Increasing the Financial Transparency of European Commission Project Funding

Michael Martin [a,*], Claus Stadler [a], Philipp Frischmuth [a], and Jens Lehmann [a]

[a] *University of Leipzig, Institute of Computer Science, AKSW, Postfach 100920, D-04009 Leipzig, Germany*
E-mail: {lastname}@informatik.uni-leipzig.de

Abstract. The Financial Transparency System (FTS) of the European Commission contains information about grants for European Union projects starting from 2007. It allows users to get an overview on EU funding, including information on beneficiaries as well as the amount and type of expenditure and information on the responsible EU department. The original dataset is freely available on the European Commission website, where users can query the data using an HTML form and download it in CSV and most recently XML format. In this article, we describe the transformation of this data to RDF and its interlinking with other datasets. We show that this allows interesting queries over the data, which were very difficult without this conversion. The main benefit of the dataset is an increased financial transparency of EU project funding. The RDF version of the FTS dataset will become part of the EU Open Data Portal and eventually be hosted and maintained by the European Union itself.

Keywords: Linked Open Data, Financial Transparency System, European Commission, OntoWiki, LIMES

1. Introduction

The *Financial Transparency System of the European Commission* (FTS) contains information about commitments granted and supported by the European Commission of the last years since 2007. Users are able to obtain insights about who receives funds and where the beneficiaries are located, the amount and type of expenditure as well as the information about responsible EU departments and further details on the project budget. Originally, this dataset is published on the European Commission website, where users are able to select data by using an HTML form and to download the datasets as CSV and XML separately for each year. In order to publish this data in a re-usable way, we analysed and transformed the data into RDF which is being sketched in Section 2. We also inter-

linked the data to increase its added value. Another important aspect is sustainability: We are collaborating with the European Commission and it is planned to integrate the dataset in the upcoming EU Open Data Portal. The RDF version of the dataset will eventually be hosted by the EU, i.e. the actual dataset owners, themselves, which ensures a long term availability of the data. The resulting vocabulary and a set of statistics about the dataset is also described in Section 2. The publication of the dataset is described in Section 3. Section 4 shows some usage examples for the dataset and Section 5 covers related work. We conclude the article in Section 6.

2. Triplification of the Raw Data

In this section, we showcase how the original dataset was triplified and describe a summary of the analysis of the dataset. Based on the analysis, we were able to create a set of requirements and rules containing def-

---

*Corresponding author. E-mail: martin@informatik.uni-leipzig.de.

initions for the T-Box (classes, properties) as well as guidelines for the A-Box creation (entity recognition and URI generation).

2.1. Analysis of the Original Dataset

The original datasets are published by the European Commission. On their website[1], the data can be filtered by using several attributes, including: (a) *name of the beneficiary*, (b) *year of the grant*, (c) *country of beneficiaries*, (d) *geographical zone*, *postal code*, *amount of funding*, *action type*, *expense type* and *budget line name or number*. The correct usage of the filter form leads to a result list containing information about beneficiaries.

It is also possible to download the dataset(s) in XLS, CSV and XML format[2]. At the moment, interested users are able to download five different datasets containing information for each year the described grants have started (from 2007 to 2011).

All representation formats of these datasets contain data items as listed above and — in addition — the *subject of grant or contact*, the *co-financing rate*, the *commitment position key* and the *responsible EU department*. Project proposals contain a list of all participating beneficiaries, the respective monetary distribution as well as the coordinator of the project.

To prepare the vocabulary development, as well as the respective data conversion and linking processes, we specified how the data items in the raw datasets have to be converted. We analysed the datasets in order to detect suitable RDF resource candidates, and created rules how the respective resource URIs have to be generated. Subsequently, we specified object relations (such as $1:1$ or $1:n$ relations) to connect these resources. Finally, we determined the language tags or datatypes of the remaining data items, which are encoded as literals.

2.2. The FTS vocabulary

As a result of the initial analysis and as preparation for the data conversion, we created the schema depicted in Figure 1. All schematic elements, which could not be reused from other vocabularies, were encoded using the FTS ontology namespace[3]. The core element of the schema is the class *fts:Commitment*, which is used to type RDF resources describing EU funded projects. As illustrated in Figure 1, a set of predicates is available to express facts using literals about the respective commitment such as *fts:subject* (used to encode the subject of the grant), *fts:commitmentPositionKey* (which is the commitment unique identifier) and *fts:totalAmount* (used to encode the amount), which is the sum of the funding of all participating beneficiaries. Temporal dependencies of commitments are represented by using a year relation which refers to the respective DBpedia [10] resource.

