Toward a Statistical Data Integration Environment – The Role of Semantic Metadata

Ba-Lam Do
TU Wien, Vienna, Austria
ba.do@tuwien.ac.at

Tuan-Dat Trinh
TU Wien, Vienna, Austria
tuan.trinh@tuwien.ac.at

Peb Ruswono Aryan
TU Wien, Vienna, Austria
peb.aryan@tuwien.ac.at

Peter Wetz
TU Wien, Vienna, Austria
peter.wetz@tuwien.ac.at

Elmar Kiesling
TU Wien, Vienna, Austria
elmar.kiesling@tuwien.ac.at

A Min Tjoa
TU Wien, Vienna, Austria
a.tjoa@tuwien.ac.at

ABSTRACT

In most government and business organizations alike, statistical data provides the foundation for strategic planning and for the management of operations. In this context, the use of increasingly abundant statistical data available on the web creates new opportunities for interesting applications and facilitates more informed decision-making. For the majority of end users, however, viable means to explore statistical data sets available on the web are still scarce. Gathering and relating statistical data from multiple sources is hence typically a tedious manual process that requires significant technical expertise. Data that is being published with associated semantics, using standards such as the W3C RDF Data Cube Vocabulary, lays the foundation to overcome such limitations. In this paper, we develop a semantic metadata repository that describes each statistical data set and develop mechanisms for the interconnection of data sets based on their metadata. Finally, we support users in exploring data sets through interactive mashups that facilitate data integration, comparisons, and visualization.

Categories and Subject Descriptors

D.1.7 [**Programming Techniques**]: Visual Programming; E.2 [**Data Storage Representations**]: Linked representations; H.2.8 [**Database Management**]: Database Applications—*Spatial databases and GIS, Statistical databases*

Keywords

Semantic Metadata, Data Integration, Statistical Data, Spatial Dimension, Temporal Dimension, RDF Data Cube Vocabulary, Mashup

1. INTRODUCTION

The proliferation of Open Data policies has encouraged governments and business organizations to publish data on the web. Statistical data, which embodies a large portion of published data, has attracted great interest of users. This data comprises a wide range of domains such as finance, demographics, and transportation and plays an increasingly important role in strategic planning, strategy implementation, and management of the activities of an organization.

To increase the value of data, many large organizations such as the European Environment Agency (EEA)[1] or the European Commission (EC)[2] have adopted and started to use the RDF Data Cube vocabulary [4], a standard of the W3C, to publish their data as Linked Open Data (LOD). This vocabulary represents data in a standardized manner, thereby facilitates the integration of disparate data sources. However, viable means to explore and integrate such statistical data sources are still scarce due to the following reasons:

1. Without a single point of access, users will find it difficult to discover relevant statistical data.
2. Gathering and identifying co-reference information from multiple sources is difficult because the same entity can be represented by different identifiers in different data sources [1]. For example, the EEA and the EC utilize different URIs to describe the same geographical entities (e.g., a country), but these URIs are not linked to each other or referred to a common URI. Attempts to deal with this issue through crawling *owl:sameAs* relationships in data sources exist [8, 22], but this approach cannot detect equivalent entities if the data sources lack *owl:sameAs* relationships, which often is the case. Another approach is to manually provide *owl:sameAs* relationships to integrate data from heterogeneous data sources [12, 19]. This approach, while helpful, is cumbersome and time consuming and therefore only applicable to a limited amount of data sources.
3. Available statistical data exploration applications [16, 21, 2, 19, 11, 10, 13] support users in exploring data sources in various different ways. To exploit the available functionality, users would require a seamless combination of these existing applications. Those do not, however, allow users and developers to extend them with new functionality.

In this paper, we address the identified issues by introducing a novel approach based on semantic metadata. To

SEMANTiCS '15, September 15-17, 2015, Vienna, Austria

DOI: http://dx.doi.org/10.1145/2814864.2814879

[1] http://semantic.eea.europa.eu/sparql
[2] http://digital-agenda-data.eu/data/sparql

leverage users' creativeness and to satisfy their information needs, we aim to provide a flexible and powerful environment that allows users to explore data by combining existing and novel functionality. In particular we contribute the following:

1. To deal with the first issue of discovering relevant statistical data sets, we analyze each data set to construct semantic metadata. The metadata relates components (i.e, dimensions, measures, and attributes) and values (i.e., values of each dimension, attribute) of a data set to their equivalent entities that are used consistently among all data sets. Metadata hence provides a sound foundation for integrating different data sets. We publish the resulting metadata repository[3] as LOD and allow data publishers to contribute new metadata for their data sets. This repository, therefore, acts as a single point of access to query and integrate different statistical data sets.
2. We focus particularly on identifying equivalent entities for spatial and temporal dimensions, because those are essential in virtually any statistical data set. To this end, we match the URIs of these dimensions to corresponding concepts of SDMX (Statistical Data and Metadata eXchange)[4], an ISO standard for processing and exchanging statistical data. We design two algorithms to match values to consolidated URIs. The first algorithm uses Google's Geocoding API[5] to generate a unique and consistent URI for same areas that are previously represented by different URIs. Furthermore, a hierarchy of areas represented by *sw:broader* and *sw:narrower* relationships[6] is built to support integration at different levels of granularity. The second algorithm uses time patterns to match temporal values (e.g., URIs or literal values) with URIs used in the data.gov.uk time reference service[7]. This service provides semantics annotation for a wide range of measure intervals (e.g., year, month). Furthermore, using the predicate *time:intervalContains*[8] this service describes relationships between an interval and its sub-intervals that facilitate integration at different levels of temporal granularity.
3. To allow users to proactively explore and integrate statistical data, we make use of the Linked Widget platform, which is a flexible, interactive mashup platform [25]. We introduce use cases to highlight the role of using semantic metadata in the mashup platform. The examples use geographical locations as an input and trigger the data set discovery process. Based on given locations, we query the metadata repository to find relevant statistical data sets, which allows users to compare, integrate, and visualize them. The repository and the platform are open and users are able to implement their own use cases.

