Journal of Web Engineering, Vol. 4, No. 3 (2005) 263-278

SHARING LEARNER INFORMATION THROUGH A WEB SERVICES-BASED LEARNING ARCHITECTURE

DANIELA LEAL MUSA

Federal University of Rio Grande do Sul - Institute of Informatics
musa@inf.ufrgs.br

JOSÉ PALAZZO MOREIRA DE OLIVEIRA

Federal University of Rio Grande do Sul - Institute of Informatics
palazzo@inf.ufrgs.br

Received October 30, 2004
Revised March 22, 2005

This paper introduces the architecture developed for the exchange of learners model information among e-learning systems in the AdaptWeb Project. This Web-learning environment offers an adaptive content associated with a particular student's profile. Hypermedia teaching applications may explore different educational strategies and tactics, including guided navigation, hierarchical contents presentation, examples, exercises, and so on. Also the teaching task can be optimized if the teacher prepares a content material that can be shared for different target public. AdaptWeb was designed concerning these aspects and allows the generation of different presentations from an ample learning material developed for a specific discipline based on the learner model. The Web Service technology is used as it yields an easy communication between Web-applications through the HTTS protocol allowing also secure personal data interchange. Our goal in this work is to provide a standard communication protocol that makes possible different e-learning systems cooperate in order to gather a set of learner model information, richer than that found in a standalone e-learning system. As result, the course content, in a compliant federated e-learning system may be better adapted and presented to students, according to each student's program, cognitive characteristics, and navigation preferences. Once the student profile is determined all the courses will be consistently offered by the e-learning systems using the same shared profile.

Key words: Web services, learner model, adaptability, AdaptWeb, e-learning

1 Introduction

Due to the Internet worldwide distribution, a huge research is being developed in the use of the Web technology to create universally accessible e-learning systems. Such systems deal with important management issues related to learning content, learner profile information, and learning resources in general. In this paper we are particularly interested in investigating how learners' information can be shared among different e-learning systems over the Internet.

Often, new e-learners are required to update their personal information before proceeding in one of the offered courses. Some systems are able to adapt its course content presentation using some techniques that discover the learner's preference [1], level of previous knowledge [2], and cognitive style [3]. However, this important learner information is not shared among different e-learning system, which may forces the learner to fill cumbersome forms in each new system, and also forces each new system to analyze and process behavioral information of each new learner. Current systems do not collaborate to enrich the information related to users of several different e-learning systems. With richer learner information shared among a federation of e-learning systems a better course content adaptation may be achieved.

Currently, international bodies, such as IEEE, IMS, and ADL, are working on the standardization of e-learning information defining metadata for learning objects, learner model, and learner assessment among others [4,5]. For example, LIP (IMS Learner Information Package) [5] allows the description of learners based on personal information, interests, or activities. However, LIP does not define any method to exchange such information among different e-learning systems. We developed a Web Service solution to exchange learner information following a learner model. The use of Web Services is justified as they provide a well-defined mechanism that set up inter-application communication over the Web. We expect that a federated e-learning system will implement a Web Service that will be able to export the internal learners information. Thus, other systems that require learner information are able to retrieve this information accessing the Web Service implemented by the e-learning system that holds such kind of information about the student. The federated e-learning systems will be able to share a larger amount of information on the cognitive learner's abilities. It is clear that some legal citizen privacy defense mechanism, specific for the e-learning environment, must be offered before such a proposal may be accepted. This last point is being under investigation in the area of e-commerce and standards for safe and authorized personal data interchange are available. It is clear that the sensibility to personal data interchange and publicity vary a lot from country to country as Orkut [6] demonstrated.

An e-learning system may store the learner's personal data while other e-learning system stores some information about the cognitive style of the same learner and a third-one store data on the previous acquired abilities in some specific training. Using Web Services it is possible to search for learners profile federated data (personal data + cognitive style) in each of the participant system in the environment. Any participant system may use this information to improve its learner model. To make this exchange possible, an interchange standard must be created to identify data through different participants, that is, a learning environment that makes with all federation participants can process in the same way the fragments of stored information. In this work the information in the learner model is composed of two parts: information supplied by the learner and information collected through the learner's behavior when using the e-learning system. The more data the system obtains on the learner behavior, the more personalized functionalities it can offer. Based on this view, the learner model here depicted follows the idea developed at the SeLeNe [7] project, which joins some of the categories defined by the PAPI [4] standard with some of the LIP [5] standard categories. The reason for this union is the fact that some categories are included and richer in one specific standard while other categories are better detailed in the other. In addition, for our purposes, some extremely important information for the adaptation work is not contemplated in any standard such as the cognitive style and the learning style of each learner. We included these characteristics in the learner model. It is clear that all proposals in this area must be discussed and submitted to experimental evaluation before a wide accepted standard may be consolidated.

This paper is organized as follows: Section 2 presents the main learner model standards existent. This section also provided a short presentation of related work and describes the Web Service technology. Section 3 describes the system architecture and the learner model used. Section 4 concludes the article and presents the plans for future research.

2 Background

This section presents the definitions and standards associated with this work. The Web Services technology used in this work for communication among e-learning systems is described in section 2.1. Section 2.2 introduces the existing learner model standards. Section 2.3 describes research projects that present solutions for the sharing of information between different e-learning systems, which are basically concerned about the reusability of the educative material.

2.1 Web Services

The Web Services technology [8] can be described as an architecture that has platform-independent components, yielding applications interoperability. For this reason, Web Services are being used for exchange of data and messages between applications through a Web protocol-based infrastructure, such as HTTP, SMTP and FTP.

W3C has some groups that deal with the standardization of Web Services and other related technologies such as SOAP, WSDL and UDDI [8]. The SOAP (Simple Object Access Protocol) protocol defines a way of communication between applications very similar to the RPC (Remote Procedure Call), however, with the ability to move through different administrative domains. SOAP basically encapsulates a procedure call into an XML structured request and returns the execution result. The procedure call, the data passed as parameter and the return value are also structured in a textual format through XML.