For co-financed EU funded projects, the property *fts:cofinancingRateType* references instances of type *fts:CofinancingRateType* that represent *lump sum / flat rate*, *mixed financing* or *explicit*. If *explicit* is set as *fts:cofinancingRateType*, the corresponding percentage value (encoded as *xsd:decimal*) is attached to the respective commitment resource using the property *fts:cofinancingRate*. If a project is not co-financed, the referred resource indicates that co-financing does not apply to the commitment.

Every commitment refers to at least one participating beneficiary, which is represented by resources of type *fts:Beneficiary*, which can be reached via the property chain *fts:benefit* and *fts:beneficiary* from the corresponding commitment. Typically for research projects with more than one beneficiary, one of the beneficiaries is also coordinating the project. The coordinator is usually responsible for redistributing the funds among the participants. We encoded this information by using the object relation *fts:coordinator* between the commitment resource and the respective beneficiary.

Geographical information of beneficiaries is encoded by using the according properties for street, city, postal code and countries. All of these properties are defined as sub properties of the respective DBpedia relations which enables on the one hand vocabulary specific domain and range relations and on the other hand a close connection to the well known DBpedia vocabulary.

For development aid grants paid by EuropeAid[4], the location of the action is represented by using the object relation *fts:geoZone* between a commitment resource and the specific geographical zone resource. The class *fts:GeographicalZone*, is used to type such special resources. It is a subclass of *shv:SpatialArea* [5] and can
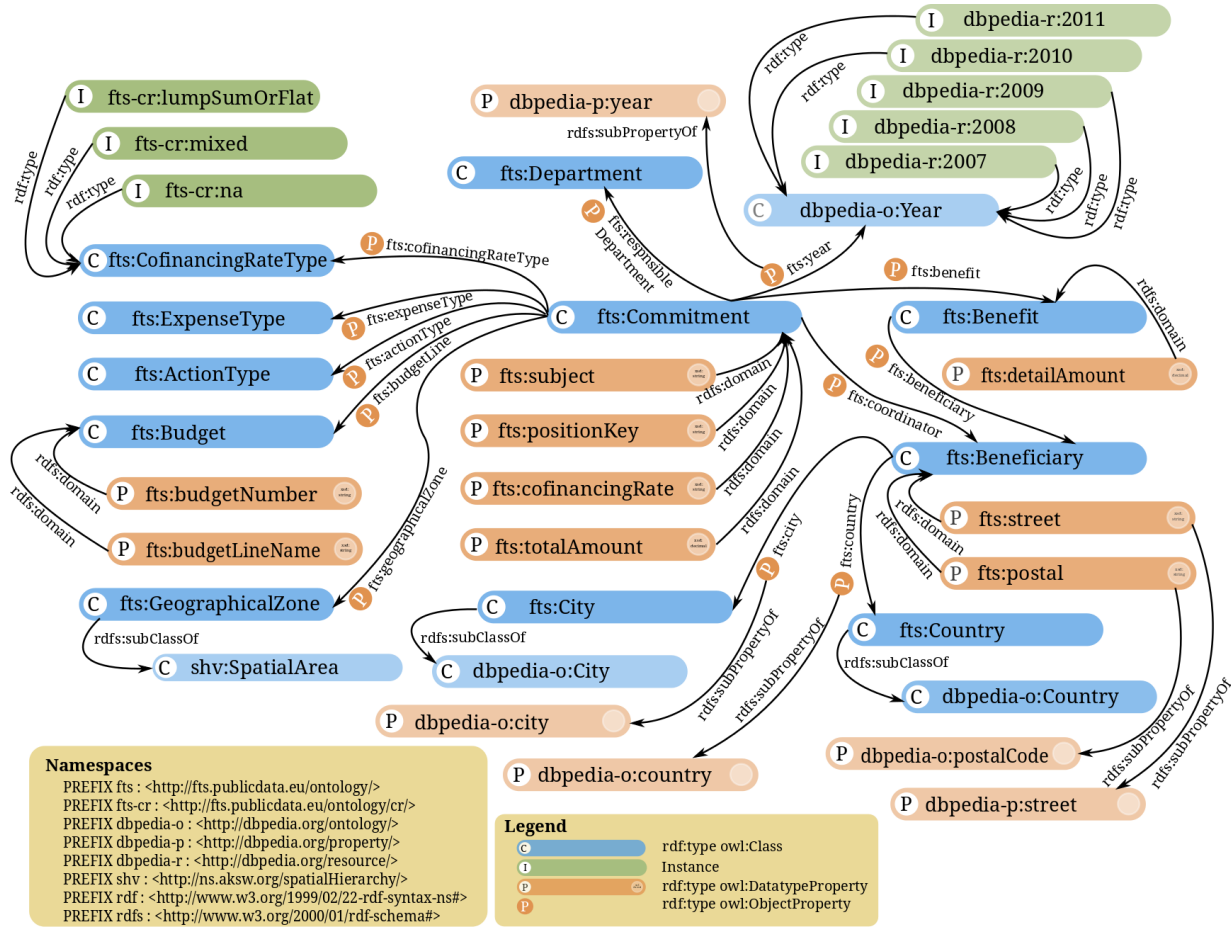

