Towards an Open-Governmental Data Web

Ivan Ermilov[1], Claus Stadler[1], Michael Martin[1], and Sören Auer[2]

[1] Universität Leipzig, D-04109 Leipzig, Germany
{iermilov|martin|cstadler}@informatik.uni-leipzig.de
[2] Technische Universität Chemnitz, D-09107 Chemnitz, Germany
soeren.auer@informatik.tu-chemnitz.de

Abstract. Up to the present day much effort has been made to publish government data on the Web. However, such data has been published in different formats. For any particular source and use (e.g. exploration, visualization, integration) of such information specific applications have to be written. This limits the overall usability of the information provided and makes it difficult to access information resources. These limitations can be overridden, if the information will be provided using a homogeneous data and access model complying with the Linked Data principles. In this paper we showcase how raw Open Government Data (OGD) from heterogeneous sources can be processed, converted, published and used on the Web of Linked Data. In particular we demonstrate our experience in processing of OGD on two use cases: the Digital Agenda Scoreboard and the Financial Transparency System of the European Commission.

1 Introduction

Up to the present day much effort has been made to publish government data on the Web. However, such data has been published in different formats. For any particular source and use (e.g. exploration, visualization, integration) of such information specific applications have to be written. This limits the overall usability of the information provided and makes it difficult to access information resources.

In the article "Government Data and the Invisible Hand" [16] the authors discuss interfaces provided by US government websites. One of the web resources mentioned in the article – *regulations.gov* contains information on the development of federal regulations and other related documents issued by the US government. This critical information for US citizens can be accessed through a set of interfaces: an internal search engine, RSS feeds as well as files in HTML or PDF format. Here, all the provided interfaces are not machine-readable. In the end a user obtains the raw text document, that she can read, but not her computer. Thus, related information about regulations of the economic sector published in the year 2001 with fluctuations of the gross domestic product during this and subsequent years, for instance, is hardly possible. Similar problems occur in any other scenario, that requires integrating several information sources. However, (semi-)automatic integration and visualization is possible if the data is available adhering to the RDF model (i.e. converted to Linked Data).

The benefits of exposing government information as Linked Data [3] can be generalized into two categories: (a) *universal access* for the different information sources (i.e. access via a SPARQL endpoint) – enables to use existing visualization, analysis and information mining techniques; (b) *enrichment and integration* of information (i.e. interlinking datasets with existing tools like LIMES [15]) – makes it possible, for instance, to analyze information from new viewpoints.

In this paper we showcase how raw Open Government Data (OGD) from heterogeneous sources can be processed, converted, published and used on the Web of Linked Data. The contributions of our work are in particular:

- We outline well-known steps for publishing Linked Open Government Data (LOGD) [17] on the Web on Linked Data (section 2) and supplement it with an extensible visualization framework, that enables new applications for existing LOGD (section 3).
- We showcase how the whole transformation-exploitation workflow can be used in two case studies: the Digital Agenda Scoreboard and the Financial Transparency System of the European Commission (section 4).

We provide an overview on related work in the areas of LOGD publishing approaches and principles of eGovernment in section 5 and conclude with an outlook on future work in section 6.

2 Linked Open Government Data Publishing

In order to exploit the benefits of Open Government Data, governments all around the globe started to create and maintain the OGD dataset catalogs. For instance, the US government publishes dataset information on the *data.gov* web portal. To maximize the benefits, the datasets from such portals have to be processed and transformed to LOGD.

We adopt the lifecycle of OGD from [17]. In essence, constituents of the OGD lifecycle are:

1. *Collecting and cleansing.* We collect the government data from heterogeneous resources and clean it from noise and duplicates. The result of this step is the data ready for analysis and conversion.
2. *Modeling and converting.* The domain vocabulary for cleaned data is modeled and the data is converted to RDF documents using the developed vocabulary. RDF documents are already comply with the Linked Data access model. However, the next steps are a prerequisite for the successful utilization of the obtained information.
3. *Publishing and licensing.* RDF documents are published as dataset dumps, individual resource descriptions or in an RDF triple store. Licensing information should be provided together with the published dataset.
4. *Interlinking.* We interlink published dataset with other datasets over the Web of Linked Data. Interlinking raises the value of the dataset and enables new viewpoints on the information contained in a dataset.

5. *Disseminating.* The dataset is advertised in relevant dataset catalogs and stakeholder communities.
6. *Exploiting.* The result of this step is a set of applications and visualizations of the dataset information.