WSDL (Web Service Description Language) [8] is a standard that describes a Web Service through XML, making possible that client applications access and validate the Web Service in a well-defined way. Besides, it is possible to publish the description of a Web Service. Once the description is published, the applications can look for it dynamically, download the description and then create a client in execution time. The UDDI (Universal Description, Discovery and Integration) [8] standard is one of the most frequently used standards for this purpose.

A solution that involves data distribution and communication on an unsafe network structure requires an efficient security mechanism. Web Services allow implanting various security related services, such as authentication, access policies and cryptography, which can be used isolated or together. The users accessing the service must be identified in order to establish roles, permissions or access levels and therefore restrict the access to data and services offered.

Although solutions that aim the independence of platform and interoperation already exist, the main advantage of the Web Services technology is the use of Web protocols to exchange messages instead of the proprietary standards, such as RMI. The Web Services technology was used in the present work as a consequence of the standardized functionalities it offers. Besides that, the SOAP protocol may operate over HTTPS, and then the learners' information cannot be observed without advanced hacking techniques providing a good level of data privacy.

2.2 Standards for Learner Model

The two most important standards for learner modeling are IEEE LTSC Personal and Private Information Standard (PAPI) [4], and IMS Learner Information Package (LIP) [5]. Both standards deal with several related categories of information about a learner; some of them are used in this work. The characteristics of the main standards of learner models are presented in the next paragraphs.

The IMS LIP standard contains several categories for data about a user. The *identification* category presents demographic and biographic data about a learner. The *goal* category presents learner targets, career expectation and other objectives. The *QCL* category is used for the identification of qualifications, certifications, and licenses from recognized authorities. The *activity* category contains learner-related activity in any state of completion. The *interest* category maintains any information describing learner hobbies and recreational activities. The *relationship* category aims for relationships between core data elements. The *competency* category serves as slot for skills, experience and knowledge acquired. The *accessibility* category aims for general accessibility to learner information by means of language capabilities, disabilities, eligibility, and learning preferences. The *transcript* category presents summary of academic achievements. The *affiliation* category presents information about membership in professional organizations. The *security key* is used for setting learner passwords.

The PAPI standard distinguishes *personal*, *relations*, *security*, *preference*, *performance*, and *portfolio* learner information. The *personal* category contains information about names, contacts and addresses of a learner. *Relations* serve as a category for relationships of a specific learner to other people (e.g. classmate). *Security* aims to provide access rights. *Preference* indicates the types of devices and objects that the learner technological environment is able to recognize. *Performance* contains information about measured performance of a learner through learning material. *Portfolio* accesses the previous experiences of a user.

The learner model used in this work follows the PAPI and LIP standards. Each one of these standards presents deficiencies in some characteristics and none of them includes the definitions of learning styles and cognitive styles, which are extremely important for this work as these styles are the keystone for content adaptation. These definitions have being included in our model.

2.3 Related Work

In this section, some solutions that have already been found for the exchange of learning data and resources among e-learning systems are described. As no solution for learners model data sharing has been found, the works we describe here bring some information about sharing of data stored in learning objects repositories. Learning objects repositories (LOR) are collections of learning resources stored in databases or file systems. They have metadata associated that, in general, are available and can be searched through the Web. The metadata of a learning object (LO) describe LO features that are used to index them in the repositories, making possible to retrieve them through search systems or use them in e-learning systems.

The Edutella Project [9] provides a RDF-based metadata infrastructure for P2P (Peer-to-Peer) applications exchange the educational resources (using standards like IEEE LOM, IMS, and ADL SCORM to describe course materials). In the same way, the CANDLE Project (Collaborative and Network Distributed Learning Environment) [10] uses metadata to describe course material to make it more reusable. CANDLE extends the metadata set defined by the Learning Object Metadata (LOM) [4] standard of the IEEE. These additional metadata describe learning objects by their purpose (learning goals, assessment methods), complexity level, type of learners (face-to-face, distance), setting (corporate, university), estimated time for completion, and others.

The Open Learning Repository (OLR) [11] is intended to support metadata-based course portals, which structure and connect modularized course materials on the Web. The modular content can be distributed anywhere over the Internet, and is integrated by explicit metadata information in order to build courses and connected sets of learning materials. Modules can be reused for other courses and in other contexts. IEEE LOM metadata is used by authors to help them choose modules and to connect them into course structures.

Elena [12] is an operational learning services network based on the interoperable communication infrastructure named “smart spaces for learning”. A personal learning assistant is a component of a smart space for learning, which helps learners in searching and selecting learning services. The personal learning assistants are also able to recommend learning services based on the learner profile.

The SeLeNe (Self e-Learning Networks) [7] project offers advanced services for the discovery, sharing and collaborative creation of learning resources, as well as a personalized access to such resources.

The Edutella, CANDLE and ORL projects deal with problems related to the reusability of the educational material and are concerned with making learning objects (LOM) available at repositories on the Web, in a way that they can be accessed and used by other systems.

The Selene and Elena projects are also concerned about the reusability of the educational material; besides, they offer personalized access to the educational material according to the learning model. In these works, each project defines its own learner model based on the IMS-LIP and PAPI standards.

3 The project approach

The works described in section 2.3 use the Web Service technology to perform the communication between the repositories of learning objects and the e-learning systems. The Web Services are used for searching and making available the learning objects in the repositories. Metadata are used to describe the objects, following the IEEE-LOM standard. However, research projects that present a solution for the sharing of learner model data, which would help the systems to have more information about the users, is not currently available. This work presents a solution for sharing these data. The section 3.1 shows a new learner model for data exchange between e-learning systems. Section 3.2 describes the architecture. Section 3.3 explains how to integrate e-learning systems and example scenarios respectively. Section 3.4 describes the service-based architecture.