The remainder of this paper is organized as follows. In Section 2, we introduce our approach grounded in semantic metadata. Section 3 illustrates our mashup approach with a collection of widgets and practical use cases. Section 4 discusses related work and we conclude in Section 5 with an outlook on future research.

[3]http://ogd.ifs.tuwien.ac.at/sparql
[4]http://sdmx.org/
[5]https://developers.google.com/maps/documentation/geocoding/
[6]sw:http://linkedwidgets.org/statisticalwidgets/ontology/
[7]http://reference.data.gov.uk/id/gregorian-interval
[8]time:http://www.w3.org/2006/time#

2. SEMANTIC METADATA

In this section, we introduce selected statistical data sources used in our running example, outline the spatial and temporal mapping algorithms that automatically generate equivalent entities, and discuss the construction of semantic metadata.

2.1 Examples of Statistical Data Sources

We select four popular LOD statistical data sources from the European Environment Agency (EEA), European Commission (EC), European Open Data Portal (EODP)[9], and our own Linked Data version of Vienna Open Government Data (VOGD)[10] [24]. These data sources include a large number of statistical data sets on topics such as economy, education, environment, and health. Gathering and connecting statistical data of organizations at different scales, e.g., country scale (VOGD) and continent scale (EEA, EODP, EC) highlights the potential of our data integration approach and allows users to obtain a more comprehensive view on the data.

2.2 Running Example

To illustrate the construction of semantic metadata, we chose a data set of the average EC funding per participation in FP7-ICT projects[11].

Table 1 shows a set of observations in the data set. In this data set, *eg-p:country* and *eg-p:year* are two dimensions, while *eg-p:value* is a measure; and *eg-p:unit* is an attribute of the observed values.

eg-p:country	eg-p:year	eg-p:value	eg-p:unit
eg-c:Austria	eg-y:2007	358279	euro
eg-c:Germany	eg-y:2007	414531	euro
eg-c:Austria	eg-y:2008	358133	euro

Table 1: Excerpt from the running example data set[12]

2.3 Spatial Dimension Mapping

The spatial dimension (i.e., *eg-p:country*) describes geographical area(s) where statistical observations were made. Although this dimension appears in most statistical data sets, it still can be missing sometimes. This is the case if a data set describes data of only one specific geographic area. Another data set, for example EAE contains data on landings of fishery products in Germany[13], but does not describe a spatial dimension, because it implicitly refers to Germany

[9]http://open-data.europa.eu/en/linked-data
[10]http://ogd.ifs.tuwien.ac.at/sparql
[11]http://data.lod2.eu/scoreboard/ds/indicator/FP7ICT_afxp_All_partners_euro
[12]`eg-p`: http://data.lod2.eu/scoreboard/properties/
`eg-c`: http://data.lod2.eu/scoreboard/country/
`eg-y`: http://data.lod2.eu/scoreboard/year/
[13]http://rdfdata.eionet.europa.eu/page/eurostat/data/fish_ld_de

via its label. In this case, it is necessary to explicitly add a spatial dimension to the structure of the data set as well as identify a unique value for it (i.e., *Germany*). This addition allows users to compare and integrate data of Germany in this data set with other data sets. To add required dimension, we compare the label and URI of a data set with a list of countries and their ISO codes to identify the country which it represents.

Ideally, all LOD data sources would make use of a shared URI to represent the spatial dimension. However, in practice, each data publisher has a tendency to define a local URI belonging to their own domain name, e.g., *eg-p:country* in the EODP. Although this allows data publishers to customize spatial dimensions with different associated semantics, e.g., its label or range of values, this causes severe difficulties in, for instance, identifying spatial dimensions. To alleviate this situation, the EEA uses the concept *sdmx-dimension:refArea*[14] from SDMX which allows data publishers to represent spatial dimensions in a standardized way. As a first step towards consolidation of spatial dimensions, we match the different URIs of spatial dimension to *sdmx-dimension:refArea*.