3 Vision: an Ecosystem of LOD Visualizations

Open Government Data provides added value for the stakeholders (i.e. citizens, journalists, policy makers, researchers etc.) if they can browse and explore the data and thus gain new knowledge and insights. This is one of the currently most crucial and challenging aspects for LOGD. In this section we describe our vision of an ecosystem of LOD visualizations, which targets this issue by providing generic visualization possibilities.

The ecosystem will comprise generic and domain specific visualization tools (e.g. for spatial and statistical data) based on HTML, CSS and JavaScript which can accept and handle output from the Linked Data API and generic web APIs. For example, such visualizations can include map views of spatial information (e.g. for WMS/WFS endpoints, geocoded data) as well as common graphs and charts for statistical information (e.g. statistical data in the DataCube RDF vocabulary as well as CSV time series data).

Such a visualization and exploration strategy is based on an analysis of the datasets hosted on the Web of Linked Data. According to the structure of the data (i.e. the vocabularies used) different visualization and exploration widgets can be automatically launched, configured and offered to the user for exploration.

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. Ultimately, we aim to realize an ecosystem of LOGD and visualizations, which can be bound together in a dynamic and unforeseen way. This will enable users to explore LOGD datasets even if the publisher of the data does not provide any exploration or visualization means.

Our envisioned architecture, depicted in Figure 1, is composed of three layers. It resembles a classical three-tier architecture, however at Web scale. We argue that this will lead to an infrastructure of reusable components, eventually lowering development times of LOGD, and thereby providing added value to end users more quickly. Its components are:

- At the base there is the *LOD cloud* which fuels the other tiers with data. The three main data access methods are downloads, Linked Data and SPARQL. The latter is usually preferred by application developers, as complex queries can be posed directly. In the case of dataset downloads one usually has to

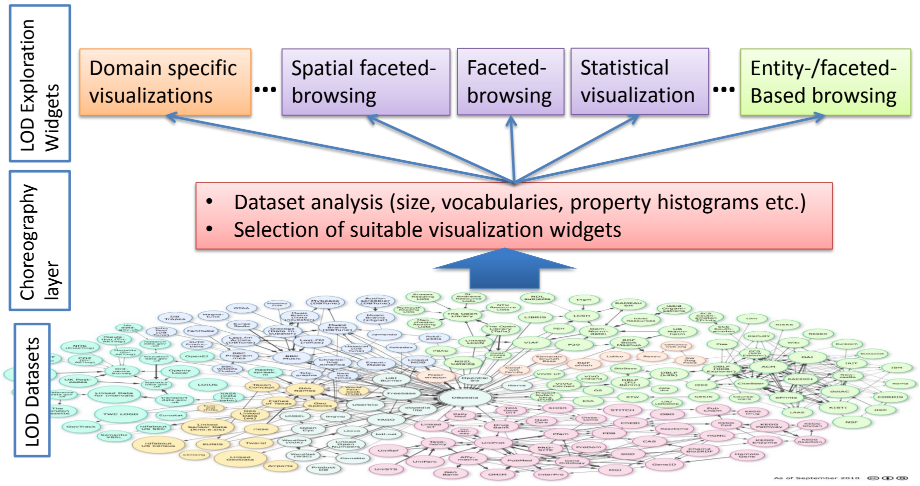

LOD Exploration Widgets
Choreography layer
LOD Datasets
Domain specific visualizations
...
Spatial faceted-browsing
Faceted-browsing
Statistical visualization
...
Entity-/faceted-Based browsing
• Dataset analysis (size, vocabularies, property histograms etc.)
• Selection of suitable visualization widgets
As of September 2010

Fig. 1. Vision: an Ecosystem of LOD Visualizations.

load them into a local RDF store, or in the case of Linked Data a dataset first has to be created by crawling.

- The *Choreography layer* is formed by a set of services that are capable of suggesting suitable visualizations for a given dataset. In general, such suggestions could be presented in both human and machine readable formats, such as HTML and RDF. The content may include several pointers, such as to: project pages of visualization software, visualization APIs, dynamically or precomputed images, and (preconfigured) links to visualization services (e.g. GoogleMaps API). Thereby these choreography services may base their choice on meta data services, such often provided by data set inventories (e.g. *the data hub*) or generic LOD2 analysis services (e.g. *lodstats* [7]).
- The LOD *exploration layer* consists of the tools and services capable of visualizing LOD data. Ideally, these services should offer flexible configuration parameters, so that the choreography services can interface with them by creating configurations dynamically.