3.1 PAPI_LIP learner model

In order to develop a solution for data exchange between e-learning systems it is necessary to establish a widely accepted data model. Our model follows the learner model defined by the SeLeNe project [7], which suggests the mutual use of PAPI and LIP standards. Figure 1 presents the elements used for each standard in our solution (PAPI_LIP).

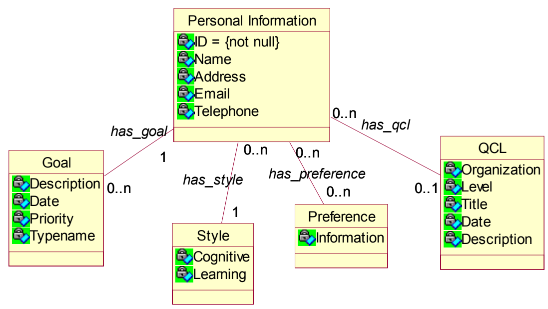

Fig 1. PAPI_LIP Learner Model

The Personal Information and Preferences categories are PAPI standards and follow its parameters. The Personal Information category comprises the following elements: 1) ID: student's single identifier, it can be the social security number or other country unique identifier (mandatory); 2) Name: student's full name; 3) Address: student's full address (street, number, and district); 3) Email: student's electronic mail address; 4) Telephone: student's contacts phone numbers. The Preferences category contains a single element named List. It comprises a list of the student's preferences like accessibility, interest area, and favorite authors. The List field is open and accepts any kind of information.

The IMS-LIP Standard provided two categories: QCL and Goal. QCL stands for Qualifications, Certificates and Licenses. Each entry in the QCL category has the following elements: Organization: institution that has given the certificate; 2) Level: level of certificate, it can be graduation, specialization, etc; 3) Title: title awarded; 4) Date: certificate's date; 5) Description: it contains additional information about the qualification certified. The other category, Goal, contains the student's objectives. The elements of this category are: 1) Typename: type of student's goals. They can be professional, educational or personal; 2) Description: description of the goal that follows the standard defined in IMS-LIP; 3) Date: deadline to reach a goal; 4) Priority: indicates priority level.

These categories were selected with the aim of fulfilling the needs of the AdaptWeb environment's adaptability techniques. Style category was added to our model and comprises the following elements: cognitive and learning.

The cognitive style of learning is an individual aspect that describes the way a person usually approaches or responds to the learning task [13]. According to Gregorc [14], a person's cognitive style is considered one of the most stable user characteristics overtime that influences a person's general attainment or achievement in learning situations. This stability is manifested in the use of hierarchies' processes in the treatment of the information and on the strategies that the learner uses when acquiring new information with a hypermedia system. The cognitive style taxonomy used in this work was the one defined by Gregorc [14]. The learning style is a collection of individual skills and preferences that affect how a person perceives, gathers, and processes information. In this work, we used the learning style classification by Felder [15]. Table 2 presents the learning and cognitive style taxonomies and their respective style descriptors used in this work.

Besides these psychological characteristics, matching the cognitive style to the domain content in hypermedia systems is a pedagogical method to make understanding easier and lead to a preferred behavioral mode of information processing [3, 13]. This occurs because the cognitive style interacts with the content structure and processes the information in a quite different way, which induces the utilization of a specific learning strategy to each cognitive style. The cognitive style related to the learning strategy requires the design of the learning resources closely tied with the learner's cognitive profiles.

Table 1. Learning and Cognitive Style Taxonomies

Cognitive Style Taxonomy	Cognitive Style Descriptor
Gregorc	Concrete Sequential (CS)
	Abstract Sequential (AS)
	Concrete Random (CR)
	Abstract Random (AR)
Learning Style Taxonomy	Learning Style Descriptor
Felder- Silverman	Active/Reflective
	Sensing/Intuitive
	Visual/Verbal
	Sequential/Global

Currently we have been using the proposed model in the context of the AdaptWeb [16] environment. The goal of this environment is to adapt the instructional contents based on the learners' model. In AdaptWeb, the learners' model is formed by information explicitly provided by the learner and information collected by monitoring the learner's behavior.

When login for the first time in the AdaptWeb environment, the learner is required to provided some personal data (Personal Information category in the PAPI_LIP model), as well as the navigation preferences (Preferences category), which can be either *free* or *tutorial*. In *tutorial* navigation, pre-requirement among concepts determines the learner's navigation. The navigation adaptation is based on the register of concepts studied by the learner. In *free* navigation, the learner can study any concepts, independent of whether a pre-requirement exists or not. The preferences category also stores the way in which the learner access the computer network (e.g. dial-up, local access network and ADSL).

In AdaptWeb, the student's Cognitive Style is taken into account to define the first element to be displayed to the student. For example, if the learner cognitive style is "Abstract Random", he usually prefers taking a simple example prior to taking the theoretical exposition of a new topic. This cognitive style is determined by analyzing the learner's navigation behavior. Details on this analysis and the definitions of the others styles can be found in [3]. In the PAPI_LIP model, this information is stored in the Cognitive element of the Style category.

