Semantic Web Technologies in Software Engineering ICSE 2009 Tutorial

Harald Gall
University of Zurich
Department of Informatics
Binzmuehlestrasse 14
CH-8050 Zurich, Switzerland
gall@ifi.uzh.ch

Gerald Reif
University of Zurich
Department of Informatics
Binzmuehlestrasse 14
CH-8050 Zurich, Switzerland
reif@ifi.uzh.ch

ABSTRACT

Over the years, the software engineering community has developed various tools to support the specification, development, and maintainance of software. Many of these tools use proprietary data formats to store artifacts which hamper interoperability. On the other hand, the Semantic Web provides a common framework that allows data to be shared and reused across application, enterprise, and community boundaries. Ontologies are used to define the concepts in the domain of discourse and their relationships and as such provide the formal vocabulary applications use to exchange data. Besides the Web, the technologies developed for the Semantic Web have proven to be useful also in other domains, especially when data is exchanged between applications from different parties. Software engineering is one of these domains in which recent research shows that Semantic Web technologies are able to reduce the barriers of proprietary data formats and enable interoperability.

In this tutorial, we present Semantic Web technologies and their application in software engineering. We discuss the current status of ontologies for software entities, bug reports, or change requests, as well as semantic representations for software and its documentation. This way, architecture, design, code, or test models can be shared across application boundaries enabling a seamless integration of engineering results.

Categories and Subject Descriptors

D.2.0 [**Software Engineering**]: General; H.4.0 [**Information Systems Applications**]: General

Keywords

Semantic Web, Software Engineering, Data Exchange, Interoperation

1. MOTIVATION

Software engineering artifacts can exhibit diverse representations and formats, ranging from architecture and design descriptions to code and tests. Currently, the integration of such artifacts mainly is done within tool environments such as IBM's Rational tool suite or Borland's Together, or between tools with XMI as a data exchange format. But what about reusing artifacts across tools, exchanging results of tools and feeding them into the next tool in the chain? This problem has particularly emerged in the area of software analysis, in which many tools exist that focus on specific analyses. Their integration has not been solved on a semantic level but rather been addressed in XML or GXL as data exchange formats.

As such, software tools keep their proprietary data despite their exportability into XML or GXL (or similar). References from code to design and other artifacts (ie. traceability links) are not represented in a way that enable cross-tool leverage. This is mainly due to the lack of formal semantic representation of the artifacts and their way of "linkage".

Code entities, architectural components, connectors and configurations, patterns, tests, etc. should be represented in a formal semantics to enable their effective interlinking. In the Semantic Web we would have all necessary ingredients for such a "meaningful" representation of software artifacts: ontologies, resource description, ontology mapping and intermediation, description logic, and standards such as RDF, OWL, or SPARQL are used for describing entities on the Web.

The goal of using Semantic Web technologies in software engineering can be summarized as follows:

- Uniform description of software entities on different levels of abstraction
- Exchangeability of software artifacts across tool boundaries
- Facilitation of tool interoperability, especially for a seamless integration of tool results
- Support collaboration of people across organizational boundaries for distributed software engineering

This tutorial addresses the above challenges and aims at presenting a survey of the foundations for Semantic Web enabled software engineering.

2. THE SEMANTIC WEB

In the current Web, Web pages are intended for human consumption. The content is encoded in HTML which provides manly formatting information and makes it difficult for machines to access the semantics of the content. Natural language processing would be needed to do so. To overcome this limitation the Semantic Web aims to make the Web's content machine-processable. Tim Berners-Lee *et al.* define the Semantic Web as an extension of the Web, in which information is given well-defined meaning, better enabling computers and people to work in cooperation [2].

Since the Semantic Web provides machine-processable information based on a formal semantics, data can be shared and reused across application, enterprise, and community boundaries [11]. Therefore, two applications that support the same ontology are able to exchange data even if they were not meant to interoperate in the first place.

To bring the vision of the Semantic Web into being the research community came up with standards, W3C recommendations, development frameworks, APIs, and databases. The Resource Description Framework (RDF) [8] is the data-model for representing meta-data in the Semantic Web. RDF is domain independent in that no assumptions about a particular domain of discourse are made. It is up to the users to define ontologies for the user's domain in an ontology definition language such as OWL [3]. An ontology formally describes the concepts found in the domain, the relationships between these concepts, and the properties used to describe the concepts. For querying RDF repositories, the W3C defined the query languages SPARQL [9].

