Creating and Utilizing Linked Open Statistical Data for the Development of Advanced Analytics Services

Evangelos Kalampokis[1,2], Areti Karamanou[1,2], Andriy Nikolov[3], Peter Haase[3],
Richard Cyganiak[4], Bill Roberts[5], Paul Hermans[6], Efthimios Tambouris[1,2],
Konstantinos Tarabanis[1,2]

[1] Centre for Research & Technology - Hellas, 6th km Xarilaou-Thermi, 57001, Greece
[2] University of Macedonia, Egnatia 156, 54006 Thessaloniki, Greece
{ekal, akarm, tambouris, kat}@uom.gr
[3] fluid Operations AG, Altrottstraße 31, 69190 Walldorf, Germany
{andriy.nikolov, peter.haase}@fluidops.com
[4] Insight Centre for Data Analytics, Galway, Ireland
{richard.cyganiak}@insight-centre.org
[5] Swirrl IT Limited, 20 Dale Street, Manchester, M1 1EZ, United Kingdom
{bill}@swirrl.com
[6] ProXML BVBA, Narcisweg 17, 3149 Keebergen, Belgium
{paul}@proxml.be

Abstract. A major part of Open Data concerns statistics such as population figures, economic and social indicators. The adoption of the Linked Data principles and technologies has promised to enhance the analysis of statistical data at a Web scale. Statistical data, however, is typically organized in data cubes where a numeric fact is categorized by dimensions. Both data cubes and Linked Data introduce complexity that raises the barrier for opening up and reusing statistical data. In this paper we describe the first release of the OpenCube toolkit that aims at supporting the whole lifecycle of linked data cubes. In particular, the OpenCube toolkit supports transforming raw data into RDF data cubes, attaching metadata, and providing query access to them. In addition, the toolkit enables linked data cube browsing and exploration as well as performing data analytics in an easy manner.

Keywords: Linked Data, statistics, data cube, multi-dimensional data, analytics, visualization, OLAP.

1 Introduction

Governments, organisations and companies are increasingly opening up their data for others to reuse. They launch data portals that operate as single points of access for data they produce or collect [1].

A major part of this Open Data concerns statistics such as population figures, economic and social indicators. For example, the vast majority of the datasets published on the open data portal[1] of the European Commission are of statistical nature. In addition, many international organizations provide statistical data about countries or regions. Major providers of statistics on the international level include Eurostat[2], World Bank[3], OECD[4] and CIA's World Factbook[5]. Analysis of statistical open data can provide value to both citizens and businesses in various areas such as business intelligence, epidemiological studies and evidence-based policy-making.

Recently, Linked Data emerged as a promising paradigm to enable the exploitation of the Web as a platform for data integration. As a result, Linked Data has been proposed as the most appropriate way for publishing open data on the Web. Statistical data needs to be formulated as cubes characterized by dimensions, slices and observations in order to unveil its full potential and value. Linked data cubes could open up new possibilities in performing data analytics at a Web scale (e.g. by integrating disparate datasets and extracting of interesting and previously hidden insights or even by incorporating learning models into the Linked Data Web) [2].

However, both Linked Data and data cubes introduce complexity that raises the barrier for opening up and reusing statistical data. Here, the RDF data cube vocabulary [3] that provides the fundamental background for modelling the data has been widely accepted and used (e.g. in [4]). As regards software components and tools, it was only recently that components and tools for publishing and reusing linked data cubes were developed e.g. [5-6]. These components and tools, however, present some limitations regarding (a) the functionalities they provide, (b) their licenses that hamper commercial exploitation, (c) their dependencies to specific platforms and environments, and (d) the capability to be used in complex scenarios in an integrated manner.

The objective of this paper is to present the OpenCube approach for working with linked open statistical data. It describes a set of software components that support different steps of the lifecycle of the linked statistical data in a holistic manner. The components can be either used as standalone tools to support specific steps of the lifecycle or integrated based on two platforms (i.e. fluidOps' Information Workbench and Swirrl's PublishMyData) to support complex scenarios.

The remaining of the paper is organized as follows. In section 2 we present tools for publishing and reusing of linked data cubes. In section 3 we present the OpenCube approach including the OpenCube lifecycle, the implementation alternatives, and the evaluation approach. In section 4 we present the OpenCube component and describe how they support the lifecycle. Finally, in section 5 we draw conclusions along with future work.

