a consequence: which is the most suitable service? How can I locate it? How should I use it?

In this paper we have presented a Web services *Broker* for the E-learning domain that acts on behalf of *Learners* to discover and invoke services offered by different services providers. In this way, the *Broker* makes transparent to the user the underlying processing, the used protocols and infrastructure and the location where Web services are running.

Future trends aimed to enrich the Web service brokerage platform with additional features. Among them, we are planning to introduce provided-oriented services as support for billing based on service usage or marketing-related services. We are also studying the introduction of push services for providers. Presently, the proposed platform only supports pull services. However, at this time, the need for push services in this kind of platform is not clear for our application domain.

Additionally, we also plan to provide notification mechanisms for customers. Notification will complement service discovery in the sense that, instead of being the customer who initiates the search for services, the broker will inform users about available services, according to the user's profile.

Acknowledgments. We want to thank “Ministerio de Ciencia y Tecnología” for their partial support to this work under grant “CORBALearn: Interfaz de Dominio guiada por Estándares para Aprendizaje Electrónico” (TIC2001-3767).

References

1. Santos, J., Caeiro, M., Rodríguez, J., Anido, L.: Standardization in Tele-learning. A Critical Analysis. In: Tele-Learning. The Challenge for the Third Millennium. 17th IFIP World Computer Congress, Montreal (Canada). Kluwer Academic Publishers (2002) 321–328
2. Microsoft Cooperation: DCOM Technical Overview (1996) Electronic version available at http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dndcom/html/msdn_dcomtec.asp.
3. OMG: Common Object Request Broker Architecture: Core Specification (2002) Electronic version available at <http://cgi.omg.org/docs/formal/02-12-06.pdf>.
4. Sun Microsystems: Universities and Solution Providers Showcase the Use of Sun Technology at this Year's EDUCAUSE Conference (2002) Electronic version available at <http://www.sun.com/smi/Press/sunflash/2002-10/sunflash.20021002.3.html>.
5. Champion, M., Ferris, C., Newcomer, E., Orchard, D.: Web Services Architecture. W3C working draft, W3C (2002) Electronic version available at <http://www.w3.org/TR/2002/WD-ws-arch-20021114>.
6. Stencil Group: Defining Web Services. Analysis memo (2001) Electronic version available at http://www.stencilgroup.com/ideas_scope.200106wsdefined.html.
7. Bray, T., Paoli, J., Sperberg-McQueen, C., Maler, E.: Extensible Markup Language (XML) 1.0 (Second Edition). W3C Recommendation 6, W3C (2000) Electronic version available at <http://www.w3.org/TR/2000/REC-xml-20001006>.
8. Gudgin, M., Hadley, M., Mendelsohn, N., Moreau, J.J., Nielsen, H.F.: SOAP Version 1.2 Part 1: Messaging Framework. Candidate recommendation, W3C (2002) Electronic version available at <http://www.w3.org/TR/soap12-part1>.
9. Chinnici, R., Gudgin, M., Moreau, J., Weerawarana, S.: Web Services Description Language (WSDL). Working Draft Version 1.2, W3C (2002) Electronic version available at <http://www.w3.org/TR/2002/WD-wsdl12-20020709/>.

10. Bellwood, T., et al.: UDDI (Universal Description, Discovery & Integration). Published Specification Version 3, Accenture, Ariba, Inc., Commerce One, Inc., Fujitsu Limited, Hewlett-Packard Company, i2 Technologies, Inc., Intel Corporation, International Business Machines Corporation, Microsoft Corporation, Oracle Corporation, SAP AG, Sun Microsystems, Inc., and VeriSign, Inc (2002) Electronic version available at <http://www.uddi.org/pubs/uddi-v3.00-published-20020719.htm>.
11. Leyman, F.: Web Services Flow Language (WSFL 1.0). Technical Report Version 1.0, IBM Software Group (2001) Electronic version available at <http://www-3.ibm.com/software/solutions/webservices/pdf/WSFL.pdf>.
12. P. Dodds (Editor): Sharable Content Object Reference Model (SCORM). The SCORM Runtime Environment. Technical Report Version 1.2, Advanced Distributed Learning Initiative (2001) Electronic version available at http://www.adlnet.org/ADLDOCS/Documents/SCORM_1.2_RunTimeEnv.pdf.
13. Hyde, J.: CMI Guidelines for Interoperability. Technical Report Version 3.5, AICC CMI Subcommittee (2001) Electronic version available at <http://www.aicc.org/docs/tech/cmi001v3-5.pdf>.
14. Cooper, A., Ostyn, C.: IMS Reusable Definition of Competency or Educational Objective - Information Model. Final Specification Version 1.0, IMS Global Learning Consortium (2002) Electronic version available at http://www.imsglobal.org/competencies/rdceov1p0/imsrdceo_infov1p0.html.
15. Smythe, C., Tansey, F., Robson, R.: IMS Learner Information Package Information Model Specification. Final Specification Version 1.0, IMS Global Learning Consortium (2002) Electronic version available at <http://www.imsglobal.org/profiles/lipinfo01.html>.
16. Collett, M., Farance, F.: Draft Standard for Learning Technology Public and Private Information (PAPI) for Learners (PAPI Learner). Technical Report Version 7, IEEE LTSC (2000) Electronic version available at http://edutool.com/papi/drafts/07/papi_learner_07_main.pdf.
17. Howes, T., Smith, M., Dawson, F.: A MIME Content-Type for Directory Information. RFC 2425, IETF, Network Working Group (1998) Electronic version available at <http://www.imc.org/rfc2425>.
18. Dawson, F., Howes, T.: vCard MIME Directory Profile. RFC 2426, IETF, Network Working Group (1998) Electronic version available at <http://www.imc.org/rfc2426>.
19. Hodgins, W.: Draft Standard for Learning Object Metadata. Draft 6.4, IEEE Learning Technologies Standards Committee (2002) Electronic version available at http://ltsc.ieee.org/doc/wg12/LOM_WD6.4.pdf.