Several kinds of data about the learner can be captured from his curriculum. For example, in Brazil, the whole research community, from undergraduate students to senior researchers, must put their curriculum available in the Lattes[a] platform. Figure 1 shows an example of Lattes curriculum exported in XML format.


```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<CURRICULO-VITAE SISTEMA-ORIGEM-XML="LATTES_OFFLINE" DATA-ATUALIZACAO="10102004" HORA-ATUALIZACAO="202511"
  xmlns:lattes="http://www.cnpq.br/2001/XSL/Lattes">
- <DADOS-GERAIS NOME-COMPLETO="Daniela Leal Musa" NOME-EM-CITACOES-BIBLIOGRAFICAS="MUSA, D. L." NACIONALIDADE="B"
    CPF="1111111111" PAIS-DE-NASCIMENTO="Brasil" UF-NASCIMENTO="RS" CIDADE-NASCIMENTO="Porto Alegre" DATA-
    NASCIMENTO="1111111111" SEXO="FEMININO" NUMERO-IDENTIDADE="1111111111" ORGAO-EMISSOR="SSP" UF-ORGAO-
    EMISSOR="RS" DATA-DE-EMISSAO="25091989" NUMERO-DO-PASSAPORTE="" NOME-DO-PAI="Musa" NOME-DA-MAE="Musa"
    PERMISSAO-DE-DIVULGACAO="SIM" OUTRAS-INFORMACOES-RELEVANTES="">
  + <ENDERECO FLAG-DE-PREFERENCIA="ENDERECO_RESIDENCIAL">
  - <FORMACAO-ACADEMICA-TITULACAO>
      <GRADUACAO SEQUENCIA-FORMACAO="1" NIVEL="1" TITULO-DO-TRABALHO-DE-CONCLUSAO-DE-CURSO="" NOME-DO-
        ORIENTADOR="" CODIGO-INSTITUICAO="000600000001" NOME-INSTITUICAO="Pontifícia Universidade Católica do Rio
        Grande do Sul" CODIGO-CURSO="90000000" NOME-CURSO="Bacharel em Informática" CODIGO-AREA-CURSO="10300007"
        STATUS-DO-CURSO="CONCLUIDO" ANO-DE-INICIO="1991" ANO-DE-CONCLUSAO="1995" FLAG-BOLSA="NAO" CODIGO-
        AGENCIA-FINANCIADORA="" NOME-AGENCIA="" />
    + <MESTRADO SEQUENCIA-FORMACAO="2" NIVEL="3" CODIGO-INSTITUICAO="019200000005" NOME-
 →      INSTITUICAO="Universidade Federal do Rio Grande do Sul" CODIGO-CURSO="42000041" NOME-CURSO="Computação"
        CODIGO-AREA-CURSO="10300007" STATUS-DO-CURSO="CONCLUIDO" ANO-DE-INICIO="1999" ANO-DE-CONCLUSAO="2001"
        FLAG-BOLSA="SIM" CODIGO-AGENCIA-FINANCIADORA="045000000000" NOME-AGENCIA="Coordenação de Aperfeiçoamento
        de Pessoal de Nível Superior" ANO-DE-OBTENCAO-DO-TITULO="2001" TITULO-DA-DISSERTACAO-TESE="Um Sistema de
        Alertas Inteligentes para Ambientes de Ensino na Internet" NOME-COMPLETO-DO-ORIENTADOR="José Palazzo Moreira
        de Oliveira">
    + <DOUTORADO SEQUENCIA-FORMACAO="3" NIVEL="4" CODIGO-INSTITUICAO="019200000005" NOME-
        INSTITUICAO="Universidade Federal do Rio Grande do Sul" CODIGO-CURSO="42000041" NOME-CURSO="Computação"
        CODIGO-AREA-CURSO="10300007" STATUS-DO-CURSO="EM_ANDAMENTO" ANO-DE-INICIO="2002" ANO-DE-CONCLUSAO=""
        FLAG-BOLSA="SIM" CODIGO-AGENCIA-FINANCIADORA="045000000000" NOME-AGENCIA="Coordenação de Aperfeiçoamento
        de Pessoal de Nível Superior" ANO-DE-OBTENCAO-DO-TITULO="" TITULO-DA-DISSERTACAO-TESE="" NOME-COMPLETO-DO-
        ORIENTADOR="José Palazzo Moreira de Oliveira">
    </FORMACAO-ACADEMICA-TITULACAO>
  - <ATUACOES-PROFISSIONAIS>
    + <ATUACAO-PROFISSIONAL CODIGO-INSTITUICAO="000100000991" NOME-INSTITUICAO="TRUE Administração de
        Informações Ltda" SEQUENCIA-ATIVIDADE="2">
    - <ATUACAO-PROFISSIONAL CODIGO-INSTITUICAO="000200000993" NOME-INSTITUICAO="Tribunal Regional do Trabalho da 4ª
        Região" SEQUENCIA-ATIVIDADE="4">
        <VINCULOS SEQUENCIA-HISTORICO="1" TIPO-DE-VINCULO="OUTRO" ENQUADRAMENTO-FUNCIONAL="LIVRE" CARGA-
          HORARIA-SEMANAL="20" FLAG-DEDICACAO-EXCLUSIVA="NAO" MES-INICIO="8" ANO-INICIO="1994" MES-FIM="6" ANO-
```


Fig. 2. CV Lattes in XML format

Data such as those of the Personal Information category and QCL of the PAPI_LIP model can be retrieved from Lattes in the XML format. For example, in figure 2, the element "Instituição" is related to element Organization in QCL category.

3.2 Architecture

Some alternatives were studied for the development of an architecture that would be appropriate to solve the learner model exchange problem.

Concerning the learners' model, two approaches are found in the literature: centralized and decentralized methodology. According to Yimam & Kobsa [17], the centralized methodology allows the integration of existing sources of learners' information in a model. In such centralized methodology, the learner model is maintained and processed in a central or virtual integrated repository and the learner information retrieved from one application can be used by other applications.