[1] http://open-data.europa.eu

[2] http://epp.eurostat.ec.europa.eu/portal/page/portal/statistics/themes

[3] http://data.worldbank.org

[4] http://www.oecd.org/statistics/

[5] https://www.cia.gov/library/publications/the-world-factbook/index.html

2 Related Work

Processing of linked statistical data has only become a popular research topic in the recent years and several practical solutions have been developed in this domain. The LOD2 Statistical Workbench[6] (SWB) brings together components developed in the LOD2 project by means of the OntoWiki[7] tool. Example tools of the LOD2 SWB include the CSV2DataCube [5], CubeViz [6] and the RDF Data Cube Validation tool [7]. However, LOD2 SWB presents some limitations:

- It is packaged as a set of Debian packages, which binds it to Linux environments. This does not facilitate making the developed tools approachable by as large an audience as possible.
- It introduces additional dependencies on OntoWiki (PHP-based), thus requiring installation of more components for users, and require extra development effort.
- Some LOD2 SWB components are licensed under restrictive licenses such as the GPL that make their use in a commercial environment extremely difficult.

Moreover, a mechanism for applying statistical models to distributed RDF data cubes is presented in [8] (the demo system[8] demonstrates applying regression models to time series from multiple data sources). Finally, Tablinker[9] is another component that enables creation of RDF data cubes from Excel files by manually annotating the spreadsheets.

In general in comparison with existing tools, OpenCube provides the following contributions:

- application development SDK allowing customized domain-specific applications to be built to support various use cases;
- new functionalities enabling users to better exploit linked data cubes;
- components supporting the whole lifecycle of linked statistical data in an integrated manner.

3 OpenCube Approach

The aim of the OpenCube project is the development of tools that would support the data user along the whole lifecycle of linked data cubes.

3.1 OpenCube Lifecycle

The OpenCube lifecycle describes a lifecycle of linked data cubes in terms of steps that raw data cubes should go through in order to create value by means of Linked

[6] http://wiki.lod2.eu/display/LOD2DOC/LOD2+Statistical+Workbench

[7] http://aksw.org/Projects/OntoWiki.html

[8] http://stats.270a.info

[9] https://github.com/Data2Semantics/TabLinker

Data technologies. The steps are categorized into two phases (a) the publish phase that includes creating linked data cubes out of raw data, and (b) the reuse phase that includes utilizing linked data cubes in advanced analytics and visualizations. In particular, the publish phase comprises the following steps:

- Discover & pre-process raw data: This step involves exploiting open data catalogues to discover raw data sets in formats such as CSV and XLS as well as processing raw data through e.g. filtering, sorting, cleansing etc.
- Define structure & create cube: The identified raw data sets are then transformed to RDF. Although the RDF Data Cube vocabulary is used to structure a data cube as an RDF graph, other Linked Data vocabularies can be also used to define the values of dimensions, measures and attributes of the cube. This introduces an extra requirement related to the management of controlled vocabularies that could be reused across different datasets.
- Annotate cube: This step refers to the enrichment of RDF data cubes with metadata to facilitate discovery and reuse. Sources of metadata include raw data files, the cube's structure and/or standard thesaurus of statistical concepts.
- Publish cube: At this step, the generated data cubes are made available to the public through different interfaces e.g. Linked Data API, SPARQL endpoint, downloadable dump etc. and is also publicized in data catalogues.

The reuse phase includes the following steps:

- Discover & explore cubes: At this step, the users that aim to consume data from data cubes exploit the mechanisms set up at the previous step in order to discover the appropriate data cubes for a task at hand. At this step we consider that the user is also able to browse the data in order to better understand the data cube and proceed with the following steps.
- Transform cube: At this step, the actual values of the observations and thus the whole data cube are transformed. This enables users to perform a number of more advanced operations (e.g. OLAP browsing) on top of the RDF data cubes.
- Analyze cube: In this step the data cubes that were resulted from the previous step are employed in order to compute simple summaries of the data or to produce learning or predictive models.
- Communicate results: This step involves the visualization of the results in order to communicate them. This step may feed back to the first step of the lifecycle as the results may call for discovering new raw data and performing a comparative analysis with existing data.