Two data sources (EODP, EC) only contain statistical data of European countries, whereas the EEA collects data for geographical areas in a more detailed manner using the NUTS (Nomenclature of territorial units for statistics)[15] territorial breakdown system. In particular, the EEA contains data at three levels: (i) major socio-economic regions - NUTS1 such as Ostösterreich (Eastern Austria), Südösterreich (Southern Austria), Westösterreich (Western Austria); (ii) basic geographical regions - NUTS2 such as Burgenland, Vienna, Salzburg; and (iii) smaller regions - NUTS3 such as Nordburgenland (Northern Burgenland), Mittelburgenland (Central Burgenland), and SüdBurgenland (Southern Burgenland). The aim of NUTS is to provide a coherent and common classification system which can be used by all EU member countries. The Viennese city government, by contrast, collects statistical data based on so-called administrative areas. This means that it does not contain, for instance, regions classified by cardinal direction such as Eastern Austria.

Each spatial URI (e.g., *eg-c:Austria*) is often attached with a textual description of that area (e.g., "Austria"). The same geographical areas therefore can be mentioned using different labels in different languages. We use Google's Geocoding API to resolve spatial areas into hierarchical information based on the label of the URI of the spatial value. It will return the same results for different area names such as "Austria", "Österreich", or "Autriche" and hence provide a common reference point.

However, there are three issues which need to be overcome when using this service: (i) For one input, the service is likely to return a large number of different areas due to its ambiguity; (ii) The same area can exist at different administrative levels. Vienna, for instance, exists at both state level (administrative level 1) and city level (administrative level 2); and (iii) The service works with actual administrative areas, hence, regions classified based on cardinal directions (e.g., Eastern, Northern) do not yield correct results.

In order to solve these issues, we develop an approach

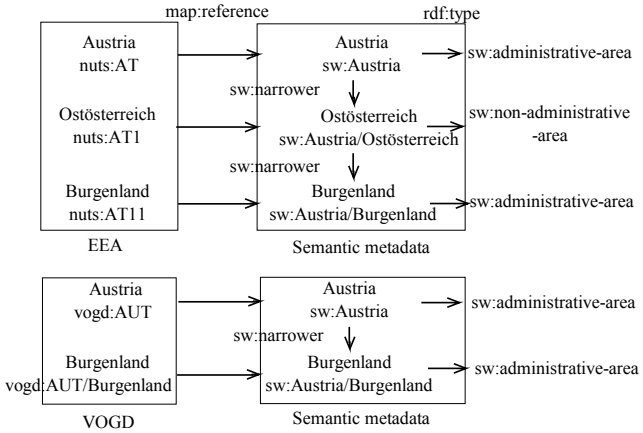

Figure 1: Example of Geographical Dimension Mapping[16]

to consolidate spatial values based on their administrative level. Particularly, we exploit our knowledge of the geographical hierarchy (e.g., Vienna (state) is one level below Austria (country)) to resolve ambiguities which may arise when performing the geocoding. To determine whether an area_k is one level higher than area_i in a geographical classification system, we define a heuristic function to estimate this relationship. For example, EEA uses the NUTS classification where higher territorial levels are denoted based on substrings of the lower level URI plus a difference in character length. We can see in Figure 1 that *nuts:AT* (Austria) is one level of detail higher than *nuts:AT1* and that *nuts:AT1* is one level of detail higher than *nuts:AT11* (Burgenland). Furthermore, we order the areas such that broader areas are positioned before narrower areas. To solve the first issue, we use this ordering to reduce the ambiguity by adding the label of the broader area to the queries of its narrower areas. The second issue can be solved by using the similarity of ordering. For example, in VOGD, Austria is one level higher than Vienna, hence we choose Vienna in Google's results at one level narrower, i.e., in level 1. To address the last issue, we create a new URI combining the URI of its broader area and its label.

Assume that $L = \{l_1, \ldots, l_n\}$ is a set of spatial values associated with the spatial dimension in a data source and $G = \{g_1, \ldots, g_n\}$ is the output of the algorithm (Algorithm 1). Each $l_i \in L$ is a pair (uri_l_i, label_l_i) that contains its URI and label. Our algorithm solves the unification of spatial URIs by mapping each area $l_i \in L$ to g_i using Google's Geocoding Service. Given l_i as an input, the service can return a set of probable areas. The algorithm selects one area from this result. In addition, we use the predicates *sw:broader* and *sw:narrower* to build relationships between areas in the resulting set G. In our algorithm, both triples *(area_k, sw:narrower, area_i)* and *(area_i, sw:broader, area_k)* mean that in a geographical classification system, area_k is one level higher than area_i. We designed Algorithm 1 to match areas in L to G as follows:

- To query an area l_i, we combine its label with the label of area l_k, in which l_k is the broader area of l_i.

[14]http://purl.org/linked-data/sdmx/2009/dimension#
[15]http://ec.europa.eu/eurostat/web/nuts/overview

[16]`map`: http://linkedwidgets.org/statisticalwidgets/mapping/
`vogd`: http://ogd.ifs.tuwien.ac.at/vienna/geo/
`nuts`: http://dd.eionet.europa.eu/vocabulary/common/nuts/

- We filter the results in g_i via two steps (assume that $g_i = \{r_1, \ldots, r_m\}$). First, because l_k is a broader area of l_i, we deduce that g_k is also a broader area of g_i. Results which do not satisfy this requirement are removed from g_i. Second, we select result r_j which has a minimal distance to adjacent areas g_{i-1} and g_{i-2} (if $i \geq 2$).
- Direction-classified regions are assigned new URIs based on the URI of its broader area (i.e., uri_k) and its label (i.e, $label_i$). To distinguish these regions, we set their type to *non-administrative area*. As a result, data can be aggregated from narrower areas through two distinctive groups: administrative areas (e.g., Burgenland, Vienna) and non-administrative areas (e.g., Östosterreich, Südösterreich).