In the following we describe two of our own development efforts that contribute to this vision, and which are aimed at enabling users to explore governmental data.

CubeViz: statistical visualization application. CubeViz is a facetted browser and visualization tool for statistical RDF data. CubeViz is applicable for browsing and visualizing datasets utilizing the RDF Data Cube vocabulary [6]. The Data Cube vocabulary is the state-of-the-art in representing statistical data in RDF. It is compatible with SDMX[3] and increasingly being adopted.

[3] http://sdmx.org/

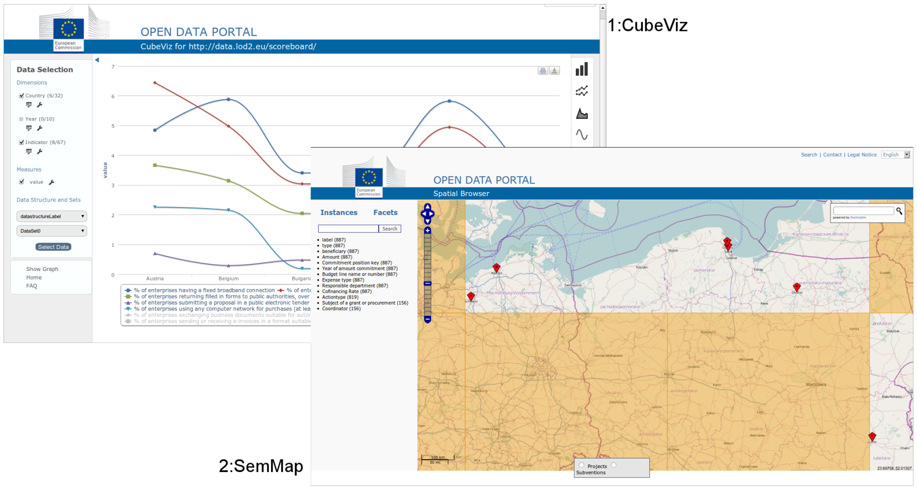

1:CubeViz
OPEN DATA PORTAL
CubeViz for http://data.lod2.eu/scoreboard/
Data Selection
Dimensions
☑ Country (6/32)
Year (0/10)
☑ Indicator (8/67)
Measures
☑ value
Data Structure and Sets
datastructureLabel
DataSet0
Select Data
Show Graph
Home
FAQ
value
7
6
5
4
3
2
1
0
Austria
Belgium
Bulgaria
% of enterprises having a fixed broadband connection
% of ente
% of enterprises returning filled in forms to public authorities, over
% of enterprises submitting a proposal in a public electronic tender
% of enterprises using any computer network for purchases (at lea
% of enterprises exchanging business documents suitable for autor
% of enterprises sending or receiving e-invoices in a format suitable
2:SemMap
Search | Contact | Legal Notice
English
OPEN DATA PORTAL
Spatial Browser
Instances
Facets
Search
label (887)
type (887)
beneficiary (887)
Amount (887)
Commitment position key (887)
Year of amount commitment (887)
Budget line name or number (887)
Expense type (887)
Responsible department (887)
Cofinancing Rate (887)
Actiontype (819)
Subject of a grant or procurement (156)
Coordinator (156)
○ Projects ○ Subventions
100 km
50 mi
23.69708, 52.01307

Fig. 2. Screenshots of CubeViz(1.) and SemMap(2.)

CubeViz can be virtually divided into two layers: (a) the back-end interacts with the datasets directly via SPARQL queries and forwards query results in the JSON format to (b) the front-end, which processes the JSON data and display it as elements of the facetted browsing cmponent or as a chart. The CubeViz back-end takes advantage of the OntoWiki [9] framework by using its API for interactions with datasets. The CubeViz front-end - facetted browser and chart visualization components - is written in JavaScript and CSS.

An exemplarily selected output of CubeViz visualizing the Digital Agenda Scoreboard dataset is depicted on Figure 2.

