daQ, an Ontology for Dataset Quality Information

Jeremy Debattista
University of Bonn /
Fraunhofer IAIS, Germany
name.surname@iais-extern.fraunhofer.de

Christoph Lange
University of Bonn /
Fraunhofer IAIS, Germany
math.semantic.web@gmail.com

Sören Auer
University of Bonn /
Fraunhofer IAIS, Germany
auer@cs.uni-bonn.de

ABSTRACT

Data quality is commonly defined as *fitness for use*. The problem of identifying the quality of data is faced by many data consumers. To make the task of finding good quality datasets more efficient, we introduce the Dataset Quality Ontology (daQ). The daQ is a light-weight, extensible vocabulary for attaching the results of quality benchmarking of a linked open dataset to that dataset. We discuss the design considerations, give examples for extending daQ by custom quality metrics, and present use cases such as browsing datasets by quality. We also discuss how tools can use the daQ to enable consumers find the right dataset for use.

Categories and Subject Descriptors

The Web of Data [**Vocabularies, taxonomies and schemas for the web of data**]

General Terms

Documentation, Measurement, Quality, Ontology

1. INTRODUCTION

The Linked (Open) Data principles of using HTTP URIs to represent things have facilitated the publication of interlinked data on the Web, and their sharing between different sources. The commonly used Resource Description Framework (RDF) provides both publishers and consumers of Linked Data with a standardised way of representing data. A substantial amount of facts has already been published as RDF Linked Open Data.[1] These facts have been extracted from heterogeneous sources, which also include semi-structured data, unstructured data, documents in markup languages such as XML, and relational databases. The use of such a variety of sources could lead to problems such as inconsistencies and incomplete information. A common open data user's perception[2] is that the *five star scheme* for open data[3] would automatically approve a dataset's quality. This statement is incorrect as the five star scheme serves as a guide in order to lead data to reach increasing levels of interlinkage, openness and standardisation. Therefore, although it is favourable to have a five star linked open dataset, dataset *quality* issues might still be unclear to data publishers. Various works promote quality measurements on linked open data [5, 8, 13]. Zaveri et al. [15] goes a step further by providing a systematic literature review.

To put the reader into the context of this work, we introduce a use case:

> Bob is a medical doctor and a computer enthusiast. During his free time he is currently working on a mobile application that would help colleagues to find out possible medicines to treat patients. Currently he is experimenting with a popular data management platform, hoping to find a suitable medical dataset for reuse. Fascinated by the views, especially the faceted filtering techniques available on this platform, Bob is particularly interested in reputable medical datasets. As he downloaded some datasets and viewed them in a visualisation tool, he found out that most of the data is either irrelevant for his work or contains many incorrect and inconsistent facts.

Data quality is commonly defined as *fitness for use* [14]. The problem of identifying the quality of data is faced by many data consumers. A simple approach would be to rate the "fitness" of a dataset under consideration by computing a set of defined quality metrics. On big datasets, this computation is time consuming, even more so when multiple datasets are to be filtered or compared to each other by quality. Apart from this, the identification of the quality of a dataset cannot be reused; other data consumers would have to do this process all over again.

To make the task of finding good quality datasets more efficient, we introduce the Dataset Quality Ontology (daQ). The daQ is a light-weight ontology that allows datasets to be "stamped" with its quality measures. In contrast to related vocabularies that represent quality *requirements* (cf. Section 5), our ontology allows for expressing concrete, tangible values that represent the quality of the data. Having this metadata available in the datasets enables data publishers and consumers to automatically perform tedious tasks such as filtering and comparing dataset quality. With the Dataset Quality Ontology we aim to add another star to the LOD *five star* scheme, for data that is not just linked and open, but of a high quality.

1.1 Terminology

To prepare the reader for the discussions carried out within the

[1] `http://lod-cloud.net`

[2] This is following a discussion with some Open Data enthusiasts at the University of Malta.

[3] `http://5stardata.info`

LDOW2014, April 8, 2014, Seoul, Korea.

following sections, we define some terminology, paraphrasing definitions by Zaveri et al. [15]:

- A **Quality Dimension** is a characteristic of a dataset relevant to the consumer (e.g. Availability of a dataset).
- A **Quality Metric** is a procedure for measuring a data quality dimension, which is abstract, by observing a concrete quality *indicator*. This assessment procedure returns a *score*, which we also call the *value* of the metric. There are usually multiple metrics per dimension; e.g., availability can be indicated by the accessibility of a SPARQL endpoint, or of an RDF dump. The value of a metric can be numeric (e.g., for the metric "human-readable labelling of classes, properties and entities", the percentage of entities having an *rdfs:label* or *rdfs:comment*) or boolean (e.g. whether or not a SPARQL endpoint is accessible).
- A **Quality Category** is a group of quality dimensions in which a common type of information is used as quality indicator (e.g. Accessibility, which comprises not only availability but also dimensions such as security or performance). Grouping the dimensions into categories helps to arrange a clearer breakdown of all quality aspects, given their large number. Zaveri et al. have identified 23 quality dimensions (with almost 100 metrics) and grouped them into 6 categories [15].