2.4 Temporal Dimension Mapping

The temporal dimension (i.e., *eg-p:year*) represents the time periods in which data publishers collected observations such as day, month, quarter, or year. It plays an important role in comparing changes of observed values over time. Despite this importance, there are cases where data sets lack a temporal dimension. For example, VOGD contains statistical data of election results at different areas of Austria in 2013, but there is no explicit temporal dimension in the structure of this data set. By matching URIs and labels of such data sets with time patterns, we add a temporal dimension to their structures. As a result, after a pre-processing step, all statistical data sets of our four selected LOD sources contain a temporal dimension.

The LOD data sources EEA, EODP, EC, and VOGD define their own URIs for representing the temporal dimension. A reference to a common concept of temporal dimension is necessary to handle this variation. To this end, we look into existing concepts about time intervals from the SDMX standard. SDMX offers two concepts to refer to time, i.e., *sdmx-dimension:refPeriod* and *sdmx-dimension:timePeriod*. The first concept represents a period which an observation is intended to refer to, whereas the latter represents the actual period of an observation. For example, the GDP (Gross domestic product) of a country can be introduced by calendar year, but the data may only be available by fiscal year, which does not necessarily start on January 1. In this context, we use *sdmx-dimension:refPeriod* to refer to calendar year, instead of using *sdmx-dimension:timePeriod* which refers to a fiscal year.

To consolidate different URIs for the temporal dimension, we chose *sdmx-dimension:refPeriod* because of two reasons. First, this concept provides good opportunities for data integration. Different countries can collect their GDP based on fiscal years which can be different from one country to another. Using the *sdmx-dimension:refPeriod* concept, we can still compare them based on the calendar year which they intend to refer to. Second, values of *sdmx-dimension:refPeriod* can be arbitrary text, e.g., winter semester, summer semester. This is not possible in *sdmx-dimension:timePeriod*, because its values have to be specific dates.

To describe the values of temporal dimensions, each data source follows a different approach. First, EODP utilizes its own URIs such as *http://data.lod2.eu/scoreboard/year/2007*, whereas EEA and VOGD use literal values, e.g., *2007-01-01^^http://www.w3.org/2001/XMLSchema#date*. Finally,

Algorithm 1 Geographical Area Mapping

Input: $L = \{l_1, \ldots, l_n\}$, $l_i = (uri\_l_i, label\_li)$
Output: Mapping L to G, $G = \{g_1, \ldots, g_n\}$,
$g_i = (uri\_g_i, label\_gi, lat\_g_i, lng\_gi, type\_gi)$
procedure GEOGRAPHICALAREAMAPPING(L)
  sortInAscendingOrder(L)
  for each area $l_i \in L$ **do**
    $k \leftarrow indexOfBroarderArea(L, l_i)$
    if k!=-1 **then**
      if *type_g_k is administrative-area* **then**
        $queryString \leftarrow label\_l_i + label\_l_k$
      else
        $queryString \leftarrow label\_li$
        $uri\_b \leftarrow uriOfBroaderArea(uri\_g_k)$
      end if
    else
      $queryString \leftarrow label\_li$
    end if
    $g_i \leftarrow queryGoogleAPI(queryString)$ ▷ $g_i = \{r_1, \ldots, r_m\}$, $r_j = (uri\_r_j, label\_r_j)$
    for each result $r_j \in g_i$ **do**
      if ($type\_g_k$ is administrative area and !isUriOfBroaderArea(uri_g_k, uri_r_j)) or ($type\_g_k$ is non-administrative area and !isUriOfBroaderArea(uri_b, uri_r_j) **then**
        remove r_j from g_i
      end if
    end for
    if size(g_i)>1 **then**
      removeByDistance(G, g_i)
    end if
    if size(g_i)=1 **then**
      set type_g_i is administrative area
    else
      uri_$g_i \leftarrow$ uri_g_k + label_l_i
      set type_g_i is non-administrative area
    end if
    if k!=-1 **then**
      set (uri_g_i, *sw:broader*, uri_g_k)
      set (uri_g_k, *sw:narrower*, uri_g_i)
    end if
  end for
end procedure
procedure SORTINASCENDINGORDER(L)
  ▷ *sort areas in L in ascending order of uri*
end procedure
procedure QUERYGOOGLEAPI(QUERYSTRING)
  ▷ *return query results of Google's Geocoding API*
end procedure
procedure REMOVEBYDISTANCE(G, g_i)
▷ *retain only one result in g_i, that is, the one which has the minimal distance to area(s) g_{i-1} and g_{i-2} (if i>=2)*
end procedure
procedure URIOFBROADERAREA(URI_g_k)
▷ *return uri of the area which is the broader area of the input uri*
end procedure
procedure ISURIOFBROADERAREA(URI_g_k,URI_g_j)
▷ *return true if uri_g_k is a broader area of uri_g_i, else return false*
end procedure
procedure INDEXOFBROADERAREA(L, l_i)
▷ *return index of the area which is a broader area of l_i in list L*
end procedure

EC makes use of a Gregorian URI set provided by the data.gov.uk time reference service. This service returns semantic descriptions for a wide range of temporal values. Adoption and use of this service as a common reference point for temporal values allows the data to be represented in a consistent way.

