Representing statistical indexes as linked data including metadata about their computation process

Jose Emilio Labra Gayo[1], Hania Farham[2], Juan Castro Fernández[1], and Jose María Álvarez Rodríguez[3]

[1] WESO Research Group
{jelabra,juan.castro}@weso.es
[2] The Web Foundation
hania@webfoundation.org
[3] Dept. Computer Science
Carlos III University
josemaria.alvarez@uc3m.es

Abstract. In this paper we describe the development of the Web Index linked data portal that represents statistical index data and the computations from which it has been obtained.
The Web Index is a multi-dimensional measure of the World Wide Web's contribution to development and human rights globally. It covers 81 countries and incorporates indicators that assess several areas like universal access; freedom and openness; relevant content; and empowerment.
In order to empower the Web Index transparency, we established as an internal requirement that every published data could be externally verified. The verification could be that it was just raw data obtained from a secondary source, in which case, the system must provide a link to that data source or that the value has been internally computed, in which case, the system provides links to the values from which it has been calculated. The resulting portal contains data that can be tracked to its sources so an external agent can validate the whole index computation process.
We describe the different aspects involved in the development of the WebIndex data portal that also offers new linked data visualization tools. Although in this paper we concentrate on the Web Index development, this approach can be generalized to other projects which involve the publication of externally verifiable computations.

1 Introduction

Statistical indexes are a widely accepted practice that have been applied to numerous domains like economics and Bibliometrics (Impact factor), research and academic performance (H-Index or Shanghai rankings), cloud computing (Global Cloud Index, by CISCO), etc. Those indexes will benefit from a Linked Data approach where the rankings can be seen, tracked and verified by their users linking each rank to the original values and observations from which it has been computed.

As a motivating example, we will employ the Web Index project (http:// thewebindex.org), which created an index to measure the World Wide Web's contribution to development and human rights globally. Scores are given in the areas of access;

freedom and openness; relevant content; and empowerment. First released in 2012, the 2013 Index has been expanded and refined to include 20 new countries and features an enhanced data set, particularly in the areas of gender, Open Data, privacy rights and security.

The 2012 version offered a data portal[4] whose data was obtained by transforming raw observations and precomputed values from Excel sheets to RDF. The technical description of that process was described in [2] where we followed the methodology presented in [5].

In this paper, we describe the development of the 2013 version of that data portal, where we employ a new validation and computation approach that enables the publication of a verifiable linked data version of WebIndex results.

We defined a generic vocabulary of computational index structures called *Computex* which could be applied to compute and validate any other kind of index and can be seen as an specialization of the RDF Data Cube vocabulary [9].

Given that the most important part of a data portal about statistical indexes are the numeric values of each observation we established the internal requirement that any value published should be justified either declaring from where it had been obtained or linking it to the values of other observations from which it had been computed.

The validation process employs a combination of SPARQL [10] queries and Shape Expressions [3] to check the different integrity constraints and computation steps in a declarative way. The resulting data portal http://data.webfoundation.org/webindex/2013 contains not only a linked data view about the statistical data but also a machine verifiable justification of the index ranks.

In the rest of the paper we will use Turtle and SPARQL notation and assume that the namespaces have been declared using the most common prefixes found in http://prefix.cc.

2 WebIndex Computation Process

The Web Index is a composite measure that summarizes in a single (average) number the impact and value derived from the Web in various countries. There are serious challenges when attempting to measure and quantify some of the dimensions the Index covers (e.g. the social and political), and suitable proxies were used instead.

Two types of data were used in the construction of the Index: existing data from other data providers (*secondary data*), and new data gathered via a multi-country questionnaire (*primary data*) which was specifically designed by the Web Foundation and its advisers. These primary data will begin to fill in some of the gaps in measurement of the utility and impact of the Web in various countries.

As the Web Index covers a large number of countries, some of which have serious data deficiencies or were not covered by the data providers, some missing data had to be imputed.

