

# Balloon Synopsis: A Modern Node-Centric RDF Viewer and Browser for the Web

Kai Schlegel<sup>(✉)</sup>, Thomas Weißgerber, Florian Stegmaier, Christin Seifert, Michael Granitzer, and Harald Kosch

University of Passau, Innstrasse 41, 94032 Passau, Germany

{kai.schlegel,thomas.weißgerber,florian.stegmaier,christin.seifert,  
michael.granitzer,harald.kosch}@uni-passau.de

**Abstract.** Nowadays, the RDF data model is a crucial part of the Semantic Web. Especially web developers favour RDF serialization formats like RDFa and JSON-LD. However, the visualization of large portions of RDF data in an appealing way is still a cumbersome task. RDF visualizers in general are not targeting the Web as usage scenario or simply display the complex RDF graph directly rather than applying a human friendly facade. *Balloon Synopsis* tries to overcome these issues by providing an easy-to-use RDF visualizer based on HTML and JavaScript. For an ease integration, it is implemented as jQuery-plugin offering a node-centric RDF viewer and browser with automatic Linked Data enhancement in a modern tile design.

## 1 RDF Data Is Online - Most RDF Visualizations Are Not

Linked Data and the Semantic Web increase in volume leading to huge amounts of RDF data. Today, even non-experts manage open data, extract semantic information from text and integrate these sources easily in their information system [1]. However, displaying graph-based RDF data in a human-friendly way is challenging.

This paper demonstrates a modern tile-based visualization of RDF data. While there are many tools available to present RDF data to an user [2–4], most of them are not available in web-browsers and use tree- or graph-based layouts. RDF visualizations should support the user to conceive information from the data quickly. Furthermore, aggregation of different knowledge sources should be possible to enable a simplified view. However, the common node-link layouts can be very space consuming and may show node or edge overlappings. As a result, even simple RDF graphs can look confusingly complex to humans. Additionally, a RDF visualization can exploit the fact, that RDF data is interlinked with other online resources. Browsing a RDF graph can be automatically extended by

---

The presented work was developed within the CODE and MICO project partially funded by the EU Seventh Framework Programme, grant agreement number 296150 and 610480.

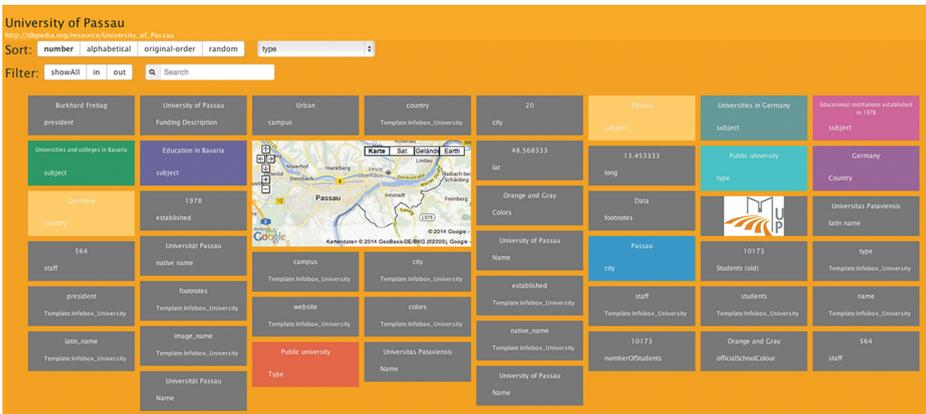
considering Linked Data resources. *Balloon Synopsis* was developed to allow Web developers an easy-to-use and embeddable RDF visualization for their website, enabling simple access and re-use of Linked Data. The key-highlights of this approach are as follows:

- Human friendly presentation of RDF utilizing a *node-centric view* and *ontology templating*.
- *Automatic enrichment* using SPARQL endpoints or automatic query federation over Linked Data.
- *HTML and JavaScript based*, to ensure a simple integration and extension by web developers.

The remaining paper discusses these in more detail.

## 2 Node-Centric and Human Friendly RDF Visualization

The user interface of *balloon Synopsis* is inspired by modern operation systems and design trends. A main principle is to focus on the content and direct relationship between semantic entities rather than showing a global and complex context. Semantic entities (graph nodes) are considered as autonomous tiles, containing information about the entity itself. Clicking on a tile gives the user the possibility to view the represented resource in detail, leading to a new node-centric overlay. Figure 1 shows a screenshot of a detail view. All related information or entities are again arranged as clickable tiles, allowing an iteratively browsing through the data.



**Fig. 1.** Viewing an example entity with enriched information and ontology templating

At any time, the user has the possibility to sort, filter and search (using keywords and regular expressions) the displayed tiles to gather information, according to the need of an users. Typically a tile represents low-end information like

resource, blank node or literal. This could result in overloaded visualizations, because a specific entity can be included in many different triples. To overcome this issue, *balloon Synopsis* implements customizable filters. These filters, for example, combine equal predicates and nodes in a common tile or remove irrelevant nodes (blacklist) to simplify the view. Furthermore, developers can easily configure ontology templates to transform data to a human friendly representation. As an example, Fig. 1 shows an instance of ontology templating by viewing a map instead of longitude and latitude coordinates.

