

Extending Ruby on Rails for Semantic Web Applications

Cédric Mesnage¹ and Eyal Oren²

¹ Faculty of Informatics
University of Lugano, USI
Lugano, Switzerland

`cedric.mesnage@lu.unisi.ch`

² Digital Enterprise Research Institute
National University of Ireland, Galway
Galway, Ireland
`eyal.oren@deri.org`

Abstract. We extend the Ruby on Rails framework towards a more complete Semantic Web application framework. The SWORD plugin provides developers with a set of tools and libraries for managing Semantic Web data and rapid Semantic Web Application development. We describe the functionality of the SWORD plugin and demonstrate its use for rapid development of a social networking application.

1 Introduction

The Semantic Web is a web of data that can be processed by machines, enabling them to interpret, combine and use Web data [1,2]. The Semantic Web offers a uniform way of identifying and describing resources in a distributed environment, and thus increases the interoperability between applications.

On the other hand, web development has evolved recently to a more structured and abstract way of engineering web applications with the appearance of frameworks such as Struts¹, Ruby on Rails² and Django³. These frameworks overcome the problem of the separation of concerns regarding data manipulation, user interaction and business logic by constructing web applications on the model–view–controller design pattern [7,3].

Although these frameworks ease the development of web applications, they offer only limited support for interoperability. Since Web applications are mostly data-driven, we see increasing interoperability and data reuse through “mash-ups” that combine data from multiple Web applications into new functionality. Such “mash-ups” are supported only to a limited extent by existing frameworks. We improve the possibilities of data reuse in Web application frameworks by using Semantic Web data as a basis for application development.

¹ <http://struts.apache.org>

² <http://rubyonrails.org>

³ <http://www.djangoproject.com>

We focus on the Ruby on Rails, an agile development platform for Web applications. Ruby on Rails provides solutions for rapid prototyping and is supported by an active community. Ruby on Rails is currently a popular platform for development of Web 2.0 applications; we show how it can be extended to support Semantic Web application development. In this paper, we review the related work, we present the functionalities and architecture of the SWORD⁴ plugin and demonstrate how SWORD enables rapid prototyping of a social networking application.

2 Related Work

In [5], Lima and Schwabe present the SHDM (Semantic Hypermedia Design Method) design approach, an extension of their earlier OOHD (Object Oriented Hypermedia Design Method) approach. SHDM is targeted for the design of Web applications for the Semantic Web, replacing the conceptual models of OOHD with Semantic Web ontologies. To our understanding, applications designed using the SHDM method keep a relational database in the backend and use ontologies to structure the metadata used for navigating the data. Our solution is different as we rely only on semantic store as a data backend.

In [9], Vdovjak *et al.* present Hera, a methodology which supports the design and engineering of Semantic Web Information Systems. They decompose an application into three layers, the semantic layer, the application layer and the presentation layer which correspond to the model–view–controller design pattern we use in SWORD. Hera focuses mainly on presentation and the interaction with users is mainly through navigation. In our solution, users can navigate the semantic web as well as adding information, editing information and deleting.

Corcho *et al.* introduce a Semantic Web portal using the MVC design pattern [4], but, similarly to Hera and SHDM, do not integrate it with an existing framework. We extend an existing popular framework (Ruby on Rails), thus connecting to an existing Web development community, leveraging the existing ecospace of plugins and extensions, and reducing the adoption barrier. Using Ruby on Rails and its ecospace leverages its standard functionality such as authentication, Web Services, AJAX support, Web 2.0, templating etc.

3 Semantic Web on Rails Development

We add the following functionality to the Ruby on Rails framework:

Prototyping: we provide a set of generators, inspired by the Ruby on Rails “scaffolding” [8], which generates models, views and controllers based on a database schema. In our case, we generate the model, views and controller based on a given RDF(S) ontology.

Interaction: the generated MVC handles showing, fetching, editing, searching, and versioning of Semantic Web resources.

⁴ <http://wiki.activerdf.org/SWORD/>

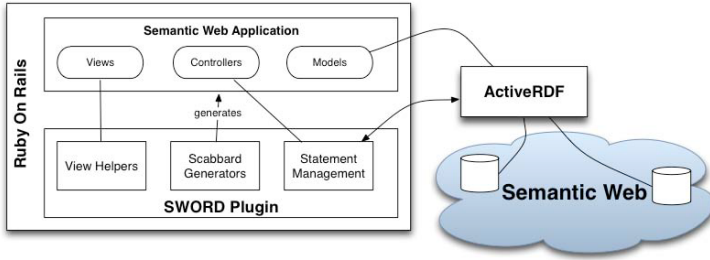


Fig. 1. SWORD architecture

Versioning and provenance of data: we track the versioning history and provenance of each statement, allowing human readers to include or exclude certain sources from their application views.

Semantic Web resource management: we provide libraries to manipulate Semantic Web resources and statements.

Figure 1 shows the relationships between SWORD and different components. SWORD integrates with Ruby on Rails as a plugin, and uses the ActiveRDF⁵ Ruby library [6] to maps RDF(S) resources to Ruby objects.