Fig. 1. Overview about the FTS vocabulary

be used in a generic fashion, e.g. whole continents as well as small areas can have this type.

Due to the fact that the European Commission is structuring their budget around titles, chapters, articles and posts, a budget line name or number is encoded in the dataset encoded with its corresponding relations.

Grants and procurements are under administration of specific departments of the European Commission called *Directorates General* (DG) in charge of implementing policies & EU funded programmes. Specific responsible departments and action types of funded projects are encoded with the object relations *fts:responsibleDepartment* and *fts:actionType*. A further object relation encodes the expense type of a project funding (*fts:expenseType*). Most of the project fundings published are for operational expenditures. Only the datasets from 2009 onwards contain commitments for administrative expenses. Resources refer-

enced by these relations can be used to group fundings by responsible European Commission departments as well as types of action and expenditures.

```
@prefix rdf:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .    1
@prefix xsd:   <http://www.w3.org/2001/XMLSchema#> .              2
@prefix rdfs:     <http://www.w3.org/2000/01/rdf-schema#> .       3
@prefix fts:      <http://fts.publicdata.eu/ontology/> .          4
@prefix fts-cm:   <http://fts.publicdata.eu/resource/cm/> .       5
@prefix fts-et:   <http://fts.publicdata.eu/resource/et/> .       6
@prefix fts-bt:   <http://fts.publicdata.eu/resource/bt/> .       7
@prefix fts-at:   <http://fts.publicdata.eu/resource/at/> .       8
@prefix fts-by:   <http://fts.publicdata.eu/resource/by/> .       9
@prefix fts-bl:   <http://fts.publicdata.eu/resource/bl/> .       10
@prefix fts-cr:   <http://fts.publicdata.eu/resource/cr/> .       11
@prefix fts-de:   <http://fts.publicdata.eu/resource/de/> .       12
@prefix dbpedia-r: <http://dbpedia.org/resource/> .               13
                                                                  14
fts-cm:SI2.ACPROCE040592100.1                                     15
  rdf:type         fts:Commitment ;                               16
  rdfs:label       "Commitment SI2.ACPROCE040592100.1" ;          17
  fts:expenseType  fts-et:administrative ;                        18
  fts:benefit      fts-bt:SI2.ACPROCE040592100.1-IMPRIMERIE       19
      +CENTRALE+SA-d31930f949619daf93b55a9d0ee19e26 ;
  fts:year         dbpedia-r:2011 ;                               20
  fts:actionType   fts-at:Research%3A+Framework++programme+       21
      n%C2%B07 ;
  fts:totalAmount  "600.34"^^xsd:decimal ;                        22
```

```
fts:coordinator    fts-by:IMPRIMERIE+CENTRALE+SA-    23
    d31930f949619daf93b55a9d0ee19e26 ;
fts:budgetLine     fts-bl:08.01.05.03 ;              24
fts:subject        "PRODUCTION OF INFORMATION CARDS FORMAT  25
    A6 INNOVATION UNION (COMPL&#xc9;MENTAIRE DU CONTRAT
    30-CE-0404470/00-13)" ;
fts:responsibleDepartment fts-de:Directorate-General+for+  26
    Research+and+Innovation ;
fts:cofinancingRateType    fts-cr:na .               27
```

Listing 1: Exemplified selected FTS resource

Listing 1 shows a simple example resource in FTS. The resource is an instance of the `Commitment` class, i.e. an EU grant. The `benefit` property points to a resource, which contains all beneficiaries as well information about them such as their country and address. Further information is available, such as the coordinator of the funded project, the responsible EU department, the amount of funding, etc.