Whenever it is necessary to subsume all of these three concepts, we will use the term **Quality Protocols**.

1.2 Structure of this Paper

The remainder of this paper is structured as follows: in Section 2 we discuss use cases for the daQ vocabulary. Then, in Section 3 and 4 we discuss the vocabulary design and give examples of how this vocabulary can be extended and used. Finally, in Section 5 we give an overview of similar ontology approaches before giving our final remarks in Section 6.

2. USE CASES

Linked Open Data quality has different stakeholders in a myriad of domains, however the stakeholders can be cast under either *publishers* or *consumers*.

Publishers are mainly interested in publishing data that others can reuse. The five star scheme, which we propose to extend by a sixth star for quality, defines a set of widely accepted criteria that serve as a baseline for assessing data reusability. The reusability criteria defined by the five star scheme and by quality metrics are largely measurable in an objective way. Thanks to such objective criteria, one can assess the reusability of any given dataset without the major effort of, for example, running a custom survey to find out whether its intended target audience finds it reusable. (Such a survey may, of course, still help to get an *even better* understanding of quality issues.)

Without an objective rating that is easy to determine, data *consumers* – both machine and human – may find it challenging to assess the quality of a dataset, i.e. its fitness for use. Machine agents, e.g. for discovering, cataloguing and archiving datasets, may lack the computational power required to assess some of their quality dimensions, e.g. logical consistency. Tools for human end users, such as semantic web search engines [12] or Web of Data browsers [4, 10, 11], do not currently focus on quality when presenting a list of search results or an individual dataset.

2.1 Cataloguing and Archiving of Datasets

Software such as CKAN[4], which is best known for driving the datahub.io platform[5], makes datasets accessible to consumers by providing a variety of publishing and management tools and search facilities. A data publisher should be able to upload to such platforms, whilst on the other hand the platform should be able to automatically compute metadata regarding the dataset's quality. With the knowledge from this metadata, the publisher can improve the quality of the dataset. On the other hand, having quality metadata available for candidate datasets, consumers would be given the opportunity to discover certain quality aspects of a potential dataset.

2.2 Dataset Retrieval

Tools for data consumers, such as CKAN, usually provide features such as faceted browsing and sorting, in order to allow prospective dataset users (such as Bob, introduced in the previous section) to search within the large dataset archive. Using faceted browsing, datasets could be filtered according to tags or values of metadata properties. The datasets could also be ranked or sorted according to values of properties such as relevance, size or the date of last modification. Figure 1 shows a mockup of a modified *datahub.io* user interface to illustrate how quality attributes and metrics could be used in a faceted search with ranking.

With many datasets available, filtering or ranking by quality can become a challenge. Talking about "quality" as a whole might not make sense, as different aspects of quality matter for different applications. It does, however, make sense to restrict quality-based filtering or ranking to those quality categories and/or dimensions that are relevant in the given situation, or to assign custom weights to different dimensions, and compute the overall quality as a weighted sum. The daQ vocabulary provides flexible filtering and ranking possibilities in that it facilitates access to dataset quality metrics in these different dimensions and thus facilitates the (re)computation of custom aggregated metrics derived from base metrics. To keep quality metrics information easily accessible, we strongly recommend that each dataset contains the relevant daQ metadata graph in the dataset itself.

Alexander et al. [1] provide the readers with a motivational use case with regard to how the voID ontology (cf. Section 5) can help with effective data selection. The authors describe that a consumer can find the appropriate dataset by basing a criteria for content (what is the dataset mainly about), interlinking (to which other dataset is the one in question interlinked), and vocabularies (what vocabularies are used in the dataset). The daQ vocabulary could give an extra edge to "appropriateness" by providing the consumer with added quality criteria on the candidate datasets.

3. VOCABULARY DESIGN

The Dataset Quality Ontology (daQ) is a vocabulary for attaching the results of quality benchmarking of a linked open dataset to that dataset. The idea behind daQ is to provide a core vocabulary, which can be easily extended with additional metrics for measuring the quality of a dataset. The benefit of having an extensible schema is that quality metrics can be added to the vocabulary without major changes, as the representation of new metrics would follow those previously defined.

daQ uses the namespace prefix *daq*, which expands to `http://purl.org/eis/vocab/daq`.