The following steps summarise the computation process of the Index:

[4] http://data.webfoundation.org

1. Take the data for each indicator from the data source for the 81 countries covered by the Index for the 2007-2012 time period (or 2013, in the case of the Web Index expert assessment survey).
2. Impute missing data for every secondary indicator for the sample of 81 countries over the period 2007-2012. Broadly, the imputation of missing data was done using two methods: country-mean substitution if the missing number is in the middle year (e.g. have 2008 and 2010 but not 2009), or taking arithmetic growth rates on a year-by-year basis.
3. Normalise the full (imputed) dataset using z-scores, making sure that for all indicators, a high value is *good* and a low value is *bad*.
4. Cluster some of the variables, taking the average of the clustered indicators post-normalisation. For the clustered indicators, this clustered value is the one to be used in the computation of the Index components.
5. Compute the component scores using arithmetic means, using the clustered values where relevant.
6. Compute the min-max values for each z-score value of the components, as this is what will be shown in the visualisation tool and other publications containing the component values (generally, it is easier to understand a min-max number in the range of 0 – 100 rather than a standard deviation-based number). The formula for this is: $\frac{x-min}{max-min} \times 100$
7. Compute sub-index scores by calculating the weighted averages of the relevant components for each sub-Index and the min-max values for each z-score value of the sub-Indexes.
8. Compute overall composite scores by calculating the weighted average of the sub-indexes and the min-max values.

The computation process was originally done by human experts using an Excel file although once the process was established, the computation was automated to validate the whole process.

3 WebIndex workflow

The WebIndex workflow has been depicted in figure 1. The Excel file was comprised of 184 Excel sheets and contained a combination of raw, imputed and normalized data created by the statistical experts.

That external data was filtered and converted to RDF by means of an specialized web service called *wiFetcher*[5].

Although some of the imported values had been pre-computed in Excel by human experts, we collected only the raw values, so we could automatically compute and validate the results.

In this way, another application called *wiCompute*[6] took the raw values and computed the index following the computation steps defined by the experts. *wiCompute*

[5] https://github.com/weso/wiFetcher
[6] https://github.com/weso/wiCompute

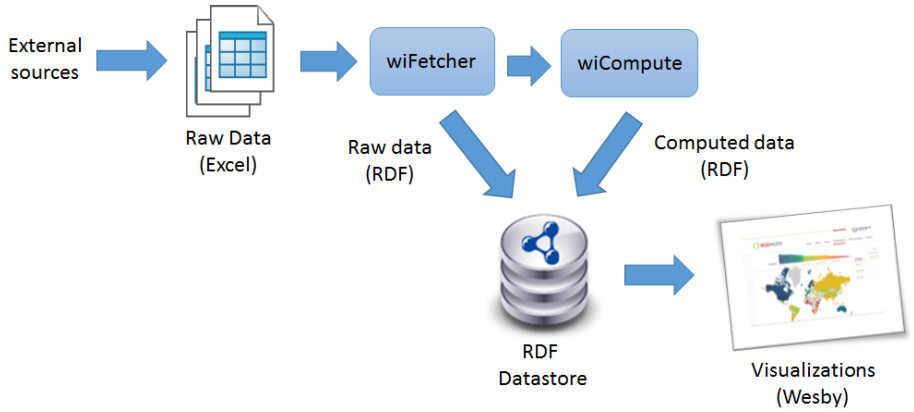

Fig. 1. Web Index data portal WorkFlow

carried out the computations generating RDF datasets for the intermediary results and linking the generated values to the values from which they had been computed.

Finally, the RDF data generated was published to a SPARQL endpoint from which we created a specialized visualization tool called *Wesby*[7].

4 WebIndex data model

Given the statistical nature of the data, the WebIndex data model is based on the RDF Data Cube vocabulary. Figure 4 represents the main concepts of the data model

As can be seen, the main concept are observations of type `qb:Observation`, which can be raw observations, obtained from an external source, or computed observations derived from other observations. Each observation has a float value `cex:value` and is related to a country, a year, a dataset and an indicator.

A dataset contains a number of slices, each of which also contains a number of observations.

Indicators are provided by an organization of type `org:Organization` from the Organization ontology[18]. Datasets are also published by organizations.

