SOFTWARE ARCHITECTURE AND DESIGN WITH SEMANTIC WEB TECHNOLOGIES IN VIRTUAL LEARNING ENVIRONMENTS

Rubén Peredo, Escuela Superior de Cómputo (ESCOM)-Instituto Politécnico Nacional (IPN), rperedo@ipn.mx
Iván Peredo, Universidad Politécnica de Querétaro (UPQ), ivan.peredo@upq.mx

ABSTRACT

Software architecture is a principal focus in the software engineering, the development of the software industry, a robust architecture allows to communicate a detailed design, which can be implemented in the coding. Any software independently of its application domain should have in general an architecture design guide, which guides its structure and development, without being excluded the Virtual Learning Environments. The Semantic Web is a crusade to represent data in formats to automate processing, integrating, and reasoning. The Semantic Web is a new programming viewpoint that converts distributed, confusing, and massive data into real solutions, giving to the metrics a new dimension. The proposal based on Semantic Web allows acquisition of information, automating the persistence and retrieval, allowing the integration of different views about the Learner, providing an enhanced visualization of the state about the Learner to the Coach/ Coach Agent, improving the decision making based on Learner's metrics.

Keywords: Software Architecture, Software Design Patterns, Semantic Web, Virtual Learning Environments

INTRODUCTION

The concept Virtual Reality (VR) was apparently coined by Jaron Lanier, the progress of the digital age developing the concept what has come to be called VR, and subsequently conforming the concept called Virtual Learning Environment (VLE), that are described as learning environments mediated by computers and digital technologies called VLE [1]. Web Based Education (WBE) is actually a main research and development area within VLE, some of the advantages of the WBE are: time and place are no longer obstacles.

Software architecture is fundamental in the Software Engineering (SE), the development of the software industry, it has shown that a good design of an architecture can reduce the failure rates in software projects, a good architecture allows a detailed design, allowing a better coding. All software systems continue to grow in complexity, any software independently of its application domain, should have in general an architecture design guide, which guides its structure and development, without being excluded the VLEs, the success of a software depends of its architecture design.

The Semantic Web (Web 3.0) is a crusade to represent data in formats to automate processing, integrating, and reasoning. The Semantic Web is a new programming viewpoint that converts distributed, confusing, and massive data into real solutions, giving to the metrics a new dimension. The applications can add semantics via programming instructions, but there is no a standard to add semantic. The metadata vocabularies are called ontologies, these allow representing and sharing knowledge based on a concrete vocabulary, for the exchange of knowledge, using a concrete communication protocol. The Semantic Web separates domain specific business logic from the program itself. The Semantic Web has a main goal, formalizing the expression of the semantics of the information, each application automatically will be able to manage, understand and reason.

The knowledge base allows incorporating multiples information sources in the proposal, decoupling the knowledge model from the application, allowing sharing among applications and users.

The proposal uses a framework for personalized Knowledge Management (KM), with a set of methods, techniques, and tools with 5 activities: acquire, store, sharing, application and creation; these actions are enabling via software agents, human knowledge processing in computer systems in an environment based on the paradigm WBE [2]. The Multi-Agent System (MAS) currently one of the most important and fast growth areas of Computer Science (CS) [3]. MAS are loosely coupled networks of problem solvers, collaborating to solve problems, beyond the individual capabilities. The problem solvers are called agents, one of the most accepted definitions of agent is given by Jennings & Wooldridge: "An agent is a computer system, situated in some environment, which is capable of

autonomous actions in this environment in order to meet its design objectives" [3]. MAS are used to build distributed applications that enable us to create artificial social systems. Software agents have among others the following characteristics: autonomy, proactivity and communication.

The proposal has the following components: knowledge base, inference engine, knowledge engineering tools, and user interface. The knowledge base includes the domain specific knowledge; it can be represented and stored in the knowledge base using the Resource Description Framework (RDF) [4]. The knowledge is modeled in the proposal as a set of declarations called triples, constituted by three parts: subject, predicate, and object, RDF is a data model used for sharing, it is used to represented triples. The inference engine is based on an inference rule, an inference rule is a form to build new knowledge based on the existing knowledge base using rules and facts. The knowledge engineering tools allow users to store their knowledge within the knowledge base to infer new knowledge based on the existing knowledge base via a machine learning process. And finally the user interface is used to act together user and system.