Based on the OpenCube lifecycle an architecture was developed that describes how the lifecycle can be implemented. For each step of the lifecycle five architecture layers were defined: (a) user interface, (b) data management, (c) infrastructure, (d) storage, and (e) model.

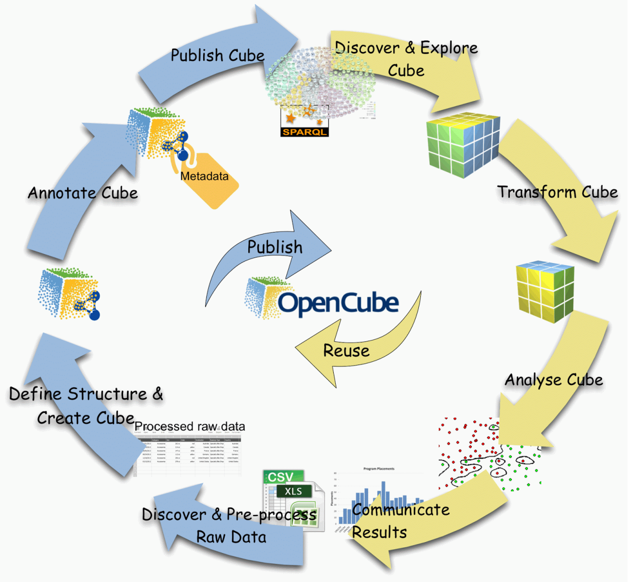


Fig. 1. The OpenCube Lifecycle

3.2 Implementation

Different steps of the lifecycle are realized by separate components. These components are integrated together by means of a common platform constituting a toolkit providing a single work environment to the user. Two different implementation approaches of this toolkit are considered based on the underlying platform. In particular OpenCube components have been included in two platforms i.e. Information Workbench and PublishMyData.

The Information Workbench (IWB) platform [9] serves as a backbone for the open source toolkit[10]. The components are integrated into a single architecture via standard interfaces provided by the IWB SDK: widgets (for UI controls) and data providers (for data importing and processing components). The overall UI design is based on the use of wiki-based templates providing dedicated views for RDF resources: an appropriate view template is applied to an RDF resource based on its type. All components of the architecture share the access to a common RDF repository (local or remote) and can retrieve data by means of SPARQL queries. Given the potentially large scale of data, which has to be processed, different data cubes can be stored in separate data repositories and queried using the SPARQL 1.1 federation capabilities.

PublishMyData is a Linked Open Data publishing platform provided as Software as a Service (SaaS). It incorporates tools for data publishers to create and manage RDF datasets in a triple store, and to provide a user friendly interface to consumers of

[10] http://opencube-toolkit.eu

the data, allowing them to navigate, search, browse and download data, as well as accessing the data programmatically through APIs and a SPARQL endpoint.

3.3 Evaluation approach

OpenCube tools aim at providing a user-friendly interaction environment and thus users feedback is of vital importance for the development of the tools. Although the evaluation of the tools has not been performed yet, in this sub-section we briefly describe the evaluation approach.

The OpenCube Toolkit will be evaluated in four pilots: (a) the Department for Communities and Local Government UK (DCLG), (b) the Central Statistics Office in Ireland (CSO), (c) the Research Unit of the Flemish government (SVR), and (d) the Global Macroeconomic Research Unit of a major Swiss bank.

DCLG's products typically comprise multiple outputs in spreadsheet and other formats: more than 4,000 documents in all. During the pilot a statistical dataset on local government finance will be transformed to linked data cube using Grafter. On the consuming side, OpenCube components deployed in PublishMyData platform will help users to answer important questions such as "for a given location, which organisation provides which service".

The CSO is the government body responsible for compiling the Irish official statistics. Their main datasets are the Census 2011 results and the datasets available via Statbank. During the first year of the pilot emphasis is put on the publishing of linked data cubes. CSO will use the TARQL and data cube R2RML extension modules of the open source version of the OpenCube Toolkit.

The research department of the Flemish government has 2 main publications: (a) de VRIND, flemish regional indicators, comprising 1200 spreadsheets published at the Flemish Open Data portal, and (b) local statistics published via a Cognos web interface. The main dimension of the datasets used is the administrative geographical dimension with four hierarchical levels. Hence, they expect to get a map view on which the different hierarchical levels can be visualized with the observations and measurements correctly aggregated on the respective hierarchical levels.

