Leveraging the Construction of Semantic Web Applications Using the Model Driven Architecture

Leonardo M. Cunha, Simone D. J. Barbosa, Carlos J. P. de Lucena

Computer Science Department – Catholic University of Rio de Janeiro (PUC-Rio)
R. Marquês de São Vicente, 225, Gávea – 22453-900 – Rio de Janeiro – RJ – Brazil
{leocunha, simone, lucena}@ inf.puc-rio.br

Abstract

This paper describes an ongoing research work carried out at the Software Engineering Laboratory of the Computer Science Department at the Catholic University of Rio de Janeiro. This work tries to leverage the construction of Semantic Web applications through the use of OMG's Model Driven Architecture (MDA). An annotation tool was developed to manage the knowledge of a group of researchers and now some tools to access the semantically annotated data need to be developed. For that purpose, the use of MDA seems quite applicable.

Introduction

Knowledge Management (KM) has recently become an important success factor (Sure, Staab and Studer 2003) for several kinds of knowledge-intensive organizations, from enterprises to universities and research groups. However, implementing a KM initiative is both a hazardous and complex task. In this context, an interesting solution is to add semantics to the available data. This will allow applications to reason about the available data and provide more personalized service to users. This solution will probably be achieved through the Semantic Web (Berners-Lee, Hendler and Lassila 2001) (Berners-Lee 2002).

Efficient and effective work on the Semantic Web will be provided by a range of tools that enable the full power of this technology. Fensel *et al.* (2003) identify that it requires the following elements: formal languages to express and represent ontologies; editors to build, merge and reuse ontologies; reasoning services to enable advanced querying and help map between different terminologies; annotation tools to link unstructured and semi-structured information sources with metadata; tools for information access and navigation that enable intelligent information access for human users and translation; and integration services between different ontologies that enable multistandard data interchange (especially for B2B electronic commerce and KM).

This paper describes an ongoing research work carried out at the Software Engineering Laboratory of the Computer Science Department at the Catholic University of Rio de Janeiro. The work tries to leverage the construction of Semantic Web applications through the use of OMG's Model Driven Architecture. An annotation tool was developed to manage the knowledge of a group of researchers. Our next step is to develop some tools to access the semantically annotated data. We believe that using MDA seems quite applicable in this context.

The paper is structured as follows: in the next section, the case study describing the development of the annotation tool is presented. In the following section, some concepts about MDA are presented. Finally, some questions are raised about how to take advantage of MDA and Semantic Web concepts in order to enable the development of Semantic Web applications.

The TecComm Case Study

A case study of the application of the Semantic Web concepts in KM was developed by the TecComm Group – a research group at the Software Engineering Laboratory (LES) of the Catholic University of Rio de Janeiro. This group is composed of undergraduate, graduate and post-graduate students and researchers who carry out several state-of-the-art research projects on multi-disciplinary areas such as E-Business, E-Learning, KM and Multi-Agents Systems, to name a few. Since the group is mainly composed of students and they are frequently completing their courses, the research team has a fast turnover. This sometimes causes a lack of continuity on TecComm projects, since researchers' knowledge can be lost when one of them leaves the group. In order to face this challenge, this group requires a more comprehensive and systematic management of Research and Development (R&D) knowledge.

The case study was based on the design and development of a Semantic Knowledge Portal for research projects to support R&D Knowledge Management for the TecComm Group. The project is based on ontologies, web services and other features of the Semantic Web. The case study encompasses questions about how to implement KM while

redesigning the TecComm web site, so that researchers can share R&D knowledge about their projects.

The semantic annotation allows web applications and software agents to use the TecComm (meta)data in order to infer additional data, to provide services to the TecComm group or to the research community in general. The first step was the creation of the Computer Science Research Projects ontology. This ontology was then mapped into Portalware Framework models followed by the implementation of the Semantic Portalware Web Service.

Once the TecComm Semantic Knowledge Portal was implemented and deployed, TecComm researchers were expected to share information much more frequently, providing up-to-date information about their projects, about themselves and about associated technologies. The development of software agents and services that access the data semantically annotated will leverage the goals accomplished with this case study, adding new functionalities so users can better access and maintain R&D knowledge. But how can it be done? Is there any data that can be inferred from the ontology that can help in constructing those systems? In the next section some concepts about MDA are presented and then questions like those are raised and some future directions of this work are presented.

MDA Concepts

Homogeneity on hardware platforms, operating systems, network protocols and programming languages is something almost impossible to be achieved in the rapidly evolving market. Some of these features are always more appropriate to treat specific problems and to attend to different kinds of missions stated by organizations such as companies or government.

The OMG's Model Driven Architecture (MDA) (Miller and Mukerfi 2003) initiative emerges as a possible solution to tackle the aforementioned problem. The MDA is based on the well-known idea of separating the specification of system functionalities from the details of their implementation on a specific platform. The main goals to be achieved are portability, interoperability and reusability through architectural separation of concerns.

MDA offers an approach to solve the problem by: specifying a system independently of the platform in which it will developed; specifying platforms; choosing a specific platform; and transforming the system specification into one of a particular platform.

