

# DashSearch LD: Exploratory Search for Linked Data

Takayuki Goto<sup>1</sup>, Hideaki Takeda<sup>1,2</sup>, and Masahiro Hamasaki<sup>3</sup>

<sup>1</sup> National Institute of Informatics,  
tygoto@nii.ac.jp, takeda@nii.ac.jp

<sup>2</sup> The Graduate University for Advanced Studies

<sup>3</sup> National Institute of Advanced Industrial Science and Technology (AIST)  
masahiro.hamasaki@aist.go.jp

**Abstract.** Although a large number of datasets gathered as Linked Open Data (LOD) is better for data sharing and re-using, the datasets themselves become more difficult to understand. Since each dataset has its own data structure, we need to understand datasets individually. In addition, since the entities in datasets are interconnected, we need to understand the interconnections between datasets. In other words, understanding the data is crucial for exploiting LOD. In this paper, we show a novel system called DashSearch LD to understand and use LOD with an exploratory search approach. The user interactively explores datasets by viewing and selecting entities in the datasets. Specifically, the user manipulates widgets on the screen by moving and overlapping them with a mouse to check entities, draw detail data on them, and obtain other entities linked by the widgets.

**Keywords:** Linked Open Data, SPARQL, Exploratory Search, Search Interface, Facet Search, Human-Computer Interaction

## 1 Introduction

Linked Open Data (LOD) is now rapidly growing in size and variety. LOD has great potential to raise the value of data by sharing and re-using data that has been hidden behind websites.

Although the increasing number of datasets gathered as LOD is better for data sharing and re-using, the datasets themselves become more difficult to understand. Since each dataset has its own data structure, we need to understand datasets individually. In addition, since the entities in datasets are interconnected, we need to understand the interconnections between datasets. In other words, LOD as a whole consists of a huge and complicated database, and understanding the data is crucial for exploiting LOD.

Understanding data is twofold; one aspect is understanding the schema and their relationships. For example, users need to know what kinds of classes exist, how classes are represented, and how they are interconnected. The other aspect is understanding the data distribution. Among the defined classes in a dataset,

some are heavily used, so many entities exist. Conversely, classes not heavily used have few entities. The same is true for properties: some properties always have values but others often lack values. Such skewed distributions are observed in real datasets.

In this paper, we show a novel system called DashSearch LD to understand and use LOD with an exploratory search approach, which can help users to understand data with respect to their structure and distribution. DashSearch LD is an LOD version of a widget-based desktop search system [2]. The user interactively explores datasets by viewing and selecting entities by manipulating widgets, e.g., moving and overlapping them, on the screen with a mouse, to check entities, draw detail data on them, and obtain other entities linked by the widgets. Through browsing and exploring data, the user can understand data *as is*, i.e., the distribution as well as the structure of the data.

## 2 Related Work

In LOD, although understanding both the distribution and the structure is important, existing studies mainly focus on one or the other. iSPARQL [6] and NITELIGHT [8] help users to compose SPARQL queries by providing a graph interface. A graph interface is suitable because it conforms to the nature of the Resource Description Framework (RDF) and is intuitive for users. These interfaces rely on pre-registered vocabularies, but many vocabularies can occur when traversing LOD graphs. So, it is difficult to apply graph interfaces to LOD. In addition, understanding the schema is required. DERI Pipes [7], which is a semantic version of Yahoo! Pipes<sup>1</sup>, enables rapid development of mashup applications by providing a graphical user interface for flow diagrams. Instead of RSS, DERI Pipes can manipulate RDF triples and SPARQL queries. DERI Pipes is user friendly and convenient for composing applications, but understanding the schema is again required. MashQL [4], although similar to DERI Pipes, is a more sophisticated approach to construct SPARQL queries. In MashQL, the schema is detected by the system and specified graphically by the user. It is an advantage to detect the schema from the data, but its interpretation is still left to the users.

Understanding data is another issue. Deligiannidis et al. proposed a system called Paged Graph Visualization [1], which visualizes RDF triples so that the user can explore them by traversing graphs. This system helps the user to understand both the data and the data structure. However, showing all individual relations is often troublesome and prohibits the user from comprehensive understanding. gFacet [3] is the most relevant work of applying graphical user interface techniques to browse databases. Whereas gFacet enables tree-like exploration of a dataset, DashSearch LD emphasizes the flexibility of the exploration of datasets. DashSearch LD offers more flexible data operations, such as combining the search results from different datasets, clipping search results, as well as providing more extensive interfaces using widget metaphors.

---

<sup>1</sup> <http://pipes.yahoo.com/pipes/>

### 3 DashSearch Interface

DashSearch LD is a Flash web application (Fig. 1). DashSearch LD provides a workspace and visual objects called widgets for operations of Linked Data. Widgets include an “endpoint widget” with the function of the SPARQL endpoint and a “metadata widget” that displays properties and their values.

