The Linked Data Visualization Model

Josep Maria Brunetti[1], Sören Auer[2], Roberto García[1]

[1] GRIHO, Universitat de Lleida, Jaume II, 69. 25001 Lleida, Spain
{josepmbrunetti,rgarcia}@diei.udl.cat, http://griho.udl.cat/
[2] AKSW, Computer Science, University of Leipzig, Germany
auer@informatik.uni-leipzig.de, http://aksw.org/

Abstract. The potential of the semantic data available in the Web is enormous but in most cases it is very difficult for users to explore and use this data. Applying information visualization techniques to the Semantic Web helps users to easily explore large amounts of data and interact with them. We devise a formal Linked Data Visualization model (LDVM), which allows to *dynamically* connect data with visualizations.

1 Introduction

In the last years, the amount of semantic data available on the Web has increased dramatically, especially thanks to initiatives like Linked Open Data (LOD). The potential of this vast amount of data is enormous but in most cases it is very difficult *and* cumbersome for users to visualize, explore and use this data, especially for lay-users without experience with Semantic Web technologies.

Applying information visualization techniques to the Semantic Web helps users to explore large amounts of data and interact with them. Visualizations are useful for obtaining an overview of the datasets, their main types, properties and the relationships between them. Compared to prior information visualization strategies, we have a unique opportunity on the Data Web. The unified RDF data model being prevalent on the Data Web enables us to bind data to visualizations in an *unforeseen* and *dynamic* way. An information visualization technique requires certain data structures to be present. When we can derive and generate these data structures automatically from reused vocabularies or semantic representations, we are able to realize a largely automatic visualization workflow. This will enable users to explore datasets even if the publisher of the data does not provide any exploration or visualization means.

The *Linked Data Visualization Model* (LDVM) we propose allows to connect different datasets with different visualizations in a dynamic way. In order to achieve such flexibility and a high degree of automation the LDVM is based on a visualization workflow incorporating analytical extraction and visual abstraction steps. Each of the visualization workflow steps comprises a number of transformation operators, which can be defined in a declarative way. As a result, the LDVM balances between flexibility of visualization options and efficiency of implementation or configuration.

2 Linked Data Visualization Model

We use the *Data State Reference Model* (DSRM) proposed by Chi [1] as conceptual framework for our *Linked Data Visualization Model* (LDVM). While the DSRM describes the visualization process in a generic way, we instantiate and adopt this model with LDVM for the visualization of RDF and Linked Data. The names of the stages, transformations and operators have been adapted to the context of Linked Data and RDF. Figure 1 shows an overview of LDVM. It can be seen as a pipeline, which originates in one end with raw data and results in the other end with the visualization.

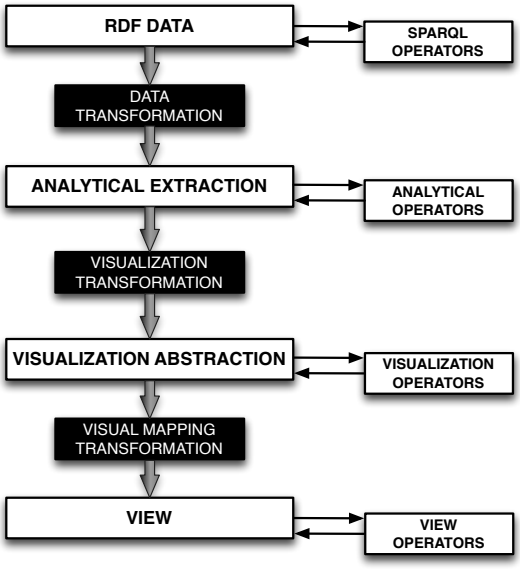

Fig. 1. High level overview of the Linked Data Visualization Model.

The LDVM pipeline is organized in four stages that data needs to pass through:

1. *RDF Data:* the raw data, which can be all kinds of information adhering to the RDF data model, e.g. instance data, taxonomies, vocabularies, ontologies, etc.
2. *Analytical extraction:* data extractions obtained from raw data, e.g. calculating aggregated values.
3. *Visual abstraction:* information that is visualizable on the screen using a visualization technique.
4. *View:* the result of the process presented to the user, e.g. plot, treemap, map, timeline, etc.