In MDA, a viewpoint on a system is "a technique for abstraction using a selected set of architectural concepts and structuring rules, in order to focus on particular concerns within that system. Here 'abstraction' is used to mean the process of suppressing selected details to establish a simplified model" (Miller and Mukerfi 2003). MDA specifies three viewpoints on a system:

(i) a computation-independent viewpoint which focuses on the environment of the system, and the requirements for the system. The details of the structure and processing of the system are hidden or has not been determined;

(ii) a platform-independent viewpoint which focuses on the operation of a system while hiding platform-specific details.

(iii) a platform-specific viewpoint which combines the platform-independent viewpoint with an additional focus on the detail of the use of a specific platform by a system.

A view of a system is also known as a viewpoint model and is a representation of the system from the perspective of a chosen viewpoint. From each of the viewpoints presented before, a view shows a perspective of the system. From (i) the Computation Independent Model (CIM) is "derived", and it does not provide the structure of the systems. But it is assumed that the user of the CIM, the domain practitioner, is not knowledgeable about the models or artifacts used to implement the CIM. A CIM is also called a domain model and a vocabulary that is familiar to the practitioners of the domain in question. This way, the CIM can help those that are experts on the design and development of the artifacts to meet the domain requirements.

From (ii) the Platform Independent Model (PIM) is "derived", and it shows a degree of platform independence so as to be suitable for use in a number of different platforms of similar types.

A Platform Specific Model (PSM) is a view of (iii). It combines the specification of the PIM and the details of a specific platform defined in a specific Platform Model(PM) and how the system uses that platform.

In the next section, we draw a parallel between a CIM and an ontology as a means to describe the knowledge available on the Semantic Web and define how that knowledge can help specify and construct applications to this new area of research through the use of MDA.

Putting Things Together and Emerging Questions

As stated before, an ontology is a formal specification of shared conceptualizations (Borst, 1997), so it is very similar to a CIM. But how can we relate these two terms?

> "An ontology defines the basic terms and relations comprising the vocabulary of a topic area, as well as the rules for combining terms and relations to define extensions to the vocabulary. " (Neches et al., 1991)

Based on it, the Semantic Web expects that systems will be able to infer new (meta)data about statements. Those systems should be "traceable" to (i.e., they should reflect) the ontology or the ontologies that they are working with.

A CIM is a model that presents the system in the environment in which it will operate, and thus it helps reveal exactly what the system is expected to do. In a MDA specification of a system, CIM requirements should be traceable to the PIM and PSM constructs that implement them, and vice versa.

When we approach the TecComm case study from a MDA perspective as depicted in Fig. 1, we find some points that can help leverage the construction of Semantic Web systems through the use of MDA. A CIM looks very similar to an ontology except for the case in which it also presents the requirements of the system being developed. Can we infer these requirements only from the ontology, or do we need more data such as the competence questions defined for the ontology construction? Is it necessary more specific system data? Supposing that an ontology is equivalent to a CIM, how can we map it onto a PIM? Would some other data be necessary? If so, which?

Based on the experience acquired when creating a portal for the Semantic Web, we provided insights to the area and now we are able to face new challenges in the development of applications that can use semantically annotated data. We mean to investigate the use of MDA as a possible form of leveraging the construction of generic Semantic Web applications. The purpose of this paper was to present the MDA while trying to make a parallel between an ontology and a CIM. In doing so, it was possible to identify interesting questions to be answered by research in the project in the near future.

Acknowledgments

This work is partially financed by CNPq: Brazilian National Research Council, under the project Software Engineering for Multi-Agents Systems, number 552068/2002-0 and by individual grants from CNPq.

References

Berners-Lee, T. 2002. Semantic Web Road map. Presentation available at http://www.w3.org/DesignIssues/Semantic.html, last visit: 05/09/2002.

Berners-Lee, T., Hendler, J. and Lassila, O. 2001. The Semantic Web. Scientific American, (May 2001).

Borst, W.N. 1997. Construction of Engineering Ontologies, Techinical Report. University of Twente, Enschede, NL. Center for Telematica and Information Technology.

Fensel, D. Hendler, J., Lieberman, H., Whalster, W. 2003. Introduction. In Fensel, D. Hendler, J., Lieberman, H., Whalster, W. Eds. 2003. Spinning the Semantic Web: bringing the World wide Web to its full potential. Cambridge, Mass: The Mit Press.

Miller, J. and Mukerfi, J. 2003 MDA Guide Version 1.0. Object Management Group. Document Number: omg/2003-05-01.

Neches, R., Fikes, R., Finin, T., Gruber, T., Patil, R., Senator, T., Swartout, W.R. 1991. Enabling Technology for Knowledge Sharing. AI Magazine. 36-56.

Sure, Y., Staab S. and Studer, R. 2003. On-To-Knowledge Methodology. In Staab, S. and Studer, R. Eds. Handbook on Ontologies. Series on Handbooks in Information Systems, Springer, forthcoming.