Vassileva [18] presents a decentralised learner modelling methodology, which consists of a group of learners' models developed and kept by a variety of software agents in the context of multi-agents environments. This approach was proposed as a distributed alternative to server-based architecture. The model is used to compute relevant information about one or more learners, depending on the purpose of adaptation. In this kind of environment, there is no single monolithic learner model associated with each learner. Rather, the learner models are fragmented and distributed throughout the

[a] The Lattes software is dedicated to update and publish on-line curricula of all the Brazilian researchers and is provided cost-free by the Brazilian National Research Council (CNPq) at http://lattes.cnpq.br/.

system. In this approach, the learner models can be stored anywhere – in a centralized or distributed database or files. This approach is the opposite of the Yimam & Kobsa approach cited before.

While the centralized approach aims to collect as many data about the learner as possible, the decentralised approach focuses on the process of collecting and integrating information about the learner at a particular time and with specific purposes.

In both approaches presented above, the applications that want to access information contained in the learner's model should know explicitly the storage schema of these data and their localization. In a web-services-based architecture, this is not necessary since the application that wants to retrieve or share data need only the existing services and a standard metadata structure to access these data, and it is not necessary to know how data are physically stored or internally structured.

In a typical Web services architecture a service is made available so that other systems can use it. The services provider creates a WSDL service description that defines the service interface, that is, the operations of the service and the input and output messages for each operation. The provider publishes its WSDL service description to a discovery agency. Service requesters find services via discovery agencies and use the WSDL description to interact with the service description that corresponds to a discovery.

The architecture of our organization is composed of e-learning systems that collaborate to enrich the learners data model, these e-learning systems compose a federation and act as services providers or service requesters. Each e-learning system that has student's data to share specifies their respective services using WSDL and registers it at a central repository. The central repository is a repository of WSDL specifications, which may be mapped to UDDI for publishing, and discovery of existing services. The central repository acts as a broker system and can store data gathered in the e-learning systems. Figure 3 presents the components of this architecture.

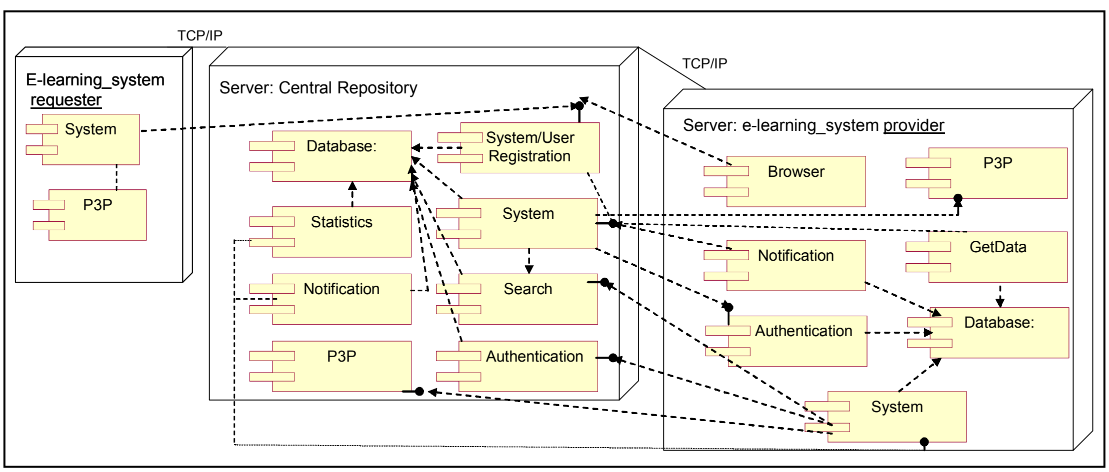

Fig. 3. Architecture of the project approach

Although figure 3 shows a repository as a unique element, its implementation could be a complex distributed system with data replication and fault tolerance. However, the access to a repository is unique through the Web Service.

As the repository is dealing with private data, an authentication mechanism is needed so that e-learning systems can access it. This authentication permits e-learning systems to retrieve data from the repository. Moreover, the central repository needs a specific authorization from each of the e-learning systems in order to be able to retrieve data from its local databases. The repository can implement a privacy policy that describes how the information it will receive will be handled. When the e-learning system provides information to the repository, it can determine how the repository will handle this information. The solution we have been studying for these cases is the P3P standard (Platform for Privacy Preferences Project) [19]. This standard provides the formats, by which two parties, client and server, describe and enforce their privacy policy.

Communication between the repository and the component systems is carried through a Web Service, which exchange data by means of the new learner model defined in this work: PAPI_LIP.

3.3 Integrating E-learning systems

Any e-learning system that needs the data stored in the repository should require access to it. After the initial access, the definition of the operations available from the web service is sent to the e-learning system in the WSDL format. The component e-learning system needs to implement an interface responsible for calling the functions of the repository web service. The communication will always be made through SOAP.

There are two types of systems that can use the data stored in the central repository. The first type is represented by the e-learning system requester in our architecture (Figure 3). This system only collects data from the repository to enrich its learner model. The exchanged data are represented in XML and follow the PAPI_LIP model. The communication is made from e-learning system to Web Service through SOAP. A wrapper is necessary to convert data for the system database. If the database used by the systems stores data in XML following the PAPI_LIP format, then the wrapper is not necessary. The second type of system represented by repository can retrieve data in any e-learning system database. To do that, the e-learning system needs to implement a Web service. This type is represented by the e-learning system provider in our architecture (Figure 3).

The Web Service accesses the e-learning database, the communication is made from Web Service to Web Service through SOAP and the received data are in PAPI_LIP format.

For example, two scenarios are described where the presented solution can be used. Scenario 1: a learner L1, for example, logs into the learning system requester, provides the security number and registers in a course. After, the learning system retrieves the learner model from the repository and inserts it into system database.

Scenario 2: the learning system provider, for example, is able to discover learner's cognitive style observing the learner's navigation pattern during the course. After that, the learning system inserts the just discovered cognitive style into the central repository.

3.4 Services

Functionalities in the system are offered as services. The central repository services are related with users' records and records of e-learning systems that will provide data. The e-learning systems that need to be associated with the student's model federation should provide, at least, the following services *P3P, Access* and *Get,* described below.