The Global Macroeconomic Research Unit of a Swiss bank focuses on investment advice and on provision of expertise to clients, with a special support for senior management. In particular, they try to forecast economic variables (GDP growth, inflation, short-term interest rates, etc.), central bank decisions, economic policy decisions with different time horizons ranging from monthly or quarterly over (bi-) yearly to longer-term predictions. These forecasts are used in investment decision-making, risk management and treasury amongst others.

4 OpenCube Tools

In this section, we discuss how different stages of the OpenCube lifecycle are supported by OpenCube components. Here we should note that this is the first release of OpenCube tools.

4.1 Creating Linked Data Cubes

At the publishing stage, the main focus is on supporting the user in transforming legacy data (such as CSV or relational databases) into RDF data cubes, attaching metadata allowing further search & discovery of relevant data, and providing query access to the them. To this end, the OpenCube tools include three software components: (a) the Grafter Extract Transform Load (ETL) framework, (b) the TARQL adaptation for data conversion, and (c) the D2RQ extension for data cubes.

Grafter is an ETL framework designed specifically to create RDF for linked data publishing purposes. It can handle a range of inputs, including of course the CSV and Excel files commonly encountered in statistical data.

There are many ETL tools in existence, but a review of existing tools found that none were a good match to the needs of OpenCube. The most important requirements were:

- the tool must support both automation and interactive use via a graphical user interface
- it must be capable of processing large datasets with good performance
- it must provide specific support for RDF Data Cube construction
- it must be able to deal with the range of input formats encountered in every day statistical data processing

For automation, we need reliability, ability to compose re-usable modules, quality and consistency in output, validation of input data, good data processing performance and a common framework for tool development.

For interactive use, we need the ability to react quickly to user actions even with large datasets, the ability to develop a processing pipeline interactively, but later run it in an automated way and clear error reporting, helping a user identify and fix any problem.

Important design choices for Grafter include:

- don't support loops and conditional branching in transformation pipelines, as this turns the DSL into a complete programming language, which becomes too complex to support effectively in a user interface. (If these features are required, simply use an existing programming language together with the DSL).
- lazy execution of pipelines - allowing a pipeline to be executed and tested on small samples of data, rather than the entire input dataset (which could be very large). This is important for testing, efficient use of system memory, and use of Grafter via a user interface.
- avoid intermediate file formats where possible: make it possible to direct the output of a Grafter pipeline straight into a triple store for example.
- flexible framework supporting many tools:
    - a pipeline builder user interface
    - an import service that can inspect a pipeline, create a data upload form for the required inputs, then execute the pipeline
    - automated execution of a pipeline
    - an API that can be called from other software

The *OpenCube TARQL* component enables cubes construction from legacy data via TARQL (Transformation SPARQL): a SPARQL-based data mapping language that enables conversion of data from RDF, CSV, TSV and JSON (and potentially XML and relational databases) to RDF. TARQL[11] is tool for converting CSV files to RDF using SPARQL 1.1 syntax. It is built on top of Apache ARQ[12]. The OpenCube TARQL component includes the new release of TARQL. It brings several improvements, such as: streaming capabilities, multiple query patterns in one mapping file, convenient functions for typical mapping activities, validation rules included in mapping file, increased flexibility (dealing with CSV variants like TSV).

The R2RML[13] language is a W3C standard for mappings from relational databases to RDF datasets. D2RQ[14] is a platform for accessing relational databases as virtual, read-only RDF graphs. *D2RQ Extensions for Data Cube* cover the functionality of importing of raw data as data cubes by mapping raw data to RDF. The process of mapping the data cube with a relational data source includes: (a) mapping the tables to classes of entities, (b) mapping selected columns into cube dimensions and cube measures, (c) mapping selected rows into observation values, and (d) generate triples with data structure definition. The user, by providing information about the dataset, such as the data dimensions and related measures, will receive an R2RML mapping file, which as a result will be used to generate and store the output.