2.3. Transformation workflow

The FTS datasets are published in Excel, CSV and XML formats. In the following, we describe (a) our RDF transformation using a lightweight Maven-2-based Java program[6] that operates on the XML, (b) design decisions about selected properties and the URI design and (c) give a statistical summary of the generated RDF.

For the XML format, there exists an XML schema definition (XSD) from which a corresponding Java class model is automatically created during compile-time using the *maven-jaxb2-plugin*[7]. The generated classes are *Fts*, *Commitment* and *Beneficiary*. Fts holds a list of commitments, and each commitment holds a list of beneficiaries. With *JAXB*, each FTS XML data set file can be directly unmarshalled into an Fts Java object. We had to address the following two issues: First, the English 2010 dataset was not well formed due to encoding problems, which we repaired using *tidy*[8]. Second, we had to update the XSD with some missing declarations of tags that were used in the XML. Otherwise, some data would not have been available in the Fts object.

The generation of RDF from an Fts object is then accomplished by emitting triples during the iteration of the commitments and beneficiaries. In the following, we present the most important transformation rules:

---

[6] https://github.com/AKSW/FTS-EC-2-RDF
[7] http://java.net/projects/
maven-jaxb2-plugin/pages/Home
[8] http://tidy.sourceforge.net/, also available as a Debian package.

Beneficiaries may have lists of labels, whereas we refer to the first one as the *primary name*. We generate *rdfs:label* and *skos:prefLabel* triples for the primary name, and *skos:altLabel* triples for all other names. We expect this to be a useful approach, as data consumers do not necessarily have to deal with multiple labels.

For beneficiaries that participate alone in a commitment, we assume its *detail amount* to be the total amount of the commitment. The XML source data leaves the detail_amount field empty in this case.

The expense type is defined per beneficiary in the XML, whereas in the CSV it is defined per commitment. We chose the per-commitment approach for our ontology because all beneficiaries always had the same expense type.

The URI design of the three main classes is as follows: *Commitment* URIs follow the pattern *http://fts.publicdata.eu/resource/cm/{positionKey}*. For beneficiaries we generate URIs based on their primary names and MD5-hashes of their address fields. Lastly, a *Benefit* URI depends on a concatenation of a commitment's position key and a beneficiary's primary name.

Table 1

Summary about the number of FTS resources belonging to particular classes/types.

Year	2007	2008	2009	2010	2011
Benefits	29,014	33,725	87,638	91,456	83,338
Commitments	28,114	32,617	53,905	55,192	51,753
Beneficiaries	10,231	11,791	30,688	30,851	23,737
Cities	3,780	4,241	7,457	7,845	6,892
Countries	157	170	199	189	189
Departments	31	33	46	50	48
Resources	71,780	83,053	180,644	186,330	166,715
Triples	509,590	620,168	1,205,058	1,243,038	1,108,909

Table 1 summarizes the following statistics: The upper part shows the number of instances for selected classes. The lower part lists the number of distinct resources and the total amount of triples per year. The overall amount of statements in FTS (without inter-links) is 4,686,763. In addition, we extracted 22,610 triples from the datasets about the beneficiaries of the European Development Fund (EDF). The EDF expenditures are not part of the EU budget which is the reason why they are published separately. The EDF datasets are hosted at the original FTS site, and are available for the years starting from 2010.

3. Dataset Publishing and Linking

Rather than providing the RDF dataset as a dump for download only, we also published it using the Linked Data principles [3]. All URIs minted during the transformation process are within a dedicated FTS namespace[9]. After publishing FTS data, we added related meta information to the *owl:Ontology* node of the FTS graph such as authors, contributors, the last modification date (current : *October, 10, 2012*) and the license (Creative Commence License *CC-BY-3.0*). To improve the visibility of the FTS dataset, a respective entry on `theDataHub.org`[10], the meta data repository for *Open Data* datasets, was registered.