An e-learning system must offers services available through Web Services and also need to describes them in the central repository in the WSDL format in order to have data of the student's model shared. Any e-learning system (user or client) that intends to enter the federation should have an access license. These services are illustrated in figure 3, in the following text the services are referred with their name in the same figure. The least set of services that the e-learning systems provider should offer are: *Authentication, P3P* and at one *GetData* service. *Authentication*(1) allows the repository to access its database. *P3P*(2) returns privacy policies of the e-learning system, which will be compared against the repository's privacy policies.

The *GetData*(3) services are directly linked to the categories and elements of the PAPI_LIP standards. The *GetPersonal* returns all elements to the repository (id, name, address, e-mail, telephone) in the PAPI_LIP format; it can be of one or more students. The *GetAllQCL, GetAllGoal* and *GetPreferenceList* service return all elements of the respective category. *GetQCL* may return only one register that can be searched by *Organization, Level, Title* or *Date*. *GetGoal* returns only the goals that are related with the date sent as parameter. *GetCognitiveStyle* and *GetCognitiveLearning* return the student's cognitive style and his learning style, respectively. The format of data sent to the repository is always in the PAPI_LIP standard.

The e-learning system should be registered in the repository using the operation *SystemRegistration* if it intends to supply data to the central repository. The register of an e-learning system involves the description of its services in WSDL and its privacy policy in P3P. An e-learning system should require the repository's privacy police before its registering. The *P3P* service provides the repository's privacy policies to the e-learning system.

The central repository offers the *UserRegistration* operation for users registered in the repository. On subscription, the user should provide his privacy police in *P3P*. Only registered users can search or research data through the repository. If a registered user wants to search the services registered in the repository, he will use the *Search* service, which returns the services the user requested. The search for a service is carried out through categories of the PAPI_LIP model categories. For example, a user may want to search that supplies services that offer the styles of a given student (Category Style of the PAPI_LIP model). An advanced search can also be made, that is, the user can discover the systems that provide the cognitive style to the students through the *Search* service.

The *Notification* service included in the repository sends notifications to registered users warning that new information that interests him have arrived. The information can be: 1) input or output of a new base in the repository; and 2) inclusion or alteration of a value.

The *Statistics* service is an especial type of search that returns a summary of the access to the repository. Examples of statistical search are: who searched such information, which received the notification before searching data (%), which received the notification and has not searched data (%), etc.

The process of Central repository fetches data in e-learning system provider is illustrated in activity diagram on Figure 4.

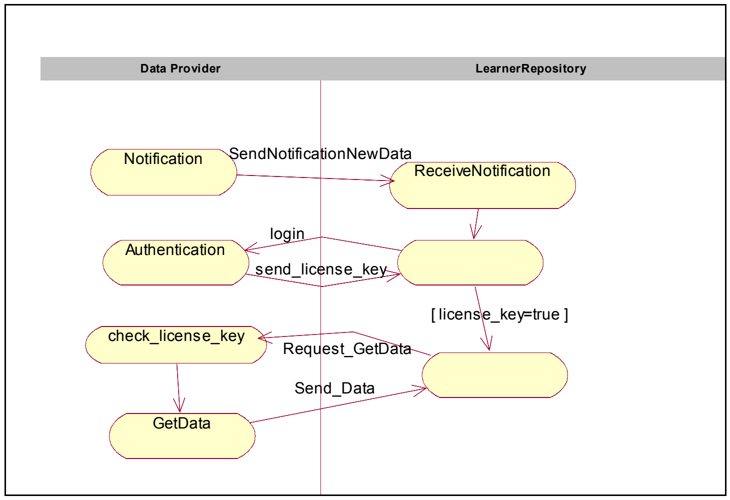

Fig. 4. Activity Diagram for Central Repository fetches data in data provider

Figure 5 shows the activity diagram for an e-learning system client while using the functionalities offered by central repository.

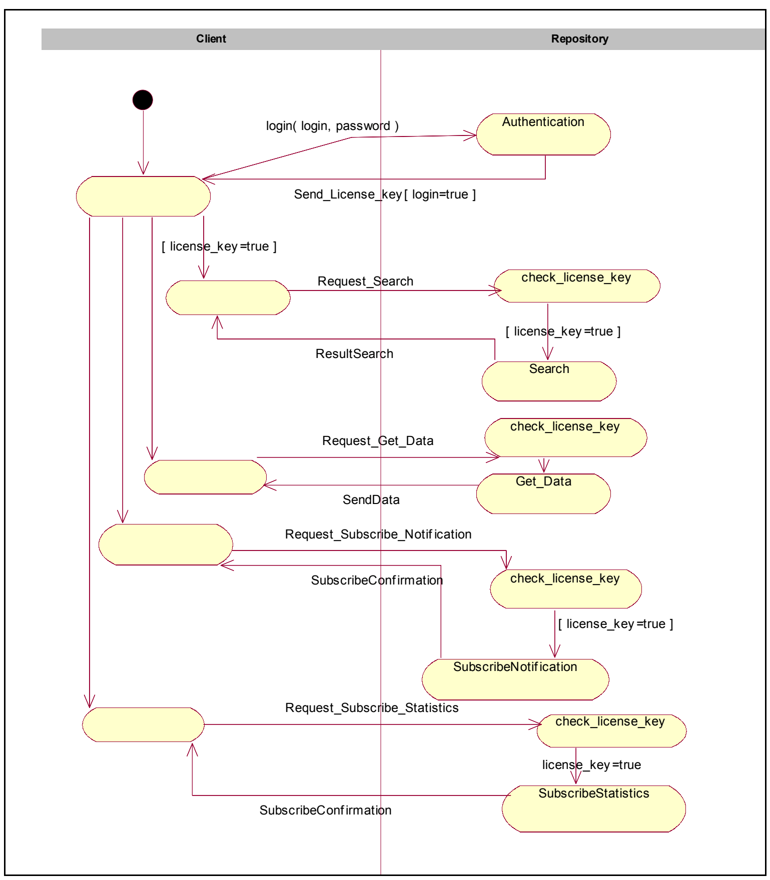

Fig. 5. Activity Diagram for system client access

4. Implementation

With the purposes of validating the Web Services-based proposed architecture, a system prototype has been developed using the PHP language. The prototype retrieves students' data from two different relational databases (MySQL) and stores such data in a central repository. A Web Service for data retrieval (Get_Data) was developed for each database, as described in section 3.