The basic and most fundamental concept of daQ is the *Quality Graph* (Figure 2 – Box A), which is a subclass of *rdfg:Graph*. daQ

[4] `http://www.ckan.org`
[5] `http://www.datahub.io`

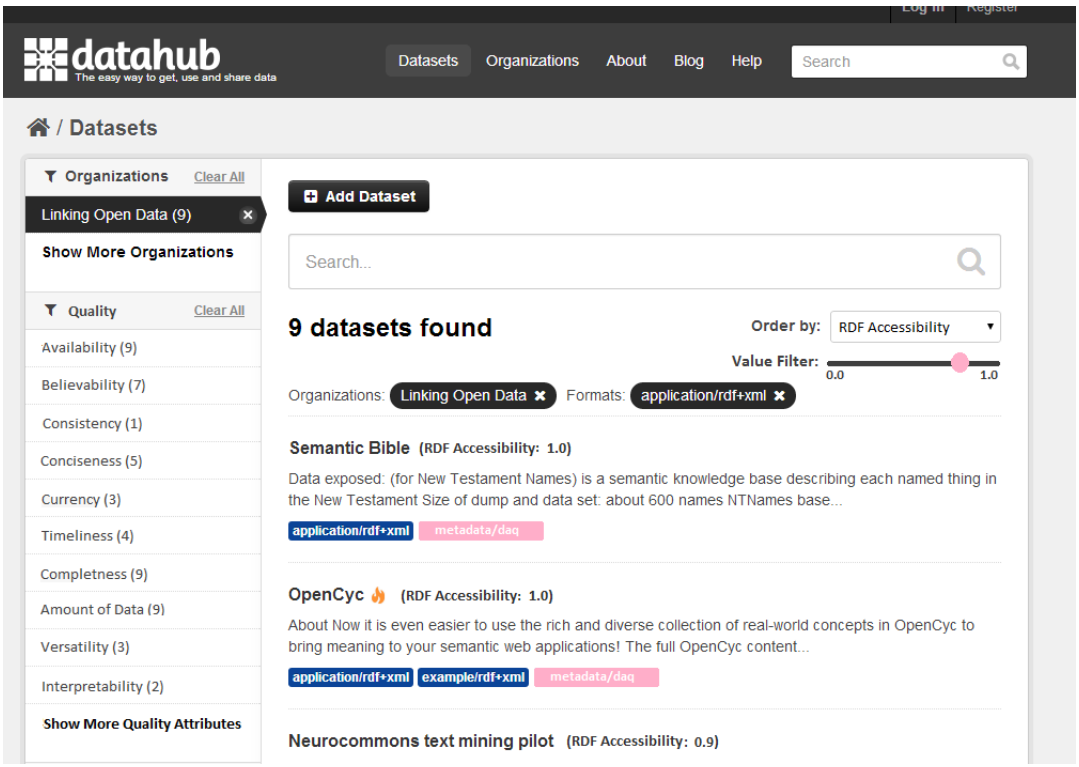

Figure 1: *datahub.io* Mockup having Quality Attributes available in the Faceted Browsing and Ranking

