Widget-based Exploration of Linked Statistical Data Spaces

Ba-Lam Do, Tuan-Dat Trinh, Peter Wetz, Amin Anjomshoaa, Elmar Kiesling, A Min Tjoa

Institute of Software Technology and Interactive Systems, Vienna University of Technology, Austria

{*ba.do, tuan.trinh, peter.wetz, amin.anjomshoaa, elmar.kiesling, a.tjoa*}*@tuwien.ac.at*

Keywords: Statistical Data, RDF Data Cube Vocabulary, Widget, Mashup, Linked Data

Abstract: Today, public statistical data plays an increasingly important role both in public policy formation and as a facilitator for informed decision-making in the private sector. In line with the increasing adoption of open data policies, the amount of data published by governments and organizations on the web is growing rapidly. To increase the value of such data, the W3C recommends the RDF Data Cube Vocabulary to facilitate the publication of data in a more structured and interlinked manner. Although important first steps toward building a web of statistical Linked Datasets have been made, providing adequate facilities for end users to interactively explore and make use of the published data remains an unresolved challenge. This paper presents a widget-based approach to deal with this issue. In particular, we introduce a mashup platform that allows users lacking advanced skills and knowledge of Semantic Web technologies to interactively analyze datasets through widget compositions and visualizations. Furthermore, we provide mechanisms for the interconnection of datasets to support sophisticated knowledge extraction.

1 INTRODUCTION

In recent years, the number of available open data sources has increased substantially (Brunetti et al., 2013; Hoefler et al., 2014). A considerable share of the data published is statistical data, comprising a wide range of domains including finance, population, transportation, employment, etc. Statisticians, scientists and researchers accumulate these data through observations and experimentation to report overall trends, identify risks and opportunities, and to conduct planning.

Statistical data becomes much more useful when published as Linked Data, which can be consumed and manipulated without proprietary tools. A Linked Data approach allows users to combine data from different sources in order to gain new insights and to obtain higher data quality, completeness, and level of detail. The RDF Data Cube Vocabulary (Cyganiak and Reynolds, 2011) is a W3C standard for the publication of multi-dimensional Linked Data on the web. This vocabulary follows the same principles as SDMX (Statistical Data and Metadata eXchange), an ISO standard for exchanging and processing statistical data. By concretizing the general syntax of the RDF standard for statistical data, this vocabulary enables data providers to publish their data as Linked Data on the web.

A large number of organizations and governments such as the European Commission[1], the United Kingdom Department for Communities and Local Government[2], and the Scottish Government[3] has adopted this vocabulary and use it to publish their data sources via dedicated SPARQL endpoints. This proliferation of available statistical data has created enormous potential for interesting applications, but it has so far resulted only in limited adoption by end users, including developers and knowledge workers. These users need appropriate tools to analyze, combine, remix, visualize and make sense of the data. At present, however, the means to obtain access to such data are limited to three options.

First, users may write SPARQL queries directly. This is a powerful information retrieval approach that facilitates the extraction of a variety of information. However, users are exposed to raw data output, which is not necessarily easy to comprehend and may be of limited use for inexperienced users aiming to deduce insights from statistic data. These users also cannot be expected to learn the SPARQL query language and formulate queries by themselves. Even Semantic Web experts typically have to invest considerable effort to understand a dataset's structure and its components

[1] `http://digital-agenda-data.eu/sparql`

[2] `http://opendatacommunities.org/sparql`

[3] `http://cofog01.data.scotland.gov.uk/sparql`

before forming a query to obtain relevant information of interest.

Second, each SPARQL endpoint would require development of customized applications, which is highly inefficient. A typical example is the European Commission, which not only provides a SPARQL endpoint, but also visualizations of statistical data by means of ten types of visual charts. Since applications can provide elaborate and highly customizable interfaces, this option may be the most suitable alternative for software developers. However, such applications are frequently proprietary and integrate only a single static data source.

Finally, some researchers attempted to deal with this problem by developing generalized solutions (Maali et al., 2012; Salas et al., 2012; Hoefler et al., 2014; Helmich et al., 2014; Kämpgen and Harth, 2014). The common idea of these approaches is to build a web-based application which can analyze components in each dataset and provide visualization for this dataset.

However, all of these options are associated with considerable disadvantages:

1. Dataset exploration is typically limited to viewing raw data or using limited graphical visualization. This makes it difficult for users to identify trends and study datasets in detail.

2. It is typically not possible to combine or compare data from different datasets, which is an important requirement in data analytics.

