DBpedia Viewer - An Integrative Interface for DBpedia Leveraging the DBpedia Service Eco System

Denis Lukovnikov
University of Leipzig
lukovnikov@informatik.uni-leipzig.de

Claus Stadler
University of Leipzig
cstadler@informatik.uni-leipzig.de

Dimitris Kontokostas
University of Leipzig
kontokostas@informatik.uni-leipzig.de

Sebastian Hellmann
University of Leipzig
hellmann@informatik.uni-leipzig.de

Jens Lehmann
University of Leipzig
lehmann@informatik.uni-leipzig.de

ABSTRACT

With the growing interest in publishing data according to the Linked Data principles, it becomes more important to provide intuitive tools for users to view and interact with resources. The characteristics of Linked Data pose several challenges for user-friendly presentation of the information. In this work, we present the DBpedia Viewer as one method to address this problem. The DBpedia Viewer is the new DBpedia Linked Data user interface, which makes DBpedia data more accessible to non-experts while integrating the DBpedia service eco system as well as external Linked Data services.

1. INTRODUCTION

The Linked Data principles provide guidelines for publishing structured data in order to make it easily accessible to both machines and humans. In contrast to the common natural language representation of information on the Web, dereferencing Linked Data resources in many cases does not provide an intuitive view for humans. The fine granularity of triple-based data representation standardized by the RDF format requires mental effort to integrate information about the described entity, even for experts.

This work describes DBpedia Viewer, the new DBpedia [7] interface, which aims to present information from DBpedia in an engaging way while adhering to the Linked Data principles. DBpedia Viewer integrates existing DBpedia services as well as external Linked Data visualization tools to improve user-friendliness. The DBpedia interface was originally envisioned to serve data from DBpedia datasets, but could lead to a customizable framework that can easily be configured for other datasets. With the new DBpedia interface, we aim to show that even a generic representation of RDF data can be user-friendly and offer relevant services. A

distinguishing feature of DBpedia Viewer is the Triple Action Framework, a framework which allows to dynamically associate action types with triples.

The remainder of the article is structured as follows: Section 2 provides an overview of DBpedia and the DBpedia service eco system. An overview of DBpedia Viewer is given in Section 3 and Section 4 gives a more detailed view of the DBpedia Viewer components. Section 5 elaborates on related work and we conclude with Section 6.

2. DBPEDIA

DBpedia is a pioneer project in Linked Data publishing. It was one of the first Linked Open Data datasets available in 2007 and is a hub in the Linked Open Data cloud[1]. The data in DBpedia originates from Wikipedia and is extracted using the DBpedia extraction framework. The latest DBpedia release provides data for 4.0 million *entities* out of which 3.2 millions are classified according to the DBpedia ontology.[2]

DBpedia provides different kinds of information about entities. Entities typically have types, labels, links, Linked Data links and textual descriptions associated with them. DBpedia contains links to equivalent entities in other datasets (such as YAGO) and links to the same entities in DBpedia datasets for other languages (such as nl.dbpedia.org). In addition to the general information about entities, e.g. types and categories, DBpedia contains properties and classes specific for particular domains. For example, the entity (of type Person) `dbpedia:Barack_Obama` has a property `dbo:spouse` which refers to the entity `dbpedia:Michelle_Obama`.

Since the start of the DBpedia project, several tools and services were developed around DBpedia. DBpedia Spotlight [9] performs Entity Linking (or Named Entity Resolution and Disambiguation, NERD) in text by linking entity mentions to DBpedia entities. DBpedia Lookup[3] is an

---

[1] Linking Open Data cloud diagram, by Richard Cyganiak and Anja Jentzsch. `http://lod-cloud.net/`
[2] `http://blog.dbpedia.org/2013/09/17/dbpedia-39-released-including-wider-infobox-coverage\-additional-type-statements-and-new-yago-and-wikidata\-links/`
[3] `http://wiki.dbpedia.org/lookup/`

additional service which allows to search for DBpedia entities using either strings or prefixes (for auto-completion). The *DBpedia mappings wiki*[4] is an effort to crowdsource the mappings between the Wikipedia infoboxes and the DBpedia ontology. Apart from DBpedia specific tools, external vizualization tools have been developed that use DBpedia, such as RelFinder [4] and LodLive [2]. RelFinder explores the knowledge graph to find paths between two entities. LodLive provides a visually appealing way to explore information associated with an entity.