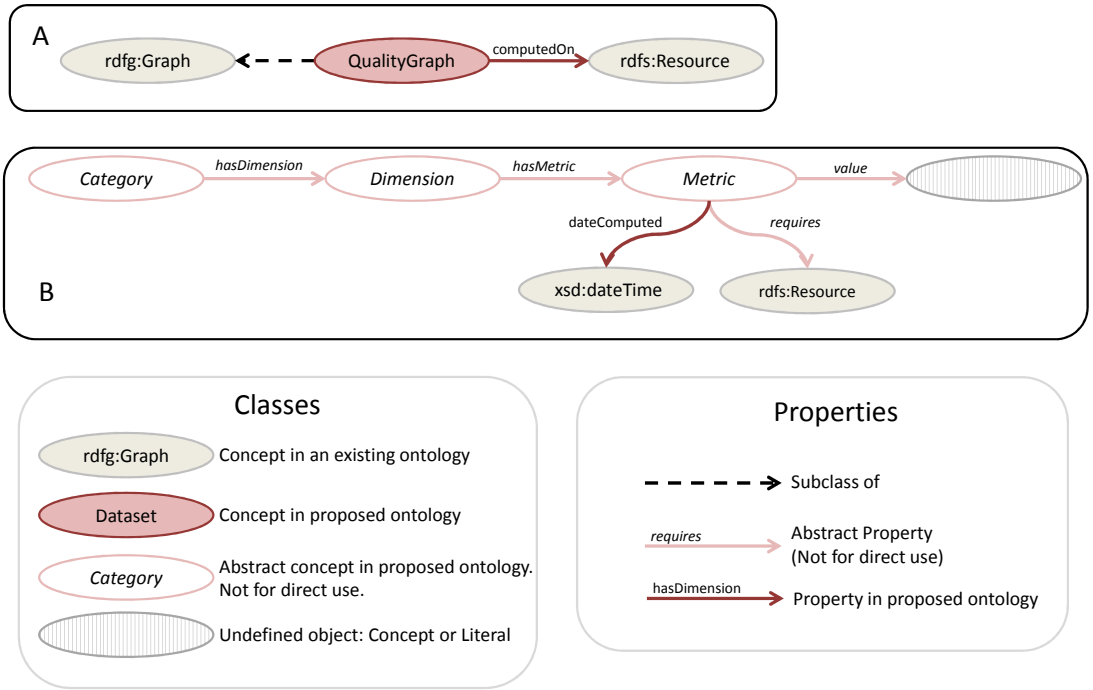

Figure 2: The Dataset Quality Ontology (daQ)

instances are stored as RDF Named Graph [7] in the dataset whose quality has been assessed. Named graphs are favoured due to

- the capability of separating the aggregated metadata with regard to computed quality metrics of a dataset from the dataset itself;
- their use in the Semantic Web Publishing vocabulary [6] to allow named graphs to be digitally signed, thus ensuring trust in the computed metrics and defined named graph instance. Therefore, in principle each *daq:QualityGraph* can have the following triple `:myQualityGraph swp:assertedBy :myWarrant .`

The daQ ontology distinguishes between three layers of abstraction, based on the survey work by Zaveri et al. [15]. As shown in Figure 2 Box B, a quality graph comprises of a number of different *Categories* (C), which in turn possess a number of quality *Dimensions* (D). A quality dimension groups one or more computed quality *Metrics* (M). To formalise this, let G represent the named Quality Graph (*daq:QualityGraph*), $C = \{c_1, c_2, \ldots, c_x\}$ is the set of all possible quality categories (*daq:Category*), $D = \{d_1, d_2, \ldots, d_y\}$ is the set of all possible quality dimensions (*daq:Dimension*) and $M = \{m_1, m_2, \ldots, m_z\}$ is the set of all possible quality metrics (*daq:Metric*); where $x, z, y \in \mathbb{N}$, then:

DEFINITION 1.

$$G \subseteq C,$$
$$C \subset D,$$
$$D \subset M;$$

Figure 3 shows this formalisation in a pictorial manner using Venn diagrams.

Quality metrics can, in principle, be calculated on a collection of statements - datasets or graphs. This vocabulary allows a data publisher to create multiple graphs of quality metrics for different data. For example, if one dataset consists of a number of graphs, quality metrics can be defined for each graph separately. The property *daq:computedOn* with domain *daq:QualityGraph* allows a data publisher to define a quality graph for different *rdfs:Resources*. The resource should be the URI of a dataset (including instances of *void:Dataset*[6]) or an RDF named graph.

3.1 Abstract Classes and Properties

This ontology framework (Figure 2) has three abstract classes/concepts (*daq:Category*, *daq:Dimension*, *daq:Metric*) and four abstract properties (*daq:hasDimension*, *daq:hasMetric*, *daq:hasValue*, *daq:requires*) which should not be used directly in a quality instance. Instead these should be inherited as parent classes and properties for more specific quality protocols. The abstract concepts (and their related properties) are described as follows, assuming the definitions given in Section 1.1:

daq:Category represents the highest level of quality assessment. A category groups a number of dimensions.

daq:Dimension – In each dimension there is a number of metrics.

daq:Metric – The smallest unit of measuring a quality dimension is a metric. Each metric has a value, representing a score for the assessment of a quality attribute. Since this value is multi-typed (for example one metric might return true/false whilst another might require a floating point number), the value's *daq:hasValue* range is inherited by the actual metric's attribute. A metric might also require additional information (e.g. a gold standard dataset to compare with). Therefore, a concrete metric representation can also define such properties using subproperties of the *daq:requires* abstract property. Each metric can record the date when it was actually computed using the *daq:dateComputed*.

[6] `http://www.w3.org/TR/void/#dataset`

4. USING THE ONTOLOGY

We start this section by first showing how the daQ vocabulary can be extended, and then proceed by giving general recommendations on how to publish daQ metadata records with datasets. We then continue by showing how a typical daQ instance is represented and we give some SPARQL examples to demonstrate how a data consumer (including tools such as the filtering UI presented in Section 2.2) can query the daQ vocabulary and a graph instance. We conclude this section by describing an application which will use the proposed vocabulary to filter and rank datasets.