RESEARCH METHODOLOGY

Our research centered upon the following research question: How can the educational institutions take advantage of the Semantic Web technologies to develop their educational materials and platform?

The research hypothesis to be tested is the following:

H_1: Educational institutions are not taking advantage of the Semantic Web technologies to develop their educational materials and platform due to its high complexity.

The hypothesis is accepted, to test this hypothesis shows our innovative proposal to develop educational materials and platform with Semantic Web technologies, making emphasis on the advantages of our innovative approach.

RESULTS

The Agent technology currently is one of the most important and fast growth areas of Computer Science, in recent times, it has recently seen a degree of utilization in business applications. MAS are used in a ample diversity of applications, but implementing the Agent Oriented Programming (AOP) paradigm is not simple, there are a large number of domain independent issues that must be solved. The Java Agent DEvelopment (JADE) framework provides the domain independent of the infrastructure, allowing developers to focus on the problem, JADE is probably the most common middleware implementing the AOP [5]. JADE Extension (JADEX) is a framework for writing rational agents based on Belief-Desire-Intention (BDI) architecture, using eXtended Markup Language (XML) and the Java programming language [6]. The Beliefs in agents based on JADEX are any object stored in a beliefbase, the agent's beliefbase represent its knowledge about the world. The Goals represent the motivations on agent's behavior, to carry out the goals the agent executes plans. JADEX uses the XML file called Agent Definition File (ADF), and Java classes for implementation, to build MAS.

Another import framework is JADEX Webbridge, it allows to combine agent technology and content Web, enabling a better integration between agents and Web application via a Web interface [6]. Webbridge uses the MVC pattern for Web applications.

The MAS has five agents: Learner, Virtual Coach, Coach, Evaluation and Delivery. The Learner agent gathers information concerning the Learner, and persists in RDF triples through the IRLCOO components, they were developed by Peredo et al. [7], they persisting the Learner's information in knowledge bases, and this information depicts the Learner's behavior in the VLE, enabling us to build new knowledge based on the existing knowledge base inferring via inference engine. The Virtual Coach agent contributes with dynamic and personalized feedback at Run-Time automatically to the Learners, based on metrics gathered on data bases and knowledge bases (Learning Resources and Learner Records), and their progress. The Coach agent is a assistant for the real Coach, supporting in their basic daily responsibilities, and warning him of students with low achievement of goals. The Evaluation agent constantly evaluates the student and meeting goals. Finally Delivering agent works reconfigure dynamically the sequence and level of course.

Friend Of A Friend (FOAF) ontology is used in our proposal to describe learners, their relations, activities, institutions, and organizations. The FOAF profiles once learners have created them, are automatically added as instances of our ontology.

The FOAF ontology works together within the Semantic Web module mainly implemented with: Java Software Development Kit (SDK) [8], and JENA Semantic Web Framework [9]. The JENA was selected because is a programmatic environment based on RDF, Resource Description Framework Schema (RDFS), Web Ontology Language (OWL) and SPARQL Protocol and RDF Query Language (SPARQL), and has a rule based inference engine, both use the Java programming language.

The RDF persistence is implemented via the JENA framework using IRLCOO components. The abstract model is transformed in a specific format, based on FOAF ontology. FOAF project is creating a Web of machine readable pages describing people, the links between them and the things they create and do; it is a contribution to the linked information system known as the Web [10]. The FOAF documents are serialized in XML syntax, and follows conventions of RDF. The proposal builds a RDF file with Learner's metrics, enabling in our proposal machine readable pages describing Learners, their activities online and their relations with other Learners, so, with Semantic Web technologies enabling us dynamic machine processing. The JENA framework together with IRLCOO components provide the RDF storage and retrieval, as well the interpretation of semantics. The knowledge bases are components that represent our collection of information accessed, described and processed ontologically in our proposal, these are collection of Learner's facts of its experience online, collected using IRLCOO components.

The FOAF module is used for building the model to describe learners, it comprehends personal information, and relationships. The model assists to build a knowledge base based on social attributes, it makes instance data of each learner and their relations through the JENA framework based on FOAF model. The FOAF ontology includes social information that FOAF module shares and combines. It includes learner's information gathered using Learner Agent and IRLCOO components, supplying to the proposal with social information. The FOAF module loads the ontology and instance data, recuperating defined sets of social information from our Web application via SPARQL.

The proposal uses the reasoned JENA to infer social information from models based on the FOAF ontology. The Web application performs inference through the inference engine based on JENA framework, that enable infers new information based on knowledge bases. The Web application gathers learner's scores from different module.