Such tools and services exist independently, but together form an eco system, which provides added value for the DBpedia datasets. Our new interface integrates several tools in a generic manner – partly to increase user-friendliness and partly to showcase the achievements in the Linked Data space obtained so far.

3. DBPEDIA VIEWER USER INTERFACE

The new DBpedia Linked Data interface (DBpedia Viewer) co-exists with the previous interface which serves Linked Data content to machines and browsers without JavaScript support. While we can default to the new interface when it is supported by the visiting agent, users can easily switch between the two interfaces.

DBpedia Viewer brings several improvements on the presentation of information. In addition to many cosmetic changes, DBpedia Viewer provides improved layout choices and new functionalities that aim to make the UI more useful.

The layout has been adapted to provide additional functionality as naturally as possible. As illustrated in Figure 1, several new features are added, each discussed in Section 4.2. DBpedia Viewer is provided as open source on Github[5].

4. SYSTEM OVERVIEW

The architecture can be divided in three levels. The base level is the triple store, which is accessible using the SPARQL query language. The web server and server-side code are built on top of the database. The third level is the client-side code.

The old interface was implemented with server-side code that generated a very simple HTML/RDFa page. DBpedia Viewer provides two modes: A simple HTML/RDFa view for machines and clients without JavaScript support and a rich web interface for humans based on JavaScript. A switch between the two interfaces is also available. The rich web interface is based solely on client-side code to construct the web page. All SPARQL queries are executed directly from JavaScript which minimizes the page generation time and reduces the load on the web server.

Following, we provide an overview of DBpedia Viewer technology stack (Section 4.1). We elaborate on the interface features (Section 4.2) with an emphasis on the Triple Action Framework (Section 4.3). Finally, we discuss the extensibility of the tool(Section 4.4).

4.1 Technology Stack

We use the Open Source edition of the Virtuoso server[6] as our triple store. Virtuoso has a modular architecture and supports plugins. These plugins use the VSP (Virtuoso Server Pages) language to communicate with the triple store, serve web requests and provide Linked Data. The previous DBpedia interface was built as a Virtuoso plugin and the DBpedia Viewer is built on top of the existing code.

Most of the new interface is client-side logic written in JavaScript. As explained above, the client-side code constructs the page by issuing SPARQL queries to the configured endpoints. Therefore, it is easy to use the new interface for non-Virtuoso deployments by only serving the client-side code. However, in this case, the old server-generated HTML/RDFa interface is not preserved.

For the client-side logic, we used AngularJS [7], an open-source JavaScript framework that implements the Model View Controller (MVC) paradigm for web applications. For styling and layout, we used the CSS from Twitter Bootstrap[8].

4.2 Features

Following we provide an elaborate discussion of the new DBpedia Viewer features.

4.2.1 Pretty Box

The *pretty box* (part one of Figure 1) displays important properties of the viewed entity. There is a predefined set of facts we provide, namely: (1) a picture, (2) the title, (3) the types, (4) a short description and (5) links to other resources. These data are generated from the set of triples describing the viewed resources using predefined mappings. The DBpedia datasets provide most of this general information for all entities. In some cases, however, the picture or links to other resources are not available. The DBpedia Viewer does not perform automatic selection of relevant properties to display. This is out of the scope of this project. Our goal was to develop a UI that is customizable to configure relevant properties used to adapt the view.

In the top right corner of the pretty box (Figure 1), three icons trigger the *entity actions*. The currently available actions are links to alternate data representations (XML/RDF, n3, JSON-LD, ...) as well as links to consult the resource using alternative Linked Data browsers.