4.2 Utilizing Linked Data Cubes

In order to support the steps of the reuse phase of the lifecycle a number of tools have been created. In particular, the first release of OpenCube tools include: (a) the Browser, (b) the Map View, (c) the Chart-based Visualization component, and (d) the Statistical Analysis component.

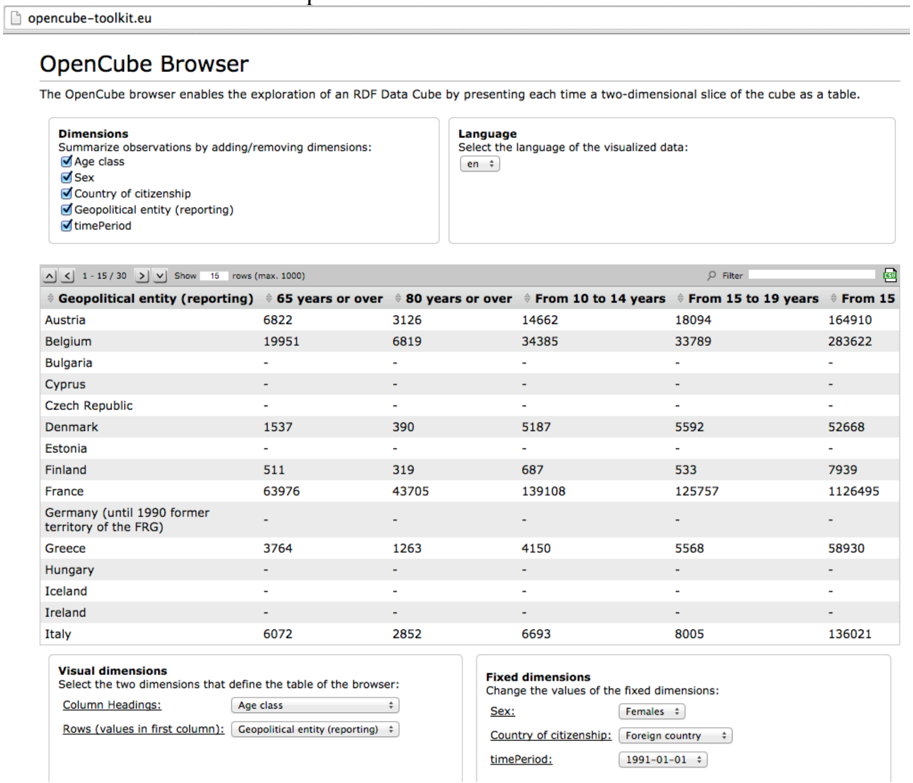
The *OpenCube Browser* enables the exploration of an RDF data cube by presenting two-dimensional slices of the cube as a table. Currently, the browser enables users to change the two dimensions that define the table of the browser and also change the values of the fixed dimensions and thus select a different slice to be presented. Moreover, the browser supports roll-up and drill-down OLAP operations through dimensions reduction and insertion respectively. The user can also create and store a two-dimensional slice (based on RDF data cube vocabulary) of the cube based on the data presented in the browser. Finally, the browser supports multilingual browsing based on multilingual *skos:labels* that are included in a dataset.

Initially, the browser sends a SPARQL query to retrieve the dimensions of the cube along with the values of the dimensions. The browser then selects two dimensions to present in the table and sets up a fixed value for all other dimensions (these can be latter changed by the user). Based on these it creates and sends a SPARQL query to the store to retrieve the appropriate data. We should also note that two approaches could be followed for extracting the values of the dimension properties: (a) from the observations that have the dimension property as a predicate and the value as an object, and (b) from coded lists. The performance of the browser is directly connected

[11] https://github.com/cygri/tarql
[12] http://jena.apache.org/documentation/query/
[13] http://www.w3.org/TR/r2rml/
[14] http://d2rq.org/

to the approach that will be followed to get the values of the dimension properties. In the case of getting the values from code lists the execution time ranges from 0.1 sec (for 3MB datasets) to 6 sec (for 75MB) while in the case of getting the values from the observations the execution time ranges from 1.4 sec (for 3MB dataset) to 29.5 sec (for 75MB dataset).