As a sample of some data, an observation can be that Italy has -0.80 as the normalized value using Z-Scores in 2007 for the indicator WEF_L (*Impact of ICT on organizational models*) provided by the World Economic Forum. This information can be represented in RDF using Turtle syntax as[8]:

```
obs:computed_26549 a qb:Observation ;
```

[7] https://github.com/weso/wesby

[8] The real observation is http://data.webfoundation.org/webindex/v2013/observation/computed_2007_1386752461095_26549. The real URIs also include an internal long number used to uniquely identify each entity

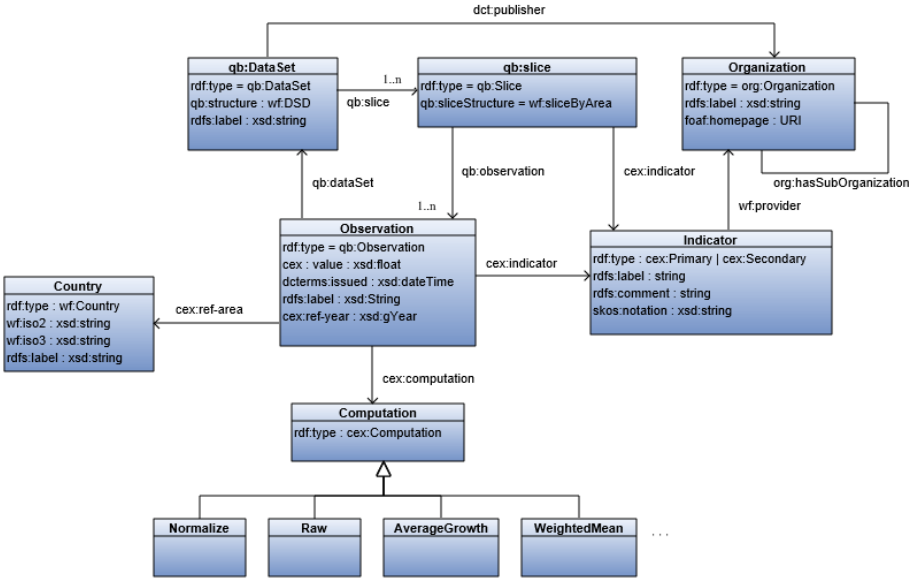

Fig. 2. WebIndex data model

```
 cex:indicator indicator:WEF_L ;
 qb:dataSet dataset:d_52 ;
 cex:value "-0.80"^^xsd:double ;
 cex:ref-area country:Italy ;
 cex:ref-year 2007 ;
 sdmx-concept:obsStatus cex:Normalized ;
 cex:computation computation:c26550
 ...other properties omitted for brevity
 .
```

Notice that the WebIndex data model contains data that is completely interrelated. Observations are linked to indicators, datasets and computations. Datasets contain also links to slices and slices have links to indicators and observations again. Both datasets and indicators are linked to the organizations that publish or provide them.

The following example contains a sample of interrelated data for this domain.

```
dataset:d_52 a qb:DataSet ;
 qb:structure wf:DSD ;
 qb:slice slice:computed_54 , slice:computed_55, ...
 ...
slice:computed_54 a qb:Slice ;
 qb:sliceStructure wf:sliceByArea ;
 qb:observation obs:computed_26549, obs:computed_26941, ... ;
 ...
indicator:WEF_L a cex:SecondaryIndicator ;
```

```
 rdfs:label "Impact of ICT on organizational models"@en ;
 wf:provider-link org:WEF ;
 ...
org:WEF a org:Organization ;
 rdfs:label "World Economic Forum" ;
 foaf:homepage <http://www.weforum.org/>
 .
country:ITA a wf:Country ;
 wf:iso2 "IT" ; wf:iso2 "ITA" ;
 rdfs:label "Italy" .
...
computation:c26550 a cex:Normalize ;
 cex:slice slice:WEF_L2007-Imputed ;
 cex:stdDesv "0.75"^^xsd:double ;
 cex:mean "4.39"^^xsd:double ;
 cex:observation obs:obs29761
 .
```

Computed observations and datasets contain a property `cex:computation` that associate them to a node of type `cex:Computation` which links the computed observation to the observations from which it has been obtained. In the above example, the computation `c26550` indicates that it is a normalization of the observation `obs:obs29761` using the observations in slice `slice:WEF_L2007-Imputed` which has a standard deviation of 0.75 and a mean of 4.39. Including these declarations, an external agent can verify if the value of the observation has been well computed or if it has been tampered. We also noticed that these declarations had another positive effect to debug the computation process in the development phase of the data portal.