4.1 Extending daQ

The classes of the core daQ vocabulary can be extended by more specific and custom quality metrics. In order to use the daQ one should define the quality metrics which characterise the "fitness for use" [14] in a particular domain. However, we are currently in the process in defining the quality dimensions and metrics described in [15], as the standard set of quality protocols for Linked Open Data in daQ.[7] **Extending** the daQ vocabulary means adding new quality protocols that inheriting the abstract concepts (Category-Dimension-Metric). In Figure 4 we show an illustrative example of extending the daQ ontology (TBox) with a more specific quality attribute, i.e. the RDF Availability Metric as defined in [15], and an illustrative instance (ABox) of how it would be represented in a dataset.

The *Accessibility* concept is defined as an *rdfs:subClassOf* the abstract *daq:Category*. This category has five dimensions, one of which is the *Availability* dimension. This is defined as a *rdfs:subClassOf daq:Dimension*. Similarly, *RDFAvailabilityMetric* is defined as a *rdfs:subClassOf daq:Metric*. The *daq:hasValue* property is also extended with a sub-property called *daq:hasDoubleValue* which types its range as *xsd:double*. The specific properties *hasAvailabilityDimension* and *hasRDFAccessibilityMetric* (sub-properties of *daq:hasDimension* and *daq:hasMetric* respectively) are also defined (Figure 4). The advantage of extending the abstract ontology concepts in Figure 2 Box B is that the domain and range of *daq:hasDimension* and *daq:hasMetric* are restricted to the appropriate quality protocols.

Extensions by custom quality metrics do not need to be made in the daQ namespace itself; in fact, in accordance with LOD best practices, we recommend extenders to make them in their own namespaces. Extending the daQ vocabulary with additional metrics assumes that their exact semantics (such as how they are to be computed) is understood by some software implementation, because daQ is intended to remain light-weight and thus not capable of expressing such semantics by its own means. Therefore, a user extending the daQ would not normally need to specify the technical requirements of the quality metric, although pointers to such requirements descriptions can be given via specialisations of the *daq:requires* abstract property.

4.2 Publishing daQ Metadata Records

[7] The metrics defined so far can be found under the namespace URI given above, or in the source files at `https://github.com/diachron/quality/blob/master/vocab/`.