SemMap: Visualizing geo-related data. SemMap[4], depicted in Figure 2, is a facetted browser for spatial RDF data. Its key features are to enable users to view, navigate and filter *indirectly* geo-related data in flexible way on a map. SemMap is written almost entirely in JavaScript and runs in browsers from where it directly interacts with SPARQL endpoints. This means, that virtually any spatial data set in the LOD cloud for which there exists a SPARQL endpoint can be visualized. Thereby our development efforts aim on the one hand at making the system capable of incorporating the re-use of vocabularies, by means of automatic configuration for displaying markers or polygons on a map. At the moment SemMap support the WGS84[5] and the LinkedGeoData[6] vocabulary. On the other hand the goal is to make it easy to add support for new or alternative behaviour via plugins.

[4] http://aksw.org/Projects/SemMap

[5] http://www.w3.org/2003/01/geo/wgs84_pos#

[6] http://linkedgeodata.org

4 Case Studies

In this section we sketch two use cases that operate with OGD in the Linked Data Web to showcase the applicability of the publishing approach (see section 2). On the one hand we present the *Digital Agenda Scoreboard of the European Commission* in subsection 4.1 which was published in 2011 visualizing statistics about European countries from different sources. On the other hand we give an overview about the RDF representation and its visualizations of the *Financial Transparency System of the European Commission* in subsection 4.2

4.1 Digital Agenda Scoreboard

As an outcome of requirements from *Digital Agenda for Europe* (DAE) to publish an annual scoreboard for monitoring and benchmarking main developments of information society in European countries, the *Digital Agenda Scoreboard of the European Commission* (DAS) [14] was developed. The *Directorate Generale* (DG) *Information Society*[7] collected statistical data from different sources, such as Eurostats[8], and combined them into one dataset represented as a spreadsheet. This dataset was converted and stored in a relational database containing the tables (a) indicators, (b) sources, (c) observations. To support the requirements (monitoring and benchmarking) the following four different scenarios were defined:

1. Compare countries by means of one selected indicator and year
2. Evolution of an indicator over time
3. Comparison of two selected indicators regarding a specific year
4. Country profile containing all indicators for a selected country and year

In preparation for the visualization, necessary data items are queried by using SQL, converted to JSON and sent to the client web interface where they are presented in specific chart types (such as bar chart, spline chart and scatter plot). With regard to the specific scenario, users are able to filter the data with respective filter widgets, such as country, year and indicator selectors as depicted in Figure 3.

In addition to the DAS HTML/JS output of filtered observations, users are able to download them as raw data files represented as CSV or RDF, which enables re-usage with other tools such as CubeViz mentioned in section 3. Besides that, the whole dataset is published using an OntoWiki-SPARQL endpoint[9] and is provided for download in different formats[10]. The representation of the DAS data model as RDF was created by using the RDF vocabulary *DataCube*[11],

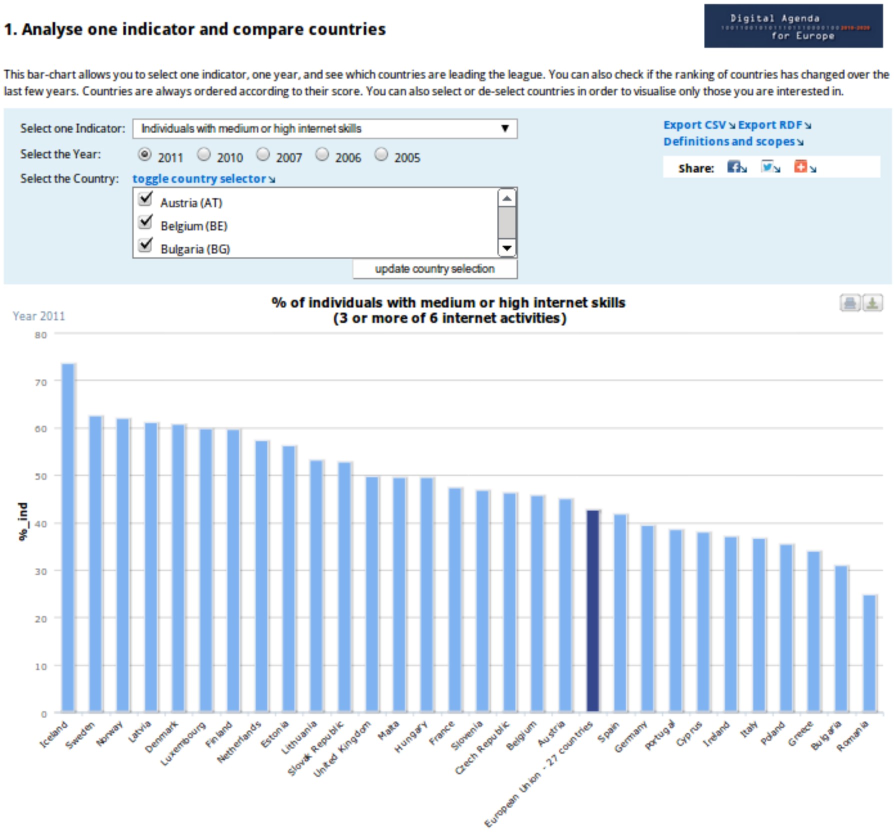
[7] `http://ec.europa.eu/information_society/`