5 Computex vocabulary

The *Computex*[9] vocabulary defines terms related to the computation of statistical index data and can be seen as a specialization of the RDF Data Cube vocabulary for this kind of statistical computations. Some terms defined in the vocabulary are:

- **`cex:Concept`** represents the entities that we are indexing. In the case of the Web Index project, the concepts are the different countries. In other applications it could be Universities, journals, services, etc.
- **`cex:Indicator`**. A dimension whose values add information to the Index. Indicators can be simple dimensions, for example: the mobile phone suscriptions per 100 population, or can be composed from other indicators.
- **`cex:Computation`**. It represents a computation. We included the main computation types that we needed for the WebIndex project, which have been summarized in Table 1. That list of computation types is non-exhaustive and can be further extended in the future.

[9] http://purl.org/weso/ontology/computex

Computation	Description	Properties
Raw	No computation. Raw value obtained from external source.	
Mean	Mean of a set of observations	`cex:observation` `cex:slice`
Increment	Increment an observation by a given amount	`cex:observation` `cex:amount`
Copy	A copy of another observation	`cex:observation`
Z-score	A normalization of an observation using the values from a Slice.	`cex:observation` `cex:slice`
Ranking	Position in the ranking of a slice of observations.	`cex:observation` `cex:slice`
AverageGrowth	Expected average growth of N observations	`cex:observations`[10]

WeightedMean	Weighted mean of an observation	`cex:observation`
		`cex:slice`
		`cex:weightSchema`

Table 1. Some types of statistical computations

- **`cex:WeightSchema`** a weight schema for a list of indicators. It consists of a weight associated for each indicator which can be used to compute an aggregated observation.

6 Development and Validation approach

The validation approach employed in the 2012 WebIndex project was based on ad-hoc resource templates and a MD5 checksum field. Apart from that, we did not verify that the precomputed values imported from the Excel sheets really matched the value that could be obtained by following the declared computation process.

In the 2013 version, we did a step forward on the validation approach. The goal was not only to check that a resource contained a given set of fields and values, but also that those values really matched the values that can be obtained by following the declared computations.

The proposed approach was inspired by the integrity constraint specification proposed by the RDF Data Cube vocabulary, which employs a set of SPARQL `ASK` queries to check the integrity of RDF Data Cube data. Although `ASK` queries provide a good means to check integrity, in practice their boolean nature does not offer too much help when a dataset does not accomplish with the data model.

We decided to use `CONSTRUCT` queries which, in case of error, contain an error message and a list of error parameters that can help to spot the problematic data.

We transformed the `ASK` queries defined in the RDF Data Cube specification to `CONSTRUCT` queries. In order to make our error messages compatible with EARL [1],

[10] This is in plural because the value of this property is an ordered list of observations

we have defined `cex:Error` as a subclass of `earl:TestResult` and declared it to have the value `earl:failed` for the property `earl:outcome`.