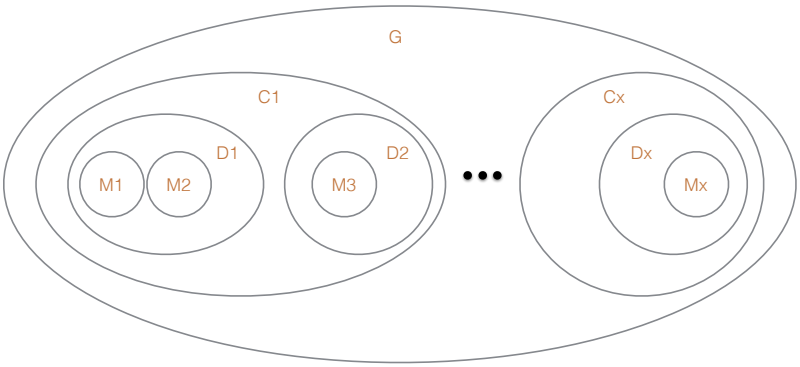

Figure 3: Venn Diagram depicting Definition 1

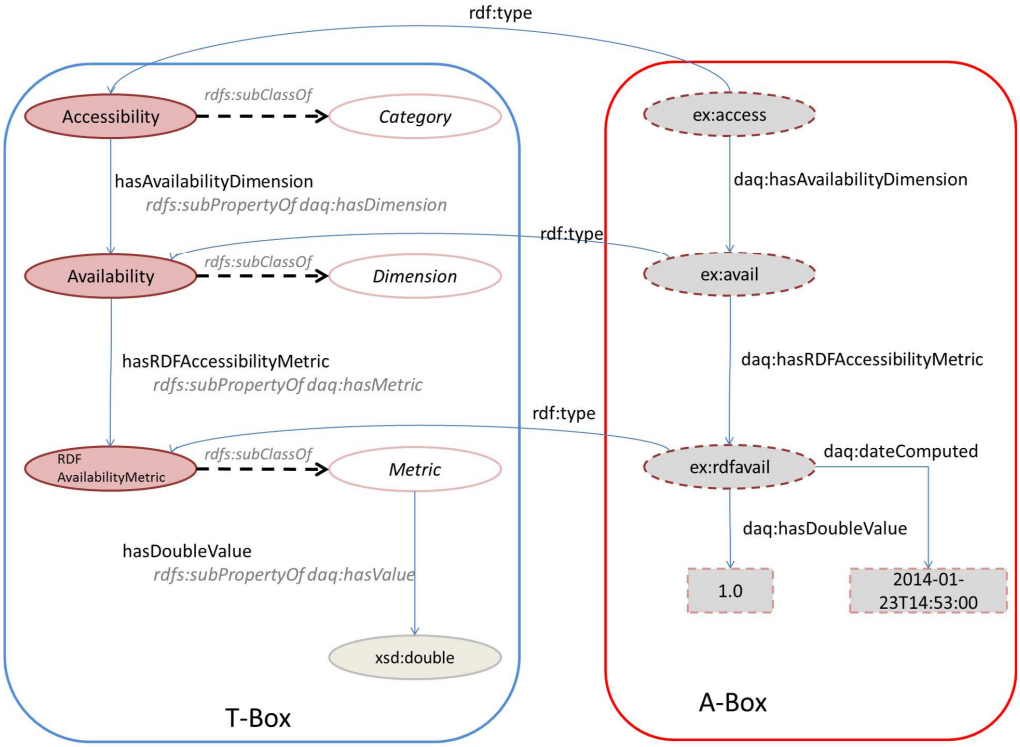

Figure 4: Extending the daQ Ontology – TBox and ABox

Dataset publishers should offer a daQ description as an RDF Named Graph in their published dataset. Since such a daQ metadata record requires metrics to be computed, it is understandable that it is not easy to author manually. Therefore, as suggested in Section 2, publishing platforms should offer such on-demand computation to dataset publishers. Another possible tool would be a pluggable platform which calculates quality metrics on datasets. Since the daQ vocabulary can be easily extended by custom quality metrics, as shown in the previous section, such a pluggable environment would allow users to import such custom metrics. One must keep in mind that the computation of metrics on large datasets might be computationally expensive; thus, platforms computing dataset quality must be scalable.

4.3 Representing Quality Metadata Instances

Listing 1 shows an instance of the *daq:QualityGraph* in a dataset. *ex:qualitygraph1* is a named *daq:QualityGraph*. The triples show that quality metrics were computed on the whole dataset. Consumers' queryies of the dataset for *daq:QualityGraph* instances will resolve to the named graph. In this named graph, instances for the *daq:Accessibility*, *daq:Availability*, *daq:EndPointAvailabilityMetric* and *daq:RDFAvailabilityMetric* are shown. A metric instance specifies the metric value and the date when it was last computed.

```
# ... prefixes

# ... dataset triples

ex:qualityGraph1 a daq:QualityGraph ;
  daq:computedOn <> .

ex:qualityGraph1 {

  # ... quality triples
  ex:accessibilityCategory a daq:Accessibility ;
    daq:hasAvailabilityDimension ex:availabilityDimension
        .

  ex:availabilityDimension a daq:Availability ;
    daq:hasEndPointAvailabilityMetric ex:endPointMetric ;
    daq:hasRDFAvailabilityMetric ex:rdfAvailMetric .

  ex:endPointMetric a daq:EndPointAvailabilityMetric ;
    daq:dateComputed "2014-01-23T14:53:00"^^xsd:dateTime
        ;
    daq:doubleValue "1.0"^^xsd:double .

  ex:rdfAvailMetric a daq:RDFAvailabilityMetric ;
    daq:dateComputed "2014-01-23T14:53:01"^^xsd:dateTime
        ;
    daq:doubleValue "1.0"^^xsd:double .

  # ... more quality triples
}
```

Listing 1: A *Dataset Quality Graph* N3 instance

4.4 Retrieving Metadata using SPARQL Queries

Listings 2 and 3 show typical SPARQL queries, which could be performed by data consumers. The first query retrieves all category and dimension instances from the quality graph. This query could be useful, for example, for those consumers who require to visualise all categories and dimensions available in a faceted manner. The second query retrieves all metric instances whose value (in this case a double-precision floating point number) is less than 0.5. This might be useful for identifying those metrics w.r.t. which the dataset needs serious improvement. Listing 4 shows a SPARQL query that retrieves and ranks all datasets by the Entity Trust [15] metric. This query is useful for consumers who would require a visible ranking of the datasets.

```
select ?catInst, ?dimInst where {
  ?qualGraph a daq:QualityGraph .
  graph <http://purl.org/eis/vocab/daq> {
    ?category rdfs:subClassOf daq:Category .
    ?property rdfs:subPropertyOf daq:hasDimension .
  }
  graph ?qualGraph {
    ?catInst a ?category ;
             ?property ?dimInst .
  }
}
```

