

SPARCool: SPARQL made easy

Alexandre Passant

Digital Enterprise Research Institute,
National University of Ireland, Galway
alexandre.passant@deri.org

Abstract. This paper introduces the motivations and technical details behind SPARCool, a web service that provides an easy way to run SPARQL queries on the Web by calling a simple URL pattern¹.

Key words: SPARQL, Linked Data, SPARCool, JSONP

1 Motivations

Thanks to the Linking Open Data² community project, million of resources, identified by their URI, are now described on the Web thanks to RDF(S)/OWL technologies. To efficiently take advantage of it and go further than simple browsing, developers must use either SPARQL or RDF(S)/OWL APIs to query and use this data. Yet, it can be complex for Web developers that are not necessarily aware of such languages. SPARCool³ aims to solve this gap by providing a way to let people easily run SPARQL queries on any URI that follows the Linked Data principles⁴ [1] without learning the SPARQL syntax, and then build applications using the query answers.

2 Using SPARCool

In order to run SPARQL queries with SPARCool, one just have to call the URL pattern `http://sparcool.net/format/predicate[;l=lang]/URI`, that is translated into a SPARQL query applied to the URI given as a parameter. The `format` parameter indicates the required output and can be (1) XML SPARQL Query results, (2) JSON SPARQL Query results, (3) HTML view of the results, either plain-HTML (for a single result) or using a `` list (for multiple results) and (4) a redirect to the first query of the answer, for instance for hyperlinks or images. The `predicate` must be written using a QName syntax and a set of prefixes are supported by SPARCool, while the `lang` parameter is optional.

¹ The work presented in this paper has been funded in part by Science Foundation Ireland under Grant No. SFI/08/CE/I1380 (Lion-2).

² <http://linkeddata.org>

³ <http://sparcool.net>

⁴ Especially the three first ones, are the fourth one is not a requirement for SPARCool.

While the two first types of results (XML and JSON serialization of SPARQL query results) imply additional work if developers want to integrate it into webpages, the HTML view and the redirect can be used to directly expose results in HTML pages. Especially, SPARCool also supports JSONP⁵ callbacks, so that the HTML results of SPARCool calls can be embedded directly in (X)HTML pages on client side, without having to face AJAX cross-domain security issues. That way, it enables to construct dynamic and constantly up-to-date webpages: if the information about a particular URI changes (e.g. a `foaf:homepage` value), the webpage will be automatically adapted.

Examples of SPARCool queries, using various result formats, can be found on the service website: <http://sparcool.net>.

3 Query translation

<http://sparcool.net/j/foaf:name;l=fr/http://example.org/Bob#b> will be translated into the following query⁶, and results will be serialized in JSON.

```
FROM <http://example.org/Bob#b>
SELECT ?value
WHERE {
  <http://example.org/Bob#b> rdf:label ?value .
  FILTER (lang(?value) = 'en')
}
```

Since the FROM parameter is applied to the same URI used in the query pattern, this URI must follow the Linked Data principles, especially being dereferencable and returning (machine-readable) information about itself, whatever the RDF serialization it (RDFa, RDF/XML, etc.).

4 Implementation

SPARQL is provided thanks to a PHP script (less than 100 lines of code) that translates the called URI (thanks to Apache `RewriterRule`) to a `SPARCool` object that contains the generated SPARQL query and additional parameters (output format and JSONP callback if any). The query is then launched using `roquet` (part of Redland⁷) and the results are intercepted by the `SPARCool` object that sets the proper content-type and returns the results in the browser.

References

1. Tim Berners-Lee. Linked Data. Design issues for the world wide web, World Wide Web Consortium, 2006. <http://www.w3.org/DesignIssues/LinkedData.html>.

⁵ <http://bob.pythonmac.org/archives/2005/12/05/remote-json-jsonp/>

⁶ Prefixes omitted

⁷ <http://librdf.org>