3. Available tools are typically not open, i.e., they do not allow users and developers to reuse solutions and extend them with new functionalities and visual presentations. In the context of open data, it is crucial to stress that the means to process and recombine such open data should themselves be open to maximize benefits and foster widespread (re-)use.

4. Existing solutions typically do not cope well with data from available SPARQL endpoints that do not strictly follow the RDF Data Cube Vocabulary. In Section 6, we show that available faceted browsers and tools can only analyze a small number of available endpoints.

In this paper, we address these issues by introducing a novel approach based on widgets and mashups that allow end users to effectively explore statistical data sources available through SPARQL endpoints. We model and expose each dataset of a source as a statistical widget with five salient characteristics: (i) *effective querying*, (ii) *standard format*, (iii) *automatic chart generation*, (iv) *openness*, and (v) *linkage*.

Effective querying means that end users can quickly and easily query a dataset via an interactive interface. Next, widgets return their results in standard JSON-LD (JSON for Linked Data) format (Sporny et al., 2013), even if the data source only partly complies to the vocabulary. Based on the result, the widget will automatically identify suitable charts that provide meaningful views on the dataset. In addition, end users can extend widgets with additional interface components and functionality. Finally, the system allows users to link widgets and thereby establish relationships between statistical datasets. A prototypical implementation of the proposed approach is available at `http://linkedwidgets.org/widget-generation`.

The remainder of this paper is organized as follows. Section 2 provides background information on the Data Cube Vocabulary, widgets, and mashups; Section 3 discusses related work. Section 4 then introduces our widget creation algorithm and Section 5 outlines our approach. Finally, we evaluate our approach and contrast it to existing alternatives in Section 6 and conclude with an outlook on future research in Section 7.

2 BACKGROUND

2.1 Data Cube Vocabulary

The Data Cube Vocabulary (Cyganiak and Reynolds, 2011) is a recently developed mechanism for enriching and transforming statistical datasets and publishing them on the web as Linked Data (Maali et al., 2012). To illustrate the approach, we provide a brief example of a statistical dataset, which represents a collection of observations. A set of dimensions, defining the foundations of the observation (e.g., the time that the observation applies to, or a geographic region that the observation covers), together with measures, which describe objects of the observation (e.g., the number of bus users during this time, or the income of employees at a specific region) semantically describe these collections. Such a statistical dataset is typically presented as a table in which a table's rows represent observations. Furthermore, dimensions typically correspond to primary keys in databases whereas measures represent the remaining columns.

Table 1 shows an example of a Bus Vehicle dataset. *Year* is a dimension, while *Pas* (the number of passengers taking the bus - unit is people in million) and *Kmh* (average speed of bus - unit is km/h) are measures.

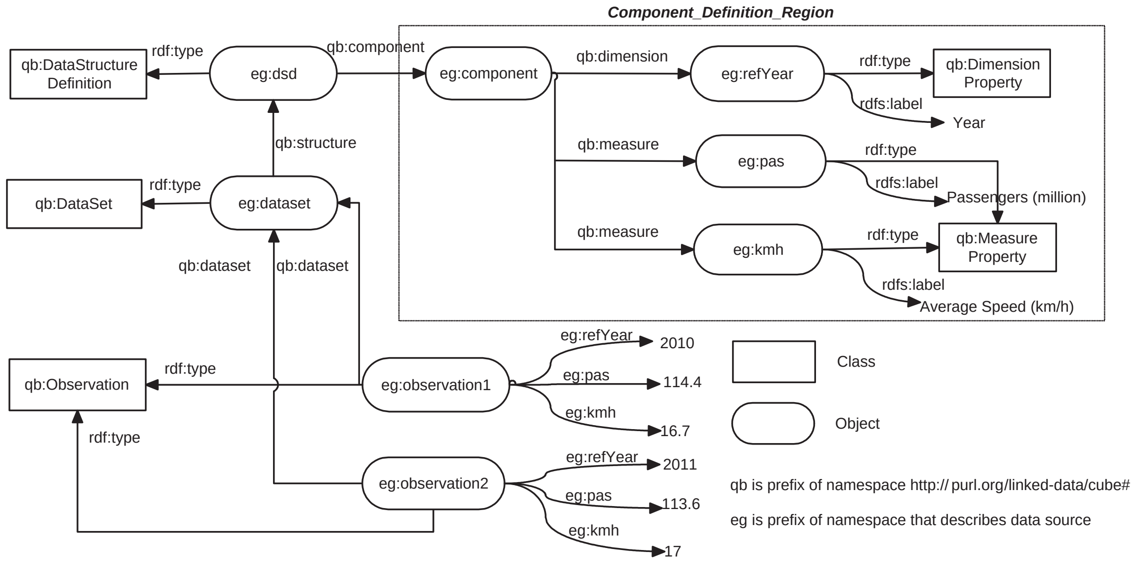

Figure 1: A description according to the Data Cube Vocabulary.