For the drill-down and roll-up operations the browser assumes that a set of data cubes has been created out of the initial cube by summarizing observations across one or more dimensions. The 2^n cubes (where n is the number of dimensions) created through this process along with the initial cube define an Aggregation Set. So, the user selects which of the available dimensions of the initial cube will be included in the browser and the browser presents the observations of a new cube that contains the selected dimensions and belongs to the same Aggregation Set with the initial one. Figure 2 depicts the OpenCube browser where the observations are presented along with the functionalities that are provided to the users.



opencube-toolkit.eu

OpenCube Browser

The OpenCube browser enables the exploration of an RDF Data Cube by presenting each time a two-dimensional slice of the cube as a table.

Dimensions
Summarize observations by adding/removing dimensions:
☑ Age class
☑ Sex
☑ Country of citizenship
☑ Geopolitical entity (reporting)
☑ timePeriod

Language
Select the language of the visualized data:
en

1 - 15 / 30 Show 15 rows (max. 1000) Filter

Geopolitical entity (reporting)	65 years or over	80 years or over	From 10 to 14 years	From 15 to 19 years	From 15
Austria	6822	3126	14662	18094	164910
Belgium	19951	6819	34385	33789	283622
Bulgaria	-	-	-	-	-
Cyprus	-	-	-	-	-
Czech Republic	-	-	-	-	-
Denmark	1537	390	5187	5592	52668
Estonia	-	-	-	-	-
Finland	511	319	687	533	7939
France	63976	43705	139108	125757	1126495
Germany (until 1990 former territory of the FRG)	-	-	-	-	-
Greece	3764	1263	4150	5568	58930
Hungary	-	-	-	-	-
Iceland	-	-	-	-	-
Ireland	-	-	-	-	-
Italy	6072	2852	6693	8005	136021

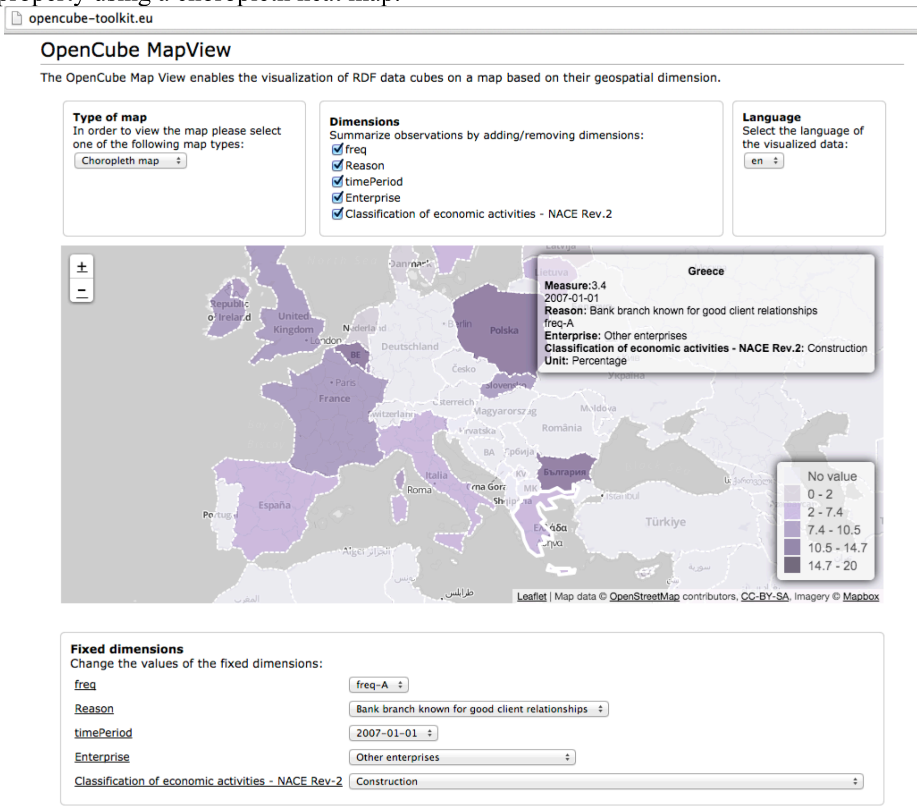
Visual dimensions
Select the two dimensions that define the table of the browser:
Column Headings: Age class
Rows (values in first column): Geopolitical entity (reporting)

Fixed dimensions
Change the values of the fixed dimensions:
Sex: Females
Country of citizenship: Foreign country
timePeriod: 1991-01-01

Fig. 2. The OpenCube Browser

The *OpenCube Map View* enables the visualization of RDF data cubes on a map based on their geospatial dimension. The Map View assumes that the geospatial dimension is defined by the or a sub-property of the sdmx-dimension:refArea dimension property.