4.2.2 Search Bar

DBpedia Viewer provides search functionality with auto-complete capabilities by re-using the DBpedia Lookup service. DBpedia Lookup enables searching for DBpedia entities using strings or provides prefix-based suggestions.

4.2.3 Language Filtering

The language filtering system allows the user to choose a preferred display language. This filters all literal values based on the user preferences and displays only the relevant values. This feature is helpful on dbpedia.org where labels and
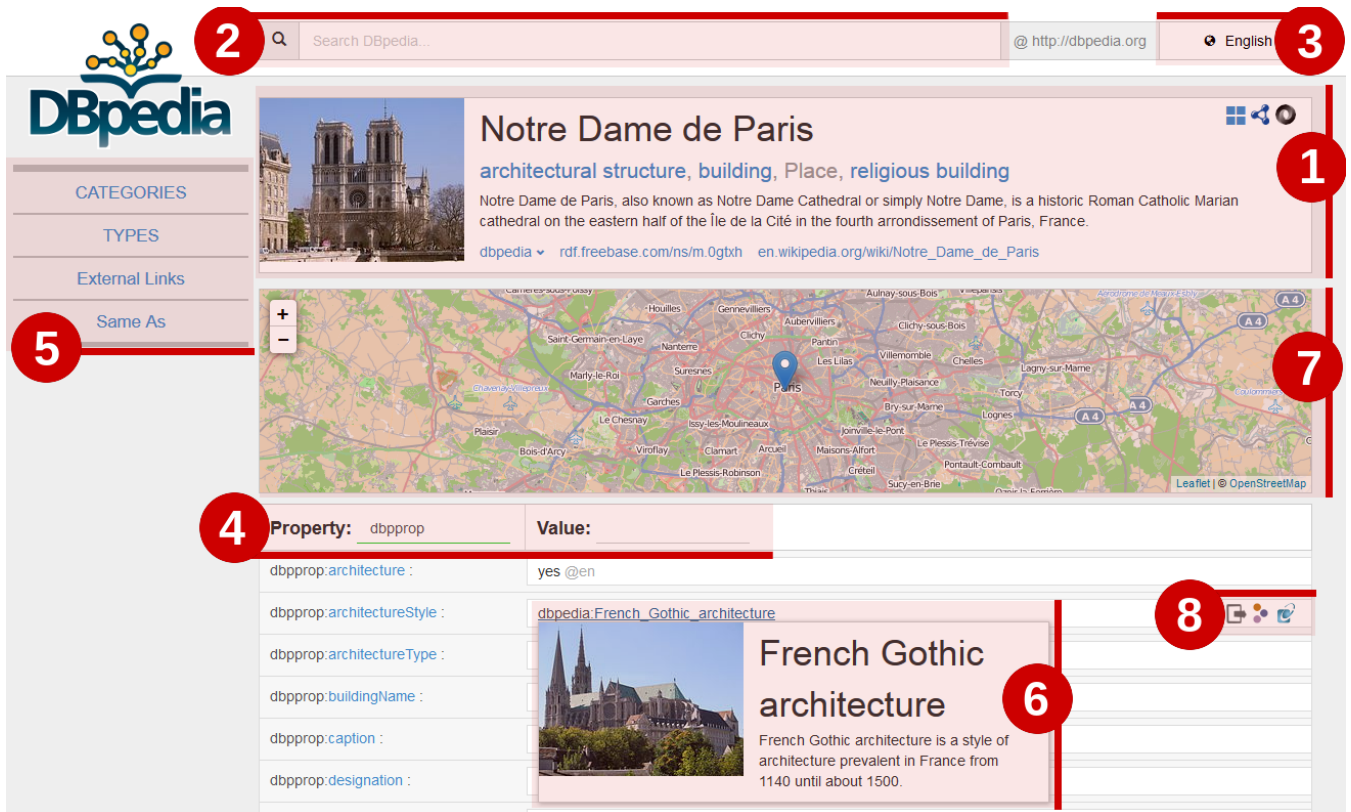

Figure 1: Screenshot of the new interface. The transparent red areas highlight new features, with the associated number in the red circle corresponding to its subsection number within Section 4.2. A quick overview: (1) pretty box, (2) search bar, (3) language switcher, (4) triple filter, (5) shortcuts, (6) preview box, (7) map and (8) triple actions.

abstracts exist in 12 different languages. In the case a literal does not exist in the preferred language, a fallback language (usually English) is chosen by default.

