

A Linked Data Web Programming Framework

Matthias Quasthoff, Christoph Meinel

Hasso Plattner Institute, University of Potsdam
{matthias.quasthoff, christoph.meinel}@hpi.uni-potsdam.de

1 Abstract

Exchange of information across different information systems is often desirable. Following linked data principles [1] is suitable for various reasons: they build upon the versatile URI Web standard and the generic RDF information model.

Many modern Web applications are usually being developed with the help of so-called next-generation Web programming frameworks. These frameworks strictly follow the Model View Controller (MVC [2]) pattern and usually support relational persistence. If at all, many developers choose to “triplify” the relational data stored by the Web application in order to produce Linked Data [3]. While this approach has been necessary to start producing increasing amounts of Linked Data, it does not allow for a tighter integration of Linked Data in the application itself. In our contribution we show that using the known method of Object Triple Mapping (OTM, [4]), such tight integration of Linked Data into Web programming frameworks can be achieved.

2 Related work

This work builds upon the RDF data model and the PathLog language for describing object-oriented concepts, which has been used in the Linked Data context first by Oren et al. [5]. There is an RDFa plugin for the MVC-based Rails Web framework¹, which however does not really integrate with the underlying application. The idea of mapping object-oriented domain models to RDF schemas is inspired by object-relational mapping [6] and has been described and implemented among others by Story [7] and Oren et al. [5]. An overview on Object Triple Mapping (OTM) implementations is found at the Tripresso site². OTM helps developers to focus on implementing actual business logic [4].

3 Contribution

Formal description. Given a set O of objects, a set F of field names, and a set U of URI, an object triple mapping is characterized by a vocabulary map

¹ <http://rdfa.rubyforge.org/>

² <http://semanticweb.org/wiki/Tripresso>

$m_t : F \rightarrow U$, an instance map $m_a : O \rightarrow U$, and an interpretation $I : O \times F \rightarrow 2^O$ [4]. This mapping can now be extended for MVC applications as follows.

The vocabulary map m_t is extended to $F \cup A$ for the set A of controller actions to relate information resources (MVC views) to non-information resources (elements of the MVC model). Also, Linked Data principles require each $o \in O$ being dereferenceable via URI. Hence, we need a fixed action $\text{id} \in A$ such that (id, o) identifies $o \in O$ and the Web application framework redirects requests to (id, o) to an appropriate information resource, e. g., $(\text{show}, o) \in A \times O$.

Implementation. The RDFa plugin for the Grails Web application framework³ conveniently links the domain model of MVC Web applications to RDF vocabularies using the proposed formalism. Consequently, the plugin dramatically simplifies the generation of RDFa from standard HTML templates as follows.

No explicit addressing. Referencing an object itself (`<div about="\${object}">`) will actually render the object's URI using the instance map m_a .

No redundant evaluation. Putting `` will look up the RDF property of the `name` field of the current object using m_t and the actual value for the current resource. Putting `` will look up the property of the `photo` action and link to (photo, o) for the current resource o .

4 Conclusion

Using the formal link from MVC to OTM, Web applications developed with popular MVC Web application frameworks can easily be turned into real Linked Data applications. Also, our extension adds the missing "Web" behavior to OTM, as it has been a purely internal design pattern so far.

References

1. Berners-Lee, T.: Linked data. <http://www.w3.org/DesignIssues/LinkedData.html> (2006)
2. Reenskaug, T.: Thing-Model-View-Editor. an Example from a planning system. Xerox PARC technical note (1979)
3. Bizer, C., Seaborne, A.: D2rq – treating non-rdf databases as virtual rdf graphs. In: Proc. of the 3rd International Semantic Web Conference, Springer (2004)
4. Quasthoff, M., Sack, H., Meinel, C.: How to simplify building semantic web applications. In: Proc. of the 5th International Workshop on Semantic Web Enabled Software Engineering, CEUR-WS.org (2009)
5. Oren, E., Heitmann, B., Decker, S.: Activerdf: Embedding semantic web data into object-oriented languages. J. Web Sem. **6**(3) (2008) 191–202
6. Fowler, M., Rice, D.: Patterns of Enterprise Application Architecture. Addison-Wesley (2003)
7. Story, H.: Java annotations and the semantic web. http://blogs.sun.com/bblfish/entry/java_annotations_the_semantic_web (2005)

³ <http://grails.org/plugin/rdfa>