SEMANTIC WEB IN EEG/ERP PORTAL
Extending of Data Layer using Java Annotations

Petr Ježek and Roman Mouček
Department of Computer Science and Engineering, University of West Bohemia
Univerzitní 8, 306 14 Pilsen, Czech Republic

Keywords: Semantic web, EEG, ERP, Experiment, OWL, RDF, Object-oriented code, Mapping, Semantics, Transformation, EEG/ERP portal.

Abstract: Because the Semantic Web uses its technologies for presenting data/metadata on the web and common systems are based on object-oriented languages a need for suitable mapping is emerging. This paper describes the difficulties during transformation of data layer represented by object-oriented code into the semantic web structures (OWL, RDF). Since there is difference between semantic expressivity of these data representations it is necessary to fill this semantic gap. Authors investigate these differences in semantics and provide a preliminary idea to add missing semantics into the Java code using Java annotations. These annotations are consequently processed by the proposed framework. The transformation is demonstrated within the EEG/ERP Portal.

1 INTRODUCTION

Our research group at Department of Computer Science and Engineering, University of West Bohemia specializes in the research of brain activity using the methods of electroencephalography (EEG) and event related potentials (ERP). Because large data are obtained during EEG/ERP experiments we were facing a problem with their long term storage and management. We solve this problem by developing a custom solution (named EEG/ERP Portal) based on a relational database, application logic and a web interface. The EEG/ERP Portal enables registered users to store and manage experiments (including raw data, metadata or experimental scenarios).

Since we currently work on the registration of the EEG/ERP Portal as a recognizable neuroscience data source within the NIF portal (NIF, n. d.) we need to describe EEG/ERP experiments using a suitable ontology.

Since the current World Wide Web lacks a description of data semantics (expressed by ontologies) an extension called the Semantic web (Berners-Lee, 2001) is being developed. The Semantic web uses a triple oriented representation described by Resource Description Framework (RDF). Since expressivity of RDF schema is insufficient, an extension called Ontology Web Language (OWL) was introduced. Both RDF and OWL can use XML syntax; richer semantic relationships can be expressed by OWL.

Despite advantages that the Semantic web brings, current software systems are usually developed using object oriented languages. Data are usually stored in a relational database. Since there are differences in semantic richness between these approaches it is necessary to ensure a suitable mapping.

This paper introduces EEG/ERP domain ontology with online data/metadata serialization. Some semantic gaps between object-oriented code and its semantic web representation are bridged by extension of object-oriented code using Java annotations. The mapping of these annotations to the Semantic web output is provided by the transformation framework described in this paper.

2 EEG/ERP PORTAL

The EEG/ERP Portal is a solution using a layered architecture according to the MVC design pattern. Data are stored into the relational database (Oracle 11g). The persistent layer is created by POJO

350 Ježek P. and Mouček R..
SEMANTIC WEB IN EEG/ERP PORTAL - Extending of Data Layer using Java Annotations.
DOI: 10.5220/0003731303500353
In *Proceedings of the International Conference on Health Informatics* (HEALTHINF-2012), pages 350-353
ISBN: 978-989-8425-88-1

objects. Object-relational mapping is ensured by the Hibernate framework.

The application layer ensures processing of user requests, managing user accounts and user groups according to user privileges. The layer is implemented using the Spring framework (includes Spring Core, Spring Security and Spring MVC). This layer also contains a module for the dynamic generation of the Semantic web structure from data stored in the database mapped to POJOs according to a search agent request.

The presentation layer is created by a set of JSP pages translated into a HTML output shown in a user web browser.

The EEG/ERP Portal preview is in Figure 1, the full system specification is in (Ježek and Mouček, 2009).

Figure 1: EEG/ERP Portal

3 OBJECT ORIENTED CODE TO SEMANTIC WEB MAPPING

3.1 OWL and OOP Differences

There are fundamental differences in richness of semantics between OWL and OOP. OWL is based on description logic and an open world assumption while OOP deals with closed world assumptions. It means that if some information is not present in an object oriented code it is considered to be false. By contrast if some information is not present in an OWL document, it may be true.

3.2 Framework Selection

We tested several frameworks that provide a mechanism for transforming an object oriented code (classes and their attributes) to an RDF or OWL output. From the tested tools we selected JenaBean (JenaBean, n. d.) and OWL API (OwlApi, n. d.). This selection and integration was described in (Ježek and Mouček, 2010) and (Mouček and Ježek, 2010). Using selected tools we are able to sufficiently transform data from our EEG/ERP portal to the Semantic web output (one side transformation is used). Since the object oriented model has poorer semantics, capabilities of OWL are not used. The next sections of the paper describe a semantic extension of object-oriented code.