3.1. Publishing FTS with OntoWiki

We used OntoWiki [2] to publish the FTS dataset. OntoWiki is a data wiki as well as a framework for building knowledge-intensive web applications [9]. The tool builds directly upon the RDF data model and keeps all information in a triple store – specifically OpenLink Virtuoso in the case of the FTS dataset. Stored data can be accessed by other applications via a Linked Data interface as well as a SPARQL endpoint. Furthermore, data can be serialized in a variety of RDF formats (e.g. RDF/XML, Turtle as well as RDF/JSON) both on resource level as well as on graph level.

Human consumption.   One of the primary goals of OntoWiki is to visualize structured information for humans and to provide easy browsing mechanisms suitable for users who are not familiar with RDF. To achieve this goal, the OntoWiki user interface hides much of the complexity of the RDF data model wherever feasible. Multiple RDF resources are usually visualized as a list, e.g. as a result of full-text search, faceted search or user generated SPARQL queries, or, if a single resource is selected, as an adequate view visualizing all locally available information about the resource in a user-friendly format.

For the purpose of FTS, we deployed the current version of OntoWiki[11] and made some FTS specific optimizations. Due to the fact that the FTS OntoWiki instance should not be used as a data acquisition Wiki, but only as a consumption interface, we disabled all authentication and editing interfaces. To present details about the dataset such as content descriptions, original sources, examples and contact information, a static landing page[12] was created.

3.2. Linking FTS resources using LIMES

In addition to the generation of RDF from the original FTS data, we geocoded the beneficiaries on the country, city and the address level, enabling visualizations of the data on maps. For countries and cities we created LIMES [11] link specifications which match countries and cities in FTS with those in LinkedGeoData [14] (LGD) based on the tri-gram distance on the country and city names[13]. A few cases where FTS-cities linked to multiple LGD-cities were manually corrected. For the address level, we generated 48,860 unique address strings based on a beneficiary's country, city and street attribute. For about 27,000 address strings we were able to obtain coordinates using MapQuest's Nominatim API[14]. Future work will be to use additional geocoding services, such as GeoNames[15], Google Maps API[16] and Yahoo! Placefinder[17] in order to increase the coverage and possibly the quality of data by cross-checking. Due to the large amount of data, we envision a crowd-sourcing approach for cleansing the data.

4. Dataset Benefits and Usage Scenarios

The publication of the FTS RDF representation with OntoWiki - especially its integrated SPARQL endpoint - supports different use cases than the original FTS website. For instance, it offers multifaceted views on the data whereas previously data had to be queried by using a static HTML form. It also decreases the barrier for developers to reuse that data and to create own applications and mash-ups. As described in Section 3, the FTS graph contains references to other knowledge bases such as DBpedia. These interlinks offer the potential to create queries using FTS data in combination with its referenced information. As an example, the query given in Listing 2 presents such a view.

---

[9]`http://fts.publicdata.eu/`
[10]`http://thedatahub.org/dataset/`
`beneficiaries-of-the-european-commission`
[11]`https://github.com/AKSW/OntoWiki`

[12]`http://fts.publicdata.eu/`
[13]Both datasets have relations between cities and corr. countries
[14]`http://open.mapquestapi.com/nominatim/v1/`
`search`
[15]`http://www.geonames.org/`
[16]`https://developers.google.com/maps`
[17]`http://developer.yahoo.com`