We have also created our own set of SPARQL `CONSTRUCT` queries to validate the *Computex* vocabulary terms, specially the computation of index data. For example, the following query validates whether every observation has at most one value.

```
CONSTRUCT { [ a cex:Error ; cex:errorParam # ... omitted
           cex:msg "Observation has two different values" . ]
} WHERE { ?obs a qb:Observation .
 ?obs cex:value ?value1 . ?obs cex:value ?value2 .
 FILTER ( ?value1 != ?value2 ) }
```

Using this approach, it is possible to define more expressive validations. For example, we are able to validate whether an observation has been obtained as the mean of other observations.

```
CONSTRUCT { [ a cex:Error ; cex:errorParam # ...omitted
  cex:msg "Mean value does not match" ] .
} WHERE { ?obs a qb:Observation ;
        cex:computation ?comp ;
        cex:value ?val .
        ?comp a cex:Mean .
  { SELECT (AVG(?value) as ?mean) ?comp WHERE {
    ?comp cex:observation ?obs1 .
       ?obs1 cex:value ?value ;
    } GROUP BY ?comp }
 FILTER( abs(?mean - ?val) > 0.0001) }
```

Validating statistical computations using SPARQL queries offered a good exercise to check SPARQL expressiveness. Although we were able to express most of the computation types, some of them had to employ functions that were not part of SPARQL 1.1 or had to be defined in a limited way. We described these limits in [13].

We implemented an online validation tool called *Computex*[11] which takes as input an RDF graph and checks if it follows the integrity constraints defined by Computex. The validation tool can also check if the RDF graph follows the RDF Data Cube integrity constraints and it can also do the index computation for RDF Graphs. Although this declarative approach was very elegant, computing the webindex using only SPARQL queries was not practical (it took around 15 minutes for a small subset), so the computation process was finally done by a specialized program implemented in Scala [12].

7 Visualizing the data portal

We developed a visualization tool called Wesby [13] which takes as input an SPARQL endpoint and offers a linked data browsing experience. Wesby was inspired by

[11] http://computex.herokuapp.com/

[12] Source code is available here: https://github.com/weso/wiCompute

[13] http://wesby.weso.es

Pubby [8] and was developed in Scala using the Play! Framework. Wesby combines the visualization with a set of templates to offer specialized views for different types of resources. For example, figure 3 contains the WebIndex visualization of Italy[14]. The interactive visualization graphics use a javascript library called WesCountry that we have also developed [15].

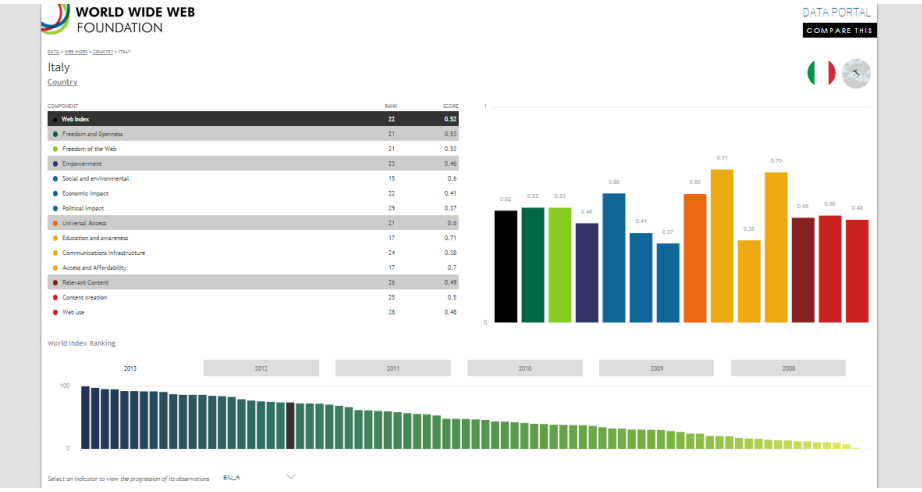

Fig. 3. Web Index visualization of country Italy

When there is no template for a given type of node, Wesby shows a table of properties and values similar to Pubby. Wesby also handles content negotiation so it can return different representations depending on the `ACCEPT` header.

In order to document the resulting data portal we created a set of templates using Shape Expressions [16]. We consider that this approach offers a good balance between human readability and machine processable specification.

8 Related work

There is a growing interest in developing solutions to improve the quality of linked data [11, 15, 12]. We consider that it is very important to publish linked data that is not only of high quality, but also that can automatically be validated. Validating RDF has also attracted a number of approaches. Most of them were presented at the W3c Workshop on RDF Validation [17] and can be classified as inference based, SPARQL queries or grammar based.

