Measuring the Quality of Relational-to-RDF Mappings

Darya Tarasowa(✉), Christoph Lange, and Sören Auer

Institute of Computer Science, University of Bonn & Fraunhofer IAIS,
Bonn, Germany
{darya.tarasowa,math.semantic.web}@gmail.com, auer@cs.uni-bonn.de

Abstract. Mapping from relational data to Linked Data (RDB2RDF) is an essential prerequisite for evolving the World Wide Web into the Web of Data. We propose a methodology to evaluate the quality of such mappings against a set of objective metrics. Our methodology, whose key principles have been approved by a survey among RDB2RDF experts, can be applied to evaluate both automatically and manually performed mappings regardless of their representation. The main contributions of the paper are: (1) assessing the quality requirements for mappings between relational databases and RDF, and (2) proposing methods for measuring how well a given mapping meets the requirements. We showcase the usage of the individual metrics with illustrative examples from a real-life application. Additionally, we provide guidelines to improve a given mapping with regard to the specific metrics.

Keywords: Data quality · Quality assessment · Quality metrics · RDB2RDF

1 Introduction

Translating the data stored in relational databases (RDB) to the linked data format is an essential prerequisite for evolving the current Web of documents into a Web of Data. In order to be effectively and efficiently reusable, linked data should meet certain data quality requirements [21]. Assessing the quality of a linked dataset created from RDB may be a laborious, repetitive task if the dataset is frequently recreated from its RDB source, e.g. after any update to the RDB. We therefore claim that it is possible to positively influence the quality of a linked dataset created from RDB by improving the quality of the RDB2RDF *mapping* that produces the linked data.[1] To the best of our knowledge there has not been prior research towards collecting and describing quality requirements

[1] Here, we use the term "mapping" to refer to the function that maps relational database columns to RDF properties, but not to materialisations of such mappings in concrete languages or representations, nor to tools that would execute such a function.

P. Klinov and D. Mouromtsev (Eds.): KESW 2015, CCIS 518, pp. 210–224, 2015.
DOI: 10.1007/978-3-319-24543-0_16

for RDB2RDF mappings. Thus, this paper aims to fill the gap by providing a system of metrics to measure the quality of RDB2RDF mappings.

A large number of tools for mapping relational data to RDF exist already[2]; they implement different mapping approaches, often allowing to customize the mapping. To standardize the description of mappings, the W3C RDB2RDF Working Group has released two recommendations in 2012 (more details in Section 3.1): Direct Mapping [1], which produces RDF graph representations directly from a relational database (data and schema) and the Relational Database to RDF Mapping Language (R2RML) [5] for expressing customized mappings.

In this paper, we aim at developing a system of quality metrics that can be applied both to direct and customized mappings that works with different representations of mappings (R2RML or proprietary formats, visual diagrams, tables, etc.). Any of these representations are suitable for being evaluated with the proposed system as long as one can derive a list of database columns (including unmapped ones) and their corresponding RDF properties (for mapped ones).

To determine the scope of requirements that can be posed to RDB2RDF mappings, we studied the state of the art in related research fields such as ontology matching, linked data quality assessment and RDB2XML mappings. However, as Sect. 2 shows, these metrics collected from the scientific community do not solve the given problem entirely. In order to fulfil the list, we followed the Direct and R2RML Mapping recommendations. Some of the RDB2RDF features standardized in the documents are connected with the output data quality dimension, while others influence the quality of the mapping itself. For example, the ability to define R2RML views allows to incorporate domain semantics into the mapping an opportunity that we consider to increase the quality of mapping. We propose not only a system of metrics but also define means of measuring them, along with guidelines to improve the rating of a dataset w.r.t. each metric.

The paper is structured as follows: in Sect. 2 we discuss the existing approaches, in Sect. 3 we summarize the quality requirements for mappings and the metrics for measuring the quality. Section 4 presents the results of the survey conducted in order to collect the community feedback. In Sect. 5, we conclude and propose future research directions.

2 Related Work

RDB2RDF Mapping Approaches. Although previous research has not explicitly focused on collecting requirements for RDB2RDF mappings, many such requirements can be found in best practice and mapping approach descriptions. For instance, in [10], the importance of reusing existing vocabularies to enhance the interoperability of the output dataset is explained. We extend this statement to a set of requirements combined in the *interoperability* dimension, taking into account not only quantitative, but also qualitative metrics.

[2] One listing is available at https://github.com/timrdf/csv2rdf4lod-automation/wiki/Alternative-Tabular-to-RDF-converters.

