

Design and Implementation of a Standpoint-based Linked Data Visualization Approach

Hiroaki Morikawa and Fumihito Nishino

Fujitsu Laboratories Limited, 4-1-1 Kamikodanaka, Nakahara-ku, Kawasaki, Kanagawa, Japan

Abstract. In this paper, we propose a standpoint-based Linked Data visualization system, as a step in the development of a Linked Data visualization system that receives an entity and visualizes the data related to it in various ways based on multiple “standpoints”. Here, a standpoint denotes the data generated from a user’s perspective of interest in an entity and its visualization. For example, “volcano” and “World Heritage Site” are standpoints with respect to Mt. Fuji. Considering these standpoints, a user interested in “volcano” in our system is shown various types of volcanoes and the history of volcanic eruptions, whereas a user interested in “World Heritage Site” is shown a list of viewing spots and the list of visitor numbers per year. This feature is not found in existing Linked Data browsers which can be defined as “single” standpoint browsers. In this paper, we present our proposed approach and describe a prototype system based on the proposed approach.

1 Introduction

In recent years, there has been increasing use of Linked Data as a method for publishing data on the Web[1]. Traditional Web data is written in natural language and exists in a network of documents (Web of documents), which is easy for people to read and understand. In contrast, Linked Data is written in a resource description framework (RDF) as recommended by the World Wide Web Consortium (W3C), and a network of data (Web of Data), which is difficult for people to read and understand. Data visualization is an approach for solving this problem, and a number of visualization approaches have been proposed[2]. Now, we consider a visualization for an entity. At this time, the content and method of visualization differ depending on the person despite the same data. This means a different “standpoints” depending on the person. In this paper, we present our proposed approach to visualize the data in various ways based on multiple “standpoints”.

2 Related work

A number of visualization approaches have been proposed. Pubby[3], Tabulator[4], Marbles[5] and Disco[6] are Linked Data browsers that were developed as early-stage Linked Data visualization tools; such browsers explore Linked Data

using semantic links and provide an HTML view. These browsers display an entity via a fixed view defined by the system; therefore, developers using Linked Data cannot make views and visualization freely according to a custom design. Sgvizler[7], D3SPARQL[8] and SemwidgJS[9] can be used to construct visualized parts by SPARQL query and visualization definition via a programming language such as javascript. The visualized parts can be reused as widgets. Information Workbench[10], a dynamic semantic publishing[11] platform for Linked Data applications in the enterprise, provides a mashup function that uses a template (based on Semantic MediaWiki[12]) and widgets. A mashup consists of processes for both aggregating data from several data sources and displaying the data through a graphical user interface (GUI). Information Workbench can be used to visualize data by aggregating it from multiple data resources and then display the results side-by-side. The Linked Data Reactor[13] can be used to a view for each Dataset, Resource, Property, Value (called "Scope"), and then provide a mashup of the visualization parts according to the type of the entity automatically or manually to build the desired application. The page construction method in our proposed approach imitates the approaches of Information Workbench and Linked Data Reactor. To the best of our knowledge, the existing visualization tools described above can consider only a "single" standpoint. In other words, these tools visualize an entity in the same way at all times.

3 Approach

Our proposed approach is to incorporate a mechanism for handling multiple "standpoints" in the method adopted by Information Workbench and Linked Data Reactor. In this paper, a standpoint denotes the data generated from a user's perspective of interest regarding an entity and its visualization.

Consider the case of visualizing the entity Mt. Fuji; in this case, "volcano" and "World Heritage Site" are example standpoints with respect to Mt. Fuji. Considering these standpoints, to a user interested in "volcano" our proposed approach will show various types of volcanoes and the history of volcanic eruption; to a user interested in "World Heritage Site" it will show a list of viewing spots and a list of visitor numbers per year.

Our proposed approach adopted a dynamic semantic publishing publishing receives a "viewpoint" and "standpoints" and then displays the data using a view corresponding to each of them. Here, "viewpoint" an entity used to start the search; for example, in the current case, the viewpoint is a mountain such as "Mt. Fuji", "Mt. Aso", "Sakurajima", "Yakushima", and each of these examples are represented as a Uniform Resource Identifier (URI). The term "standpoints" refers to a class of an entity, for example, "volcano", "World Heritage Site", and "island".

Page is constructed manually. First, we develop a view part called by a gadget. A gadget consist of a pair of a query and a visualization code same as Sgvizler, D3SPARQL or SemwidgJS etc. Next, set constructed gadgets in a page, finally, correspond a class to a page. Gadgets can be reused in other pages. If our proposed approach receive an entity as a "viewpoint", then it visualizes a page corresponding a class of an entity as "standpoints".

4 Prototype system

In this section, we describe our prototype system. The prototype is equipped on LOD4LL¹; for more details, the viewer can be found at <https://lod4all.net/frontend/>.

The prototype system receives a URI corresponding to an entity as a “viewpoint” or an entity related a keyword as a “viewpoint”. By using keyword search, the prototype system presents a list of URIs related to the keyword. Therefore, even if the user does not know the entity’s URI that corresponds to the “viewpoint”, the user can determine the entity’s URI by using the keyword search function. The prototype system uses a *rdf:type* in an entity as “standpoints”, and does not receive “standpoints” as a user input.