[14] It can be seen here: http://data.webfoundation.org/webindex/v2013/country/ITA
[15] http://weso.github.io/wesCountry/
[16] http://weso.github.io/wiDoc/

Inference based approaches try to adapt OWL for validation proposes. However, the use of Open World and Non-unique name assumption limits the validation possibilities. A variation of OWL semantics using Closed World Assumption to express integrity constraints has been proposed in [6, 20, 16]. SPARQL queries can also express validation constraints and offer a great level of expressiveness [13]. Grammar based approaches like OSLC Resource Shapes [19] and Dublic Core Application Profiles [7] define a domain specific language to declare the validation rules. Recently, Shape Expressions [3] have been proposed as a new technology to describe and validate RDF data portals.

Representing statistical linked data has also seen an increasing interest. SDMX [17] is the primary format of the main statistical data organizations. The transformation of SDMX-ML to RDF/XML has been described in [4]. The RDF Data Cube vocabulary [9] has been accepted as a W3c Recommendation technology to publish multidimensional statistical data and to link it with other concepts and data. We have opted to follow the RDF Data Cube vocabulary and in fact, we consider that Computex can be seen as a further specialization of RDF Data Cube to represent statistical index computations.

Another line of related work is the representation of mathematical expressions as linked data. Lange [14] gives an overview of the different approaches. OpenMath was proposed as an extensible standard that can represent the semantic meaning of mathematical objects. Wenzel and Reinhardt [21] propose an approach to integrate OpenMath with RDF data for the representation of mathematical relationships and the integration of mathematical computations into reasoning systems. We consider *Computex* as a first step in that direction to represent statistical computations and we expect more future work to appear about how to represent statistical computations as linked data.

9 Conclusions

In this paper, we described how we were able to represent statistical index computations as linked data which include information to track the origin of any published observation. Although the number of triples were around 3,5 million, we consider that the data portal is of medium size, so we were able to play with different validation possibilities.

Although we have been able to express most of the computations using SPARQL queries, we have found some limitations in current SPARQL 1.1 expressiveness with regards to built-in functions on maths, strings, RDF Collections and performance. In fact, although we initially wanted to do the whole computation process using SPARQL CONSTRUCT queries, we found that it took longer than expected and was difficult to debug, so we opted to develop an independent program that did all the computation process in a few seconds.

After participating in the W3c RDF Validation workshop we were attracted by the Shape Expressions formalism so we developed the documentation of the WebIndex data portal using Shape Expressions. We consider that some structural parts of the data portal can be better expressed in Shape Expressions.

[17] http://sdmx.org/

Our future work is to automate the declarative computation of index data from the raw observations and to check the performance using the Web Index data. We are also improving the Wesby visualization tool and the WesCountry library for statistical graphics. We are even considering to relate visualization templates with Shape Expressions offering a better separation of concerns in the development process.

10 Acknowledgements

We would like to thank Jules Clements, Karin Alexander, César Luis Alvargonzález, Ignacio Fuertes Bernardo and Alejandro Montes for their collaboration in the development of the WebIndex project.

Bibliography

[1] S. Abou-Zahra. Evaluation and Report Language EARL 1.0 schema. http://www.w3.org/TR/EARL10-Schema/, 2011. W3C Working Draft.

[2] J. M. Alvarez Rodríguez, J. Clement, J. E. Labra Gayo, H. Farhan, and P. Ordoñez. *Cases on Open-Linked Data and Semantic Web Applications*, chapter Publishing Statistical Data following the Linked Open Data Principles: The Web Index Project., pages 199–226. IGI Global, 2013. doi:10.4018/978-1-4666-2827-4.ch011.

[3] I. Boneva, J. E. Labra, S. Hym, E. G. Prud'hommeau, H. Solbrig, and S. Staworko. Validating RDF with Shape Expressions. *ArXiv e-prints*, Apr. 2014.

[4] S. Capadisli, S. Auer, and A.-C. Ngonga Ngomo. Linked sdmx data. *Semantic Web Journal*, pages 1–8, 2013.