We notice that existing literature on the topic uses incoherent terminology. Often, the term *mapping* is used instead of mapping *language*, e.g. in [2,7]. Discussing the requirements for mapping *language*, both articles mention the following requirements for RDB2RDF *mapping* as well:

- presence of both ETL and on-demand implementation
- incorporating domain semantics that is often implicit or not captured at all in the RDB schema
- indication of time metadata about the dataset during the mapping creation process to control the dataset currency (we subsume this metric under "data quality")
- intensive reuse of the existing ontologies

Requirements for Mappings Between Relational Data and XML. The topic of mapping relational data to *XML* is related to RDB2RDF mapping due to the similarity between the XML and RDF data models. The earliest articles on the quality of RDB2XML mappings (e.g. [16,17]) only take into account the performance of the mapping algorithms. This is because RDB2XML mappings are mostly produced automatically and do not allow customization. Only few authors (e.g. [15]) propose metrics for evaluating the quality of the output data. However, the proposed metrics evaluate only the syntactic correctness of the output XML and therefore cannot be applied to documents in RDF (except for *presence of syntax errors*).

Other approaches (e.g. [18]) prove the need for customizable techniques of mapping due to the wide gap between these two formats. As their major evaluation criterion, they use the *efficiency of query processing.* We adopt this approach and base our *simplicity* metric on it. Liu et al. [13] describe an approach to design a high-quality XML Schema from ER diagrams. The authors define a list of requirements for the design; the most relevant one for our purposes is *information preservation.* However, the metrics for *measuring* such requirements are not discussed. We study the information preservation requirement in detail and provide objective metrics in the *Faithfulness of Output* dimension.

Requirements for Ontology Matching. Mapping between a database schema and an RDF vocabulary can also be viewed as a special case of ontology matching. Several studies propose requirements for measuring the quality of an ontology matching. According to [8], all proposed measures can be distinguished between compliance measures, measures concerned with system usability and performance measures focusing on runtime or memory usage. As our study focuses on evaluating the quality of mapping itself, we do not take the system usability or performance dimensions into account.

In the evaluation of ontology matching, compliance measures are based on the two classical information retrieval measures of *precision* and *recall* [20]. We adapt these metrics to the evaluation of RDB2RDF mappings and include them in the *faithfulness of output* dimension as *coverage* and *accuracy of data presentation* metrics.

Measuring Linked Data Quality. The quality of mapping correlates with the quality of output linked data produced by the mapping. Poor design decisions made at the stage of defining a mapping, such as using deprecated classes, redundant attributes or badly formed URIs, decreases the quality of the output data. Therefore, metrics for the quality of the output data can be viewed as metrics for quality of the mapping.

The field of assessing the quality of linked data is still in its infancy, but several articles have addressed it already. Our recent survey [21] collects and compares the existing papers, the most complete and detailed ones being Bizer's PhD thesis [4], Flemming's master's thesis [9] and empirical evaluations by Hogan et al. [11,12]. The survey provides formal definitions of data quality dimensions and metrics as well as evaluates existing tools for (semi-)automatic measurement of linked data quality.

The current paper assumes that the quality of a linked dataset is influenced by the mapping that produces it and thus categorizes the metrics from the survey from the perspective of the RDB2RDF mapping. We select those metrics that are related to the mapping process and adapt them to the RDB2RDF domain.

3 Quality Requirements

This section gives a detailed overview of the proposed quality requirements for RDB2RDF mappings, ways of measuring them (metrics) and guidelines for improving mappings w.r.t. these metrics. Our proposed system incorporates four quality dimensions with 14 objective metrics overall. For assessing the overall quality of a mapping, one would, in practice, assign *weights* to the metrics. Their choice depends on the goal of the mapping process. For example, when the goal of the mapping is to accurately represent the relational data, the metrics in the "faithfulness of the output" dimension should be assigned the highest weight.

3.1 Quality of the Mapping Implementation and Representation

The requirement of mapping quality implementation and representation combines the requirements for resultant data accessibility and standard compliance of the mapping representation.

Data Accessibility. Data accessibility describes how the result of the mapping is accessed. This metric is also known in the literature as "access paradigm", "mapping implementation" or "data exposition" [19]. There are two possibilities: (i) Extract Transform Load (ETL) and (ii) on-demand mapping. According to [7], ETL means physically storing triples produced from relational data in an RDF store. The disadvantage of ETL is that that, whenever the RDB is updated, you have to re-run the entire ETL process, even if just one RDB record has changed, carrying out an often redundant synchronization process. However, in the ETL case nothing more than the RDF store is needed to answer a query.