4.2.4 Triple Filtering

Part four of Figure 1 highlights the triple filtering feature. Triples can be filtered using both properties and values. This is useful for the users who quickly want to find specific properties and values. The filtering is based on string matching and supports all literal values as well as URIs.

4.2.5 Shortcut box

The shortcut box (part 5 in Figure 1) provides anchor links to some important properties of entities. However, the list of properties is currently hardcoded and contains links to categories, types, external links, etc...

4.2.6 Live Previews

When the user hovers over a DBpedia link (URI, ontology property or class) a concise, language-filtered preview is displayed. For entities, this preview contains a picture (if available), the title and a short description. Part 6 of Figure 1 shows a preview of the French Gothic architecture entity.

4.2.7 Maps

For entities having location information (latitude and longitude), a map is shown with its coordinates. OpenStreetMap is used for the map display.

4.2.8 Triple Actions

As displayed in part 8 of (Figure 1), next to each triple, different icons exist, each representing a different *triple action*. Triple actions are enabled using conditions on the triple. Thus, the set of available actions for different triples may be different. When the conditions are met, the action icon is displayed next to the triple. When the user clicks on the triple action icon, the action is executed. Below is an overview of the currently implemented user actions:

- Annotation – uses DBpedia Spotlight to annotate text. Only applicable to texts of certain length.

- RelFinder – links to RelFinder, where the connections (including indirect ones) between the viewed entity and the value entity can be explored. Only applicable to DBpedia resources.

- LodLive – opens the value entity with the LodLive browser. Only applicable to DBpedia resources.

- OpenLink Faceted Browser – view the value entity using OpenLink Faceted Browser. Only applicable to DBpedia resources.

- Wikipedia – opens the Wikipedia page associated with the value entity. Only applicable to DBpedia resources.

- DBpedia template mapping – links to the DBpedia mapping associated with the DBpedia template. Only applicable to DBpedia resources under the Wikipedia template namespace.

4.3 Triple Action Framework

The Triple Action Framework (TAF) aims to improve the integration of tools from the DBpedia eco system on the DBpedia website. The interface maintainer can easily add new actions or adapt existing ones for a particular deployment. TAF allows to define a triple action with the following core semantics, (1) bind and (2) execute.

Upon page load, for each triple, the bind method of each action is called to determine whether this action is applicable for this triple. The bind method may use any information available from the triple to decide whether the action is applicable or not. For example, the DBpedia Spotlight annotation action should only be made available for annotation of textual resources, so the bind method of this action checks whether the object of the triple is a string literal and whether it exceeds a configurable minimum length.

The execute method of an action is called when the user clicks on an action icon next to the triple. For the DBpedia Spotlight annotation action, this method uses the text in the object of the triple, sends it to the DBpedia Spotlight API for annotation and waits for a response. When the API responds, the Spotlight action changes the display value of the object of the triple to show the annotations.

In the actual implementation, TAF provides additional hooks, providing more functionality to define new actions with ease. Moreover, the Triple Action Framework allows to define locally/globally stateful actions and hidden (system-level) actions that execute upon binding and are not available to the users. Actions of the hidden type are used to detect coordinates for the map and to populate the shortcut box as well as some parts of the pretty box.

4.4 Extensibility

The triple action framework (TAF) promotes the extensibility of the new interface. The website maintainer can quickly define new actions and add them to the interface. TAF lightens adding new functionality by providing a useful abstraction where easy access to the triple is provided and displaying is already taken care of. To create a new action, one simply needs to implement the hooks with the desired action logic. We are looking for ways to make TAF action creation easier.