4 The FOAF Browser Example

We demonstrate the functionalities of SWORD by prototyping a social networking application. This application uses the FOAF⁶(friend of a friend) ontology which defines a vocabulary for representing people and their relationships. The application provides views to search, show, browse and edit personal profiles, while maintaining the history and provenance of information. The application integrates data from various, arbitrary, sources using on-demand data collection, and enables interoperability with other Web applications through the FOAF ontology and other RDF vocabularies.

We generate and run the web application as follows:

```
rails social_networking ; cd social_networking
./script/generate scabbard person foaf http://xmlns.com/foaf/0.1/
./script/server
```

Fig. 2 shows the generated files, namely a controller for `people`, some helpers, a `person` model, a shared `people` layout and several views for displaying the various actions (the underscored files are partial views used in AJAX actions).

In Figure 3, the generated application is displayed, showing the profile of Cédric. This personal profile is actually an RDF file on his web page which was automatically fetched by the application and integrated into the knowledge base.

⁵ <http://www.activerdf.org/>

⁶ <http://foaf-project.org/>



Fig. 2. Generated scaffolding files

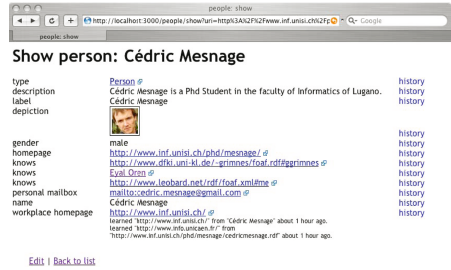


Fig. 3. Showing Cédric’s FOAF profile

When showing this person, all known statements about him are displayed, using human-readable labels, defined in an external ontology which is also integrated on-demand.

As shown at the bottom of the screenshot, the user can display the history and provenance (source) of any piece of information; in this particular example, the history of the “workplace homepage” is shown.

The generated application allows users to browse the social network by following any relationship between people, such as the “foaf:knows” relationship between acquaintances. When clicking on such a relationship, the information about that person is automatically fetched from the Semantic Web, and the display is updated to show the particular information of that person. In this example, users browse the social network; in general, the application allows users to browse the Semantic Web.

Fig. 4 displays the automatically generated overview page, showing a list of people. From this page users can search for all people (resources) related to a certain term, and for each person the standard CRUD (create, read, update, delete) actions are available, through the user interface actions: “show”, “edit”, and “delete”.

Editing as shown in Figure 5 is provided by a view which contains a form dynamically created according to the available information on the resource to be edited. Once submitted the update action of the controller is called. It adds

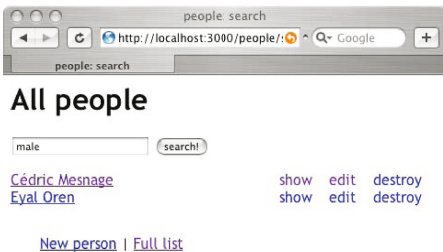


Fig. 4. Searching

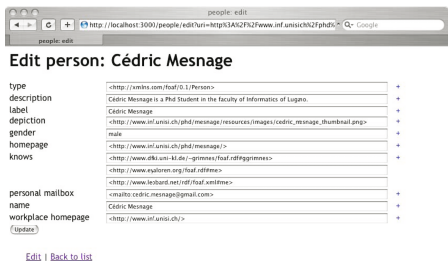


Fig. 5. Editing Cédric’s foaf profile

new statements if new information is entered (using the + button in the form) or edition statements if some existing statements have been edited.

All this behaviour is generated automatically, and can be customised by the application designer to adapt the generic behaviour to his needs, by changing actions or by adding new actions and new views.

5 Conclusion

In this paper, we presented SWORD, a plugin for Ruby On Rails which together with the ActiveRDF Ruby library transforms the popular Web application framework in a Semantic Web application framework. We demonstrated the use of SWORD to create an interoperable social networking application based on the FOAF ontology.

References

1. Berners-Lee, T.: Weaving the Web – The Past, Present and Future of the World Wide Web by its Inventor. Texere (2000)
2. Berners-Lee, T., Hall, W., Hendler, J.A.: A Framework for Web Science (Foundations and Trends(R) in Web Science). Now Publishers Inc., (2006)
3. Burbeck, S.: Applications Programming in Smalltalk-80: How to use Model–View–Controller (1987)
4. Corcho, O., López-Cima, A., Gómez-Pérez, A.: A platform for the development of semantic web portals. In: ICWE '06. Proceedings of the 6th international conference on Web engineering, pp. 145–152. ACM Press, New York (2006)
5. Lima, F., Schwabe, D.: Application modeling for the semantic web. In: Web Congress, 2003. Proceedings. First Latin American, pp. 93– 102 (2003)
6. Oren, E., Delbru, R., Gerke, S., Haller, A., Decker, S.: ActiveRDF: Object-oriented semantic web programming. In: Proceedings of the International World-Wide Web Conference (May 2007)
7. Reenskaug, T.: Models, views, controllers. Tech. rep., Xerox PARC (1979)
8. Thomas, D., Hansson, D., Breedt, L., Clark, M., Davidson, J.D., Gehland, J., Schwarz, A.: Agile Web Development with Rails. Pragmatic Bookshelf (2006)
9. Vdovjak, R., Frasincar, F., Houben, G., Barna, P.: Engineering semantic web information systems in hera. Journal of Web Engineering, pp. 3–26 (2003)