Initially, Map View presents to the user the supported types of visualization (including markers, bubbles, choropleth and heat maps) along with all the dimensions and their values in drop-down lists. The user selects the type of visualization and a map appears that actually visualizes a one-dimension slice of the cube where the geospatial dimension is free and the other dimensions are randomly "fixed". In addition, the user can click on an area or marker or bubble and see the details of the specific observation i.e. the values of all dimension properties, the value of the measure property, and the values of attribute properties (if any). The maps are created using OpenStreetMap[15], Mapbox[16] street tile layer and Leaflet[17] open-source library. In Figure 3 a data cube is visualized on a map based on its geospatial dimension property using a choropleth heat map.



opencube-toolkit.eu

OpenCube MapView

The OpenCube Map View enables the visualization of RDF data cubes on a map based on their geospatial dimension.

Type of map
In order to view the map please select one of the following map types:
Choropleth map

Dimensions
Summarize observations by adding/removing dimensions:
☑ freq
☑ Reason
☑ timePeriod
☑ Enterprise
☑ Classification of economic activities - NACE Rev.2

Language
Select the language of the visualized data:
en

Greece
Measure:3.4
2007-01-01
Reason: Bank branch known for good client relationships
freq-A
Enterprise: Other enterprises
Classification of economic activities - NACE Rev.2: Construction
Unit: Percentage

	No value
	0 - 2
	2 - 7.4
	7.4 - 10.5
	10.5 - 14.7
	14.7 - 20

Leaflet | Map data © OpenStreetMap contributors, CC-BY-SA, Imagery © Mapbox

Fixed dimensions
Change the values of the fixed dimensions:

freq	freq-A
Reason	Bank branch known for good client relationships
timePeriod	2007-01-01
Enterprise	Other enterprises
Classification of economic activities - NACE Rev-2	Construction

Fig. 3. Visualization of a data cube that includes a geospatial dimension

To allow the user explore the data in a data cube, it is important that the used visualization controls are (i) interactive and (ii) adapted to the cube data

[15] http://wiki.openstreetmap.org/wiki/Develop
[16] http://www.mapbox.com
[17] http://leafletjs.com

representation. In this way the user can easily switch between different slices of the cube and compare between them. To this end, we implemented our *Chart-based Visualization* functionality. The charts can be inserted into a wiki page of an RDF resource and configured to show data cube slices. When viewing the page, the user can change the selection of dimension values to change the visualized cube slices. The SPARQL query to retrieve the appropriate data is constructed based on the slice definition, and the data is downloaded from the SPARQL endpoint dynamically.

When working with statistical data, a crucial requirement is the possibility to apply specialized analysis methods. One of the most popular environments for statistical data analysis is R, which defines a programming language and for which there exist a plethora of library packages implementing various statistical analysis methods. To use the capabilities of R inside the OpenCube Toolkit, we integrated it with our architecture through the *Statistical Analysis of RDF Data Cubes* component. R is run as a web service (using Rserve package) and accessed via HTTP. Input data are retrieved using SPARQL queries and passed to R together with an R script *provided by the user.* Then, R capabilities can be exploited in two modes: (i) as a widget (the script generates a chart, which is then shown on the wiki page) and (ii) as a data source (the script produces a data frame, which is then converted to RDF using defined R2RML mappings and stored in the data repository).