Our algorithm (Algorithm 2), which creates consolidated URIs for temporal values of each data source, receives URIs or literal values as an input for the analysis process. Then, by using time patterns, it identifies contained intervals, and relates them to the corresponding URI according to the Gregorian URI scheme. Furthermore, using the semantics provided by the service, we create *time:intervalContains* relationships between this URI and its subintervals in the metadata.

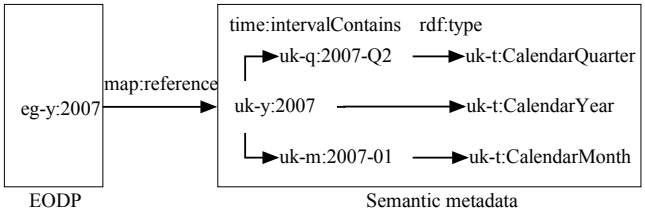

Figure 2: Example of Temporal Dimension Mapping[17]

2.5 Constructing Semantic Metadata

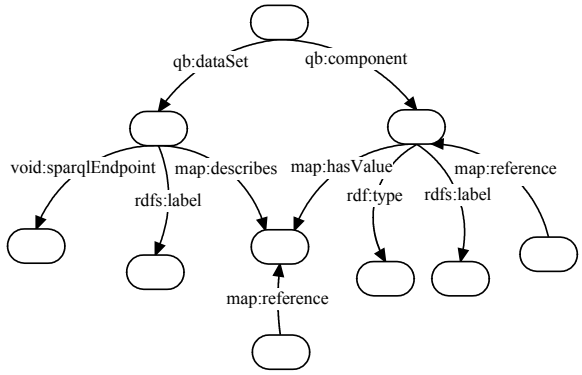

Figure 3: Structure of Semantic Metadata[18]

Figure 3 shows the structure of the metadata. It can be split into two main parts. The first part, represented by the *qb:dataSet* predicate, describes a data set's URI and label, as well as the endpoint containing it. The second part contains a list of components (i.e., *eg-p:country*, *eg-p:year*, *eg-p:value*, and *eg-p:unit*) in the data set represented by *qb:component* predicates. We describe the URI, label, and type of each component (e.g., dimension, measure, and attribute). In order to establish connections to other data sources, the metadata needs to contain links to equivalent concepts and values. In particular:

[17] `uk-y`: http://reference.data.gov.uk/id/gregorian-year/
`uk-q`: http://reference.data.gov.uk/id/gregorian-quarter/
`uk-m`: http://reference.data.gov.uk/id/gregorian-month/
`uk-t`: http://reference.data.gov.uk/def/intervals/

[18] `qb`: http://purl.org/linked-data/cube#
`void`: http://rdfs.org/ns/void#

Algorithm 2 Temporal Value Mapping

```
1: Input: T = {t_1, ..., t_n}
2: Output: Mapping T to U, U = {uri_1, ..., uri_n}
3: procedure TemporalValueMapping(T)
4:     uk      = "http://reference.data.gov.uk/id/"
5:     pYear   = "[1-9][0-9]{3}"
6:     pQuarter ="[1-9][0-9]{3}-Q[1-4]"
7:     pMonth  ="[1-9][0-9]{3}-[0-1][0-9]"
8:     pDate   ="[1-9][0-9]{3}-[0-1][0-9]-[0-3][0-9]"
9:     for each value t_i ∈ T do
10:        if pDate match t_i then
11:            v = getValue(pDate, t_i)
12:            uri_i = uk + "gregorian-date/"+v
13:        else
14:            if pQuarter match t_i then
15:                value = getValue(pQuarter, t_i)
16:                uri_i = uk + "gregorian-quarter/"+v
17:            else
18:                if pMonth match t_i then
19:                    value = getValue(pMonth, t_i)
20:                    uri_i = uk + "gregorian-month/"+v
21:                else
22:                    if pYear match t_i then
23:                        value = getValue(pYear, t_i)
24:                        uri_i = uk + "gregorian-year/"+v
25:                    end if
26:                end if
27:            end if
28:        end if
29:        if uri_i !=null then
30:            queryMeaning(uri_i)
31:        end if
32:    end for
33: end procedure
34: procedure getValue(P, t)
35: ▷ return contained interval in t through using pattern P
36: end procedure
37: procedure queryMeaning(uri)
38:    ▷ return semantics of uri through using time service
39: end procedure
```

- We use the *map:reference* predicate to link a component (i.e., a dimension, measure, or an attribute) to its reference concept. We do not use the *owl:sameAs* predicate to avoid potential contradictions. For example, the running example may declare *eg-p:country* which is a *subProperty* of *sdmx-dimension:refPeriod*. In this case, an *owl:sameAs* relationship between the two URIs would be inaccurate.