Table 1: An extract of the Bus Vehicle dataset

Year	Pas	Km/h
2010	114.4	16.7
2011	113.6	17
2012	167.1	17.3

Figure 1 presents a description of the Data Cube Vocabulary for this dataset. We will use this figure in the remaining Sections to illustrate and explain our approach.

2.2 Statistical data source exploration

To make a statistical data source available for interactive exploration through linked widgets, it is necessary to identify (i) datasets in the data source, (ii) dimensions and measures associated with each dataset, and (iii) a list of possible values for each dimension. Upon completion of these steps, users can construct meaningful data filters in order to uncover information in large datasets.

To this end, the platform provides mechanisms for *slice-selection* and *visualization* of a part of a dataset (Dadzie and Rowe, 2011), created by filtering a single or multiple dimensions by value. Two types of visualizations are available: (i) single dataset visualizations (e.g., a line chart that describes the trend of number of passengers taking the bus in the period from 2010 to 2012 – cf. Figure 2), and (ii) multiple dataset visualizations (e.g., a multiple column chart that compares the number of passengers taking the bus, tram and metro in the same period – cf. Figure 5).

2.3 Widgets and mashups

Our approach is based on *widgets* and *mashups*. A widget is *"an interactive single purpose application for displaying and/or updating local data or data on the Web, packaged in a way to allow a single download and installation on a user's machine or mobile device"* (Cáceres, 2011). Embedding widgets into a web page allows execution at the client and makes it easy for users to modify them. In addition, widgets can be connected to each other in a mashup (Trinh et al., 2013), which can convey information to the user and highlight features of the data in a fast and efficient manner. A mashup in this context is *"a Web page, or Web application, that uses content from more than one source to create a single new service displayed in a single graphical interface"* (Crupi, 2010).

3 RELATED WORK

Several researchers implemented custom browsers to facilitate exploration of statistical datasets. Using the URL of a SPARQL endpoint or a dataset as a starting point, they allow users to explore the data source.

CubeViz[4] (Salas et al., 2012) is a general purpose solution for exploring statistical data sources and provides visual presentations via five different types of charts. In our evaluation, however, we found that it only worked with two data sources from the European Commission and was not able to detect datasets using

[4] http://cubeviz.aksw.org/

other endpoints (cf. Table 4).

Data Cube faceted browser[5](Maali et al., 2012) was able to detect datasets in a larger number of endpoints in our evaluation presented in Section 6. However, it is only suitable for datasets with a small number of observations. It is also limited in that it provides only a list of observations of each dataset without any visual charts.

Linked Data Query Wizard[6] (Hoefler et al., 2014) applies the idea of presenting statistical datasets via a tabular interface. As such, end users can change the slices of a dataset by choosing one value as a filter value. However, due to the lack of a complete list of values for each dimension, this solution constrains end users to a small number of slices. In addition, this solution is restricted to a static list of specific endpoints.

Linked Data Cubes Explorer (LDCE)[7] (Kämpgen and Harth, 2014) uses an OLAP approach to validate and analyze statistical datasets. Unfortunately, this tool only works with its sample datasets and returns an error for any external dataset (cf. Table 4).

Payola[8] (Helmich et al., 2014) can receive an arbitrary RDF source and transform it to RDF conforming to the Data Cube Vocabulary. After that, it can provide user-friendly visualizations. At present, however, this tool seems unstable and cannot run its sample experiments.

The difficulties in analyzing data sources that these tools face stem from the variability in the use of the vocabulary. A considerable number of datasets only describe a part of the vocabulary. For example, in Figure 1, without using *Component_Definition_Region*, we cannot directly differentiate between dimensions and measures in the dataset. In addition, there are datasets which use *slices* (Cyganiak and Reynolds, 2011) to build subsets of observations without using the predicate *qb:dataset*. We implemented an algorithm to cope with such heterogeneity and inconsistency.

Furthermore, existing solutions use *open data* to provide *closed applications* which run on the server side. This poses a contradiction. We provide a novel approach that is *open* for adaptation and extension by end users. To this end, our approach presents data in a well-defined standard format.

[5]`http://vmsgov03.deri.ie:8080/RDF-faceted-browser/start.html`

[6]`http://code.know-center.tugraz.at/search`

[7]`http://ldcx.linked-data-cubes.org/projects/ldcx`

[8]`http://datacube.payola.cz`

4 DATA SOURCE ANALYSIS

To analyze data sources provided via SPARQL endpoints and automatically generate widgets for the identified datasets, we introduce an algorithm that involves the following steps: (i) datasets identification, (ii) dimensions and measures identification, and (iii) values and labels identification. In the following, we use Figure 1 as an illustrative example.