The first database accessed comes from the AdaptWeb environment [16]. It offers educational content according to the student's profile. ID, name, address, email and telephone are data from the AdaptWeb student's model, which is stored in a MySQL base. Figure 6 presents the mapping from AdaptWeb to PAPI_LIP.

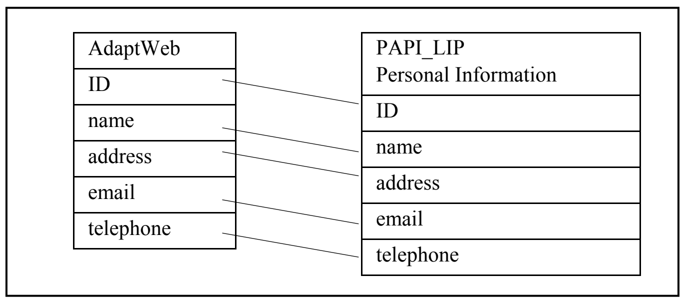

Fig.6. AdaptWeb to PAPI_LIP mapping

The second database is from the Claroline environment [20]. Claroline is a software which was developed by the Université Catholique de Louvain (Belgium) and released under Open Source licence (GPL). Figure 6 shows Claroline to PAPI_LIP mapping.

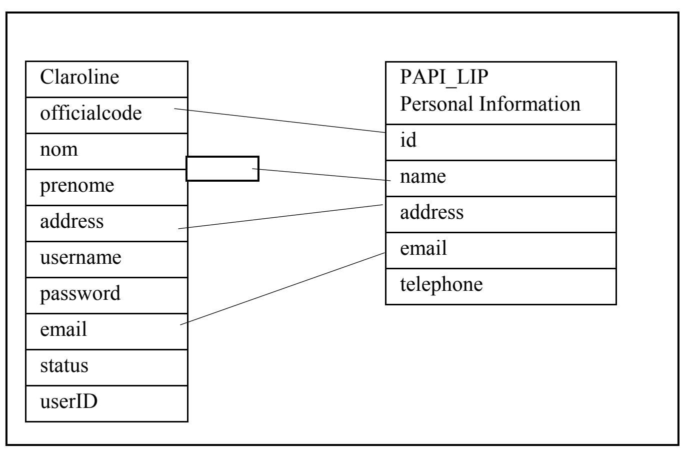

Fig.7. Claroline to PAPI_LIP mapping

We have chosen these target systems because they contain distinct data models and enough tuples of data to allow us to carry out our experiment.

4.1 GetData Service

A client log into the system and receives an access key . This key is sent at each operation request, and it is valid only for the current session, for safety reasons.

The client makes a query that can retrieve data from one of the repositories (Claroline and AdaptWeb) or from all of them (All). At this moment, the client also chooses the type of data from the PAPI_LIP model it wants, by clicking on the required option (Personal Information, Preference, QCL, Goal, Style). The fields corresponding to the type chosen are displayed. The client should fill one or more fields to request the query. In the example shown in Figure 8, the query is made under the student's name. The GetAll returns all student's personal data that match the search criteria selected.

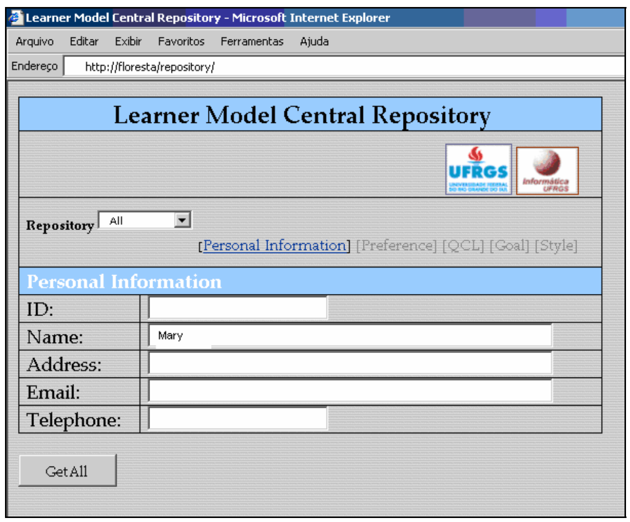

Fig. 8 - Learner Model Central Repositoy

The GetData operation of Claroline checks if the access key is valid to perform a query. After the query, data is shown on the screen and the client chooses if it wants to store them in the central repository. Figure 9 shows an example in which only data containing the word "Mary" returned. If any of those bases has information referring to Personal Information, the field would be null. The example shows the "telephone" field null for the records from the Claroline environment, as it does not have this information in its basis.

The operation receives the user's request in the PAPI_LIP format and maps to local database. After conversion between schemas, the web-service makes the query in the database. The same mapping should be performed when the query result is sent to the client. Each environment is responsible for mapping the PAPI_LIP schema in the local database.

All data are sent through the HTTPS, which confers safety to data sending procedures. Some information may be available to the public, some information may have limited accessibility to the public, some information may be private, and other combinations are possible. PAPI_LIP information may be administered and secured separated, e.g., personal learner information is private and secure, while the learner portfolio information is public. The public nature of the learner information is chosen by the learners and administrators and the requirements for this choice is outside the scope of this work.

