Web 2.0 Architecture for Services Federation in Virtual Learning Communities

Julio R. Ribón  
University of Cartagena  
Piedra de Bolívar  
Cartagena (Colombia)  
+57 5 6632567  
jcrr@dit.upm.es

Tomás P. de Miguel  
Polytechnical University of Madrid  
Ciudad Universitaria  
Madrid (Spain)  
+34 1 336 7366 + 3026  
tmiguel@dit.upm.es

Martín Monroy R.  
University of Cartagena  
Piedra de Bolívar  
Cartagena (Colombia)  
+57 5 6725678  
monroy.martin@gmail.com

ABSTRACT
Internet enables multiple organizations to share their learning expertise enabling the generation of virtual learning communities. However, current Learning Management Systems do not allow real time access to the experiences offered by other peers, which is causing a problem of technological isolation. This paper presents a case study of generating a community of e-learning for corporate staff qualification, which has been implemented on a Web 2.0 architecture for the services federation, which is useful because it allows to solve the problem of isolation, allowing the generation of virtual learning communities.

Categories and Subject Descriptors
K.3.1 [Computers And Education]: Computer Uses in Education - Collaborative learning, Distance learning, Computer-managed instruction (CMI).

General Terms
Documentation, Design.

Keywords
E-learning, virtual communities, cooperative, joint degrees.

1. INTRODUCTION
Virtual learning communities [1][2] or E-learning communities can be considered as formation services formed by a grid of learners records and learning resources (Grid expertise of peers) are accessible through Learning Management Systems – LMS [3], which make federation possible and allow collaborative spaces between each of the members that conform it.

There is a technological problem that is making impossible the realization of e-learning communities. This issue is related to the inability of current LMS to federate the expertise of peers who belong to the community, which is causing isolation.

The current problem has been detected by authors' experience in implementing e-learning communities in the Colombian Caribbean Universities [4][5][6], however, a review of the literature shows that many initiatives that are being developed worldwide are recurrent [7][8][9].

Because of this, we intend to contribute to the solution of the problem, through the presentation of an architecture for service federation, which serves as an intermediary to the current LMS to facilitate the generation of virtual communities of e-learning, as is the case of joint degrees between organizations [4][5].

A service federation system is a mechanism that allows the interoperability by alliance or pact between the software tools that support virtual learning communities, hiding the heterogeneity, redundancy and fragmentation of information.

The methodology used in the construction of the proposed solution is RUP [10], which is focused on architecture, feature that allows easily connecting and integrating processes, methods, techniques and notations in software engineering. The methodology used for modeling the cooperative behavior of joint degrees is AMENITIES [11], since it allows to model cooperative systems based on behavior and tasks model, considering static and dynamic aspects of these systems. The diagrams used in the description of the architecture are based on UML [12] [13] as modeling language, which is considered as a standard in software development.

Initially is described the system architecture through different views: the logical view of the system, continuing the process view, view of development, physical view or deployment view and the view of information, later is described a case study of a developed joint degree to qualify corporate staff, to finally draw conclusions.

2. DESCRIPTION OF ARCHITECTURE
The design model represents the system architecture of the federation of services in virtual learning communities, composed of loosely coupled subsystems, whose structure is explained below:

2.1 Logical View
Its purpose is to represent the key elements of abstraction responsible for meeting the functional requirements. To ensure the functional independence (weak coupling between the parts), the model assumes that every component defines interfaces to offer their services, which are invoked by using XML [14] and JSON [15].

These key elements are symbolized by three components, as shown in the graphic representation of the logical view (Figure 1), each of which is explained below:

- **Web Services:** Ensures the invocation of federated services from different types of devices, by the actors in
the system, allowing each of them the realization of their functions through rich user interfaces, using a lightweight programming model that facilitates communication with other system components.

- **Service Platform:** Responsible for control on data sources unique and difficult to replicate, giving the possibility that they are susceptible to mixing and processing, to be enriched as more people use them to configure new services taking advantage of the collective intelligence so they can be provided to the organizational and external users, incorporating reporting mechanisms through the syndication of such services.