4.1 Datasets Identification

The vocabulary allows to identify a dataset (i.e., `eg:dataset`) through one of the following triple patterns: $eg:observation - qb:dataset \rightarrow eg:dataset$, $eg:dataset - rdf:type \rightarrow qb:DataSet$, and $eg:dataset - qb:structure \rightarrow eg:dsd$.

The first pattern is available in almost all evaluated SPARQL endpoints, because it provides the relationship between a dataset and its observations. However, an endpoint can provide millions of observations, and therefore this query, which is essential for dataset detection, will eventually timeout. To alleviate this issue, we apply the two latter triple patterns. They represent a 1:1 relationship between a dataset and its type, and between a dataset and its data structure definition, respectively. Unfortunately, datasets can focus on describing observations without describing the remaining components of the vocabulary, rendering these triple patterns useless. Overall, however, the combined utilization of all three triple patterns results in high recall when detecting datasets.

4.2 Dimensions and Measures Identification

Ideally, the predicates `qb:dimension`, `qb:measure` or `rdf:type` – which are defined in the vocabulary – indicate dimensions and measures (i.e., `eg:refYear`, `eg:pas`, `eg:kmh`) directly. Otherwise, we can only derive dimension and measures based on their URIs and values in the observations. In statistical data, the values of measures are typically represented via numerical formats whereas the values of dimensions are often either years or strings (e.g. country code: AT for Austria, BE for Belgium, etc.). Furthermore, URIs of components may indicate their role, such as `http://.../measure/pas` for a measure.

4.3 Values and Labels Identification

A complete list of values of each dimension must be obtained, because it can serve as filter values for end users. In addition, we need to identify labels

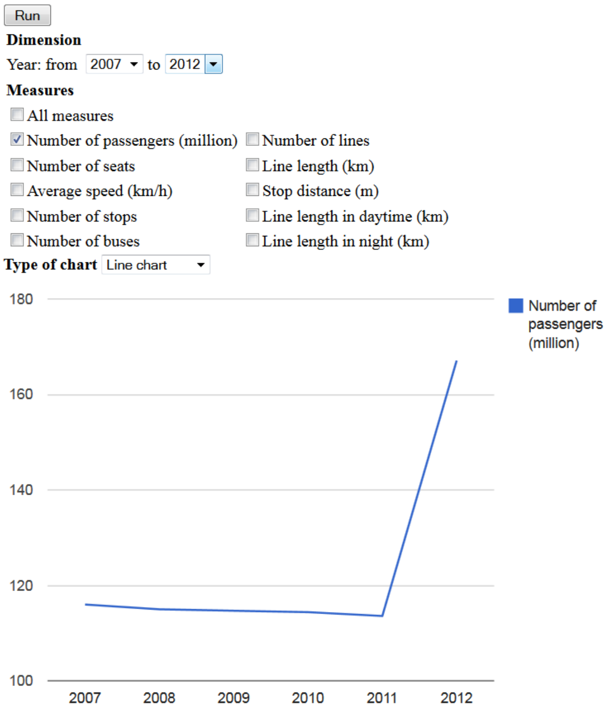

Figure 2: An automatically generated widget

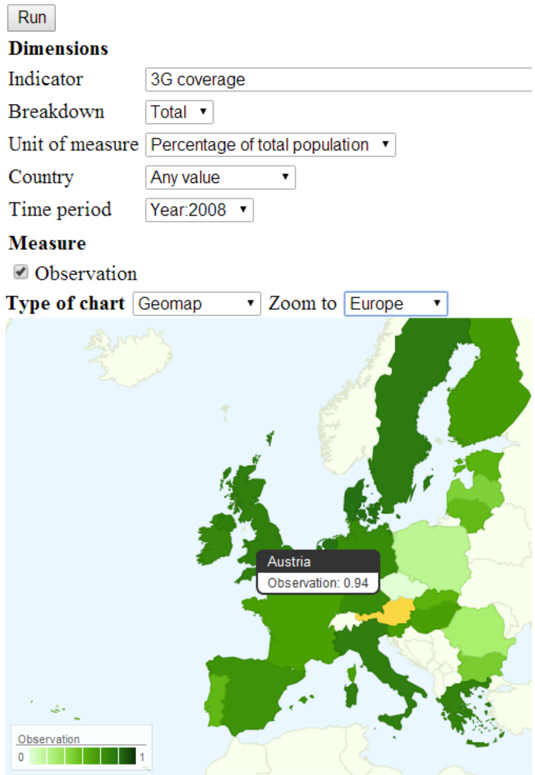

Figure 3: Automatically generated widget of the European Commission endpoint.

of dimensions and measures in each dataset, because they support users to understand the meaning of these components. For example, in the *Bus Vehicle* dataset, the measure *"Pas"* does not make sense for users, while its label – *"Number of passengers"* is a meaningful description. To overcome limitations of query time for large datasets, we use loops to retrieve data within a given time threshold.