Data is propagated through the pipeline from one stage to another by applying three types of transformation operators:

1. *Data transformation:* transforms raw data values into analytical extractions declaratively (using SPARQL query templates).
2. *Visualization transformation:* takes analytical extractions and transforms them into a visualization abstraction. The goal of this transformation is to condense the data into a displayable size and create a suitable data structure for particular visualizations.
3. *Visual mapping transformation:* processes the visualization abstractions in order to obtain a visual representation.

As illustrated in Figure 2, our model allows to connect different RDF datasets and different data extractions with different visualization techniques. Not all datasets are compatible with all data extractions and each data extraction is only compatible with some visual configurations.

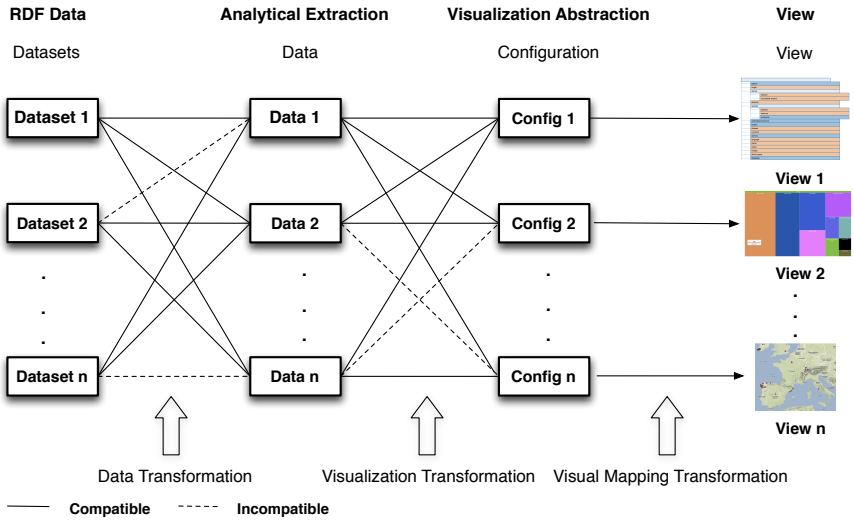

Fig. 2. Linked Data Visualization Model ecosystem, which allows to dynamically connect datasets with visualizations.

Each dataset offers different data structures to be extracted, e.g. class hierarchy, property hierarchy, geospatial data, etc. Each data extraction can be visualized with different configurations, which contain information such as the visualization technique to use, colors, etc. Then, a concrete visualization is generated depending on the data extraction and the visual configuration.

In summary, the model is divided in two main areas: data space and visual space. The RDF data stage, analytical extraction stage and data transformation

belong to the data space, while visual abstraction stage, view stage and visual mapping transformation belong to the visual space. These two main blocks are connected by a visualization transformation.

3 Demonstration

We have implemented a prototype called *LODVisualization*[1] that supports the Linked Data Visualization Model proposed. It allows to explore and interact with the Data Web through different visualizations. This way, our prototype serves not only as a proof-of-concept of our LDVM but also provides useful visualizations of RDF. These visualizations allow users to obtain an overview of RDF datasets and realize what the data is about: their main types, properties, etc.

LODVisualization is compatible with most of SPARQL endpoints as long as they support JSON and SPARQL 1.1. We have evaluated our implementation of the Linked Data Visualization Model with different datasets, data extractions and visualizations. The goal of our evaluation was to prove that the LDVM can be applied to different datasets providing different data visualizations in real time. All the visualization examples are available on the website and it easy to create new ones.

4 Related Work

Some of the existing tools available to explore and visualize Linked Data have been analyzed in [2]. However, only very few provide visualizations and they are focused on concrete data types or domains.

5 Conclusions

The Linked Data Visualization Model (LDVM) can be applied to rapidly create visualizations of RDF data. It allows to connect different datasets, different data extractions and different visualizations in a dynamic way. Applying this model, developers and designers can obtain a better understanding of the visualization process with data stages, transformations and operators. The LDVM offers user guidance on how to create visualizations for RDF data.

References

1. Ed H. Chi. A taxonomy of visualization techniques using the data state reference model. In *Proceedings of the IEEE Symposium on Information Vizualization 2000*, INFOVIS '00, pages 69–, Washington, DC, USA, 2000. IEEE Computer Society.
2. Aba-Sah Dadzie and Matthew Rowe. Approaches to visualising linked data: A survey. *Semantic Web*, 2(2):89–124, 2011.

[1] `http://lodvisualization.appspot.com/`