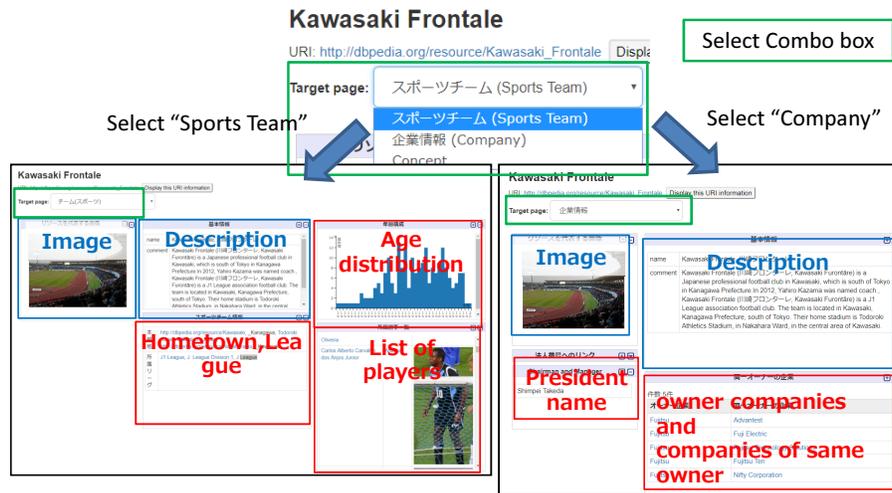


Fig. 1. Display of the same entity for different standpoints

The prototype system visualize the data by a definition corresponding to a *rdf:type* from the entity. Fig.1 shows an example of visualization for the case of “Kawasaki Frontale (http://dbpedia.org/resource/Kawasaki_Frontale)”².

When Kawasaki Frontale is regarded as “Sports Team (<http://dbpedia.org/ontology/SportsTeam>)”, the prototype system visualizes a hometown, a league name, an age distribution, and a list of players (See:https://lod4all.net/frontend/page/display?uri=http%3A/dbpedia.org/resource/Kawasaki_Frontale&page_id=SportsTeam). When Kawasaki Frontale is regarded as “Company (<http://dbpedia.org/ontology/Company>)”, the prototype system visualizes a president name, owner companies, and companies of same owner(See:https://lod4all.net/frontend/page/display?uri=http%3A/dbpedia.org/resource/Kawasaki_Frontale&page_id=Company). An image display part and a description display part are used by both pages. These are a same part on both pages.

The prototype has a similar page search function, and when there is no page for the type given to the entity, the system searches for an equivalent type

¹ <https://lod4all.net/> (accessed Sep. 2018)

² Kawasaki Frontale is the soccer club sponsored by FUJITSU in Japan .

(“<http://www.w3.org/2002/07/owl#equivalentClass>”) or a parent type (“<http://www.w3.org/2000/01/rdf-schema#subclassOf>”), if the page is defined for the type. If a page corresponding to one of these types is found, then the prototype attempts to visualize the entity according to these types. If the similar page search function does not find the page, then the prototype visualizes the entity using the default page. The entity has no class, then the prototype also visualizes the entity using the default page. The default page is a simple page, for example, provided by the Linked Data browser.

5 Conclusion

We proposed a standpoint-based Linked Data visualization approach. Our proposed approach can visualize the data related to it in various ways based on multiple “standpoints”. In this work, we confirmed that it is important to assign ontology about data, when utilizing data based on our approach using a dynamic semantic publishing. In the future, a Linked Data service developer can develop APIs by using this approach. This means that the proposal method is also a standpoint-based Linked Data reconstruction approach.

References

1. Bizer, Christian, Tom Heath, and Tim Berners-Lee. “Linked data-the story so far.” *Semantic Services, Interoperability and Web Applications: Emerging Concepts* (2009): 205-227.
2. Dadzie, Aba-Sah, and Matthew Rowe. “Approaches to visualising linked data: A survey.” *Semantic Web 2.2* (2011): 89-124.
3. Cyganiak, Richard, and Chris Bizer. “Pubby-a linked data frontend for sparql endpoints.” Retrieved from <http://www4.wiwiss.fu-berlin.de/pubby/> at May 28 (2008): 2011.
4. Berners-Lee, Tim, et al. “Tabulator: Exploring and analyzing linked data on the semantic web.” *Proceedings of the 3rd international semantic web user interaction workshop*. Vol. 2006. 2006.
5. Becker, Christian, and Christian Bizer. “DBpedia Mobile: A Location-Enabled Linked Data Browser.” *Ldow* 369 (2008): 2008.
6. Bizer, C., Gaub, T. (2007). Disco. <http://www4.wiwiss.fu-berlin.de/bizer/ng4j/disco/>. (accessed Sep. 2018)
7. Skjveland, Martin G. “Sgvizler: A javascript wrapper for easy visualization of sparql result sets.” *Extended Semantic Web Conference*. Springer, Berlin, Heidelberg, 2012.
8. Katayama, Toshiaki. “D3SPARQL: JavaScript Library for Visualization of SPARQL Results.” *SWAT4LS*. 2014.
9. Stegemann, Timo, and Jrgen Ziegler. “SemwidgJS: A Semantic Widget Library for the Rapid Development of User Interfaces for Linked Open Data.” *GI-Jahrestagung*. 2014.
10. Haase, Peter, et al. “The information workbench interacting with the web of data.” (2009).
11. Rayfield, Jem. “Dynamic semantic publishing.” *Semantic Technologies in Content Management Systems*. Springer, Berlin, Heidelberg, 2012. 49-64.
12. Krötzsch, Markus, Denny Vrandečić, and Max Völkel. “Semantic mediawiki.” *International semantic web conference*. Springer Berlin Heidelberg, 2006.
13. Khalili, Ali, Antonis Loizou, and Frank van Harmelen. “Adaptive linked data-driven web components: Building flexible and reusable semantic web interfaces.” *International Semantic Web Conference*. Springer, Cham, 2016.