- To consolidate different URIs describing measure properties, we make use of the *sdmx-measure:obsValue* concept in SDMX. However, a data set may contain multiple measures, which is a well-known issue in statistical data [4]. To deal with this case, we use multiple metadata structures to model such a data set. Each structure mentions one measure of the data set.

- At present, metadata is still missing equivalent URIs for the attribute component, such as *eg-p:unit* as well as for other dimensions, such as sex or age.

- The metadata allows to represent values of dimensions

and attributes, and relates these values to equivalent values through *map:reference* predicates.

We use the SPARQL endpoint of a published data source as input for metadata generation and perform four steps: (i) identify all data sets of the source; (ii) identify all dimensions, measures, and attributes for each data set; (iii) identify all values for each dimension and attribute; (iv) identify equivalent entities for spatial and temporal dimensions. Steps i - iii are performed automatically by a data source analysis algorithm described in our previous work [7]. The last step uses the geographical area and temporal value mapping algorithms to automatically generate consolidated values.

3. USE CASES AND MASHUP APPROACH

In this section, we outline example use cases and illustrate the applicability of the developed integration approach within a semantic mashup platform.

3.1 Example use cases

The semantic metadata repository acts as a single point of access where users find all relevant statistical data sets that have been modeled. In this repository, values of spatial and temporal dimensions are consolidated, hence users are able to compare and integrate data based on geographic or temporal input. To illustrate our approach, we introduce two use cases:

- The first use case relates to multi-scale exploration of geographical areas. Users can provide an address, e.g., *Donaufelder Strasse 54, Austria*, or simply define a point on a map, for which we detect corresponding administrative areas, e.g., *Country: Austria*, *City: Vienna*, and *District: Floridsdorf* and use those to select relevant data sets.
- The second use case allows users to compare data of different areas. First, users choose one or multiple areas on a map, e.g., Germany and Belgium. Next, we are able to identify data sets that contain data of both countries.

3.2 Linked Widget Platform

To enable users to dynamically select and combine statistical data sets and synthesize desired information, we follow a visual programming paradigm implemented in the Linked Widget platform. This mashup platform is based on widgets and wiring, as described in detail by Trinh et al. [25]. Its key elements are so-called linked widgets, which represent an extension of standard web widgets backed by a semantic model that follows Linked Data principles. This model describes data input/output and metadata, such as data provenance and licensing terms, to facilitate widget discovery and automatic widget composition.

There are three types of widgets, i.e., data, process, and visualization widgets. The platform provides a graphical interface for creating data flows and composing on-the-fly applications by connecting widgets in arbitrary but controlled ways. Stakeholders can develop widgets independently and contribute widgets to the platform to extend its functionality.

3.3 Statistical Widget Collection

In the platform, widgets are grouped into widget collections. Each collection addresses a different problem domain. We developed a collection[19] for statistical data exploration based on spatial contexts. Each widget has at least one input, but only a single output at the most. The output of a widget can serve as input for another one. Our exemplary statistical widget collection consists of the following three widgets:

Spatial Entity Recognizer: This *data* widget receives an address text or a user-defined location as its input and uses the Google Geocoding API to obtain corresponding spatial entities at different levels, e.g., country level, or administrative area levels.

Spatial Data Locator: This *process* widget returns a list of data sets related to the input entities. It contains an option to filter the output data sets based on the label of a data set or based on the label of components of this data set.

Spatial Data Visualization: This *visualization* widget presents visualizations for one area or a couple of areas, if users want to compare different areas.

The format for data exchange between widgets is JSON-LD. We describe the data structures following the W3C RDF Data Cube vocabulary [4].

3.4 Example Mashups

Sample mashups created from the widgets are shown in Fig. 4: (i) discovery of statistical information on an area based on a user-provided location[20]; and (ii) comparison of pairs of areas[21].

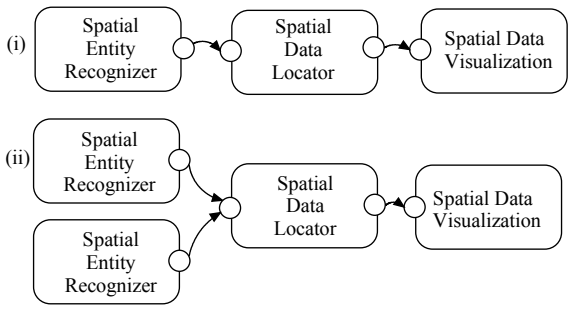

Figure 4: Sample mashup use cases

4. RELATED WORK

Related work in the area of data integration falls into two main categories: research on statistical data and open data research.

