A System to enable Relational Persistence and Semantic Web style access simultaneously for Objects

Jing Mei, Guotong Xie, Shengping Liu, Lei Zhang, Yue Pan
IBM China Research Laboratory
Zhongguancun Software Park, Beijing, 100193, China
{meijing, xieguot, liusp, lzhangl, panyue}@cn.ibm.com

ABSTRACT
Object Oriented (OO) programming is dominant in the current software development. Starting from the design of OO models for applications, developers also expect to address issues on the data of models and the semantics of models. Objects, being the data of models, could be stored in relational databases, and ontologies appear as a good candidate for capturing the semantics of models. This poster presents a method and system which elegantly generates relational schema, OWL ontology, and semantic mapping between them, for any given OO model. The resulting relational schema serves for storing objects that are defined in the input OO model, the resulting OWL ontology is assured by a semantically "close" model transformation, and the generated automation mapping between them enables relational persistence and Semantic Web style access simultaneously for objects.

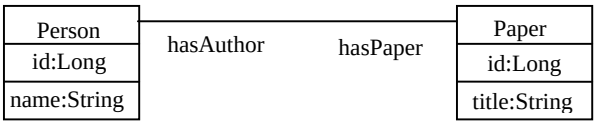
Categories and Subject Descriptors
H.4.m [**Information Systems Applications**]: Miscellaneous.

General Terms
Management, Design, Experimentation.

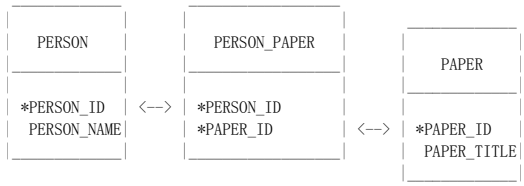
Keywords
Object Relational Mapping, Model Transformation, Semantic Mapping.

1. INTRODUCTION
Object Oriented (OO) programming is dominant in the current software development. Starting from the design of OO models for applications, developers also expect to address issues on the data of models and the semantics of models. Objects, being the data of models, could be stored in relational databases via OR (Object Relational) mapping systems [1], which first generate relational schemas for OO models and then convert objects to relational data. Capturing the semantics of models, ontologies appear as a good candidate, and the Semantic Web (SW), an emerging technology to build semantic-based systems, recommends the Web Ontology Language (OWL) for knowledge representation and reasoning. As a result, the model transformation arises, provided by ad-hoc transformation engines between OO models and OWL ontologies [3].

Although, the above two issues seem to have solutions respectively, a framework involved both is still a challenge. Given OO models, relational data via the OR mapping does not correspond to any ontology, despite there are ontologies generated by the model transformation. Admittedly, tools like the D2R server [2] are available for publishing relational databases on the Semantic Web, which means a solution to bridge the relational data and ontologies. However, such techniques do not address our problem as a whole. Because an ontology, being published from relational schema by the D2R-like tool automatically, is NOT always semantically "close" to its native OO model. For instance, we build two OO classes, viz. Person and Paper, between which a bi-directional association holds (as shown below).



Most possibly, the relational schema generated by OR mapping tools looks like:



Using the *generate-mapping* script provided by the D2R server, a D2RQ mapping would be automatically generated from the schema structure of a database, where table names become class names and column names become property names in a relatively straightforward manner. Therefore, besides two OWL classes of PERSON and PAPER, a third OWL class PERSON_PAPER also arises but unfortunately does not fit well with the OO model semantics. Undoubtedly, a more reasonable D2RQ mapping could be defined in manual by human understanding the original OO models. However, an automation semantic mapping is desirable, considering that a semantic correspondence between OWL ontologies and OO models has been established. That is, running a model transformation engine properly receives OWL classes PERSON and PAPER without PERSON_PAPER. In this respect, ontologies published by the D2R server in an automatic manner are (most possibly) different from or worse than those being transformed from OO models.

Meanwhile, we realize an ontology repository [4] might be employed to store OWL ontologies which have been transformed from OO models. However, its relational schema is now being reduced as plain as Triple(subject predicate object) which more or less charges OO programming.

Consequently, this poster presents a system, which elegantly generates relational schema, OWL ontology, and semantic mapping between them, for any given OO model. In a practical

perspective, assuming that OO models for applications have been designed well, developers of OO programming would make use of OR mapping tools to generate relational schemas for those OO models, followed by deploying generated schemas into relational databases and then storing objects into databases. Since relational databases have been established, access to such data is quite attractive for the Semantic Web community. Recalling those designed OO models could be transformed into OWL ontologies, Semantic Web users would employ semantic mapping tools to do ontology query answering on relational data for OO models. Instead of specifying the semantic mapping manually, our proposed system would generate the semantic mapping files automatically.