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-       1
    syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema   2
    #>
PREFIX fts: <http://fts.publicdata.eu/ontology/>      3
PREFIX dbpedia-o: <http://dbpedia.org/ontology/>      4
PREFIX dbpedia-p: <http://dbpedia.org/property/>      5
PREFIX dbpedia-r: <http://dbpedia.org/resource/>      6
                                                      7
SELECT * { {                                           8
  SELECT ?ftsyear ?ftscountry (SUM(?amount) AS ?       9
      funding) {
    ?com rdf:type fts:Commitment .                    10
    ?com fts:year ?year .                             11
    ?year rdfs:label ?ftsyear .                       12
    ?com fts:benefit ?benefit .                       13
    ?benefit fts:detailAmount ?amount .               14
    ?benefit fts:beneficiary ?beneficiary .           15
    ?beneficiary fts:country ?country .               16
    ?country owl:sameAs ?ftscountry .                 17
} } {                                                 18
  SELECT ?dbpcountry ?gdpyear ?gdpnominal {            19
    ?dbpcountry rdf:type dbpedia-o:Country .          20
    ?dbpcountry dbpedia-p:gdpNominal ?gdpnominal .    21
    ?dbpcountry dbpedia-p:gdpNominalYear ?gdpyear .   22
} }                                                   23
  FILTER ((?ftsyear = str(?gdpyear)) &&               24
         (?ftscountry = ?dbpcountry))                 25
                                                      26
}                                                     27
LIMIT 10
```

Listing 2: SPARQL query on FTS and DBpedia

It creates a juxtaposition of project fundings, taken as a SUM from all FTS benefits in relation to year and country, and the corresponding gross domestic products (GDP) from DBpedia[18]. It can be used to analyse the correlation of project funding with the GDP of countries, e.g. for policy decisions. Assembling this data without the RDF FTS version would require substantial work.

5. Related Work

On the technical side, triplification approaches for CSV and XML data, are related. In addition, we also describe Linked Data in E-Government efforts as the general domain of FTS.

RDF Conversion of Datasets   Currently, most work in the area of RDF conversion from legacy data focuses on generating RDF from relational database content. There is a wide range of approaches developed in this regard ranging from lightweight solutions such as *Triplify* [1] over standalone solutions such as *D2R* [5] up to integrated tools such as *Virtuoso RDF Views* [12]. Under the auspices of the W3C, the *RDB2RDF working group* has standardized the *R2RML* mapping lan-

guage for the mapping and transformation of relational data to RDF. FTS data is available in CSV and RDF format. For CSV files, which were previously (before 2012) used as source format for FTS before switching to XML, there are a number of conversion tools. Stats2RDF[19] [15] is an approach targeted mainly at statistical data. XLWrap[20] allows to directly create SPARQL endpoints using CSV (and CSV based formats) as input. csv2rdf4lod[21] provides another way to convert CSV to RDF with a focus on provenance and reasoning. Finally, the RDF extension of Google Refine[22] provides RDF conversion capabilities based on the popular Google Refine tool.

For converting XML files to RDF, which is our current approach for conversion, most methods are based on Extensible Stylesheet Language Transformations (XSLT). For instance, GRDDL is a W3C recommendation for extracting RDF data from XML documents. Krextor[23] is a tool providing a quick way of converting XML to various RDF syntaxes.

The original publishers of FTS provided an XSD as specification of the XML elements, which enables us to automatically create a Java class model using JAXB as described in Section 2.3. Based on our existing specification, we were able to easily create a conversion algorithm that offers the necessary flexibility as well as the extensibility for upcoming dataset versions. Since we needed additional logic to convert the data correctly (e.g., to extract and combine entities from multivalued columns with special rules), we developed this adapted approach for FTS conversion instead of using the CSV or XML based tools mentioned above.

Linked Data in E-Government   Several governments have started to publish governmental data on the Web as Linked Data. Tim-Berners Lee discussed a set of Design Issues [4] on how to publish governmental information in a re-usable way. One of the first Linked Data providers publishing governmental data is the UK government with `http://data.gov.uk`, hosting information about different governmental sectors of Great Britain including transport, legislation and finance [13]. A further provider of governmental data is `http://data.gov` which is hosting information

---

