

Logic and a Little Language for Heritage Resources on the Semantic Web

Andrew Russell Green

Instituto Mora (Mexico City)*

ahg@servidor.unam.mx

Digital libraries stand to benefit greatly from the Semantic Web. Catalogues modeled using semantically-rich data structures should provide a basis for novel end-user functions targeted at research and teaching. However, several problems remain to be solved before this promise is made a reality. In work to develop a scalable digital library application based on a back-end of semantically modeled data, the project *Rescate de la Memoria* (“Rescuing Memory”) has identified various issues that must be addressed, and has proposed viable solutions, currently being implemented in our application, called *El Pescador*.

The major issues we have come up against are:

1. The lack of adequate studies of algorithms for processing information as it flows from the model to the user interface, and, following user input, from the interface back into the model.¹
2. The lack of a unified mechanism for defining vocabulary, model structure and model-interface translation rules.

In response to the first problem, we analyzed model-interface translation from a general perspective and designed configurable algorithms which may be applied to a wide range of semantically modeled catalogue data and record presentation formats. This proposal, called *Knowledge Repository Logic*, applies information engineering best-practice principles, such as encapsulation and modularity, throughout. Our response to the second issue has been to create a domain-specific (or “little”) programming language with which the programmer/model designer may establish vocabulary and model structure and set the parameters of model-interface translation algorithms. This “full-stack” solution allows related aspects of a semantic catalogue implementation to be established together, avoids the repetition of information in this process, and offers simplified access to recurrent model and interface structures, thus providing both easy deployment and scalability, and applying, again, the above-mentioned best-practice principles.

The proposed demonstration will present the specifics of Knowledge Repository Logic, the related little language, and the real-world application that implements them. (The application will be in an alpha stage of development at the time of ESWC 2007.) We hope that the demonstration will provide an opportunity to contact others working on similar issues and get feedback on our proposals.

The remainder of this text explains a few more aspects of our work; preliminary but much more complete documentation is available on-line (in Spanish); see *Pescador: Lógica de RC* <http://200.67.231.185/mediawiki/index.php/Pescador:L%C3%B3gica_de_RC>.²

*Partial funding for this project is provided by the Open Network of Digital Libraries. See <<http://www.onedl.org.mx/>>.

¹One example of work in this area is the Fresnel Display Vocabulary, which provides a “simple, browser-independent vocabulary for specifying how RDF graphs are presented”, however, the mechanisms it establishes as yet cover only a fraction of the processes involved in model-interface translation. See <<http://www.w3.org/2005/04/fresnel-info/>>.

²Some aspects of this proposal will change in the near future. The source code of our application (which currently implements part of the proposal) is available under a free software license from <<http://200.67.231.185/svn/>>.

Export to standard formats

The language we have designed provides facilities for describing things (vocabulary, constraints and paths, for example) that may also be described using existing standards (such as RDFS, OWL, or SPARQL). The current version of our application automatically exports vocabulary definitions to an RDF Schema, and export of other elements to standard formats is a conceivable feature that may be implemented in the future.

Meaningful, scalable path descriptions

Current mechanisms for describing paths between nodes in a model are text-based, relatives of SQL. When the complexity of a model increases and certain segments of complex paths appear repeatedly in several parts of an application, these mechanisms provide insufficient flexibility and scalability. Our proposal allows for the encapsulation and combination of definitions of segments of paths, thus avoiding repetition in, and augmenting the semantic value of, path descriptions.

Strong distinction between interface and model logic

Some schools of thought in Cognitive Linguistics posit a strong distinction between the rules governing expression and those that apply to human mental models. Following the (limited) analogy that may be drawn between knowledge representations and mental models, we have taken a cue from this hypothesis, and have integrated in Knowledge Repository Logic the assumption that a user should not see a direct encoding of the contents of a model, but rather a complex transformation that follows the rules of human communication. In other words, while other Semantic Web projects are rooted in the assumption that language is compositional, we work from the idea that—as Gilles Fauconnier puts it—“expression is not compositional formal encoding that mirrors a compositional conceptual construction.”³

Real-World Application: Heritage Resources

The dissemination, investigation and management of heritage resources presents special challenges that traditional information technology has not been able to meet in a satisfactory manner. Our work on Knowledge Repository Logic, the related little language and *El Pescador* is part of an interdisciplinary research project that aims to use computers in new ways in the realization of these tasks, and in general promotes the conservation of and open access to historical archives.

³Turner, M. and Fauconnier, G. (1995) “Conceptual Integration and Formal Expression?” in *Journal of Metaphor and Symbolic Activity*. Vol. 10, no. 3. Mark Johnson, editor. <<http://philosophy.uoregon.edu/metaphor/turner.htm>>