Besides the Web, Semantic Web technologies have proven to be a useful means to enable the interoperation of software systems also in other domains such as the user's desktop. Semantic Desktop Systems aim to interlink data from specialized desktop applications such as calendar, address book, and email client and provide integrated views on the user's data. For example, the Semantic Clipboard [10] enables copy and paste of semantically rich RDF data between desktop applications without losing the meaning of the data.

3. SEMANTIC WEB AND SOFTWARE ENGINEERING

The problems of the WWW that were the driving force behind the development of the Semantic Web were different from the problems in software engineering as discussed above. The technologies developed for the Semantic Web, however, seem to be a promising solution also in the domain of software engineering. The research community took up this idea and organized a series of workshops such as the workshop on *Semantic Web Enabled Software Engineering* (SWESE) held the third time at the *International* and *European Semantic Web Conference* (ISWC/ESWC).

Although research in this area started recently, first ontologies, research projects and applications point out the potential of using the formal data representation of the Semantic Web in the domain of software engineering. A first overview of applications of ontologies in software engineering is given in [5]. In [7] an extension of the RDF query language SPARQL is used to analyze software repositories based on the EvoOnt ontology [4] to assess the amount of changes between versions or to detect code smells. The bug ontology Baetle [1] provides the vocabulary to represent bug history data from software project.

The goal in the OASIS project [6] is to demonstrate the use of ontologies to enable tool integration in a service oriented environment. [12] tackles the problem in software maintenance to establish and maintain the semantic connections between different software engineering artifacts.

4. SUMMARY

In this tutorial, we present Semantic Web technologies in the context of software design, development, and evolution. We discuss the potential of using RDF as data representation language, introduce ontologies for the software engineering domain, and demonstrate tools that benefit from the usage of semantically rich artifacts.

For that, the tutorial will consist of several parts:

1. Semantic Web technologies
2. Software engineering in the light of Semantic Web technologies
3. Approaches for Semantic Web enabled software engineering
4. Example scenarios and approaches.

5. REFERENCES

[1] Baetle: Bug and enhancement tracking language, Last modified March 02, 2007. http://blogs.sun.com/bblfish/entry/baetle_bug_and_enhancement_tracking.

[2] T. Berners-Lee, J. Hendler, and O. Lassila. The Semantic Web. *Scientific America*, 284(5):34–43, 2001.

[3] M. Dean and G. S. eds. *OWL Web Ontology Language Reference*. W3C Recommendation, 10 February 2004. http://www.w3.org/TR/owl-ref/.

[4] EvoOnt - a software evolution ontology, Last visited October 2007. http://www.ifi.uzh.ch/ddis/msr/.

[5] H.-J. Happel and S. Seedorf. Applications of ontologies in software engineering. In *2nd Workshop on Semantic Web Enabled Software Engineering (SWESE 2006) at the 5th International Semantic Web Conference (ISWC 2006)*, Galway, Ireland, November 11-15 2006.

[6] D. Jin and J. R. Cordy. Ontology-based software analysis and reengineering tool integration: The oasis service-sharing methodology. In *ICSM '05: Proceedings of the 21st IEEE International Conference on Software Maintenance (ICSM'05)*, pages 613–616, Washington, DC, USA, 2005. IEEE Computer Society.

[7] C. Kiefer, A. Bernstein, and J. Tappolet. Mining software repositories with isparql and a software evolution ontology. In *Proceedings of the 2007 International Workshop on Mining Software Repositories (MSR '07)*. IEEE Computer Society, 2007.

[8] G. Klyne and J. J. C. eds. *Resource Description Framework (RDF): Concepts and Abstract Syntax*. W3C Recommendation, 10 February 2004. http://www.w3.org/TR/2004/REC-rdf-schema-20040210/.

[9] E. Prud'hommeaux and A. S. eds. SPARQL query language for RDF. W3C Recommendation, 15 Jannuary 2008. http://www.w3.org/TR/rdf-sparql-query/.

[10] G. Reif, G. M. Laube, K. Möller, and H. Gall. SemClip - overcoming the semantic gap between desktop applications. In *5th Semantic Web Challenge at the 6th International Semantic Web Conference (ISWC)*, Busan, South Korea, November 11-15 2007. Springer-Verlag.

[11] World Wide Web Consortium (W3C) Semantic Web activity homepage. http://w3c.org/sw.

[12] R. Witte, Y. Zhang, and J. Rilling. Empowering software maintainers with semantic web technologies. In *4th European Semantic Web Conference (ESWC 2007)*, Innsbruck, Austria, 3-7, June 2007. Springer Verlag.