- **Reusable Business:** This component represents the data sources unique and difficult to replicate. In the view of development is explained in detail its structure and function, represented in the Federated subsystem.

2.2 **Process View**

It is used to represent the distribution of the features in each of the components identified in the logical view, highlighting the order of communication that allows evidence of compliance with the purposes of the system. For a better understanding, it is explained below taking into account the main moments that conforms the provision of services offered by the federation system.

We have identified two important moments because the conceptual model focuses on integrating services enabling interoperability across heterogeneous technology platforms that support these services. The first corresponds to the configuration of the service and the second to consumption.

**Service settings:** For each of the actors in the system [4] to be able to invoke the services to which they are entitled, it is necessary that the system administrator previously performs a process configuration of such services, which consists in defining an exported schema for each source, which contains information relating to resources that it enables as shared to be available to the federation, it describes the access rights that are available for the structure of a local schema, represented by a component schema that ensures control over data sources unique and difficult to replicate, giving the possibility that they are susceptible of mixing and processing. All this is done in component labeled Reusable Business (Figure 2).

Additionally, for the service to be fully configured, it becomes necessary through the component Services Platform, the system administrator describes the information concerning to the portfolio of services that places available to other users of the system, according to the rights allocated by the administrator. These federated services are formed by the various resources that each single data source (federated) enables to share. There is a single federated schema to which access each user of the federation.

To ensure lightweight programming models that allow loosely coupled systems, the representation of each of these schemes is done using XML [14], as explained in detail later in the view of development.

**Consumer Service:** After configuring the services, it is necessary to provide a mechanism to use them, represented in the model by the component called Web Services (Figure 2), which invokes the component Services Platform, to publish the federated services, facilitating invocation from different device types, its syndication and the creation of rich user interfaces, by applying models of development and light business models, to the extent that its invocation is done through XML.
2.3 View of Development

Also known as a view of implementation is responsible for displaying the organization of software modules, libraries, subsystems and development units (own components of the system.) The service federation system consists of two subsystems: a subsystem of the federated source-side (Federated) and a subsystem that manages the federation (Federator), each of which is loosely coupled, because the communication between them is done via XML and JSON, ensuring a high degree of functional independence by using design patterns and lightweight programming models.

2.3.1 Federator

The Federator component (Figure 3) is the core of the Federation provides the functionality to configure the services of the federation by the system administrator, maintaining control over data sources unique and difficult to replicate, which are enriched as more people use them to configure services, in order to enable these data sources to be susceptible of mixing and processing of data. It also allows the use of pre-configured services, to users of the system, administered according to the roles assigned to each.

Figure 3. Federator Component Architecture.
2.3.2 **Federated**

Correspond to the Reusable Business component of the logical view, which represents the data sources unique and difficult to replicate, providing the opportunity to be susceptible of mixing and processing (Figure 4). Used as an adapter from which are coupled the different sources of data that can later be federated; also allows them, to be configurable by the administrator of the service concerned, contributing to the autonomy of each federated source. Like the Federated component, is implemented using lightweight programming models that allow loosely coupled systems.

![Figure 4. Federated Component Architecture.](image-url)
2.4 Deployment View

This view (Figure 5) can represent the physical layout of nodes required for the deployment of processes that represent the services provided by the system. It should be clarified that the communication occurs in two forms: one that adheres to the formalism of the Web Services stack SOAP (Simple Object Access Protocol) [16] and the other simply providing XML data over HTTP and JSON, it is a lightweight approach sometimes referred to as REST (Representational State Transfer) [17], ensuring the use of Web 2.0 technologies [18].

As shown in Figure 5, the deployment view is structured into three nodes, each of which is explained below in order of use of end-user.

**WorkStation Node:** Represents the different types of devices (PC, PDA, cell, etc.). From which actors of the system perform invocations of federated services through interfaces that guarantee them rich expertise, they developed possibly themselves, through the application of design patterns and lightweight programming models that allow loosely coupled systems.