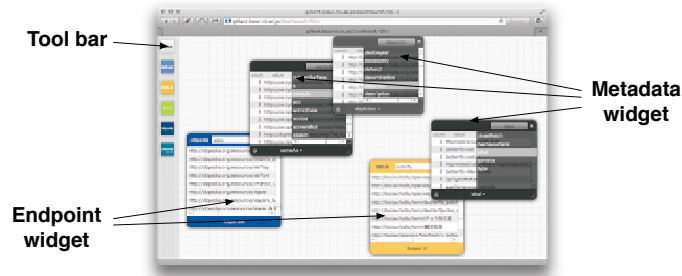


Fig. 1. DashSearch LD interface

Users can retrieve resources by typing strings on the endpoint widget. The result is a list of resources. Next, users can see all the properties by overlapping a metadata widget, as shown in Fig. 2. If they select a property in the list, all the values of the property starting from the retrieved resources are shown with the frequency of the values. Users can understand the distribution of data by browsing properties with the property values. Selecting a value of a property narrows the search results.

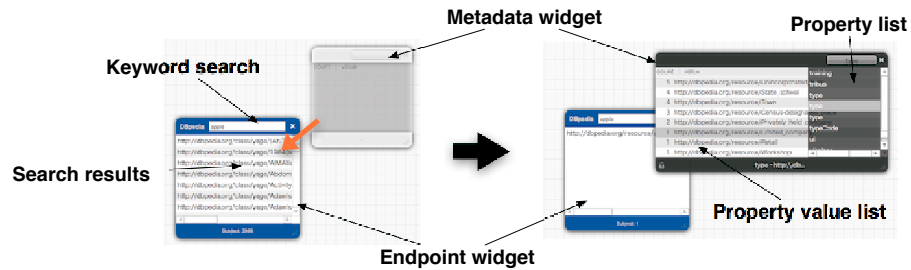
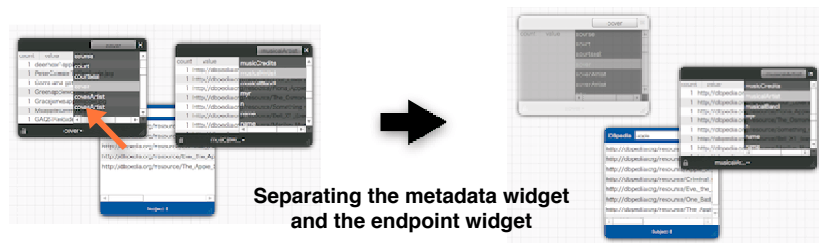


Fig. 2. Property browsing

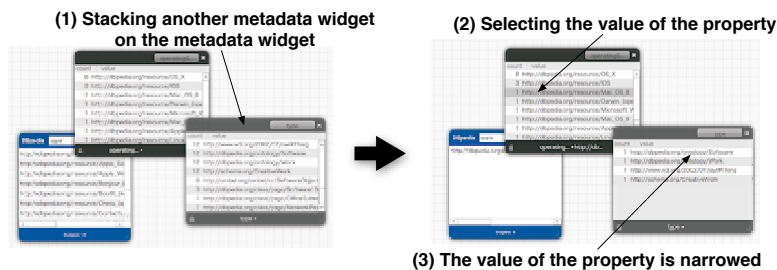
Users can compose a more complex search condition by combining metadata widgets. They can compose AND queries easily by attaching multiple metadata widgets (Fig. 3 Left). The search condition can be removed from the query and re-used by separating metadata widgets (Fig. 3 Right). The detached metadata

widget still contains the search condition (either a property or a value for a metadata widget) and can be used again. Thus, the addition, deletion, and repetition of search conditions can be facilitated by rearranging widgets by controlling the mouse, and a complex search can be formalized and performed quickly.



**Fig. 3.** AND operation by combining widgets

DashSearch LD enables a faceted search by overlaying multiple metadata widgets. First, users stack another metadata widget on one metadata widget. Next, they select the value of the property of the metadata widget, as shown in Fig. 4. Then, the metadata widget's properties and their values on the stack are narrowed with resource narrowing. By employing the facet search, users can understand the values of the properties related to the value of the selected property.



**Fig. 4.** Faceted search

With DashSearch LD, users can mix multiple datasets by using multiple endpoint widgets. The AND search with multiple endpoint widgets can yield results shared by the datasets. As mentioned, users can easily re-use a search condition of a metadata widget. Re-use of a search condition is also applicable to other endpoint widgets, and so users can apply the same widgets used for browsing other datasets. In addition, users can search different datasets by the same metadata used with another endpoint widget (Fig. 5).

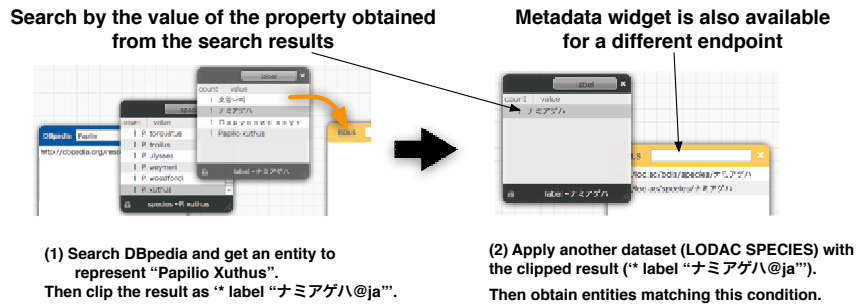


Fig. 5. Search by the same metadata widget

Thus, by using DashSearch LD, users can understand the data structures and data distribution by browsing the metadata.