Within the former group, LD-Cubes[22] [12, 13, 14] follows a similar approach in that it aims to analyze and integrate distributed multidimensional data sets. To this end, users can create mappings from LOD data sources to multidimensional models in a data warehouse. Then, they can execute OLAP (Online Analytical Processing) operations to access,

[19]http://linkedwidgets.org/MashupPlatform.html?widgetCollectionId=SpatialStatisticalCollection
[20]http://linkedwidgets.org?id=MashupSpatialDataLocator
[21]http://linkedwidgets.org?id=MashupSpatialDataComparator
[22]http://www.linked-data-cubes.org/

analyse, and integrate data. To integrate data, however, *owl:sameAs* relationships need to be established manually beforehand.

The goal of the CODE project[23] is to establish an ecosystem that enables data enrichment, querying, and integration of LOD. To this end, a visualization wizard [18] for accessing, filtering, and visualizing statistical data was developed. Simultaneous visualizations of many data sets are made possible based on similar component and values of respective data sets.

The PlanetData project[24] supports organizations in exposing their data in new and useful ways. This support allows businesses, governments, communities, and individuals to take decisions in an informed manner. Sabou et al. [19, 20] publish European statistical tourism data as LOD and establish connections to other LOD data sources through manually established *owl:sameAs* relationships. This supports tourism managers in cross-domain decisions based on the comparison of indicators from various data sources. Corcho et al. [5, 15] introduce a tool kit for visualizing geospatial data sets available in SPARQL endpoints. When a point is selected on the map, name and category of this location are shown and completed with additional information such as a statistical visualizations, equivalent URIs from services like sameas.org, etc.

Capadisli et al. [3] provide a method for publishing data sources using the SDMX-ML standard (e.g., World Bank, Eurostat, the United Nations) as LOD. These LOD sources share the same concepts and values, which allows Capdisli et al. [2] to perform integration between many data sets.

Salas et al. [17] present a mediation architecture for describing and exploring statistical data which is exposed as RDF triples, but stored in relational databases. The authors construct a catalogue of *linked data cube descriptions*, that uses concepts of W3C RDF Data Cube vocabulary [4] to model each data set.

Salient characteristics that distinguish our work from these related efforts are as follows: (i) Whereas many existing approaches [18, 2, 3] typically aim to integrate data from sources that alreay use consistent terminology, we provide an environment that supports statistical integration of heterogeneous data sources. (ii) We introduce mechanisms to automatically interconnect statistical data sets rather than manually providing co-reference information [12, 14, 19, 20] or data set description [17]. (iii) Various research prototypes are capable of providing statistical data of a geographical area selected by a user [5, 15, 19, 20], but they are typically either limited to a single statistical data source [5, 15], or they present just "raw" data returned from multiple sources [19, 20]. In our approach, we construct relationships between different values of sources through *sw:narrower, sw:broader,* and *time:intervalContains* relationships.

In the open data area, there are also a number of contributions that aim for data integration. SPARQL-based crawling of co-reference information allows Glaser et al. [8] to create a co-reference resolution service. Following a similar approach, Schlegel et al. [22] build an on-the-fly query rewriting service. Compared to these approaches, we focus on matching values in disparate data sources to their equivalent URIs without using special predicates like *owl:sameAs, skos:exactMatch, etc.*

Second, defining mappings from source data to a common ontology provides a solid foundation for on-the-fly integration services, as has been shown by [9, 23]. However, their work is aimed towards Web APIs, which have different structures and requirements than those in the Linked Data domain. For instance, end users need to write small programs to specify what part of data should be fetched via rules, together with a query that returns the final results. Furthermore, users need to manually model their data sources of interest.

5. CONCLUSION AND FUTURE WORK

Based on Linked Data principles, which aim to facilitate connecting and reusing disparate data, we present a novel approach focusing on statistical data integration. For each data set, we model semantic metadata which relates components and values of dimensions to common identifiers. The metadata repository is published as LOD and provides an opportunity to build on-the-fly integration.

At present, we consolidate spatial and temporal dimensions from disparate statistical data sets. Providing equivalent identifiers for other components is still an open challenge that we need to solve in future. Furthermore, to build semantic metadata for statistical data that is available in non-RDF formats (e.g., CSV, XSL, JSON), we plan to use RML [6] to transform data to RDF format. We also plan to automatically create links to DBpedia for each metadata structure. This would ease the integration of statistical data with existing LOD repositories for developers. Finally, it will be necessary to evaluate precision and recall of geographical area mapping algorithm. This evaluation, however, requires data publishers to clearly describe geographical areas that their URIs refer.

Furthermore, we are currently developing a *Dataset Recommender* widget which accepts a single spatial entity as input and detects all spatial entities that share the same parent. When connecting its input with the output of the *Spatial Entity Recognizer* and its output with the input of *Spatial Data Visualization*, we have a new mashup enabling users to compare their area to appropriate, automatically identified, other areas.

References

[1] C. Bizer, T. Heath, and T. Berners-Lee. Linked data - the story so far. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 5(3):1–22, 2009.

[2] S. Capadisli, S. Auer, and R. Riedl. Linked statistical data analysis. In *1st International Workshop on Semantic Statistics (SemStats 2013)*, 2013.

[3] S. Capadisli, S. Auer, and R. Riedl. Linked sdmx data: Path to high fidelity statistical linked data. *Semantic Web*, 6(2):105–112, 2015.

[4] R. Cyganiak and D. Reynolds. The rdf data cube vocabulary, 2011.