**Federation Server Node:** Contains the Federator subsystem, responsible for providing the functionalities to configure the services of the federation in charge of the system administrator, maintaining control over data sources unique and difficult to replicate, which are enriched as more people use them to configure services, in order to enable these data sources to be susceptible of mixing and processing of data.

Also allows the consumption and syndication of the services previously configured, for users of the system, administered according to the roles assigned to each. For its implementation requires the application server servlet container, such as Apache Tomcat (tomcat.apache.org). Make communication with the Federated subsystem through SOAP [16], from which it access data sources.

**Federated Server Node:** Contains the Federated subsystem, which represents the data sources unique and difficult to replicate, providing the possibility that they are susceptible of mixing and processing. It’s used an adapter from which are coupled the different sources of data that can later be federated; also allows them, to be configurable by the administrator of the service concerned, contributing to the autonomy of each federated source.

For its implementation requires the Apache Tomcat application server and Web service client called Axis (ws.apache.org / axis /), responsible for making clear requests and responses made to the Web Services representing the Federated. It is worth mentioning that for its configuration, the Federated component can be accessed from node WorkStation, using XML and JSON, which involves the use of loosely coupled programming models.

![Deployment View](image)

Figure 5. Deployment View.
2.5 Information View

This view refers to the important elements that works with the system for managing the collective knowledge in Web 2.0: It refers to the support given to the management of collective intelligence [18], these elements make reference to information which is used to allow the federal resources, which later will be made available as services of the federation. Give support to the software so that it won’t be seen as a product but as a service platform, important concept in Web 2.0 [18]. Taking as reference a 5-level architecture for federated systems [19] [20] are provided the following diagrams to describe the information that is managed by the federation system (Figure 6):

2.5.1 Local Schema (Esquema Local)

It is the own schema of each source or participant of the federation on the side of the source of the federated components. It is determined by how each technological tool that supports services in the company stores the information, for example, can be defined by a database management system (DBMS) or a set of files stored on disk.

Only the federated know how to access to their schemes, this means, that they are responsible for manage them and provide the information requested at one time a member of the federation.

In this scheme is described the logical structure of files and / or databases, referring to the metadata for each source, that is to say, that there will be a local scheme for each source that is integrated into the federation.

2.5.2 Component Schema (Esquema Componente)

Each source or origin of data, can define which part of a local schema can be enabled to provide a resource that can be consumed by the federation. This definition is stored in a format that describes the information sharing resources (metadata). In the different federated this scheme is structured in XML format.

2.5.3 Export Schema (Esquema Exportado)

Contains information relating to resources that the source enables as shared to be available to the federation, it describes the access rights that have on the structure of a given local schema. It creates a schema exported by each federated. It is described through an XML document.

2.5.4 Federated Schema (Esquema Federado)

This scheme makes the description of the information concerning the portfolio of services placed as available for the federation. These services are comprised of the various resources that each source (Federated) enables to share. There is a single federated schema to which access each user of the federation, to subscribe to a particular portfolio of services that the administrator has configured previously. It is structured through an XML document.

![Figure 6. Information View of Services Federation System](image-url)
3. CASE STUDY
The proposed architecture has been implemented to support a virtual learning community in the city of Cartagena de Indias. In this community, is established a joint degree consists of learning experiences provided by the peer: Peer ID1: University of Cartagena through its research group E-Solutions. These experiences are federated for the company Peer ID2: Colcomputo Ltda, to integrate them into their LMS and make them visible to their employees to qualify them in technologies that support business management.

The training of employees within the business field is a key factor the moment of achieving quality in the different processes within the company. The objective of supporting business training through the Federation System Services, is to support the specific function of the human resources department, strengthening and supporting the training of human talent. This enriching experience acquired by users through the Federation system is another feature provided by Web 2.0 technologies.

Besides experiences federated by Peer ID1, you can federate learning experiences that have been generated within the various departments of the Peer ID2, enabling horizontal scalability of learning content, so the instructor or teacher in this case can aggregate information from these courses (federated course), which are found on different platforms to present a curriculum appropriate to the profile of each worker. This system provides adequate control to unique data sources that are federated and growing in a measure that are appropriate through Web 2.0 technologies.