## 4 Implementation

The architecture of the system is shown in Fig. 6. The basic flow of the system is 1) acquiring data by SPARQL via SPARQL endpoints, 2) caching the acquired data, 3) filtering the data, and 4) displaying the data. An endpoint widget obtains triples, the subjects of which have a label containing the search query. Querying by SPARQL is done iteratively with consideration of the server's load, which can yield up to 10,000 triples as the result. Then, metadata widgets can filter the set of these triples. A metadata widget obtains RDF data from an endpoint widget with respect to a selected property. If a metadata widget is put on another metadata widget, it obtains RDF data from the metadata widget located below it. Thus, the RDF data is filtered sequentially. The search results of the endpoint widget are narrowed by the AND query that consists of the value of the property specified by each metadata widget. Thus, DashSearch LD enables users to browse data by filtering from various viewpoints. The system is fully implemented and openly accessible<sup>2</sup>.

Currently, DBpedia<sup>3</sup>, DBpedia Japanese<sup>4</sup>, LODAC Museum [5] and LODAC Species are listed as endpoints<sup>5</sup>. A simple query example with these datasets is searching some entities (such as universities) that have the same property and value (such as the homepage URL) in DBpedia and DBpedia Japanese. A more complicated example is exploring species information, such as searching species by name from DBpedia, obtaining all species in the same genus as the search species, selecting and clipping one of the species, then searching LODAC Species with the clipped search condition. All operations are done without typing except for the initial input for the text search (Fig. 5).

<sup>2</sup> <http://www.ahirulab.com/dashsearch/lod/>

<sup>3</sup> <http://dbpedia.org/sparql>

<sup>4</sup> <http://ja.dbpedia.org/sparql>

<sup>5</sup> <http://lod.ac>

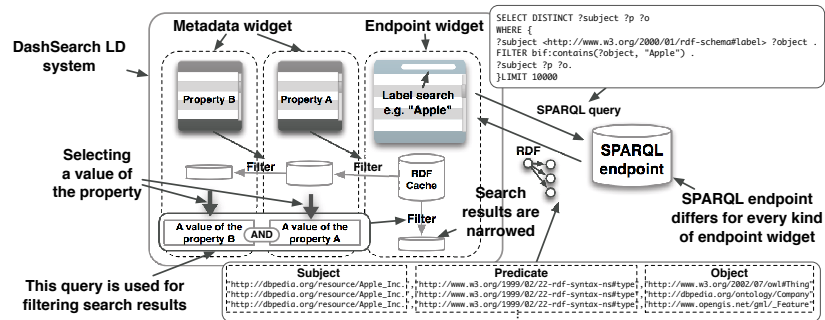


Fig. 6. DashSearch LD system

## 5 Conclusion

We described a retrieval interface called DashSearch LD that uses widgets to understand the LOD. Our main contribution is to provide a novel exploratory search environment for LOD that enables users to understand LOD through an intuitive interface. Users can interactively explore one or multiple datasets by viewing, selecting, and interacting with the entities in these datasets. In addition, users can just manipulate widgets on the screen to check entities, draw detail data on them, and obtain other entities linked by the widgets. Thus, through exploratory searching, DashSearch LD can help users to understand LOD *as is*, including not only the structure and the distribution of data but also the implicit association between datasets.

## References

1. Deligiannidis, L., Kochut, K.J., Sheth, A.P.: RDF data exploration and visualization. In: Proc. of CIMS2007. pp. 39–46 (2007)
2. Goto, T., Takeda, H., Yasumura, M.: Dash search: Dash Search: Desktop Widget based Desktop Search for Metadata Exploitation. In: Adjunct Proc. of UIST2008. pp. 37–38 (2008)
3. Heim, P., Ertl, T., Ziegler, J.: Facet graphs: Facet Graphs: Complex Semantic Querying Made Easy. In: Proc. of ESWC2010, vol. 6088, pp. 288–302 (2010)
4. Jarrar, M., Dikaiakos, M.D.: A Query Formulation Language for the Data Web. IEEE Transactions on Knowledge and Data Engineering (2011)
5. Kamura, T., Takeda, H., Ohmukai, I., Kato, F., Takahashi, T., Ueda, H.: Study Support and Integration of Cultural Information Resources with Linked Data. In: Proc. of Culture and Computing 2011. pp. 177–178 (2011)
6. Kiefer, C., Bernstein, A., Stocker, M.: The Fundamentals of iSPARQL: A Virtual Triple Approach For Similarity-Based Semantic Web Tasks. In: ASWC2007 (2007)
7. Phuoc, D.L., Polleres, A., Morbidoni, C., Hauswirth, M., Tummarello, G.: Rapid semantic web mashup development through semantic web pipes. In: WWW2009 (2009)
8. Russell, A., Smart, P.R., Braines, D., Shadbolt, N.: NITELIGHT: A Graphical Tool for Semantic Query Construction. In: SWUI2008 (2008)