[5] F. A. Cifuentes Silva, C. Sifaqui, and J. E. Labra Gayo. Towards an architecture and adoption process for linked data technologies in open government contexts: a case study for the library of congress of chile. In C. Ghidini, A.-C. N. Ngomo, S. N. Lindstaedt, and T. Pellegrini, editors, *I-SEMANTICS*, ACM International Conference Proceeding Series, pages 79–86. ACM, 2011.

[6] K. Clark and E. Sirin. On RDF validation, stardog ICV, and assorted remarks. In *RDF Validation Workshop. Practical Assurances for Quality RDF Data*, Cambridge, Ma, Boston, September 2013. W3c, http://www.w3.org/2012/12/rdf-val.

[7] K. Coyle and T. Baker. Dublin core application profiles. separating validation from semantics. In *RDF Validation Workshop. Practical Assurances for Quality RDF Data*, Cambridge, Ma, Boston, September 2013. W3c, http://www.w3.org/2012/12/rdf-val.

[8] R. Cyganiak and C. Bizer. Pubby: A linked data frontend for sparql endpoints. http://www4.wiwiss.fu-berlin.de/pubby/.

[9] R. Cyganiak and D. Reynolds. The RDF Data Cube Vocabulary. http://www.w3.org/TR/vocab-data-cube/, 2013. W3c Candidate Recommendation.

[10] S. Harris and A. Seaborne. SPARQL 1.1 Query Language. http://www.w3.org/TR/sparql11-query/, 2013.

[11] A. Hogan, A. Harth, A. Passant, S. Decker, and A. Polleres. Weaving the pedantic web. In *Linked Data on the Web Workshop (LDOW2010) at WWW'2010*, volume 628, pages 30–34. CEUR Workshop Proceedings, 2010.

[12] D. Kontokostas, P. Westphal, S. Auer, S. Hellmann, J. Lehmann, R. Cornelissen, and A. Zaveri. Test-driven evaluation of linked data quality. In *Proceedings of the 23rd International Conference on World Wide Web*, WWW '14, pages 747–758, Republic and Canton of Geneva, Switzerland, 2014. International World Wide Web Conferences Steering Committee.

[13] J. E. Labra and J. M. Alvarez Rodríguez. Validating statistical index data represented in RDF using SPARQL queries. In *RDF Validation Workshop. Practical Assurances for Quality RDF Data*, Cambridge, Ma, Boston, September 2013. W3c, http://www.w3.org/2012/12/rdf-val.

[14] C. Lange. Ontologies and languages for representing mathematical knowledge on the semantic web. *Semantic Web*, 4(2):119–158, 2013.

[15] P. N. Mendes, H. Mühleisen, and C. Bizer. Sieve: Linked data quality assessment and fusion. In *Proceedings of the 2012 Joint EDBT/ICDT Workshops*, EDBT-ICDT '12, pages 116–123, New York, NY, USA, 2012. ACM.

[16] B. Motik, I. Horrocks, and U. Sattler. Adding Integrity Constraints to OWL. In C. Golbreich, A. Kalyanpur, and B. Parsia, editors, *OWL: Experiences and Directions 2007 (OWLED 2007)*, Innsbruck, Austria, June 6–7 2007.

[17] RDF Working Group W3c. W3c validation workshop. practical assurances for quality rdf data, September 2013.

[18] D. Reynolds. The Organization Ontology. http://www.w3.org/TR/vocab-org/, 2014.

[19] A. G. Ryman, A. L. Hors, and S. Speicher. OSLC resource shape: A language for defining constraints on linked data. In C. Bizer, T. Heath, T. Berners-Lee, M. Hausenblas, and S. Auer, editors, *Linked data on the Web*, volume 996 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2013.

[20] J. Tao, E. Sirin, J. Bao, and D. L. McGuinness. Integrity constraints in OWL. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence (AAAI-10)*. AAAI, 2010.

[21] K. Wenzel and H. Reinhardt. Mathematical computations for linked data applications with openmath. In *Conferences on Intelligent Computer Mathematics, CICM 2012*, 2012.