Listing 2: A SPARQL query retrieving all Category and Dimension instances from a *daq:QualityGraph*

```
select ?metricInst where {
  ?qualGraph a daq:QualityGraph .
  graph <http://purl.org/eis/vocab/daq> {
    ?metric rdfs:subClassOf daq:Metric .
  }
  graph ?qualGraph {
    ?metricInst a ?metric ;
                daq:doubleValue ?val .
    filter(?val < 0.5)
  }
}
```

Listing 3: A SPARQL query retrieving all Metrics which have their value (double) < 0.5

```
select ?dataset where {
  ?qualGraph a daq:QualityGraph ;
             daq:computedOn ?dataset

  graph ?qualGraph {
    ?metricInst a daq:EntityTrustMetric ;
                daq:doubleValue ?val .
    order by desc(?val) .
  }
}
```

Listing 4: A SPARQL query retrieving and rank all Datasets by the Entity Trust metric value

4.5 The DIACHRON Project

The DIACHRON project ("Managing the Evolution and Preservation of the Data Web"[8]) combines several of the use cases mentioned so far. DIACHRON'S central cataloguing and archiving hub is intended to host datasets throughout several stages of their lifecycle [3], mainly evolution, archiving, provenance, annotation, citation and data quality. As a part of the DIACHRON project, we are implementing scalable and efficient tools to assess the quality of datasets. A web-based visualisation tool, to be implemented as a CKAN plugin, will

- allow data publishers to perform quality assessment on datasets, which will provide them with quality score metadata and also assist them with fixing quality problems;
- allow data consumers to filter and rank datasets by multiple quality dimensions.

The daQ vocabulary is the core ontology underlying these services. It will help these services to do their jobs, i.e. adding quality metadata to datasets, which in turn is displayed on the web frontend.

[8] http://diachron-fp7.eu

5. RELATED WORK

To the best of our knowledge, the Data Quality Management (DQM) vocabulary [9] is the only one comparable to our approach. Fürber et al. propose an OWL vocabulary that primarily represents data requirements, i.e. what quality requirements or rules should be defined for the data. Such rules can be defined by the user herself, and the authors present SPARQL queries that "execute" these requirements definitions to compute metrics values. Unlike our daQ model, the DQM defines a number of classes that can be used to represent a data quality rule. Similarly, properties for defining rules and other generic properties such as the rule creator are specified. The daQ model allows for integrating such DQM rule definitions using the *daq:requires* abstract property, but we consider the definition of rules out of daQ's own scope. As discussed in Section 2, the intention of the Dataset Quality vocabulary is to enable data publishers to easily describe dataset quality so that, in turn, consumers can easily find out which datasets are fit for their intended use. Rather than having quality rules defined using the daQ itself, the semantics of the custom metric concepts should be understood by the application implementing them. Therefore, rather than having a fixed set of classes/rules which one can extend, the daQ vocabulary gives the freedom to the user to define and implement any metrics required for a certain application domain.

Our design approach is inspired by the digital.me Context Ontology (DCON[9]) [2]. Attard et al. present a structured three-level representation of context elements (Aspect-Element-Attributes). The DCON ontology instances are stored as Named Graphs in a user's Personal Information Model. The three levels are abstract concepts, which can be extended to represent different context aspects in a concrete ubiquitous computing situation.

The voID[10] and dcat[11] ontologies recommended by the W3C provide metadata vocabulary for describing datasets. The "Vocabulary of Interlinked Datasets" (voID) ontology allows the high-level description of a dataset and its links [1]. On the other hand, the Data Catalog Vocabulary (dcat) describes datasets in data catalogs, which increase discovery, allow easy interoperability between data catalogs and enable digital preservation. With the daQ ontology, we aim to extend what these two ontologies have managed for datasets in general to the specific aspect of quality: enabling the discovery of a good quality (fit to use) datasets by providing the facility to "stamp" a dataset with quality metadata.

6. CONCLUDING REMARKS

In this paper we presented the Dataset Quality Ontology (daQ), an extensible vocabulary to provide quality benchmarking metadata of a linked open dataset to the dataset itself. In Section 2 we presented a number of use cases that motivated our idea, including cataloguing, archiving and filtering datasets, and that helped in developing the daQ ontology (Section 3). The ontology is still in its initial phases, thus further modelling will be required in the coming months to make sure that the core vocabulary covers all concepts required for the intended use cases. This will be possible by (i) exchanging ideas with interested LOD quality researchers, and (ii) making sure that the vocabulary meets the standards required to be easily adapted by both data producers and consumers.

We are currently in the process of giving more precise definitions of the quality dimensions and metrics collected in [15]. A number of quality metrics are also being implemented, with the aim of providing information about the quality of big LOD datasets. This would allow us to create meaningful daQ Named Graph instances at a large scale, i.e. creating quality metadata on real datasets. The DIACHRON platform will support the daQ by ranking and filtering datasets according to the quality metadata, like we sketched in the mockup explained in Section 2.2. Having tools and platforms supporting the daQ will finally allow us to test and evaluate the vocabulary thoroughly, to see whether the daQ itself is of a high quality, i.e. fit for use.