One of the worst enemies of software developers is the change, the best way to manage change is through a good design that allows managing changes and adaptations easily. The Software Design Patterns allow to manage better the change, because are recognized solutions, upgraded over time, to build better architectures. The Software Design Patterns capture knowledge of experts, allowing us to apply a general solution to a particular case with excellent results.

In previous work we have used Software Design Patterns, building betters architectures, implemented the following Software Design Patterns: Composite, Factory Method, Template Method, Decorator, Model-View-Controller (MVC), Observer and Singleton. The Composite pattern allows us to build complex components based on simpler components at Run-Time, assembling composite components for content and evaluation. The Factory Method patter allows to decouple the objects creation, unspecified the precise class of object to create. The Template Method pattern is a sequence of operations to achieve a objective, establishing a set of steps and blocking the sequence, enabling changes through inheritance. The Decorator pattern adds responsibilities to an object without aggregating these responsibilities to the entire class. The MVC pattern is a set of multiple patterns working together, separating the application in three parts: Model, View and Controller, maximizing the reuse of parts and providing maintenance, enabling us to build complex applications. The Observer pattern allows to keep updated the View and the Model within the MVC pattern. The Singleton pattern ensures a single instance of a class and a unique global access to the object, optimizing the resources [11].

This proposal has added two new Software Design Pattern: Command and Interceptor. The Interceptor pattern is a way to expand a normal processing cycle, allowing adding new processes, two of the key points are: the change is transparent and its use is automatic, furthermore the typical block of code to be executed does not have to know the

blocks added or changed, allowing the integration of additional services codes without the core architecture has to be modified. The interceptor pattern is a way to augment a normal processing cycle, allowing adding new processes. The Intercept pattern in the proposal allowed us to have an architecture: flexible and extensible.

The Command pattern allows to a client to make requests to an object without making any assumption about the request or the object received. The Command pattern has the following characteristics: scalability, flexibility, encapsulation of behavior in an object of portable command, decoupling the classes and methods of those classes that execute the behavior required, the creation of new behaviors dynamically create new command objects and assigning them to callers at Run-Time. The Command pattern is fundamentally based on the encapsulation. The Command pattern embeds the behavior in the command objects. The commands are executed by calling the *execute ()* method within the command object, classes are delegated to perform the behavior, and hides what methods of those classes that implement the behavior, decoupling the code that invokes the behavior of the code that implements the behavior, the decoupling makes commands extremely portable. A very important feature of the pattern is the possibility that a command object can be assigned dynamically to other callers. The Command pattern in the proposal allowed us to have an architecture: scalable and flexible. The implementation of the Command pattern was made in the navigation component.

The proposal has maintained support for the initiative Advanced Distributed Learning (ADL)- Sharable Content Object Reference Model (SCORM) [12], and developed different modules over the years: LMS, Metadata, SCORM, Metadata VoiceMXL, Metadata XML, Metadata RDF, File Upload, Dynamic Sequencing, Dynamic Composition, Dynamic Feedback, etc.

The proposal uses the IEEE 1484 architecture – LTSA. The IEEE 1484 LTSA standard is a high level architecture for Learning Management Systems (LMS) supported education, learning and training that describes the high level system design and the components of these systems. The IEEE 1484 LTSA standard is widely recognized into the Web learning community [13-16]. The processes of the IEEE 1484 LTSA standard were replaced by agents. The implementation of the system was made using the frameworks: JADE, JADEX, and Webbridge [17].

The Figure 1 shows our architecture based design patterns with Semantic Web Technologies, the top of the figure shows the client side, the middle part of the figure shows the server side, the bottom of the figure shows the MAS platform. The client side shows patterns and their relationship with the IRLCOO components, the Software Design Patterns are: Factory Method, Decorator, Template Method and Command. The server side shows patterns and their relationship with the Web applications and middleware, the Software Design Patterns are: MVC, Interceptor, Observer, and Singleton. The MAS platform shows its relations with Webbridge, database layer, knowledge base, and IRLCOO components.