2. SYSTEM ARCHITECTURE

Figure 1 is the system architecture, where the dashed rectangle is the boundary of our system. There are three components, namely OR Mapping Generator, Model Transformer, and Semantic Mapping Generator. Beginning with input of OO models, the OR Mapping Generator generates files of OR mapping and relational schema, while the Model Transformer yields a corresponding ontology file. Ending up, these three kinds of files are fed into the Semantic Mapping Generator, with output of the fourth file, namely the semantic mapping.

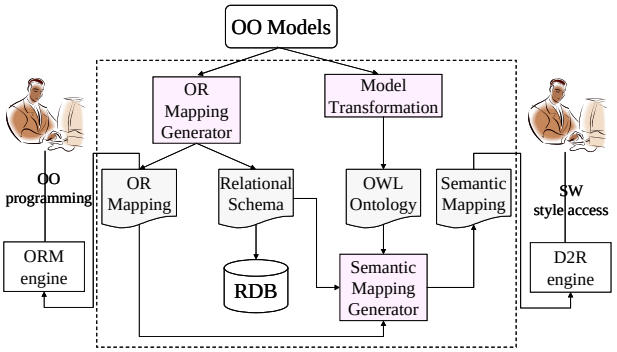


Figure 1. System Architecture.

Back to the above PERSON_PAPER example, our system would generate the relational schema as previously, which enables relational persistence for objects in OO programming. The OWL ontology would also be received as expected, without any OWL class for the intermediate tables. More importantly, this proposed system automatically generates D2RQ mappings in a novel way so as for a Semantic Web style access to objects which have stored in relational databases.

3. PROTOTYPE IMPLEMENTATION

Being a prototype implementation, we experiment on Ecore models in the Eclipse Modeling Framework (EMF) [4], using XMeta [5] for the OR mapping generator and EODM [3] for the model transformation. The starting point could be design of UML models using IBM Rational XDE tool, and then an XDE2Ecore Eclipse Plug-in would validate and convert XDE (UML) models to Ecore models. On the one hand, using XMeta, Ecore models are given as input to Java code generation (including Java interface and implementation class for Ecore model), then to package compilation and registration in the repository. On the other hand, using EODM, a specified Ecore2OWL transformer outputs an OWL file with input of an EPackage name. Finally, a D2RQ mapping file of class maps and property bridges is outputted by our semantic mapping generator.

It should be pointed out that, special for application modeling, package dependency exists, e.g., from EClass or EReference in one EPackage to EClass in another EPackage. Taking strategies into account, our system is configurable for ignoring all dependency, or (partially) including all dependency packages.

4. RELATED WORK AND CONCLUSION

Observing that OO models are still the key to development of real-world applications, many popular OR mapping products are often used for object persistence, and meanwhile, various OWL ontologies are also ready for publishing to express the underlying OO model semantics. How to enable relational persistence and Semantic Web style access simultaneously for objects is the problem being addressed in this poster, and related work could be roughly divided into: (1) approaches for publishing relational databases on the Semantic Web; and (2) approaches for storing ontologies which represent the OO model semantics. As mentioned above, using approaches (1) has to manually define semantic mappings because relational schemas, in that context, are not necessarily machine understandable for making semantics of OO models recurring. Similarly, using approaches (2) fails to an automatic communication with OO programming, because relational schemas, in this context, are more ontology-oriented rather than object-oriented. Indeed, we do not lose sight of a third candidate, that is, a direct mapping to bridge OO programming and Semantic Web style access. However, we argue that, due to a great diversity of OO programming languages, it is non trivial for a common and unified access to objects. In other words, there is none standard query language for OO programming, such as SQL for relational databases which makes D2RQ mapping doable.

As a conclusion, we propose the method and system to enable relational persistence and Semantic Web style access simultaneously for objects. Also, we conduct initial experiments on Ecore models, with experiences sharing for the prototype implementation.

5. REFERENCES

[1] HIBERNATE – Relational Persistence for Idiomatic Java. http://www.hibernate.org/hib_docs/v3/reference/en/html/index.html

[2] D2R Server – Publishing Relational Databases on the Semantic Web. http://www4.wiwiss.fu-berlin.de/bizer/d2r-server/index.html

[3] EODM – EMF Ontology Definition Metamodel. http://www.eclipse.org/modeling/mdt/?project=eodm

[4] EMF – Eclipse Modeling Framework. http://www.eclipse.org/modeling/emf/?project=emf

[5] Scheffler, L. 2006. XMeta Repository and Services. In Conceptual Modeling – ER 2006. Springer, LNCS 4215, 566. DOI= http://dx.doi.org/10.1007/11901181_43