[9] http://www.semanticdesktop.org/ontologies/dcon/

[10] http://www.w3.org/TR/void/

[11] http://www.w3.org/TR/2014/REC-vocab-dcat-20140116/

7. ACKNOWLEDGMENTS

This work is supported by the European Commission under the Seventh Framework Program FP7 grant 601043 (http://diachron-fp7.eu).

8. REFERENCES

[1] K. Alexander, R. Cyganiak, M. Hausenblas, and J. Zhao. Describing Linked Datasets – On the Design and Usage of voiD, the 'Vocabulary of Interlinked Datasets'. In *WWW 2009 Workshop: Linked Data on the Web (LDOW2009)*, Madrid, Spain, 2009.

[2] J. Attard, S. Scerri, I. Rivera, and S. Handschuh. Ontology-based situation recognition for context-aware systems. In *Proceedings of the 9th International Conference on Semantic Systems*, I-SEMANTICS '13, pages 113–120, New York, NY, USA, 2013. ACM.

[3] S. Auer, L. Bühmann, C. Dirschl, O. Erling, M. Hausenblas, R. Isele, J. Lehmann, M. Martin, P. N. Mendes, B. van Nuffelen, C. Stadler, S. Tramp, and H. Williams. Managing the life-cycle of linked data with the LOD2 stack. In *Proceedings of International Semantic Web Conference (ISWC 2012)*, 2012. 22

[4] T. Berners-Lee, Y. Chen, L. Chilton, D. Connolly, R. Dhanaraj, J. Hollenbach, A. Lerer, and D. Sheets. Tabulator: Exploring and analyzing linked data on the semantic web. In *Proceedings of the The 3rd International Semantic Web User Interaction Workshop (SWUI06)*, Nov. 2006.

[5] C. Bizer. *Quality-Driven Information Filtering in the Context of Web-Based Information Systems*. PhD thesis, FU Berlin, Mar. 2007.

[6] J. J. Carroll, C. Bizer, P. J. Hayes, and P. Stickler. Semantic web publishing using named graphs. In J. Golbeck, P. A. Bonatti, W. Nejdl, D. Olmedilla, and M. Winslett, editors, *ISWC Workshop on Trust, Security, and Reputation on the Semantic Web*, volume 127 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2004.

[7] J. J. Carroll, P. Hayes, C. Bizer, and P. Stickler. Named graphs, provenance and trust. In A. Ellis and T. Hagino, editors, *Proceedings of the 14th WWW conference*, pages 613–622. ACM Press, 2005.

[8] A. Flemming. Quality Characteristics of Linked Data Publishing Datasources. http://sourceforge.net/apps/mediawiki/trdf/index.php?title=Quality_Criteria_for_Linked_Data_sources, 2010. [Online; accessed 13-February-2014].

[9] C. Fürber and M. Hepp. Towards a vocabulary for data quality management in semantic web architectures. In *Proceedings of the 1st International Workshop on Linked Web Data Management*, LWDM '11, pages 1–8, New York, NY, USA, 2011. ACM.

[10] A. Harth. Visinav: A system for visual search and navigation on web data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 8(4):348–354, 2010. Semantic Web Challenge 2009 User Interaction in Semantic Web research.

[11] P. Heim, J. Ziegler, and S. Lohmann. gFacet: A browser for the web of data. In *Proceedings of the International Workshop on Interacting with Multimedia Content in the Social Semantic Web (IMC-SSW 2008)*, volume 417 of *CEUR Workshop Proceedings*, pages 49–58, Aachen, 2008.

[12] A. Hogan, A. Harth, J. Umbrich, S. Kinsella, A. Polleres, and S. Decker. Searching and browsing linked data with SWSE: The semantic web search engine. *Web Semantics: Science, Services and Agents on the World Wide Web*, 9(4):365 – 401, 2011. {JWS} special issue on Semantic Search.

[13] A. Hogan, J. Umbrich, A. Harth, R. Cyganiak, A. Polleres, and S. Decker. An empirical survey of linked data conformance. *J. Web Sem.*, 14:14–44, 2012.

[14] J. M. Juran. *Juran's Quality Control Handbook*. Mcgraw-Hill (Tx), 4th edition, 1974.

[15] A. Zaveri, A. Rula, A. Maurino, R. Pietrobon, J. Lehmann, and S. Auer. Quality assessment methodologies for linked open data (under review). *Semantic Web Journal*, 2012. This article is still under review.