Currently there are managed several LMS, Learning Management Systems, including Moodle and ATutor found; these in order to offer the user a tracing in their training or qualifying process, each LMS have different courses which can be developed by employees.

When workers begin a cycle of training within the company, they are obliged to enroll in several courses, which don’t need to develop all the readings and activities, in some cases a course is complementary to others at the same time and within the development they shall be submitted to other courses.

The problem that occurs in this situation lies in the dispersion of information that is posted on the course, because a course may be integrated of one or more files that are stored in any of the LMS, which means, that learning content of a course can have different data sources, which hinders access to information by employees, making it necessary to federate these sources.

With the federation system is proposed a new alternative mechanism that allows all users that are part of the federation to have an easy access to contents that are managed and are part of several virtual platforms, with Web 2.0 technologies that is possible, as detailed later.

Through the service federation system, is offered a solution that provides integration of content from the various courses of the LMS used. For this it is necessary that each of the nodes where there are installed the LMS to be included in the federation as a Federated more, and thus use the database application that handles each of them. The following graph (Figure 7) illustrates the situation:

Figure 7. Federated course through the system of services federation support by web 2.0 technology.
LMS can link to resources that are out of the database or make link to other pages for the student to perform the reading of that material. Figure 8 shows a description of federated resources and the relative url to integrate it within various LMS:

Figure 8. Information to access to federated contents from different data sources.

Thus the user the moment he access this web address will be consuming a service through the middleware provided by the service federation system. The following diagram (Figure 9), shows how a course in an LMS, Moodle for this event, which has integrated federated resources from other peers and can view them within their interfaces:

Figure 9. Federated course from Moodle, supported by the system of services federation.

4. CONCLUSION

As a result of the use of federation of services architecture has been found that is useful, for various reasons, among them are:

Web 2.0 technologies implemented through the federation model can use the collective intelligence of the masses, through the use of learning content for each organization creates and makes available to other federated. This allows virtual learning platforms are not seen only as a technological product, but as service platforms that can be structured through the various learning resources, for example, joint curriculums, educational social networks, open learning communities, among others, key fundamentals of Web 2.0.

The implementation and integration of Federated Systems Services, ensures the possibility of structuring a curriculum in an LMS with the specific information for each of the employees of the organizational levels of the company (according to their occupational profile), federating information found in any other LMS repositories used to support the training process, all transactions and information management that this process leads are grounded in the principles of Web 2.0 to use as a resource handling technologies such as XML and Web Services.

Web 2.0 technologies that support the implementation of the model of federation of services in addition to facilitating the exchange of information and resources as well as dissemination of results and work, provide new avenues for the evaluation, enhance collaboration and cooperation between those who want to learn, opens possibilities for the formation of learning networks, enables an e-learning model more participative, collaborative and social and the possibility of networking collaborative and modular tools with low-cost, a key aspect of Web 2.0 philosophy.

The use of Web 2.0 technologies, allows the implementation of content syndication as a means of propagation, which takes advantage of the potential of centralization and extension of corporate information system that provides a federation of services, providing outreach and information transmission useful for the human component of the company, which ultimately enables the organization to increase productivity, since the possibility of making decisions rises in terms of less time, reducing operational cost and simultaneously increases activity of the enterprise. Thanks to this light programming mechanism that provides Web 2.0 with the utilization and Web Services (RSS), the Federation Service System accomplishes this task easily.

Besides the above, among the advantages of syndication in such scenarios are:

- The tracing of information concerning new learning experiences; with it, users (students) can keep up with the main published results and save considerable time and effort in finding and locating this new information.
- The web retransmission, which through the XML format (Web 2.0 technology), it is possible to provide proprietary content to be displayed on other websites in an integrated manner, which increases the value of content management.
- In the specific case of virtual communities management training, its developed a new value chain in the field of organizational communication, as is changing the forms of relationship of employees with corporate information.

5. ACKNOWLEDGMENTS

Thanks to Diana Marcela Mendoza for providing language help.
6. REFERENCES