5 STATISTICAL WIDGET GENERATION AND MASHUP

Figure 2 shows a sample widget. Each widget consists of (i) a *list of dimensions*, (ii) a *list of measures*, and (iii) a *chart type*, which allows users to impose filter conditions in an easily comprehensible manner.

Next, depending on the options, each widget generates a SPARQL query to collect the desired data, which is then converted into the JSON-LD format (Sporny et al., 2013) and set as the input of the chart. The use of JSON-LD facilitates the integration of data between disparate systems, thereby supporting the combination of statistical datasets. Figure 4 provides an example of a JSON-LD description of a query.

Charts may illustrate statistical data in a meaningful way and can uncover relationships that are not obvious from studying a list of numbers. Based on the JSON-LD result of the query, the widget automatically detects which types of charts are feasible for a specific query from a list of nine common chart types, e.g. Column, Line, Pie, Bubble, and Geo map chart[9] .

Developers need appropriate means to modify auto-generated widgets in order to extend the interface and incorporate additional functionality. For example, the European Commission's endpoint offers only one statistical dataset, but it has more than 100 *indicators* and each *indicator* requires a specific value set for the dimensions *breakdown*, *unit of measure*, *country*, and *time period* as shown in Figure 3. Therefore, additional functionality is necessary to list only suitable values for remaining dimensions whenever users change the value of the *indicator*. Developers may also, for example, impose thresholds on dimension values, query a limited list of measures, or provide a new visual chart type.

Using JSON-LD, the generated widgets can collaborate in a mashup to compare and combine data from different datasets. For example, Figure 5 shows a mashup that compares the number of passengers using bus, tram, and metro vehicles.

[9] `https://developers.google.com/chart/interactive/docs/gallery`

```
{
  “@context”:{
    “vogd”: “http://ogd.ifs.tuwien.ac.at/vienna/”,
    “qb”: “http://purl.org/linked-data/cube#”,
    “xsd”: “http://www.w3.org/2001/XMLSchema#”,
    “observation”: “qb: observation”,
    “dimension”: “qb: dimension”,
    “measure”: “qb: measure”,
    “component”: “qb: component”,
    “label”: “http://www.w3.org/2000/01/rdf-schema#label”,
    “year”: “xsd: gYear”,
    “pas”: “vogd: pas”,
    “kmh”: “vogd: kmh”
  },
  “@id”: “vogd: betriebszweige2012-autobus”,
  “@type”: “qb: DataSet”,
  “label”: “Vienna Bus”,
  “component”:
  {
    “@type”: “qb: ComponentSpecification”,
    “dimension”:
    [
      {
        “@id”: “xsd: gYear”,
        “@type”: “qb: DimensionProperty”,
        “label”: “Year”
      }
    ],
    “measure”:
    [
      {
        “@id”: “vogd: pas”,
        “@type”: “qb: MeasurePropery”,
        “label”: “Number of passengers (million)”
      },
      {
        “@id”: “vogd:kmh”,
        “@type”: “qb:MeasurePropery”,
        “label”: “Average speed(km/h)”
      }
    ],
    “observation”:
    [
      {
        “@type”: “qb:Observation”,
        “@id”: “vogd: betriebszweige2012-autobus.9”,
        “year”: “2010”,
        “pas”: 114.4,
        “kmh”: 16.7
      }
    ]
  }
}
```


Figure 4: Query result in JSON-LD format

Each statistical mashup can be composed of three types of widgets: (i) *Dataset Widgets* are auto-generated widgets or modified widgets, (ii) *Merger Widgets*, which integrate two datasets from compatible input widgets, i.e. widgets with the same dimensions, to a single combined dataset, and (iii) *General Presentation Widgets*, which receive data from either a Dataset Widget or a Merger Widget and displays them visually.

We can also distinguish widget types by the cardinality of their inputs and outputs: dataset widgets do not have an input, but have an output; merger widgets have two inputs and an output; general presentation widgets have an input, but no outputs.

