

RDF Visualizer: A Tool for Displaying, Browsing and Exploring High Density RDF Data

Kostas Petrakis¹, Nikos Minadakis¹, Korina Doerr¹, Maria Theodoridou¹ and Martin Doerr¹

¹ Institute of Computer Science, FORTH-ICS, Heraklion, Greece

Abstract. RDF Visualizer is a generic browsing mechanism that gives the user a flexible, highly configurable, detailed overview of a dataset/database encoded in RDF. It is designed and developed to overcome the drawbacks of the existing RDF data visualization methods/tools with an innovative point of view. It allows users with neither previous knowledge of the data structure nor SPARQL skills to navigate and explore the data starting from an element of interest and incrementally explore further the source. The tool is currently used in several national and EU funded research projects such as HELLAS-CH and PARTHENOS and has been tested with large datasets from the British Museum and the American Art Collaborative project.

Keywords: Linked Open Data, Visualization tools, Big Data, Hierarchical and Interactive Visualizations, Incremental Exploration, Semantic Web, SPARQL, RDF Visualization.

1 Introduction

Big Data era has brought on surface the availability of a great amount of massive datasets that are dynamic, noisy and heterogeneous in nature. RDF is widely used for data integration, transformation and aggregation of diverse sources and applications of mappings between different source schemata. Having a visual overview at the time of transformation or exploration of the data is more than a crucial, highly required modern task [1]. Although a great deal of emphasis and effort has been placed on the validation of the produced RDF structure and format, the efficient visualization of the constructed database contents that enables semantic validation by domain experts has largely been neglected. Until now, visualization is achieved either by manual inspection of multiple files, by formulation and execution of complex SPARQL queries, or by custom user interfaces that work only with a particular RDF schema without supporting user intervention or configuration, and presenting only scintillas of the actual content. Data visualization has become a major research challenge involving several issues related to data storage, querying, indexing, visual presentation, interaction, personalization [2].

* Copyright © 2019 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

2 RDF Visualizer

An efficient and user-friendly RDF data visualization tool should support the following basic principles: (1) display data of any schema and RDF format; (2) display all nodes of any class/instance; (3) apply configuration rules to improve the layout or presentation for known classes and properties (e.g., hide URIs that are meaningless for the user); (4) display information of high density in one screen (which is not possible in solutions based on “object templates”).

Modern systems have to “squeeze a billion records into a million pixels” [2]. Offering a dataset overview may be extremely difficult as information overloading is a common issue [3]. RDF visualization solutions already exist and have been applied to successful projects [4], however the existing approaches fail to fulfil the complete set of principles enumerated above. Specifically, approaches that display all the nodes of any class/schema and support any schema and format (principles 1 and 2) fail to display information of high density in one screen and those that succeed in regard the latter fail in relation to the rest of the principles.

In order to meet the requirements laid out by the complete set of principles, RDF Visualizer (RDFV) has been designed and implemented as a generic browsing mechanism that gives the user a flexible, highly configurable, detailed overview of an RDF dataset/database.

RDFV presents RDF data as an indented list to handle the density and depth of information (principle 4) starting from a specified RDF resource (URI) and generating a dynamically expandable tree structure. In order to achieve this, all incoming and outgoing links are transformed and presented as nodes of every class/instance (principle 2) in a schema agnostic way (principle 1). Users are able to configure the display of schema-dependent information according to their preferences by editing a configuration file (principle 3). They are also able to define priority based rule chains that are used to define schema-dependent style and order of properties that are inherited to subclasses and sub properties by editing an xml file (principle 3). For anything not covered by a rule, default options are applied based on experience and best practices.



Fig. 1. RDF Visualizer user interface.

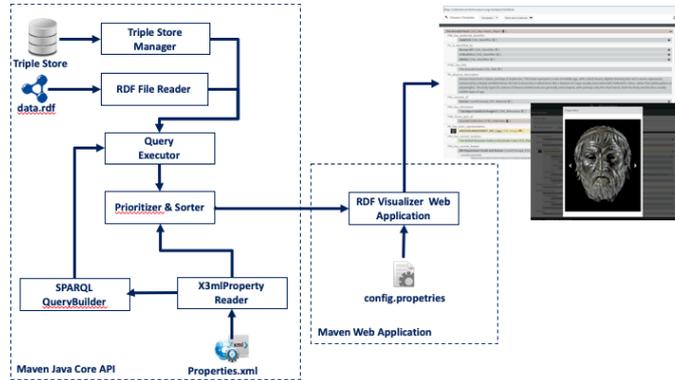


Fig. 2. The architecture of RDF Visualizer.