The Figure 2 shows compound IRLCOO components for educational materials (Spanish). The hypothesis was tested with the following results:

1.   The first significant difference when comparing our innovative proposal is that educational materials are of type Rich Internet Applications (RIA).
2.   The second significant difference is that our innovative proposal is based on Software Design Pattern, enabling an innovative flexible and extensible architecture.
3.   The third significant difference is that our proposal is that educational materials have support for reconfiguration, sequencing and dynamic feedback using Semantic Web technologies.
4.   The fourth significant difference of our innovative proposal is the support for ADL-SCORM.
5.   The fifth difference significant difference of our proposal is that educational materials and platform use Semantic Web technologies for reconfiguration, sequencing and dynamic feedback.
6.   The sixth difference significant difference of our proposal is that educational materials and platform have decoupled Knowledge Model in our innovative proposal.
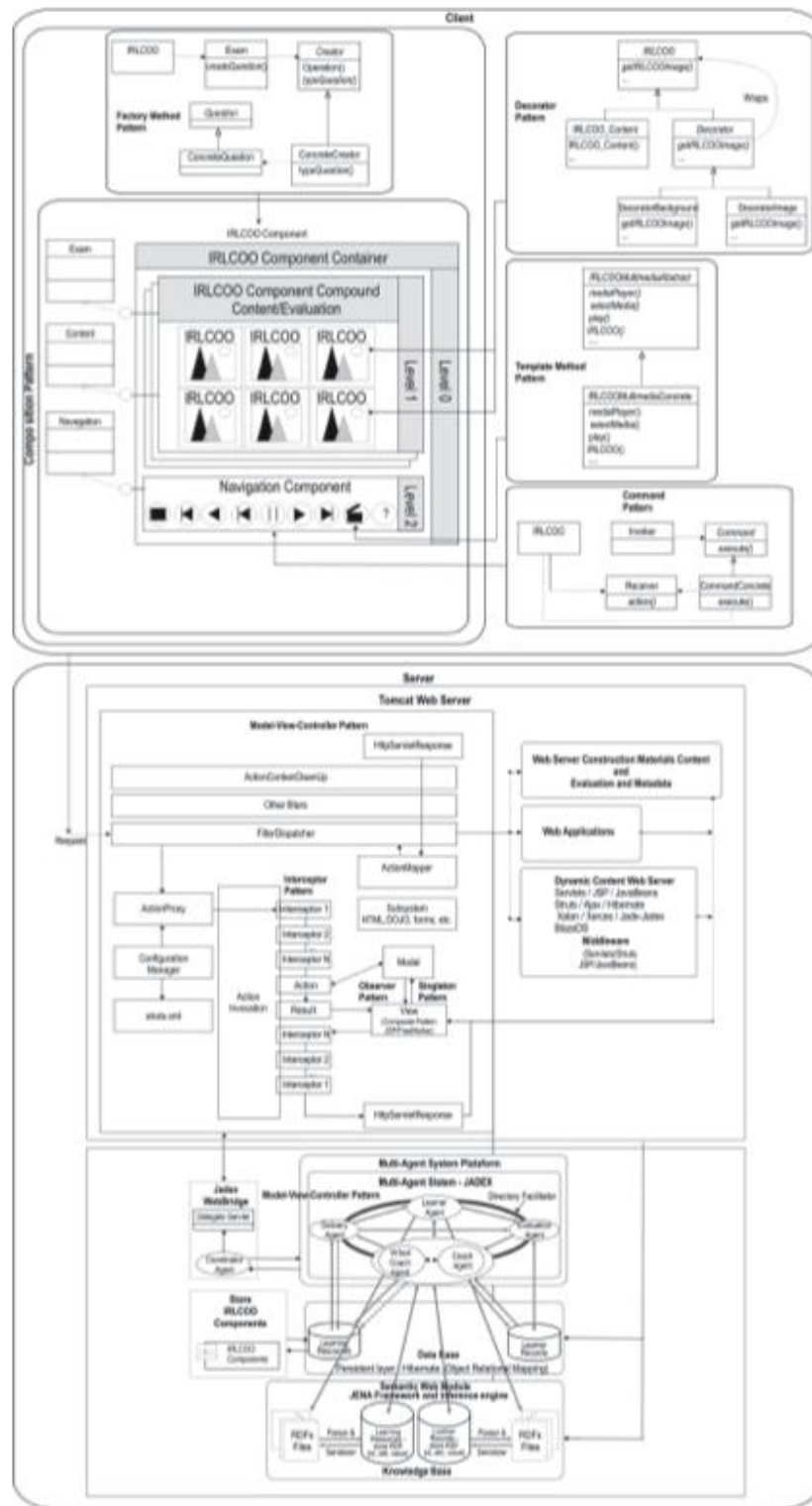
Figure 1. Architecture based design patterns with Semantic Web Technologies for our proposal.