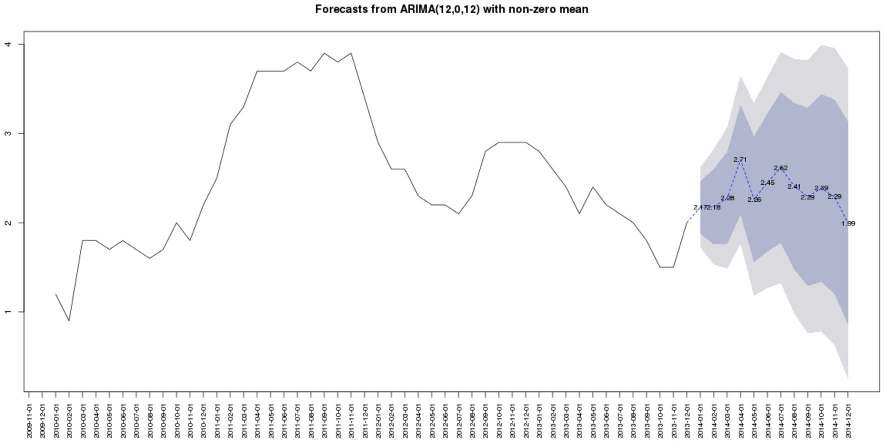


Fig. 4. Visualization of the forecasted inflation data with R

5 Conclusions

A major part of Open Data concerns statistics such as population figures, economic and social indicators. Accurate and reliable statistics provide the solid ground for performing analyses that would support businesses and governments to understand the world and make better decisions. The adoption of the Linked Data principles and technologies has promised to enhance the analysis of statistical data at a Web scale.

This paper presents the first version of the OpenCube Toolkit developed to enable easy publishing and reusing of linked data cubes. The toolkit smoothly integrates separate components dealing with different steps of the linked data cube lifecycle to provide the user with a rich set of functionalities for working with statistical semantic data. At the publishing phase, the main focus is on supporting the user in transforming legacy data (such as CSV or relational databases) into RDF data cubes, attaching metadata allowing further search & discovery of relevant data, and providing query access to the them. At the reusing phase of the lifecycle the toolkit enables linked data cubes browsing and exploration as well as performing data analytics on top of them in an easy manner.

Future work includes the evaluation of the developed tools in three pilots that involve public agencies working with statistical data in three countries i.e. the Department for Communities and Local Government in the UK, the Central Statistics Office in Ireland, the Research Unit of the Flemish government in Belgium, and the Global Macroeconomic Research Unit of one of the major banks in Switzerland. The feedback of the users will enable the improvement of the existing tools and also support the specification of the OpenCube tools that will be developed in the future.

Acknowledgments. The work presented in this paper was partially carried out in the course of the OpenCube[18] project, which is funded by the European Commission within the 7th Framework Programme under grand agreement No. 611667.

References

1. Kalampokis, E., Tambouris, E., Tarabanis, K.: A Classification Scheme for Open Government Data: Towards Linking Decentralized Data. International Journal of Web Engineering and Technology, 6(3), 266-285 (2011)
2. Kalampokis, E., Tambouris, E., Tarabanis, K.: Linked Open Government Data Analytics. In: Wimmer, M.A., Janssen, M., Scholl, H.J. (eds.) EGOV 2013. LNCS, vol. 8074, pp. 99-110. IFIP International Federation for Information Processing (2013)
3. Cyganiak, R., Reynolds, D.: The RDF Data Cube vocabulary, http://www.w3.org/TR/vocab-data-cube/ (2013)
4. Capadisli, S., Auer, S., Ngonga Ngomo, A.-C.: Linked SDMX Data: Path to high fidelity Statistical Linked Data for OECD, BFS, FAO, and ECB. Semantic Web (2013)
5. Salas, P. E. R., Martin, M., Mota, F. M. D., Auer, S., Breitman, K., Casanova, M.A.: Publishing Statistical Data on theWeb. In: IEEE Sixth International Conference on Semantic Computing (ICSC), pp. 285{292. IEEE Press, New York (2012)
6. Ermilov, I., Martin, M., Lehmann, J., Auer, S.: Linked Open Data Statistics: Collection and Exploitation. In: Klinov, P., Mouromtsev, D. (eds.) Knowledge Engineering and the Semantic Web, vol. 394, pp. 242-249. Springer Berlin Heidelberg (2013)
7. Janev, V., Mijovic, V., Vranes, S.: LOD2 Tool for Validating RDF Data Cube Models. ICT innovations 2013 Web Proceedings.
8. Capadisli, S. Towards Linked Statistical Data Analysis. SemStats 2013
9. Haase, P., Schmidt, M., Schwarte, A. The Information Workbench as a Self-Service platform for Linked Data Applications. COLD 2011, ISWC 2011, Shanghai, China (2011)

[18] http://www.opencube-project.eu