To illustrate the transformations performed by merger widgets, assume that we have separate passenger datasets for tram and bus vehicles (cf. Table 2 and 1, respectively). We use dimensions, i.e. Year, to group data by year, allowing users to easily compare the number passengers using Bus and Tram for each year (cf. Table 3).

Table 2: Subset of a Tram dataset

Year	Pas	Hast
2011	193.8	1031
2012	295.1	1056

Pas = number of passengers; Hast = Number of stops

Table 3: Combined Bus & Tram dataset

Year	Pas_Bus	Pas_Tram	Khm	Hast
2010	114.4	-	16.7	-
2011	113.6	193.8	17	1031
2012	167.1	295.1	17.3	1056

We outline the algorithm to merge dimensions, measures, and observations from two datasets in the following:

1. Dimensions of the output dataset are the same as the dimensions of the input datasets.
2. Measures of the output dataset contain all measures from two input datasets. Measures which belong to both of two input datasets, e.g., Pas, play an important role in comparing and contrasting data. However, since they have the same URI, each measure will receive a new URI. The new URI and the original URI are linked via the *owl:sameAs* predicate.
3. Observations: the algorithm merges observations of two input datasets based on the values of dimensions to build observations of the output dataset. If two observations from two input datasets have the same dimension value, e.g. Year - 2011, the algorithm will merge them into a single observation in Table 3. Otherwise, if the dimension value of an observation O_1 from one dataset does not appear in the other dataset, e.g., Year - 2010 in Bus Vehicle, the algorithm generates an empty observation O_2 for the latter dataset before merging it with O_1 to the output dataset.

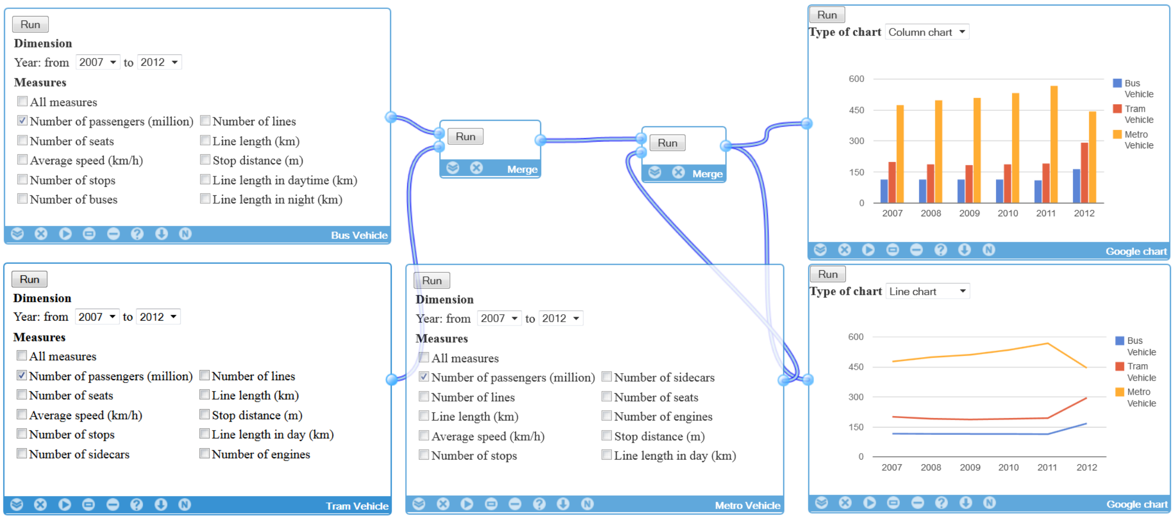

Figure 5: An example of a mashup for data comparison

6 EVALUATION

We performed two tasks for a preliminary evaluation of our approach: (i) test the data source analysis algorithm with 23 available endpoints using the Data Cube Vocabulary, and (ii) validate the features provided by auto-generated widgets.

In our experiments for the first task, we found that our prototypical implementation can analyze and create widgets for all endpoints tested. We compared our results to those obtained using existing alternatives with respect to four aspects: (i) datasets identified, (ii) dimensions identified, (iii) measures identified in each dataset, and (iv) list of values identified for each dimension. The result in Table 4 shows that the existing browsers can only handle data from less than four SPARQL endpoints correctly. The *Linked Data Cube Explorer* even cannot analyze a single dataset out of the 23 tested endpoints. The *Linked Data Query Wizard*, by contrast, can analyze the first twelve endpoints. However, since it supports only a limited number of fixed input endpoints, we cannot evaluate its capabilities for the full set of tested endpoints.