[8] `http://epp.eurostat.ec.europa.eu/portal/page/portal/information_society/introduction`

[9] `http://data.lod2.eu/scoreboard/`

[10] `http://scoreboard.lod2.eu/index.php?page=export`

[11] Prefix:Namespace qb:`http://purl.org/linked-data/cube#`



1. Analyse one indicator and compare countries

This bar-chart allows you to select one indicator, one year, and see which countries are leading the league. You can also check if the ranking of countries has changed over the last few years. Countries are always ordered according to their score. You can also select or de-select countries in order to visualise only those you are interested in.

Select one Indicator: Individuals with medium or high internet skills ▼

Select the Year: ● 2011 ○ 2010 ○ 2007 ○ 2006 ○ 2005

Select the Country: toggle country selector

☑ Austria (AT)
☑ Belgium (BE)
☑ Bulgaria (BG)

Fig. 3. Scoreboard visualization of scenario one.

which is made especially for representing statistics containing multiple dimensions. The resulting DAS-DataCube contained three component properties of type `qb:DimensionProperty` representing the country, the year and the indicator context relation of observations. The fourth component property of type `qb:MeasureProperty` is used to encode the measurement of each observation.

As part of the publishing process of the RDF model, meta data about the model were added, such as the list of responsibilities, versioning and licensing information (in this case CC-BY-3.0[12]). We further enriched the dataset with spatial and temporal information (countries and years) by interlinking with resources from the Linked Data Web, such as DBpedia [2] and a Eurostats snapshot, using SILK [18]. To conclude the publishing process, the dataset was disseminated over different channels such as the creation of a CKAN entry[13].

[12] `http://creativecommons.org/licenses/by/3.0/`

[13] `http://thedatahub.org/dataset/scoreboard`

4.2 Financial Transparency of European Commission

The Financial Transparency System (FTS) of the European Commission[14] contains information about commitments granted and supported by the European Commission (EC) of the last years since 2007. The official datasets are published as the formats CSV, XML and XLS at the EC website. Additionally, for the XML format there exists a publicly available XML schema definition (XSD). In the following we sketch the creation of RDF version of the FTS dataset according to outline given in section 2. A more thorough analysis of the dataset is given in [13].

The XML essentially makes use of two complex types (i.e. elements that are composed of other elements rather than primitive values), namely *Commitment*, and *Beneficiary*, whereas the latter are always children of the former. The commitment element represents information about a monetary amount granted to one or more beneficiaries. This information includes several fields, among them the *position key*, which is a unique id, the (total) *amount*, the *year*, and the *responsible department.* Beneficiaries may be organizations, governments and private persons. Their data consists of the address, the share they received from the total amount and the co-financing rate. The granularity of the address may vary (e.g. city level vs street address) and there are a few cases where the detail amount is not given.

Most OWL properties were derived from the XML element names by first converting them to lower camel case and then prefixing them with *ftso:*[15]. One of the exceptions is e.g. *amount* which we renamed to *totalSum* in order to establish backward compatibility with an early version of the rdfized FTS dataset. The difficult part in the transformation process is to define the ranges of the properties: We are left with the choice of whether to convert the primitive values in the XML to RDF literals or to resources. In the latter case, the resource would become an instance of an OWL class with the name of the property written in upper camel case, and the original value would become a label of that resource. For example, in our RDF, a commitment's *budget line* is modelled as a resource of type *ftso:BudgetLine*, which is referened by the commitment via the *ftso:budgetLine* property. An excerpt of the vocabulary is depicted in Figure 4.