Table 1. Summary table of proposed metrics system

Requirement	Description	Measure
	Quality of the mapping implementation and representation	
Data accessibility	Describes how the mapping result can be accessed.	ETL/on-demand/both
Standard compliance	Characterizes if the mapping representation is standard compliant.	boolean
	Faithfulness of the output	
Coverage	Characterizes the mapping completeness	percentage of DB columns mapped
Accuracy of data representation	Characterizes the mapping correctness	percentage of correctly mapped DB columns
Incorporation of domain semantics	Shows level of domain semantics incorporation	percentage of properties that link to the results of SQL queries
	Quality of the output	
Simplicity	Shows the simplicity of SPARQL queries returning the frequently demanded values	percentage of complex SQL queries results integrated into the mapping
Data quality	Characterizes the quality of output data	aggregation of linked data quality metrics (→ Table 2)
Data integration	Characterizes the interlinking degree of the output data	percentage of external instances integrated into the resultant dataset
	Interoperability	
Reuse of existing ontologies	Shows the amount of reused vocabulary elements	percentage of reused properties and classes
Quality of reused vocabulary elements	Characterizes the quality of chosen for reuse properties and classes	accumulated quality and popularity of reused vocabulary elements
Accuracy of reused properties	Characterizes the accuracy of properties reuse	percentage of accurately reused properties
Accuracy of reused classes	Characterizes the accuracy of classes reuse	percentage of accurately reused classes
Quality of declared classes/properties	Shows the quality of ontology documentation	accumulated quality of declared classes/properties

On-demand mapping is realized by translating a SPARQL query into one or more SQL queries at query-time, evaluating these against (a set of) unmodified relational database(s) and constructing a SPARQL result set from the result sets of such SQL queries. In contrast to the ETL implementation, on-demand mapping requires more resources for processing each query. However, the on-demand implementation does not face the synchronization issue and does not replicate the data. In the light of these advantages and disadvantages, we claim that the best solution is to implement a mapping with both data access approaches. Thus, the index of implementation takes a value equal to 1, if both implementations are present and 0.5 otherwise.

Standard Compliance. An RDB2RDF mapping can either be represented as a set of customized rules or through a generic set of rules (as defined by the W3C Direct Mapping standard [1,3]). Often, the output of a Direct Mapping may not be useful, that is, it may not adequately take the structural/semantic complexity of the database schema into account. Thus, while the applicability of a Direct Mapping satisfies the requirements for the "standard representation" metric, not using the customized rules will lead to a loss of points w.r.t. other metrics.

Languages for representing customized mappings have been surveyed in [19]. Until recently, no standard representation language for RDB2RDF mappings existed, however, as of 2012, R2RML [5] has been released as a W3C recommendation. As it is the only mapping language that has been standardized to date, we do not consider other (proprietary) formats reasonable.

Thus, we propose that if there is no material representation of a mapping available, it should be assumed that a Direct Mapping is carried out. We define the "standard representation" metric to be one if the mapping is represented in *R2RML* or if there is no representation (and therefore a Direct Mapping is applicable), and zero otherwise.

3.2 Faithfulness of the Output

In terms of RDB2RDF mapping we define the faithfulness of output as an abstract measure of similarity between the source data and resultant dataset.

Coverage. This metric indicates the ratio of the database columns mapped to the RDF. In general, a high coverage increases the faithfulness of the output data. However, reaching the highest level of coverage may conflict with privacy restrictions. Therefore, great care should be taken when mapping any kind of personal (sensitive) data that might be linked with other data sources or ontologies. It may even be illegal to create or to process such linked information in countries with data protection laws without having a valid legal reason for such processing [6].

Moreover, application-specific data such as statistics of usage or service tables often are not included into the mapping due to its application-related nature. Thus, in real-world applications, the coverage metric never reaches one.

Accuracy of Data Representation. When mapping relational data to RDF, the accuracy of data representation is tightly connected to data formatting issues. For example, differences in the representation of numeric values between the database and RDF can cause inaccuracies in the output data. Inaccurate dealing with not-Latin characters leads to loss of data meaning. We propose the following algorithm to evaluate the accuracy index of the mapping:

1. for each mapped database column containing numeric data, compute the average value of the numbers in this column;
2. find an average value for the corresponding property in the RDF output, and compare it to the average computed in step 1;
3. add a point for each coinciding average (considering the statistical error)
4. for each mapped database column containing literal data, analyze the readability of the corresponding properties; add a point for each completely readable property;
5. calculate a sum of the points obtained in steps 3 and 4 and divide the sum on the total number of columns mapped.