In the latter task, we compared features of auto-generated widgets for the endpoint of the European Commission with visual charts designed specifically for this endpoint[10]. Using the developed platform, users can easily impose filters on a single or multiple dimensions to explore particular slices of the dataset (cf. Figure 3 for an example). Hence, users do not need to write complex SPARQL queries such as the one listed in Figure 6, which retrieves the same data as the mashup depicted in Figure 3. Our *automatic chart* generation feature ensures that the widget can provide suitable charts for the selected data, e.g. column, bar, pie, donut, geo charts, and Geo Maps are available for the view in Figure 3). In addition, users can use mashups to compare the values between indicators or compare values from different countries. We built two widgets – *Merge widget* and *General Presentation widget* – which are suitable for arbitrary statistical *Dataset widgets*. Overall, our automatically generated widget offers visualizations which are comparable in functionality to those provided by the application of the European Commission. However, our visualizations can be used not only for this specific endpoint, but more generally for arbitrary endpoints.

```
PREFIX qb: <http://purl.org/linked-data/cube# >
PREFIX sdmx: <http://purl.org/linked-data/sdmx/2009/measure# >
PREFIX digital:<http://semantic.digital-agenda-data.eu/def/property/>
PREFIX dataset: <http://semantic.digital-agenda-data.eu/dataset/ >
PREFIX indi: <http://semantic.digital-agenda-data.eu/codelist/indicator/>
PREFIX time: <http://reference.data.gov.uk/id/gregorian-year/>
SELECT DISTINCT * WHERE {
    ?o qb:dataSet ?ds.
    ?o sdmx:obsValue ?obsValue.
    ?o digital:indicator ?indicator.
    ?o digital:breakdown ?breakdown.
    ?o digital:ref-area ?ref_area.
    ?o digital:time-period ?time_period.
    ?o digital:unit-measure ?unit_measure.
FILTER(?ds = dataset:digital-agenda-scoreboard-key-indicators)
FILTER(?indicator=indi:mbb_3gcov)
FILTER(?time_period= time:2008)
}
```

Figure 6: An example of a SPARQL query

[10] http://ec.europa.eu/digital-agenda/en/graphs

7 CONCLUSIONS

Based on the idea of Linked Data, which aims to connect and reuse data rather than storing it in isolated silos, we propose a novel approach to provide end users with efficient mechanisms to analyze, combine, remix, visualize, and make sense of statistical data available via SPARQL endpoints using the Data Cube Vocabulary. Each statistical dataset is automatically made available as a widget that allows *effective querying*, provides output in a *standard format*, facilitates *automatic chart generation*, and embodies principles of *openness* and *linkage*. Two of these characteristics – *effective querying* and *automatic chart generation* – allow end users to effectively explore the dataset. Furthermore, *standard format* and *openness* enable developers to modify and develop new functionalities and new types of visualization. Finally, *linkage* allows all automatically generated and manually modified widgets to be combined flexibly in a mashup.

We also presented a prototypical implementation of the proposed system and evaluated it using 23 SPARQL endpoints that use the Data Cube Vocabulary. We found that the approach shows great potential and handles data from all identified SPARQL endpoints well, even those that only partly follow the RDF Data Cube Vocabulary.

Due to the problems of co-reference between URIs, ontology mapping, etc. (Millard et al., 2010; Schlegel et al., 2014), widget *linkage* is currently supported only for datasets from the same endpoint. As a next step, we plan to address this limitation.

REFERENCES

Brunetti, J. M., Auer, S., García, R., Klímek, J., and Necaský, M. (2013). Formal linked data visualization model. In *IIWAS'13, International Conference on Information Integration and Web-based Applications & Services*.

Cáceres, M. (2011). Requirement for Standardizing Widgets.

Crupi, J. (2010). A business guide to enterprise mashups.

Cyganiak, R. and Reynolds, D. (2011). The RDF Data Cube Vocabulary.

Dadzie, A. S. and Rowe, M. (2011). Approaches to visualising linked data: A survey. *Semantic Web*, 2:89–124.

Helmich, J., Klímek, J., and Necaský, M. (2014). Visualizing rdf data cubes using the linked data visualization model. In *ESWC'14, Extended Semantic Web Conference - Posters & Demonstrations Track*.

Hoefler, P., Granitzer, M., Veas, E., and Seifert, C. (2014). Linked data query wizard a novel interface for accessing sparql endpoints. In *LDOW'14, Linked Data on the Web Workshop*.

Kämpgen, B. and Harth, A. (2014). Olap4ld a framework for building analysis applications over governmental statistics. In *ESWC'14, Extended Semantic Web Conference - Posters & Demonstrations Track*.

Maali, F., Shukair, G., and Loutas, N. (2012). A dynamic faceted browser for data cube statistical data. In *W3C Using Open Data Workshop*.