For converting the dataset to RDF we developed a Java program. This choice is motivated by two reasons: On the one hand, there exists the *maven-jaxb2-plugin*[16] which automatically creates Java classes from XSD files. On the other hand, Jena[17] is a powerful RDF library, which provides interfaces and classes for manipulating sets of triples and serializing them in all conventional formats. The actual conversion work therefore only consisted of looping through the commitments and their beneficiaries and emitting the corresponding triples. Of course, approaches based on other technologies and frameworks, such as XSLT (kregx-

[14] http://ec.europa.eu/beneficiaries/fts/index_en.htm

[15] `http://fts.publicdata.eu/ec/ontology/`

[16] `http://java.net/projects/maven-jaxb2-plugin/pages/Home`

[17] `http://jena.apache.org`

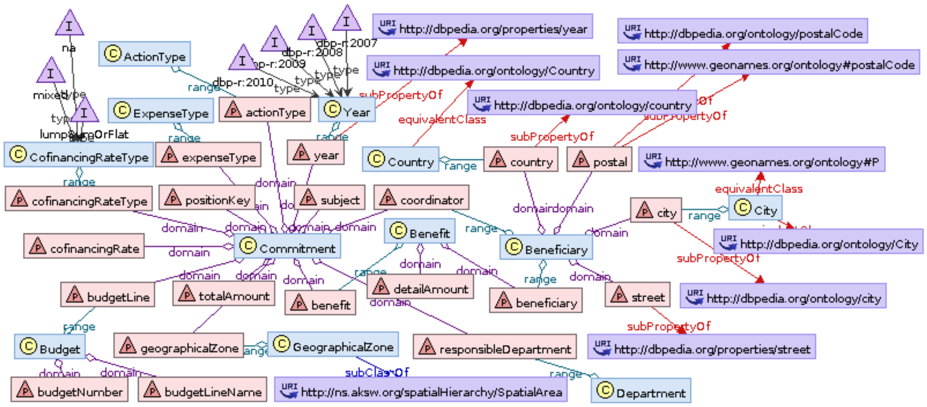


Fig. 4. Overview of the Financial Transparency System vocabulary.

tor[18]) or for instance Scala[19] may work equally well, as one of the greatest time determining factors in this process is the experience of the developers and their confidence with the available tools. Interlinking was performed between the cities mentioned in the beneficiaries' addresses and the cities in LinkedGeoData.

Following our vision of establishing a choreography layer, the resulting dataset was registered at the datahub [20]. The geo-links were published as a separate dataset (see TheDataHub entry). The reason for this is, that the *Directorate-General for Budget*[21] is responsible for the quality assurance of the their data, whereas we maintain the geo-links. Also, since we relied on automatic geocoding with Nominatim[22] and Google Maps API[23], some links may be of low quality or even wrong.

These two datasets together now enable users to browse the data on a map. Figure 5 depicts SemMap showing information about one of the more than 3000 commitments related to beneficiaries located in Luxembourg. Note that commitments themselves are not geo-tagged, however they are related to beneficiaries whose addresses were geocoded.

5 Related Work

Related work can be divided in the two categories: conceptual guidelines for publishing LOGD and technical approaches for managing the lifecycle of OGD.

Conceptual guidelines are related to eGovernment design issues. The most important document in this group is ”Putting Government Data Online” [4].