Incorporating Domain Semantics. The incorporation of domain semantics is one of the crucial aspects that indicate the quality difference between direct and customized mappings. Measuring this requirement helps to estimate the direct benefit of mapping customization. Generally, a direct mapping does not incorporate domain semantics; in this case the metric takes a value of zero. The extent of domain semantics incorporation can be measured by counting number of *class properties* which take values from any database table *other than* the table that corresponds to a class. The metric is then calculated as the ratio between this number and the count of properties summed up over all classes in the mapping.

To improve the mapping w.r.t. this metric, implicit relations between the data in the database should be explicitly modelled in the mapping. This process also increases the simplicity of the mapping, as it requires integration of the SQL query results into the mapping, thereby simplifying (future) SPARQL queries for obtaining these values.

3.3 Quality of the Output

Quality of the output combines requirements of the output data quality in aspects of simplicity of usage, level of interlinking and objective data quality metrics.

Simplicity. One often has the task of mapping highly complex data models to RDF. In that case, Direct Mapping produces an RDF output that may be difficult to use, especially when considering the limitations of the SPARQL query language. Thus, useful information from the source database may lose its value in the RDF dataset that results from mapping. Metrics for the quality of the output should therefore take the simplicity of the RDF dataset into account. By simplicity of RDF data, we mean the simplicity of *operating* on it. In other words, the simplicity of data determines the simplicity (or length in terms of abstract syntax) of the SPARQL queries returning frequently demanded values. To increase the simplicity, the mapping should aim to produce a dataset, that can be queried for frequently demanded values by relatively simple SPARQL constructions. To do this, the *frequently demanded values returning by the complex SQL queries* should be integrated into the mapping. Such preparations not only increase the simplicity of the output, but also *incorporate domain semantics* not presented in the relational database explicitly (cf. Section 3.2).

To calculate the index of simplicity, first a list of complex SQL queries should be assembled. We propose to consider a query "complex" in the following cases:

- a query joins at least two tables or views,
- a query supposes recursive computations over one table or view.

After the list is assembled, frequently demanded values, returned by the queries should be selected. This selection is subjective and should be carried out by a group of developers and administrators of the project. The simplicity metric is then calculated as the percentage of *frequently demanded values returned by complicated SQL queries* that have been integrated into the mapping.

Table 2. Data quality metrics divided into four groups according to the stage of mapping creation, on which they can be influenced.

Requirements for linked data quality and their metrics	
1. Influenced by mapping – **Completeness:** influenced by coverage – **Validity-of-documents:** influenced by quality and accuracy of reused and newly determined classes – **Interlinking:** influenced by data integration – **Availability:** influenced by dereferenceability of reused and newly defined classes – **Consistency:** influenced by reuse *2. Depend on mapping* – ***Provenance:*** *indication of metadata about a dataset* – ***Verifiability:*** *verifying publisher information, authenticity of the dataset, usage of digital signatures* – ***Licensing:*** *machine-readable/human-readable license, permissions to use the dataset, attribution, Copyleft or ShareAlike* – ***Validity-of-documents:*** *no syntax errors* – ***Consistency:*** *entities as members of disjoint classes, usage of homogeneous datatypes, misplaced classes or properties, misuse of* `owl:datatypeProperty` *or* `owl:objectProperty`*, bogus* `owl:Inverse-FunctionalProperty` *values, ontology hijacking* – ***Conciseness:*** *redundant properties/instances, not-unique values for functional properties, not-unique annotations* – ***Performance:*** *no usage of slash-URIs, no use of prolix RDF features* – ***Understandability:*** *human-readable labelling of classes, properties and entities by providing* `rdfs:label`*, human readable metadata* – ***Interpretability:*** *misinterpretation of missing values, atypical use of collections, containers and reification* ***Currency:*** *currency of data source* – ***Volatility:*** *no timestamp associated with the source*	*3. Depend on relational data* – ***Completeness:*** *values for a property are not missing* – ***Amount-of-data:*** *see [21]* – ***Provenance:*** *trustworthiness of RDF statements, trust of an entity, trust between two entities, trust from users, assigning trust values to data/sources/rules, trust value for data* – ***Believability:*** *see [21]* – ***Accuracy:*** *see [21]* – ***Consistency:*** *no stating of inconsistent property values for entities, literals incompatible with datatype range* – ***Interpretability:*** *interpretability of data* – ***Versatility:*** *provision of the data in various languages* *4. Depend on publishing* – ***Availability:*** *accessibility of the server, accessibility of the SPARQL end-point, accessibility of the RDF dumps, no structured data available, no dereferenced back-links* – ***Performance:*** *see [21]* – ***Security:*** *see [21]* – ***Response-time:*** *see [21]* – ***Conciseness:*** *keeping URIs short* – ***Understandibility:*** *indication of one or more exemplary URIs, indication of a regular expression that matches the URIs of a dataset, indication of an exemplary SPARQL query, indication of the vocabularies used in the dataset, provision of message boards and mailing lists* – ***Versatility:*** *provision of the data in different serialization formats, application of content negotiation* – ***Currency:*** *see [21]* – ***Timeliness:*** *see [21]*