Millard, I. C., Glaser, H., Salvadores, M., and Shadbolt, N. (2010). Consuming multiple linked data sources: Challenges and experiences. In *COLD'10, International Workshop on Consuming Linked Data*.

Salas, P. E. R., Mota, F. M. D., Martin, M., Auer, S., Breitman, K., and Casanova, M. A. (2012). Publishing statistical data on the web. *International Journal of Semantic Computing*, 6:373–388.

Schlegel, T., Stegmaier, F., Bayerl, S., Granitzer, M., and Kosch, H. (2014). Balloon fusion: Sparql rewriting based on unified co-reference information. In *DESWeb'14, 5th International Workshop on Data Engineering Meets the Semantic Web*.

Sporny, M., Kellogg, G., and Lanthaler, M. (2013). JSON-LD 1.0 - A JSON-based Serialization for Linked Data.

Trinh, T.-D., Do, B.-L., Wetz, P., Anjomshoaa, A., and Tjoa, A. M. (2013). Linked widgets an approach to exploit open government data. In *IIWAS'13, International Conference on Information Integration and Web-based Application & Services*.

APPENDIX

Table 4: Evaluation results for existing browsers

Endpoint	Faceted browser				CubeViz				LDCE			
	DS	D	M	V	DS	D	M	V	DS	D	M	V
`http://open-data.europa.eu/en/sparqlep`	√	√	√	χ	√	√	√	√	χ	χ	χ	χ
`http://digital-agenda-data.eu/data/sparql`	√	√	χ	χ	√	√	√	√	χ	χ	χ	χ
`http://ogd.ifs.tuwien.ac.at/sparql`	√	√	√	√	√	χ	χ	χ	χ	χ	χ	χ
`http://zaire.dimis.fim.uni-passau.de:8890/sparql`	∞	∞	∞	∞	χ	χ	χ	χ	χ	χ	χ	χ
`http://ecb.270a.info/sparql`	∞	∞	∞	∞	χ	χ	χ	χ	χ	χ	χ	χ
`http://fao.270a.info/sparql`	∞	∞	∞	∞	χ	χ	χ	χ	χ	χ	χ	χ
`http://imf.270a.info/sparql`	∞	∞	∞	∞	χ	χ	χ	χ	χ	χ	χ	χ
`http://oecd.270a.info/sparql`	∞	∞	∞	∞	χ	χ	χ	χ	χ	χ	χ	χ
`http://transparency.270a.info/sparql`	√	√	√	√	√	χ	χ	χ	χ	χ	χ	χ
`http://worldbank.270a.info/sparql`	∞	∞	∞	∞	√	χ	χ	χ	χ	χ	χ	χ
`http://datameti.go.jp/sparql`	√	√	√	√	√	χ	χ	χ	χ	χ	χ	χ
`http://semantic.eea.europa.eu/sparql`	∞	∞	∞	∞	χ	χ	χ	χ	χ	χ	χ	χ
`http://gov.tso.co.uk/coins/sparql`	∞	∞	∞	∞	√	χ	χ	χ	χ	χ	χ	χ
`http://openuplabs.tso.co.uk/sparql/gov-coins`	∞	∞	∞	∞	χ	χ	χ	χ	χ	χ	χ	χ
`http://agencies.publicdata.eu/sparql`	√	χ	√	χ	χ	χ	χ	χ	χ	χ	χ	χ
`http://unodc.publicdata.eu/sparql`	√	χ	√	χ	χ	χ	χ	χ	χ	χ	χ	χ
`http://cofog01.data.scotland.gov.uk/sparql`	∞	∞	∞	∞	χ	χ	χ	χ	χ	χ	χ	χ
`http://eur-lex.publicdata.eu/sparql`	√	χ	√	χ	χ	χ	χ	χ	χ	χ	χ	χ
`http://prelex.publicdata.eu/sparql`	√	χ	√	χ	χ	χ	χ	χ	χ	χ	χ	χ
`http://n-lex.publicdata.eu/sparql`	√	χ	√	χ	χ	χ	χ	χ	χ	χ	χ	χ
`http://eventmedia.eurecom.fr/sparql`	√	χ	χ	χ	χ	χ	χ	χ	χ	χ	χ	χ
`http://opendatacommunities.org/sparql`	∞	∞	∞	∞	χ	χ	χ	χ	χ	χ	χ	χ
`http://open-data.europa.eu/linked-data`	∞	∞	∞	∞	χ	χ	χ	χ	χ	χ	χ	χ

DS: Dataset, D: Dimensions, M: Measures, V: Values of Dimension

√:Yes; χ:No; ∞: No result after one hour