about the USA. Due to the fact that not all of this information was made available as Linked Data, external groups started to transform and publish the information according the Linked Data principles [6]. Recent research work also aims to facilitate government data ecosystems through specialized portals [7] and distributed dataset catalogs [8].

6. Conclusion

In this work, we described the RDF representation of the European Commission Financial Transparency System. We summarized the analysis of the raw data, described the vocabulary, the transformation process and presented how the data was published and linked to the LOD cloud. Possible use cases and sketched benefits for users of the RDF representation were presented as well. During the last two years we experimented with different RDF conversion approaches of the data, ranging from using CSV and PHP, up to the most recent conversion algorithm presented in Section 2.3 in order to obtain a high quality version of the dataset. Current versions of the vocabulary, the conversion algorithms and the integrated linking approach are designed to flexibly handle upcoming FTS datasets.

7. Acknowledgement

References

[1] S. Auer, S. Dietzold, J. Lehmann, S. Hellmann, and D. Aumueller. Triplify: Light-weight linked data publication from relational databases. In J. Quemada, G. León, Y. S. Maarek, and W. Nejdl, editors, *Proceedings of the 18th International Conference on World Wide Web, WWW 2009, Madrid, Spain, April 20-24, 2009*, pages 621–630. ACM, 2009.

[2] S. Auer, S. Dietzold, J. Lehmann, and T. Riechert. OntoWiki: A Tool for Social, Semantic Collaboration. In N. F. Noy, H. Alani, G. Stumme, P. Mika, Y. Sure, and D. Vrandecic, editors, *Proceedings of the Workshop on Social and Collaborative Construction of Structured Knowledge (CKC 2007) at the 16th International World Wide Web Conference (WWW2007) Banff, Canada, May 8, 2007*, volume 273 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2007.

[3] T. Berners-Lee. Linked Data - Design Issues. web page, W3C Design Issue, 2006. `http://www.w3.org/DesignIssues/LinkedData.html`.

[4] T. Berners-Lee. Putting Government Data online. web page, W3C Design Issue, 2009. `http://www.w3.org/DesignIssues/GovData.html`.

[5] C. Bizer and R. Cyganiak. D2R Server - Publishing Relational Databases on the Semantic Web. Poster at the 5th International Semantic Web Conference (ISWC2006), 2006.

[6] L. Ding, D. DiFranzo, A. Graves, J. Michaelis, X. Li, D. L. McGuinness, and J. A. Hendler. TWC data-gov corpus: incrementally generating linked government data from data.gov. In M. Rappa, P. Jones, J. Freire, and S. Chakrabarti, editors, *Proceedings of the 19th International Conference on World Wide Web, WWW 2010, Raleigh, North Carolina, USA, April 26-30, 2010*, pages 1383–1386. ACM, 2010.

[7] L. Ding, T. Lebo, J. S. Erickson, D. DiFranzo, G. T. Williams, X. Li, J. Michaelis, A. Graves, J. Zheng, Z. Shangguan, J. Flores, D. L. McGuinness, and J. A. Hendler. TWC LOGD: A portal for linked open government data ecosystems. *Journal of Web Semantics*, 9(3):325–333, 2011.

[8] J. S. Erickson, E. Rozell, Y. Shi, J. Zheng, L. Ding, and J. A. Hendler. TWC International Open Government Dataset Catalog. In C. Ghidini, A.-C. N. Ngomo, S. N. Lindstaedt, and T. Pellegrini, editors, *Proceedings the 7th International Conference on Semantic Systems, I-SEMANTICS 2011, Graz, Austria, September 7-9,2011*, pages 227–229. ACM, 2011.

[9] N. Heino, S. Dietzold, M. Martin, and S. Auer. Developing Semantic Web Applications with the OntoWiki Framework. In T. Pellegrini, S. Auer, K. Tochtermann, and S. Schaffert, editors, *Networked Knowledge - Networked Media*, volume 221 of *Studies in Computational Intelligence*, pages 61–77. Springer, Berlin / Heidelberg, 2009.

[10] M. Morsey, J. Lehmann, S. Auer, C. Stadler, and S. Hellmann. DBpedia and the Live Extraction of Structured Data from Wikipedia. *Program: electronic library and information systems*, 46:27, 2012.

[11] A.-C. Ngonga Ngomo and S. Auer. Limes - A Time-Efficient Approach for Large-Scale Link Discovery on the Web of Data. In T. Walsh, editor, *IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Catalonia, Spain, July 16-22, 2011*, pages 2312–2317, 2011.

[12] OpenLink Software. Automated Generation of RDF Views over Relational Data Sources with Virtuoso. web page, 2009. `http://virtuoso.openlinksw.com/dataspace/dav/wiki/Main/VOSSQL2RDF`.

[13] J. Sheridan and J. Tennison. Linking UK Government Data. In C. Bizer, T. Heath, T. Berners-Lee, and M. Hausenblas, editors, *Proceedings of the WWW2010 Workshop on Linked Data on the Web, LDOW 2010, Raleigh, USA, April 27, 2010*, volume 628 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2010.

[14] C. Stadler, J. Lehmann, K. Höffner, and S. Auer. LinkedGeoData: A Core for a Web of Spatial Open Data. *Semantic Web Journal*, 3(4):333–354, 2012.

[15] A. Zaveri, R. Pietrobon, S. Auer, J. Lehmann, M. Martin, and T. Ermilov. ReDD-Observatory: Using the Web of Data for Evaluating the Research-Disease Disparity. In *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence*, 2011.

---

[24] `http://latc-project.eu`
[25] `http://lod2.eu`