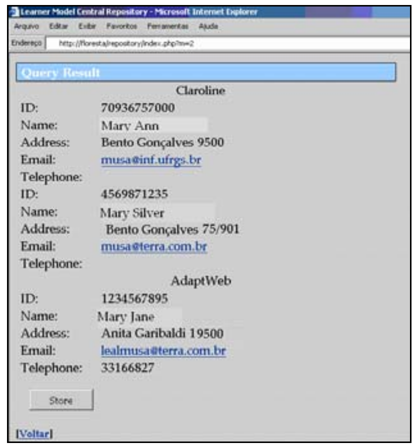

Fig. 9 - Query Result

5. Conclusions and Future Works

This paper presented an architecture for the exchange of learner's information between applications on the Web developed for the AdaptWeb[7] [16] that offers an adaptive environment for educational contents. The Web Service technology was used since it allows the simple communication between Web applications also through the HTTPS protocol. Our goal with this work was to allow different e-learning systems to cooperate with each other in order to reach a set of learner information richer than the set currently found in common stand-alone e-learning systems.

We have also defined a learner model that was used in the data exchange performed by the e-learning systems that communicate with each other. This learner model was defined based on the main characteristics of existing learner model standards such as PAPI and LIP. In addition, we have also presented a set of services required to allow the communications between the e-learning systems. Those services, defined and implemented via Web Services, allow not only data exchange but also provide security, privacy, and event notification facilities.

Some experiments have already been made aiming the technical validation of the proposed architecture. However, more research still must be carried on in order to improve the offered functionalities mainly considering the pedagogical and psychological aspects. The implementation of an active behavior at the learner models repositories is under investigation. This kind of behavior can be implemented through ECA (event-condition-action) rules in order to notify the e-learning systems every time some relevant data is updated or inserted in the common repository. The opposite is also valid, since an e-learning system can also have an active behavior, which would give a warning to the central repository. These features will improve the current implementation of the event notification service. In addition, other notifications classes that can also be defined by active rules are on research schedule. The AdaptWeb environment, where the solution presented in this paper has been implemented, is available from SourceForge [21]: all other future developments will be freely available as well.

References

1. Dolog, P., and Nicola Henze, Personalization Services for Adaptative Educational Hypermedia, In: Proc. of International Workshop on Adaptivity and User Modelling in Interactive Systems, Germany, 2003. http://km.aifb.uni-karlsruhe.de/ws/LLWA/abis/dolog.pdf
2. Freitas, V. et al, AdaptWeb: an Adaptive Web-based Courseware, ICTE, Badajoz, Espanha, 2002. http://rubens.cs.kuleuven.ac.be:8989/ariadne/CONF2003/papers/Pal2003.pdf
3. Souto, M. et al, Towards an adaptive Web training environment based on cognitive style of learning: an empirical study. In: 2nd AHA, Malaga, Lecture Notes in Computer Science v. 2347, ISSN: 0302-9743, p. 338-348, 2002.
4. Learning Technology standards committee: http://ltsc.ieee.org
5. IMS Consortium: http://www.imsproject.org.
6. Orkut, http:// www.orkut.com.
7. SeLeNe –http://www.dcs.bbk.ac.uk/selene/.
8. Tsalgatidou, A. and Pilioura, T., An Overview of Standards and Related Technology in Web Service. In Distributed and Parallel Databases, vol. 12, p. 135 – 162, ISSN:0926-8782, 2002.
9. Qu, C., and W. Neddl, Searching SCORM Metadata in RDF-based E-learning P2P Network Using XQuery and Query by Example, 3rd IEEE International Conference on Advanced Learning Technologies, 2003, Athens - Greece, 2003. Proceedings. p 81-85.
10. Wetterling, J. Re-Usability of Telematics Course Material within a European Network: The CANDLE Project, In. Proc. EDEN Annual Conference, Rhodes, Greece, p. 324-327. 2003. Proceedings.
11. Dhraief, H. et al., Open Learning Repositories and Metadata, In Proc. International Semantic Web Working Symposium - Infrastructure and Applications for the Semantic Web - July 30 - August 1, 2001, p. 495-514, California, USA.
12. Simon, B. et al., Elena: A Mediation Infrastructure for Educational Services, In Proc. of WWW Conference, Budapest, Hungary, May 2003, p. 616. http://www2003.org/cdrom/papers/alternate/P616/p616-simon.html
13. Riding, R. & Rayner, S.: Cognitive Styles and Learning Strategies. London: David Fulton Publishers, 2000.
14. Gregorc, A.F. Individual Differences: Teaching for Active Learning, Keynote Address, University of Illinois at Urbana Champaign Faculty Retreat on College Teaching, June 19,1996.
15. Felder, R.M and L.K. Silverman, Learning and Teaching Styles in Engineering Education, In Engineering Education, 78(7), 674 , 1988.
16. AdaptWeb homepage at http://www.inf.ufrgs.br/adapt/adaptweb
17. Yimam, Dawit & Kobsa, Alfred; Centralization vs. Decentralization Issues in Internet-based Knowledge Management Systems: Experiences from Expert Recommender Systems. TWIST 2000. Irvine, CA.
18. Vassileva, J. Distributed User Modelling for Universal Information Access, Proceedings of the 9th International Conference on Human-Computer Interaction, New Orleans, USA, vol.3, Lawrence Erlbaum: Mahwah, N.J., p. 122-126, 2001.
19. The Platform for Privacy Preferences 1.0 (P3P1.0) Specification. W3C Recommendation, April 2002. http://www.w3.org/TR/2002/
20. Claroline Homepage – http://www.claroline.net
21. AdaptWeb SourceForge - http://www.sourceforge.org/projects/adaptweb