3.3 Existing Approaches

There are many frameworks that try to enrich an object oriented code by adding missing semantics. (Oren et al., 2007) developed a library ActiveRDF for accessing RDF data from Ruby programs.

(Po-Huan et al., 2009) present the Semantic Object Framework (SOF) as a solution that uses embedded comments in source codes to describe semantic relationships between classes and attributes.

(Liu et al., 2007) developed a solution named eClass. EClass is a Java class with changed syntax containing additional semantic information about methods and properties.

The tested frameworks have several important disadvantages. The most of them are difficult to use either because they do not include suitable means for adding missing semantics, or they require a modification of common programming syntax.

4 ANNOTATION FRAMEWORK

The specification of requirements originated from the needs of the Semantic web output provided by our EEG/ERP Portal. We expect that our solution could serve a wide community of researchers that use object oriented systems and need to generate ontologies. To support this idea we decided to use only the standard syntactic structures of the Java programming language for data input, which is extended by missing semantics, and the standard Java Runtime Environment for the transformation process.

We introduced an idea to extend the JenaBean framework in (Ježek and Mouček, 2011). As the next step we proposed a framework using Java Annotations that extend the semantics of Java classes and fields. This approach ensures that our framework could be used not only by the experts in the Semantic web field but also by the software engineers who are able to prepare data objects suitable for the transformation to the Semantic web output. In addition, when a user has difficulties with

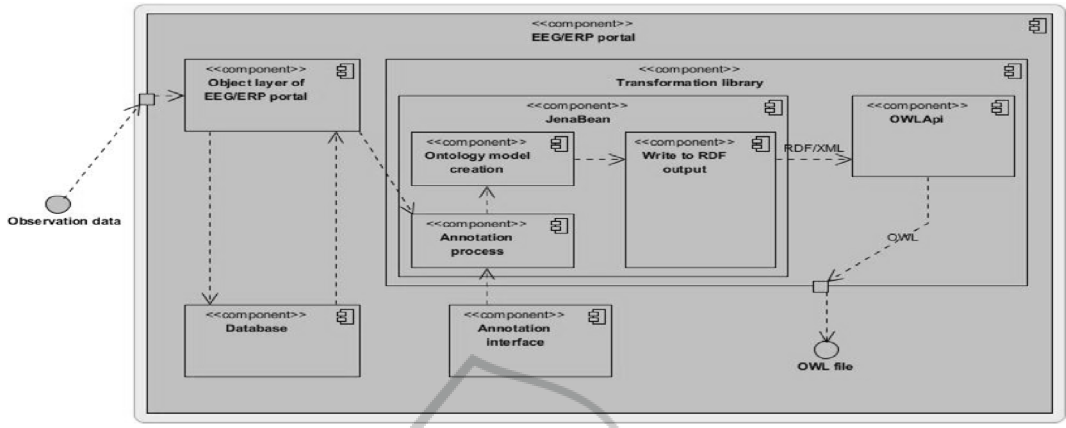

Figure 2: Library integration within EEG/ERP Portal.

a manual annotation of classes we provide a graphical tool which can be used for loading Java classes and annotating them easily.

4.1 Framework Integration

Figure 2 shows the main components of the library and its integration within the EEG/ERP Portal. The transformation library includes modified JenaBean and the OWL API. This component serves as a black-box; we provide it as a single library, so it could be used in various systems.

4.2 Transformation Library

The transformation library processes input POJO objects and serializes them into an output OWL structure. This processing includes parsing of Java classes (with their methods and attributes) using the Java Reflection API, creating an internal ontology model and writing an output RDF model. This RDF model is later processed by the OWL API, which generates the corresponding OWL output. In addition, the transformation into the internal ontology model includes processing of Java annotations and enrichment of the internal model.

5 PRELIMINARY RESULTS

5.1 OWL Constructs

We enriched the JenaBean framework by designing and implementing a new set of transformations from Java annotations into OWL constructs, e. g. Property restriction (allValuesFrom, someValuesFrom, cardinality), Versioning (versionInfo, priorVersion, incompatibleWith...), and Annotation properties (label, comment, seeAlso ...). A more detailed specification of these constructs is in (OWL, n.d.). In the next section we describe an example of mapping of the selected OWL constructs.