Figure 2. Educational materials (Spanish).

CONCLUSIONS

Educational institutions can take advantage of semantic Web technologies, incorporating them into their educational platforms, in order to improve the tailoring. The use of Semantic Web technologies and ontologies allows to represent and share knowledge based on a common vocabulary with a concrete format, allowing reuse the knowledge model, separating domain specific business logic from the program itself.

Our innovative proposal enhances architecture IEEE 1484-LTSA, integrating Semantic Web technologies and ontologies.

The Web application with Semantic Web technologies allows acquisition of information, automatic persistence and retrieval of the information, enabling the conjunction of different views related to the Learner, given us a better view about the state of Learner through Coach Agent. The use of Semantic Web Technologies, ontologies, and the incorporation of the knowledge base layer, allowed the inference of information through the inference engine.

The Semantic Web (Web 3.0) is a new vision that converts vast amounts of information meaningless, into real solutions, giving to the metrics a new dimension.

The components allow conquering the complexity, allowing the separation of content and navigation of educational materials, enabling configuration/sequencing dynamic at Run-Time, and make possible to tailor the course to the Learner's needs at Run-Time.

Our innovative proposal offers a new vision to address the challenges of developing advanced educational materials and platforms.

The main limitation of the current proposal is the context dependence of the components, depending on the plug-in Adobe Flash Player, for the implementation of Software Design Patterns. The following progress of present proposal focuses in improving of the architecture, adding more Software Design Patterns, and improving the proposed semantic platform, improving the automatic processing of the proposal adding more ontologies.

The information collected from students can help the teacher in the classroom to make better decisions, allowing more advanced students progress at their own pace, and the teacher focus on the students with problems.

ACKNOWLEDGEMENTS

REFERENCES

1. J. Weiss, J. Nolan, J. Hunsinger, P. Trifonas. The international handbook of virtual learning environments. Springer. 2006.

2. R. Peredo Valderrama, A. Canales Cruz, A. N. Ramírez Salazar, and J. C. Caravantes Ramírez. Personalized knowledge management in environments of Web-Based Education. International Journal of Information Technology & Decision Making, Vol. 12, No. 2 (2013) 277-307.

3. N.R. Jennings, K. Sycara, and M. Wooldridge. A Roadmap of Agent Research and Development. Journal of Autonomous Agents and Multi-Agent Systems, 1(1):7–38, 1998.

4. RDF - Semantic Web Standards, URL: http://www.w3.org/RDF/

5. JADE Agent Platform, URL: http://jade.tilab.com/

6. JADEX Agent Platform, URL: http://jadex-agents.informatik.uni-hamburg.de/xwiki/bin/view/About/Overview

7. Peredo, R., Balladares, L., Sheremetov, L.: Development of intelligent reusable learning objects for Web-based education systems. Expert Systems with Applications. 28(2). (2005) 273-283.

8. Java SE - Downloads | Oracle Technology Network | Oracle, URL: http://www.oracle.com/technetwork/java/javase/downloads/index.html?ssSourceSiteId=otnjp

9. Apache Jena - Home, URL: http://jena.apache.org

10. The Friend of a Friend (FOAF) project | FOAF project, URL: http://www.foaf-project.org

11. R. Peredo Valderrama and I. Peredo Valderrama. RIA components and design patterns for the development of educational materials under the paradigm of web based education. Issues in Information Systems, Volume 14, Issue 2, pp.262-268, 2013.

12. Advanced Distributed Learning Initiative, URL: http://www.adlnet.org

13. IEEE Learning Technology Standards Committee. Available at: http://ltsc.ieee.org/

14. IEEE 1484.11.2 Standard for Learning Technology – ECMAScript Application Programming Interface for Content to Runtime Services Communication. November 10, 2003 Available at: http://ltsc.ieee.org/

15. IEEE 1484.12.1-2002 Learning Object Metadata Standard. Available at: http://www.ieee.org/

16. IEEE - The world's largest professional association for the advancement of technology, URL: http://www.ieee.org/index.html

17. Rubén Peredo, Alejandro Canales, Alain Menchaca, Iván Peredo, Intelligent Web-based education system for adaptive learning, Expert Systems with Applications, 38(12): 14690–14702, Pergamon Press, 2011.