

# RExplorator - supporting reusable explorations of Semantic Web Linked Data.

Marcelo Cohen, Daniel Schwabe

Pontifical Catholic University of Rio de Janeiro  
R. M. S. Vicente 225  
Gávea, Rio de Janeiro, RJ, Brazil  
+55 21 3527-1500  
{mcohen21@gmail.com, dschwabe@inf.puc-rio.br}

**Abstract.** This demo presents RExplorator, an environment which allows non-technically savvy users, but who understand the problem domain, to explore the data until they understand its structure. They employ a combination of search, query and faceted navigation in a direct manipulation, query-by-example style interface. In this process, users can reuse previously found solutions by other users, which may accomplish sub-tasks of the problem at hand. It is also possible to create an end-user friendly interface to allow them to access the information. Once a solution has been found, it can be generalized, and optionally made available for reuse by other users. This enables the establishment of a social network of users that share solutions for problems in particular domains (repositories) of interest.

**Keywords.** RDF, exploratory search, exploration, ontology, semantic web, reuse, interface, set-based navigation

## 1 Introduction

The availability of Linked Open Data in the WWW has increased tremendously<sup>1</sup>. Currently, when building a new application, it is becoming increasingly common to first explore available data that can be leveraged to enhance and complete one's own data to provide the desired functionality. The BBC Music website<sup>2</sup> is one visible example of this approach, combining MusicBrainz and DBPedia with their own data.

In previous work we developed Explorator [1], a model for representing information processing by users in exploratory tasks, and its associated tool, which

---

<sup>1</sup> <http://linkeddata.org>

<sup>2</sup> <http://www.bbc.co.uk/music>

provides a browser interface supporting this model. Explorator is based on the metaphor of direct manipulation of information in the interface, with immediate feedback of user actions. In this demo we present RExplorator<sup>3</sup>, a significant extension of Explorator, with new features illustrated in this demo.

## 2 REXPLORATOR

Consider two simple tasks to be carried out over the “Dogfood” data server<sup>4</sup>, containing collected publication information for several conferences related to the Semantic Web - Finding all publications of a given author and finding co-workers of a given researcher, and their publications. We assume the user has no prior knowledge about the contents of this repository.

For the first task, the user has to

1. Find a class that represents persons
2. Find the desired person, “a”.
3. Find a property “p” that relates a person to publications,
4. Find all triples of the form <a p ?pub> and collect all objects from these triples.

In Explorator, this is achieved by first clicking on “Menu”-> All RDF Classes”, noticing class Person, mousing over it to click on “All Instances”, which reveals a set of all Persons. Double-clicking on a Person (e.g. “Steffen Staab”), a new box appears with all details for this resource (i.e., all triples with this resource as subject). Looking at the details, one notices the property “made”, which relates Person to Publications. To get all publications by a Person, one may click on the “Selected Person Details” box, and click on the “S” operand position at the top; click on the “made” box and click on the “P” operand position at the top, and then clicking on the “=” (“compute”) operator at the top.

For the second task, the user has to follow a similar set of steps, re-using task 1 once the co-workers have been found.

RExplorator extends Explorator by (we don’t detail all for space reasons)

1. Allowing operations to be parameterized;
2. Allowing the results of a query to be fed as input of another query, thus forming graphs of interconnected operations;
3. Allowing keeping such graphs as separate workbenches, while enabling interconnection of graphs across workbenches;

---

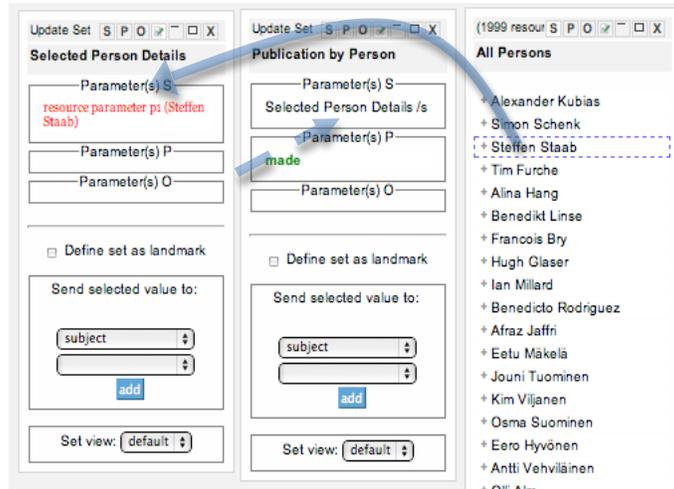
<sup>3</sup> Available at <http://www.tecweb.inf.puc-rio.br/replorator>, which includes a demo video. See also a demo of Explorator - <http://blip.tv/play/AennPpPBIw>

<sup>4</sup> <http://data.semanticweb.org>

4. Allowing the designer to import previously defined query graphs into the current workbench;
5. Allowing the designer to define additional operators beyond the set operations provided;
6. Allowing the designer to define interfaces oriented towards end users, hiding details and customizing the look-and-feel.