[18] `http://trac.kwarc.info/krextor/`

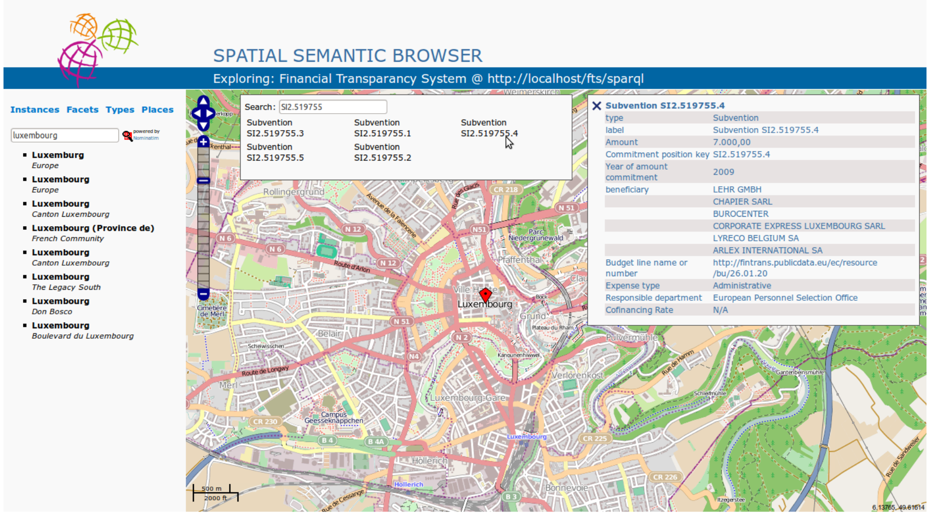
[19] `http://www.scala-lang.org/`

[20] /urlhttp://thedatahub.org/dataset/beneficiaries-of-the-european-commission

[21] `http://ec.europa.eu/dgs/budget/index_en.htm`

[22] `http://open.mapquestapi.com/nominatim/v1/search`

[23] `https://developers.google.com/maps`


SPATIAL SEMANTIC BROWSER
Exploring: Financial Transparancy System @ http://localhost/fts/sparql
Instances Facets Types Places
powered by Nominatim

luxembourg

- **Luxemburg** *Europe*
- **Luxembourg** *Europe*
- **Luxembourg** *Canton Luxembourg*
- **Luxembourg (Province de)** *French Community*
- **Luxembourg** *Canton Luxembourg*
- **Luxembourg** *The Legacy South*
- **Luxembourg** *Don Bosco*
- **Luxembourg** *Boulevard du Luxembourg*

Search: SI2.519755

Subvention SI2.519755.3 | Subvention SI2.519755.1 | Subvention SI2.519755.4
Subvention SI2.519755.5 | Subvention SI2.519755.2

Subvention SI2.519755.4

type	Subvention
label	Subvention SI2.519755.4
Amount	7.000,00
Commitment position key	SI2.519755.4
Year of amount commitment	2009
beneficiary	LEHR GMBH
	CHAPIER SARL
	BUROCENTER
	CORPORATE EXPRESS LUXEMBOURG SARL
	LYRECO BELGIUM SA
	ARLEX INTERNATIONAL SA
Budget line name or number	http://fintrans.publicdata.eu/ec/resource/bu/26.01.20
Expense type	Administrative
Responsible department	European Personnel Selection Office
Cofinancing Rate	N/A

Fig. 5. SemMap showing details about a subvention with at least one its beneficiaries located in Luxembourg.

Also, the *W3C eGovernment Interest Group*[24] describes principles behind publishing Open Goverment Data [1]. [10] looks from a theoretical viewpoint on the aspects of LOGD publishing. It considers all the steps from data acquisition to publishing, licensing and dissemination.

Technical approaches propose a particular framework for the LOGD publishing. In [5] authors evaluate a framework for collecting, cleaning and converting data to RDF. The publication step is mentioned, but not described in the detail. The main idea of the paper is that the process of collecting, cleaning and converting should be delegated to the consumer, thus making it a *self-service* approach. The idea of self-service approach is refined and extended in [12]. However, the self-service approach does not highlight the exploitation step. In [8] the authors show how to build a LOGD portal, "where stakeholders of different sizes and roles find, manage, archive, publish, reuse, integrate, mash-up, and consume open government data in connection with online tools, services and societies." It includes all necessary stages for LOGD publishing. However, extensibility of the exploitation layer is limited since developers can only apply existing visualizations for the LOGD. In [11] the authors provide a classification scheme for OGD, that includes a technological approach for making data available on the Web and an organizational approach for data provision. The technological approach is aimed at the development of government portals in the first place and does not take in account already existing and maintained information resources.

[24] http://www.w3.org/egov/wiki/Main_Page

6 Conclusion and Future Work

In this paper we showed how LOGD publication process can be applied to the raw OGD on two examples: Digital Agenda Scoreboard and Financial Transparency System of European Commission datasets. Up to now, there is no general approach for LOD dataset visualization. We outlined a possible architecture featuring a choreography layer in order to mediate between LOD datasets and the visualization ecosystem and demonstrated its applicability with two case studies. Additionally, we highlighted the current deficiency of the exploitation step of the LOGD lifecycle.

Despite the fact that publishing pipelines already exist, LOGD as a part of the LOD cloud is still in its infancy. To advance the progress in developing Linked Open Government Data, the pipelines as well as the applications require tighter integration to facilitate their use in everyday practice. Therefore, in the future we plan to create services that tie together existing dataset catalogs with visualizations, such as CubeViz and SemMap, in order to realize the visualization ecosystem.