5.2 Mapping Example

Let us suppose a Java class named Person. We can e.g. annotate this class using @RdfType that determinates the name of the class in the output OWL structure. In addition we can define e.g. Equivalent construct (@EquivalentClass annotation). Later we can e. g. determinate the corresponding class using @SameAs annotation. Specifically, suppose the following Java class:

```
@RdfType("Person")
@EquivalentClass("http://www.kiv.zcu.cz
/measured_person")
@SameAs("http://www.kiv.zcu.cz/eegerpPe
rson")
public class Person {

    @Id
    private int personId;
    private String givenname;

//get and set methods
}
```

This code is serialized into the output OWL structure where we can see the following constructs:

- Declaration of the class name:

```
    <Declaration>
        <Class
IRI="http://www.kiv.zcu.cz/Person"/>
    </Declaration>
```

- Definition of equivalent classes:

```
    <EquivalentClasses>
```

```
        <Class
IRI="http://www.kiv.zcu.cz/Person"/>
        <Class
IRI="http://www.kiv.zcu.cz/measured_per
son"/>
    </EquivalentClasses>
```

- Definition of the same individuals:

```
  <SameIndividual>
   <NamedIndividual
IRI="http://www.kiv.zcu.cz/Person"/>
   <NamedIndividual
IRI="http://www.kiv.zcu.cz/eegerpPerson
"/>
   </SameIndividual>
```

6 CONCLUSIONS

Currently the domain description using a specific ontology is discussed in many scientific fields. These ontologies can serve as structures for recognizable data sources. The Semantic web, which is based on domain ontologies, is suitable for machine processing. OWL is used for expressing the Semantic web. Although the idea of the Semantic web is promising several disadvantages have to be solved. Mostly OWL is not considered to be a software programming language in contrast with the object oriented programming that is not intended for construction of ontologies.

Since common software systems are usually based on object oriented languages, scientific community is facing the problem how to ensure a suitable mapping.

We had to deal with the same issue when ontology in EEG/ERP domain was built. Only one side transformation (from an object oriented code to an ontology web language) was solved (it is easier than any reverse transformation). However, there are several semantic gaps that are needed to be solved. We investigated several existing solutions described in this paper.

We presented a custom solution based on Java annotations that adds missing semantics into an object oriented code and developed a framework that processes such a code and provides an OWL serialization. The integrated JenaBean framework was enriched by a new set of transformations into OWL constructs. Our solution uses a common programming language and common technologies; hence it can be immediately deployed.

Our future work includes a full registration of the EEG/ERP portal as a recognizable data source within the NIF portal. Simultaneously we plan to provide and implement a full description of all OWL constructs using Java annotations.

ACKNOWLEDGEMENTS

The work was supported by the UWB grant SGS-2010-038 Methods and Applications of Bio- and Medical Informatics

REFERENCES

Neuroscience Information Framework (NIF). (n. d.). Retrieved May 20, 2011, from http://www.neuinfo.org/

Berners-Lee, T., 2001. The Semantic Web. In *Scientific American* (pp. 34-43).

Ježek, P., Mouček, R. 2010. Database of EEG/ERP experiments. *In HEALTHINF 2010 - Proceedings of the Third International Conference on Health Informatics*. (pp. 222-227).

Mouček, R, Ježek, P., 2010. System for Storage and Management of EEG/ERP Experiments – Generation of Ontology. In *12th International Conference on Enterprise Information System, Volume 1* (pp. 415-420)

Ježek, P., Mouček, R., 2010. EEG/ERP Portal – Semantic Web Extension. In *Second Global Congress on Intelligent Systems* (pp. 392-395)

Oren, E., Delbru, R., Gerke, S., Haller, A. and Decker S., 2007. ActiveRDF: object-oriented semantic web protramming, *In Proceedings of the 16th international conference on World Wide Web* (pp. 817-824)

Po-Huan, Ch., Chi-Chuan, L., Kuo-Ming, Ch., 2009. Integrationg Semanic Web and Object-Oriented Programming for Cooperative Desig, In *Journal of University Computer Science, vol. 15, no. 9*

Liu, F., Wang, J., Dillon, S. T., 2007. Web Information Representation, Extraction and Reasoning based on Existing Programming Technology, In *Computational Inteligence 37* (pp. 147-168)

JenaBean, n.d. Retrieved May, 2011, from http://www.ibm.com/developerworks/java/library/j-jenabean.html

OwlApi, n.d. Retrieved May, 2011, from http://owlapi.sourceforge.net/

Ježek, P., Mouček, R., 2011. Transformation of object-oriented code into semantic web using Java annotations. In *13th International Conference on Enterprise Information System,* (pp. 207-210)

OWL Web Ontology Language Overview (OWL), n.d. Retrieved May, 2011, from http://www.w3.org/TR/owl-features/