Based on its configuration, *balloon Synopsis* can either work with imported data alone or can make use of an automatic enrichment of the data at runtime. In the second case configured knowledge bases are queried in the background to enhance current tile views successively. All filters are then applied on the resulting data consequently. While browsing the data, a history is saved to allow jumping back to already shown view layers and to track an users behavior.

### 3 Prototypic Open Source Implementation

*Balloon Synopsis* features a HTML and JavaScript implementation and is available as jQuery-plugin. To give more detailed information, Fig. 2 highlights essential internal components. At its core, balloon Synopsis uses a **local SPARQL capable RDF store**<sup>1</sup>. To import data into the local store, a set of different possibilities are offered. Besides common RDF serializations like Turtle, N3 or

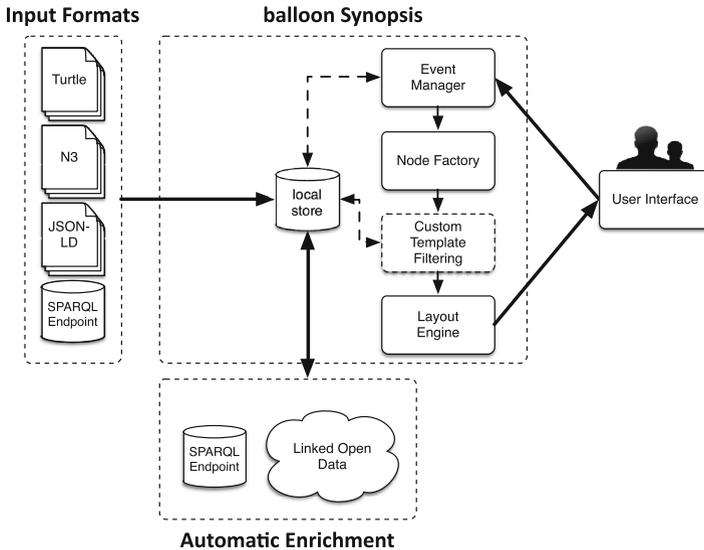


Fig. 2. Conceptual overview of internal processing steps

<sup>1</sup> <https://github.com/antoniogarrote/rdfstore-js>

JSON-LD, *balloon Synopsis* offers to specify a SPARQL endpoint and SPARQL query to load remote RDF data portions. In the whole project, cross domain problems are circumvented by using YQL<sup>2</sup>, to facilitate remote data querying. The initial view can be configured by an arbitrary initialization query to the local store or an automatic arrangement of all entities.

By clicking on a tile, the **user interface** invokes the **event manager** to show a specific detail view. The event manager is supposed to query the local store for information about the desired entity. As described above, *balloon Synopsis* offers an **automatic enrichment** of the local RDF data by querying (i) remote SPARQL endpoints or (ii) do an query federation over Linked Data endpoints utilizing the recently introduced *balloon Fusion* service [5]. The selection of proper remote endpoints is significant for a fast browsing performance. The event manager transmits all results to the **node factory**, which transforms the RDF data to corresponding JavaScript components to represent the content of a basic tile. These components are then forwarded to a **(custom) template filtering**. Besides aforementioned pre-packed filters to simplify the view, a developer can easily integrate custom ontology templates by means of a JavaScript function and a HTML snippet using handlebars syntax<sup>3</sup>. The custom filter also has access to the local store to load additional informations or alter existing tiles. The final layout is then computed by the **layout engine**, which can influence, for example, the ranking, scale and color of tiles based on importance or similarity. In addition, the user interface itself can affect the layout due to searching, sorting or responsive design events (e.g. resizing or panning). Integrated layouting mechanisms, building upon isotope<sup>4</sup>, enable flexible reordering of the view and animated transitions.

Balloon Synopsis is developed on a modular basis and available as free open source software to support contributions and extensions. Further developments will focus on the collection of ontology templates to continuously enhance the visualizations. A first demonstration as well as the prototypic sources are available at <http://schlegel.github.io/balloon/balloon-synopsis.html>.

## References

1. Gangemi, A.: A comparison of knowledge extraction tools for the semantic Web. In: Cimiano, P., Corcho, O., Presutti, V., Hollink, L., Rudolph, S. (eds.) ESWC 2013. LNCS, vol. 7882, pp. 351–366. Springer, Heidelberg (2013)
2. Camarda, D.V., Mazzini, S., Antonuccio, A.: Lodlive, exploring the web of data. In: Proceedings of the 8th International Conference on Semantic Systems, I-SEMANTICS '12, pp. 197–200. ACM, New York (2012)
3. Sayers, C.: Node-centric rdf graph visualization. Mobile and Media Systems Laboratory, HP Labs (2004)

<sup>2</sup> <http://developer.yahoo.com/yql/>

<sup>3</sup> <http://handlebarsjs.com/>

<sup>4</sup> <http://isotope.metafizzy.co/>

4. Pietriga, E.: Isaviz: a visual environment for browsing and authoring RDF models. In: Eleventh International World Wide Web Conference Developers Day (2002)
5. Schlegel, K., Stegmaier, F., Bayerl, S., Granitzer, M., Kosch, H.: Balloon fusion: SPARQL rewriting based on unified co-reference information. In: 5th International Workshop on Data Engineering Meets the Semantic Web, Co-Located with the 30th IEEE International Conference on Data Engineering (2014)