User interface of the tool (Figure 1) has been designed in cooperation with potential users, focusing on usability and readability. Moreover, rich functions are provided to control the display of data items, such as, identifying same instances, expanding collapsing big texts or big sets of results, displaying images and image galleries, removing or replacing prefixes, selecting non displayed URIs and displaying the subject's URI on label hover. With right click user can focus on a specific URI, open it as root in new tab, copy it on clipboard or retrieve the path of its sub-graph.

The architecture of RDFV is presented in Figure 2. Here the arrows depict the data flow between the two basic system components. The first is the core API which consists of micro services that are divided in three levels. RDFV is able to parse RDF data either from triple stores–graph databases (e.g. virtuoso, blazegraph) using **Triple Store Manager** or from raw .ttl files using **RDF File Manager**. The tool considers user's choices to prioritize subject's properties from an xml file with **X3MLProperty Reader**. It constructs SPARQL queries using **SPARQL Query Builder** and executes them with **Query Executor**. The output of these functions is fed as input to **Prioritizer and Sorter** producing a Map <Triple, List<Triple>> considering user's choices from two configuration files (properites.xml and config.properties).

The second component is the **Web Application** that transform this Map into JSON format and projects data in a readable and interactive tree structure on a web browser. Each node is represented by its label and type. Initially, the tool creates a tree that has as subject (root) the given URI and as leaves its direct properties and objects. If an object is by its own a subject (having properties and objects), there is an expand icon on the right. On click, the above same process runs recursively transforming the object into subject creating dynamically its own sub tree of properties and objects. So with this incremental exploration the tool is remarkably lightweight and lets users guide their exploration.

RDFV has been integrated in the 3M interface of the X3ML suite of tools for data mapping and transformation [5]. It enhances 3M with an important validation tool for transformed data. Domain experts can easily check and correct their mapping and transformations on the fly resulting, enabling an iterative and collaborative evaluation of the resultant RDF.

3 Demo

The first part of the demo is designed to show a full set of functionalities provided by RDFV in a single well-formed example². The main goal of this part is to demonstrate how the tool can visualize data starting from a given URI as input and let the user guide the exploration. In our case this root URI will be Hei Tiki³ from a British Museum dataset⁴. During data incremental exploration all functionalities mentioned above will be revealed and explained in detail. Also it will be explained how users can apply their own preferences to the tool and get different overviews of same data depending on their choices. This can be done by editing two configuration files⁵ and define the priority of the properties of the given subject and what the tool takes as input, which prefixes do not show or replace etc.

In the second part a more generic use case will be presented, showing different datasets from diverse sources to demonstrate the ability of the tool to visualize heterogeneous data in a generic way. In this example⁶ the available datasets will be from The British Museum, American Art Collaborative⁷ and PARTHENOS⁸ Project exploring the Culture Heritage field and from HELLAS CH (MIS 5002735) exploring scientific workflows of sequencing and analyzing archaeological DNA data.

4 Acknowledgments

This work has been supported by the projects PARTHENOS (H2020 GA654119) and HELLAS CH⁹ (MIS 5002735). We would like to thank the British Museum and the American Art Collaborative for making their data available publically.

References

1. Po, L., Malvezzi, D.: High-level Visualization Over Big Linked Data. In: Proceedings of the ISWC 2018 Posters & Demonstrations (2018)
2. Shneiderman, B.: Extreme Visualization: Squeezing a Billion Records into a Million Pixels," in ACM Conference on Management of Data (SIGMOD), 2008.
3. Bikakis, N.: Big data visualization tools. Encyclopedia of Big Data Technologies (2018)
4. Graziosi A., et al.: Customised Visualisations of Linked Open Data, VOILA (2017)
5. Marketakis, Y., et al.: X3ML mapping framework for information integration in cultural heritage and beyond, International Journal on Digital Libraries, pp 1-19, (2016)

² Full set of functionalities demo available at <https://demos.isl.ics.forth.gr/RDFVisualizer-demo/?resource=http://collection.britishmuseum.org/id/object/EOC10881>

³ "Hei-Tiki" URI <http://collection.britishmuseum.org/id/object/EOC10881>

⁴ BMCcollection-2013-09-25.tar.gz at <https://collection.britishmuseum.org/resource/sparql>

⁵ Available at <https://demos.isl.ics.forth.gr/RDFV-Demo/properties.xml> & [config.properties](https://demos.isl.ics.forth.gr/RDFV-Demo/config.properties)

⁶ Available at <https://demos.isl.ics.forth.gr/RDFV-Demo/>

⁷ Available at <https://github.com/american-art>

⁸ <http://www.parthenos-project.eu/>

⁹ Implemented under "Action for Strengthening Research and Innovation Infrastructures," funded by the Operational Programme "Competitiveness, Entrepreneurship and Innovation" (NSRF 2014-2020) and co-financed by Greece and the European Union (European Regional Development Fund)