[5] A. de León, F. Wisniewki, B. Villazóon-Terrazas, and O. Corcho. Map4rdf-faceted browser for geospatial datasets. In *Proceedings of Workshop on using open Data*, 2012.

[23] http://code-research.eu/

[24] http://www.planet-data.eu/

[6] A. Dimou, M. V. Sande, P. Colpaert, R. Verborgh, E. Mannens, and R. V. de Walle. Rml: A generic language for integrated rdf mappings of heterogeneous data. In *Proceedings of the Workshop on Linked Data on the Web (LDOW 2014)*. CEUR-WS.org, 2014.

[7] B.-L. Do, T.-D. Trinh, P. Wetz, A. Anjomshoaa, E. Kiesling, and A. M. Tjoa. Widget-based exploration of linked statistical data spaces. In *Proceedings of 3rd International Conference on Data Management Technologies and Applications (DATA 2014)*. SciTePress, 2014.

[8] H. Glaser, A. Jaffri, and I. C. Millard. Managing co-reference on the semantic web. In *Proceedings of the Workshop on Linked Data on the Web (LDOW 2009)*. CEUR-WS.org, 2009.

[9] A. Harth, C. A. Knoblock, S. Stadtmüller, R. Studer, and P. Szekely. On-the-fly integration of static and dynamic linked data. In *Proceedings of International Workshop on Consuming Linked Data (COLD 2013)*. CEUR-WS.org, 2013.

[10] J. Helmich, J. Klímek, and M. Necaský. Visualizing rdf data cubes using the linked data visualization model. In *The Semantic Web: ESWC 2014 Satellite Events*. Springer International Publishing, 2014.

[11] P. Hoefler, M. Granitzer, E. Veas, and C. Seifert. Linked data query wizard: A novel interface for accessing sparql endpoints. In *Proceedings of the Workshop on Linked Data on the Web (LDOW 2014)*. CEUR-WS.org, 2014.

[12] B. Kämpgen and A. Harth. Transforming statistical linked data for use in olap systems. In *Proceedings of the 7th International Conference on Semantic Systems*. ACM, 2011.

[13] B. Kämpgen and A. Harth. Olap4ld - a framework for building analysis applications over governmental statistics. In *The Semantic Web: ESWC 2014 Satellite Events*. Springer International Publishing, 2014.

[14] B. Kämpgen, S. Stadtmüller, and A. Harth. Querying the global cube: integration of multidimensional datasets from the web. In *Proceedings of 19th International Conference on Knowledge Engineering and Knowledge Management (EKAW 2014)*. Springer International Publishing, 2014.

[15] A. Llaves, O. Corcho, and A. Fernandez-Carrera. Map4rdf-ios: a tool for exploring linked geospatial data. In *Proceedings of Workshop on Linked Geospatial Data*, 2014.

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

[17] L. Ruback, S. Manso, P. E. R. Salas, M. Pesce, S. Ortiga, and M. A. Casanova. A mediator for statistical linked data. In *Proceedings of the 28th Annual ACM Symposium on Applied Computing*, pages 339–341, 2013.

[18] V. Sabol, G. Tschinkel, E. Veas, P. Hoefler, B. Mutlu, and M. Granitzer. Discovery and visual analysis of linked data for humans. In *The Semantic Web-ISWC 2014*, Lecture Notes in Computer Science, pages 309–324. Springer Berlin Heidelberg, 2014.

[19] M. Sabou, I. Arsal, and A. M. P. Brasoveanu. Tourmislod: A tourism linked data set. *Semantic Web*, 4(3):271–276, 2013.

[20] M. Sabou, A. M. P. Brasoveanu, and I. Önder. Linked data for cross-domain decision-making in tourism. In *Proceedings of ENTER2015 Conference*, 2015.

[21] P. E. R. Salas, F. M. D. Mota, M. Martin, S. Auer, K. Breitman, and M. A. Casanova. Publishing statistical data on the web. *International Journal of Semantic Computing*, 6(4):373–388, 2012.

[22] T. Schlegel, F. Stegmaier, S. Bayerl, M. Granitzer, and H. Kosch. Balloon fusion: Sparql rewriting based on unified co-reference information. In *5th International Workshop on Data Engineering Meets the Semantic Web*, pages 254–259. IEEE, 2014.

[23] S. Stadtmüller, S. Speiser, A. Harth, and R. Studer. Data-fu: a language and an interpreter for interaction with read/write linked data. In *Proceedings of the 22nd international conference on World Wide Web*. ACM, 2013.

[24] T.-D. Trinh, B.-L. Do, , P. Wetz, A. Anjomshoaa, and A. M. Tjoa. Linked widgets: An approach to exploit open government data. In *Proceedings of International Conference on Information Integration and Web-based Applications & Services (IIWAS '13)*. ACM, 2013.

[25] T.-D. Trinh, P. Wetz, B.-L. Do, A. Anjomshoaa, E. Kiesling, and A. M. Tjoa. Open linked widgets mashup platform. In *Proceedings of the AI Mashup Challenge 2014 ESWC Satellite Event*. CEUR-WS.org, 2014.