DBpedia is distributed in many language chapters [5]. DBpedia Viewer can be deployed on all DBpedia language editions by changing the configuration on the Virtuoso server. When deploying to other DBpedia chapters, the functionality supported in the DBpedia Viewer depends on whether this functionality is available for that dataset.

5. RELATED WORK

First, we discuss different tools that are integrated in the new interface. This is followed by an overview of some Linked Data browsers.

Integrated tools

RelFinder [4] allows users to explore connections between multiple entities in a intuitive and interactive way. Given two entities, RelFinder shows paths in the underlying RDF graph connecting the two entities. The relationship discovery algorithm used in [4] is based on the original DBpedia Relationship Finder algorithm [8]. The search algorithm is essentially a breadth-first search algorithm with several optimizations for the problem.

DBpedia Spotlight [9] is an Entity Linking (EL) system. Given a text, the purpose of EL is to find which parts of text refer to which entities. DBpedia Spotlight performs EL with DBpedia entities. The linking approach of DBpedia Spotlight consists of three steps: (1) the spotting stage where the phrases in the text are recognized that might refer to entities, (2) the candidate selection stage where possible "meanings" of spotted phrases are generated and (3) the disambiguation stage where the best candidate entity is chosen as the meaning of the phrase.

Another tool integrated as a triple action is LodLive [2]. LodLive is an exploratory tool that allows users to browse Linked Data in an interactive way, using a dynamic visual graph. Moreover, it integrates information available across different SPARQL endpoints. This way, it aims to showcase the principles behind Linked Data.

Linked Data browsers and integrators

Dadzie and Rowe [3] performed a survey of tools for Linked Data consumption. In their review, the authors make a distinction between visualization (e.g. RelFinder[4]) and presentation (e.g. Marbles). A wide range of tools is discussed and a comparative study of their features is performed. They also make a distinction between three kinds of users: (1) tech-users, (2) domain experts and (3) lay users. One of the conclusions of their survey is that the reviewed Linked Data consumption tools are mostly oriented at tech users. Some of the tools discussed by Dadzie and Rowe [3] are discussed in this section.

The Marbles Linked Data browser [9] is a server-side application that generates HTML from Semantic Web content using Fresnel [10] vocabularies. Marbles is used in DBpedia Mobile [1], a location-based Linked Data browser. DBpedia Mobile shows locations available from DBpedia on a map with information about the location.

Pubby [10] is a server-side Java application that can be configured to use a SPARQL endpoint and publish the data behind it as Linked Data. It also provides a simple (static) HTML user interface. The Graphity project [11] provides a framework for publishing RDF data or building applications around it. LDIF [11] is a framework aiming to integrate in-

formation about entities from different datasets but does not focus on displaying data.

6. CONCLUSION AND FUTURE WORK

The DBpedia Viewer is a step towards a customizable framework for interactive, user-friendly presentation of Linked Data. The original goal was a DBpedia-specific user interface that integrates some tools working with DBpedia. However, the Triple Action Framework also proved to be useful for defining system actions, which allow for greater and easier customization of the interface.

The interface does not try to conceal the technical philosophy behind Linked Data. Instead, it embraces the philosophy and presents the data as it is in a visually appealing fashion, highlighting the underlying ideas and demonstrating the possibilities of the integrated tools.

The Triple Action Framework (TAF) introduced with DBpedia Viewer demonstrates a method for adding interactive functionality to Linked Data, going beyond merely serving RDF facts. Such ideas may not only inspire improvements in other Linked Data interfaces but might also evolve to a standardized framework for human interaction with data across the Semantic Web in the future.

Future versions of DBpedia Viewer will enforce a stronger modularization between triples, actions and display, thus following the MVC design pattern. This will generalize the interface into a framework that can be customized for different datasets. We also plan to add triple actions for the incorporation of triple validation by the end users [6, 12]. Another triple action we are investigating is the option to automatically import DBpedia triples into WikiData.