The original Explorator metaphor lets users compose operations incrementally, seeing the results at each composition step. Each new query takes its operands from existing query results. In the end, one may regard this set of inter-related operations as a graph, similar to an Excel spreadsheet. However, the operations are all grounded, which would be akin to not having any variables in the formulas of the analogous spreadsheet. Thus, the first generalization made was to allow operations to have its operands parameterized, and to propagate values through the graph of operations when the value of the parameter is changed. This is equivalent to introducing variables in the expression that denotes the operation.

Consider step 4 in task 1, finding all publications of a Person. In Explorator, this is achieved by selecting an instance of Person (e.g., “Steffen Staab in box “All Persons”) in Figure 1, setting it as the subject parameter, selecting the relation “make” as the property parameter, and clicking on the “=” operator to find all triples of the form <<url for Steffen Staab> made? o> clicking on the  icon in each box, as shown in Figure 1 reveals the actual operations and their dependencies .



**Figure 1** – Query structure and dependencies

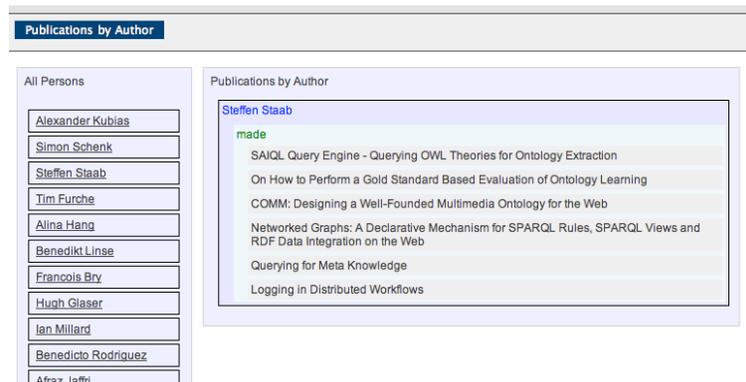
The first box, Selected Person Details, represents the query that finds out all triples with a given Person as subject. Notice that the first position, “S”, has been

parameterized, and the current parameter value is (the URI for) **Steffen Staab**. If we drag any person from the rightmost box (**All Persons**) onto the “S” position in the **Selected Person Details** box, the value is replaced and the query re-evaluated.

The **Publication by Person** query (middle box) is defined as taking its “subject” parameter from the “subject” position of the **Selected Person Details** query. Therefore, if a new value is plugged into the “S” position in the **Selected Person Details** query, it is automatically propagated to this query, which triggers its reevaluation.

RExplorator organizes the workspace into workbenches, each representing a task. A user may save workbenches for later reuse, and share it with other users as well.

The development interface of RExplorator is best suited to allow users to explore RDF repositories, and requires understanding the RDF model. RExplorator allows expert users to provide an end-user friendly interface – called the **Application Interface** - to solutions found while exploring datasets. This interface is generated by a combination of views, which are defined using the “views” menu option; generic, pre-defined views are initially available for reuse. Views make full use of CSS, which is also defined in a separate view that can be customized to change the look-and-feel of the generated interface. Figure 2 shows an example of an application interface.



**Figure 2** – User-friendly interface with All Persons and Selected Person Co-Workers.

**ACKNOWLEDGMENT.** Daniel Schwabe was partially supported by a grant from CNPq.

### 3 REFERENCES

- [1] Araújo F. C. S.; Schwabe D.; Explorator: A tool for exploring RDF data through direct manipulation, In: Proceedings of the Linked Data on the Web Workshop (LDOW2009), Madrid, Spain, April 20, 2009, CEUR Workshop Proceedings, ISSN 1613-0073, online [http://CEUR-WS.org/Vol-538/ldow2009\\_paper2.pdf](http://CEUR-WS.org/Vol-538/ldow2009_paper2.pdf)