Tutorial: An Introduction to SPARQL and Queries over Linked Data

Olaf Hartig

Humboldt-Universität zu Berlin
hartig@informatik.hu-berlin.de

Abstract. Nowadays, more and more datasets are published on the Web adhering to the Linked Data principles. Our tutorial provides a beginners' introduction on how to query this data using the query language SPARQL.

1 Motivation

Since the Linked Data principles have been proposed in 2006 [1], a grass-roots movement started to publish and interlink multiple open databases on the Web based on these principles [2]. Today an increasing number of data publishers such as the BBC, Thomson Reuters, The New York Times, the Library of Congress, and the UK government adopt this practice. This ongoing effort resulted in bootstrapping the Web of Linked Data which, today, comprises billions of statements including millions of links between datasets. The published datasets include data about books, movies, music, radio and television programs, reviews, scientific publications, genes, proteins, medicine, clinical trials, geographic locations, people, companies, statistical and census data, etc.

The availability of this data, including the existence of data-level connections between datasets, presents exciting opportunities for the next generation of Web-based applications. As a consequence, consuming Linked Data is a highly relevant topic in the context of Web engineering.

2 Topics

Our introductory tutorial aims to provide participants with an understanding of one of the basic aspects of Linked Data consumption, that is, querying Linked Data. The tutorial consists of three main parts.

Part 1: The RDF Data Model and Linked Data In the first part, we briefly introduce the concept of Linked Data and its underlying data model, RDF [3].

The idea of Linked Data is based on four principles [1]. These principles require to identify an entity as well as provide access to a structured data representation of it, via a single HTTP scheme based URI. Hence, resolving such a URI via the HTTP protocol yields data about the entity identified by the URI. This data should be represented using the Resource Description Framework (RDF). RDF is a generic data model that represents data using triples of the form (subject, predicate, object). Each element of such an RDF triple can be a URI or a local identifier for unnamed entities; objects can

also be a literal. A set of RDF triples is called an RDF graph. Furthermore, the Linked Data principles require that the provided RDF data includes *data links* pointing to data from other data sources on the Web. A data link is an RDF triple where the subject is a URI in the namespace of one data source and the object is a URI in the namespace of another source. By connecting data from different sources via such links a single, globally distributed dataspace emerges.

Part 2: The SPARQL Query Language The second and largest part provides a comprehensive introduction to SPARQL [4], the de facto query language for RDF.

SPARQL is based on RDF graph patterns and subgraph matching: The basic building block for SPARQL queries is called *basic graph pattern* (BGP). A BGP is a set of triple patterns which are RDF triples that may contain query variables at the subject, predicate, and object position. More complex query patterns are unions of pattern, optional patterns, filter expressions, etc. Query results in SPARQL are defined based on graph pattern matching: Each element of the result is a set of variable bindings that, basically, represents a matching subgraph in the queried RDF graph.

Part 3: Querying Multiple Linked Datasets In the third part of the tutorial, we discuss several approaches for executing SPARQL queries over multiple, interlinked datasets. These approaches can be classified in three categories: data warehousing, query federation, and Linked Data query processing [5].

Data warehousing is an approach where data is collected and copied into a central database. Queries are executed over this central database.

The query federation approach is based on distributing the processing of queries to query services provided by Linked Data publishers. A mediator analyzes and decomposes the user query into several sub-queries. These sub-queries are distributed to the query services which, then, execute these sub-queries and return the results.

Linked Data query processing approaches evaluate queries over the Web of Linked Data by relying only on the Linked Data principles. The prevalent example of a Linked Data query processing approach is link traversal based query execution. The idea of this approach is to intertwine the traversal of data links with the construction of the query result and, thus, to integrate the discovery of data into the query execution process [6].

References

1. Berners-Lee, T.: Linked Data. http://w3.org/DesignIssues/LinkedData (2006)
2. Bizer, C., Heath, T., Berners-Lee, T.: Linked Data – The Story So Far. Journal on Semantic Web and Information Systems **5**(3) (2009)
3. Klyne, G., Carroll, J.J.: Resource Description Framework (RDF): Concepts and Abstract Syntax. W3C Recommendation (February 2004)
4. Prud'hommeaux, E., Seaborne, A.: SPARQL Query Language for RDF. W3C Recommendation (January 2008)
5. Hartig, O., Langegger, A.: A Database Perspective on Consuming Linked Data on the Web. Datenbank-Spektrum **10**(2) (2010)
6. Hartig, O., Freytag, J.C.: Foundations of Traversal Based Query Execution over Linked Data. In: Proceedings of the 23rd ACM Conference on Hypertext and Social Media. (2012)