A potential area of future research is the analysis of user behavior on the interface to produce novel Entity Summarization and Entity Ranking data and methods. Entity Summarization scores can be useful for Question Answering and Semantic Relatedness. The scores can also be used to extend the pretty box with the most important entity-specific information (e.g.: *birth place* for persons) and to compute a better list of properties for the shortcut box.

Acknowledgements

7. REFERENCES

[1] Christian Becker and Christian Bizer. DBpedia Mobile: A Location-Enabled Linked Data Browser. *LDOW*, 369, 2008.

[2] Diego Valerio Camarda, Silvia Mazzini, and Alessandro Antonuccio. LodLive, exploring the Web of Data. In *Proceedings of the 8th International Conference on Semantic Systems*, pages 197–200. ACM, 2012.

[3] Aba-Sah Dadzie and Matthew Rowe. Approaches to visualising linked data: A survey. *Semantic Web*, 2(2):89–124, 2011.

[4] Philipp Heim, Steffen Lohmann, and Timo Stegemann. Interactive Relationship Discovery via the Semantic Web. In *Proceedings of the 7th Extended Semantic Web Conference (ESWC 2010)*, volume 6088 of *LNCS*, pages 303–317, Berlin/Heidelberg, 2010. Springer.

[5] Dimitris Kontokostas, Charalampos Bratsas, Sören Auer, Sebastian Hellmann, Ioannis Antoniou, and George Metakides. Internationalization of Linked Data: The case of the Greek DBpedia edition. *Web Semantics: Science, Services and Agents on the World Wide Web*, 15(0):51 – 61, 2012.

[6] Dimitris Kontokostas, Amrapali Zaveri, Sören Auer, and Jens Lehmann. TripleCheckMate: A Tool for Crowdsourcing the Quality Assessment of Linked Data. In *Proceedings of the 4th Conference on Knowledge Engineering and Semantic Web*, 2013.

[7] Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick van Kleef, Sören Auer, and Christian Bizer. DBpedia - A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia. *Semantic Web Journal*, 2014.

[8] Jens Lehmann, Jörg Schüppel, and Sören Auer. Discovering Unknown Connections - the DBpedia Relationship Finder. In *Proceedings of the 1st Conference on Social Semantic Web (CSSW 2007)*, volume 113 of *LNI*, pages 99–110. GI, 2007.

[9] Pablo N Mendes, Max Jakob, Andrés García-Silva, and Christian Bizer. DBpedia Spotlight: shedding light on the Web of Documents. In *Proceedings of the 7th International Conference on Semantic Systems*, pages 1–8. ACM, 2011.

[10] Emmanuel Pietriga, Christian Bizer, David Karger, and Ryan Lee. Fresnel: A browser-independent presentation vocabulary for rdf. In Isabel Cruz, Stefan Decker, Dean Allemang, Chris Preist, Daniel Schwabe, Peter Mika, Mike Uschold, and LoraM. Aroyo, editors, *The Semantic Web - ISWC 2006*, volume 4273 of *Lecture Notes in Computer Science*, pages 158–171. Springer Berlin Heidelberg, 2006.

[11] Andreas Schultz, Andrea Matteini, Robert Isele, Pablo N Mendes, Christian Bizer, and Christian Becker. LDIF - A Framework for Large-Scale Linked Data Integration. In *21st International World Wide Web Conference (WWW 2012), Developers Track, Lyon, France*, 2012.

[12] Amrapali Zaveri, Dimitris Kontokostas, Mohamed A. Sherif, Lorenz Bühmann, Mohamed Morsey, Sören Auer, and Jens Lehmann. User-driven Quality Evaluation of DBpedia. In *To appear in Proceedings of 9th International Conference on Semantic Systems, I-SEMANTICS '13, Graz, Austria, September 4-6, 2013*. ACM, 2013.

---

[12] http://www.google-melange.com/gsoc/org2/google/gsoc2013/dbpediaspotlight