Acknowledgments

We would like to thank our colleagues from AKSW research group for their helpful comments and inspiring discussions related to LOGD applications. This work was partially supported by a grant from the European Union's 7th Framework Programme provided for the project LOD2 (GA no. 257943).

References

1. J. M. Alonso et al. Improving Access to Government through Better Use of the Web. Technical report, World Wide Web Consortium (W3C), 2009.
2. S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives. DBpedia: A nucleus for a web of open data. In *Proceedings of the 6th International Semantic Web Conference (ISWC)*, volume 4825 of *Lecture Notes in Computer Science*, pages 722–735. Springer, 2008.
3. T. Berners-Lee. Linked Data. Technical report, World Wide Web Consortium (W3C), 2006.
4. T. Berners-Lee. Putting Government Data Online. W3C Design Issue, 2009. http://www.w3.org/DesignIssues/GovData.html.
5. R. Cyganiak, F. Maali, and V. Peristeras. Self-service linked government data with dcat and gridworks. In *Proceedings of the 6th International Conference on Semantic Systems*, I-SEMANTICS '10, pages 37:1–37:3, New York, NY, USA, 2010. ACM.
6. R. Cyganiak, D. Reynolds, and J. Tennison. The RDF Data Cube Vocabulary, 2010. http://www.w3.org/TR/vocab-data-cube/.
7. J. Demter, S. Auer, M. Martin, and J. Lehmann. LODStats – An Extensible Framework for High-performance Dataset Analytics. In *EKAW*, 2012.
8. L. Ding, T. Lebo, J. S. Erickson, D. DiFranzo, A. Graves, G. T. Williams, X. Li, J. Michaelis, J. Zheng, Z. Shangguan, J. Flores, D. L. M. J, and J. A. Hendler. Twc logd: A portal for linked open government data ecosystems. *Web Semantics: Science, Services and Agents on the World Wide Web*, 9(3), 2011.

9. N. Heino, S. Dietzold, M. Martin, and S. Auer. Developing Semantic Web Applications with the OntoWiki Framework. In *Networked Knowledge - Networked Media*, volume 221 of *Studies in Computational Intelligence*, pages 61–77. Springer, Berlin / Heidelberg, 2009.
10. B. Hyland and D. Wood. The Joy of Data - A Cookbook for Publishing Linked Government Data on the Web Linking Government Data. In D. Wood, editor, *Linking Government Data*, chapter 1, pages 3–26. Springer New York, New York, NY, 2011.
11. E. Kalampokis, E. Tambouris, and K. Tarabanis. A classification scheme for open government data: towards linking decentralised data. *Int. J. Web Eng. Technol.*, 6(3):266–285, June 2011.
12. F. Maali, R. Cyganiak, and V. Peristeras. A publishing pipeline for linked government data. In E. Simperl, P. Cimiano, A. Polleres, O. Corcho, and V. Presutti, editors, *ESWC*, volume 7295 of *Lecture Notes in Computer Science*, pages 778–792. Springer, 2012.
13. M. Martin, C. Stadler, P. Frischmuth, and J. Lehmann. Increasing the Financial Transparency of European Commission Project Funding. *Semantic Web Jorunal*, Special Issue on Linked Dataset descriptions, June 2012. Under Review.
14. M. Martin, B. van Nuffelen, S. Abruzzini, and S. Auer. The Digital Agenda Scoreboard: an Statistical Anatomy of Europe's Way into the Information Age. *Semantic Web Jorunal*, June 2012. Under Review.
15. A.-C. Ngonga Ngomo and S. Auer. Limes - a time-efficient approach for large-scale link discovery on the web of data. In *Proceedings of IJCAI*, 2011.
16. D. Robinson, H. Yu, W. P. Zeller, E. W. Felten, and E. W. Felten. Government data and the invisible hand. 2008.
17. B. Villazón-Terrazas, L. Vilches-Blázquez, O. Corcho, and A. Gómez-Pérez. Methodological guidelines for publishing government linked data linking government data. In D. Wood, editor, *Linking Government Data*, chapter 2, pages 27–49. Springer New York, New York, NY, 2011.
18. J. Volz, C. Bizer, M. Gaedke, and G. Kobilarov. Discovering and maintaining links on the web of data. In *ISWC*, 2009.