Data Quality. The quality of a mapping can partly be assessed by evaluating the quality of the output RDF data. Section 2 points to approaches for assessing data quality. However, when evaluating the quality of a mapping, not all aspects of output data quality need to be taken into account. This is due to the fact that not all aspects of linked data quality are affected by the mapping stage.

We confine ourselves to the objective requirements discussed in [21] and divide them into four groups (cf. Table 2):

– Requirements that are influenced by mapping quality implicitly. Increasing the quality of the mapping in turn increases the quality of data.

- Requirements that are influenced by mapping explicitly. The proposed data quality index discussed below aggregates these metrics.
- Requirements that depend on the quality of data stored in the database. We consider these metrics to be out of the scope of this paper.
- Requirements that depend on the quality of linked data publishing. We do consider these metrics to be out of the scope of this paper.

The metrics and methods of measuring them are discussed in detail in [21]. To evaluate the quality of mapping with regard to output data quality we propose to aggregate only the metrics explicitly depending on the mapping. There are two possible strategies for calculating our overall data quality metric based on the individual metrics: assigning the same weight to all individual metrics, or assigning different weights to different dimensions or even to individual metrics.

Data Integration. The key advantage of Linked Data is its ability to effectively integrate data from disparate, heterogeneous data sources. In most cases, the output dataset resulting from a mapping can be linked to existing datasets via explicitly modelled relationships between entities. As mentioned in [14], the use of domain ontologies along with user defined inference rules for reconciling heterogeneity between multiple RDB sources, is an effective integration approach for creating an RDF. A simple metric for data integration can be defined as the number of external datasets linked to the one evaluated. However, this metric cannot be mapped to a value in the interval $[0, 1]$ in an obvious way, as needed for computing the overall mapping quality from the values of the individual metrics. Thus, we propose an alternative measure to evaluate the data integration level: index of data integration. It can be calculated as the ratio between the number of external instances integrated in the resulting RDF dataset and the total number of instances.

3.4 Interoperability

The interoperability requirement tackles the aspect of a mapping to produce interoperable data, which can be easily linked to and integrated with other ontologies and datasets.

Reuse of Existing Ontologies. The reuse of existing ontology elements (i.e. classes and properties), increases the interoperability of the output dataset. Additionally, this reuse prevents one from having to introduce new elements. This requirement can be measured using two metrics: the ratio of reused properties and the ratio of reused classes. However, not only the quantity, but also the *quality* of reused properties and classes is important. This aspect is covered by the next metric.

Quality of Reused Vocabulary Elements. Measuring the quality of reused properties or classes is not trivial. As a large number of vocabularies has been published on the Web, many vocabulary elements are repeatedly declared instead of being reused.

We claim that the best choice of a class or property to reuse depends on two parameters: (i) the quality of the class or property itself and (ii) the frequency of its usage in LOD in comparison with semantically similar alternatives. Thus, to measure the quality of each individual property or class we propose the following workflow:

1. Measure the quality of the chosen vocabulary element (class/property)
2. Find alternatives to the chosen element and measure their quality
3. Compare the quality metrics of the chosen element and the best alternatives

We propose to compute the quality index (i_{qual}) of the chosen vocabulary element as the product of the following two indexes: the index of documentation quality (i_{doc}) and the index of popularity (i_{pop}). The proposed index of documentation quality is computed differently for classes and properties. Both for classes and properties, the index of documentation quality takes into account dereferenceability as well as documentation by `rdfs:label` and `rdfs:comment` (preferably in multiple languages). If the class or property is explicitly deprecated, i.e. its `owl:deprecated` annotation property equals `true`, we define its quality index as zero. Additionally, for classes, their relation to other classes should be indicated (rdfs:subClassOf or owl:equivalentClass).

For properties, `rdfs:range` and `rdfs:domain` should be defined. Based on the presence of the properties definitions mentioned above, a decision about quality of the documentation for a vocabulary element (possibly automated) should be made. The index of popularity aims to measure the frequency of usage of the class or property on the Web in comparison with semantically similar alternatives. This index can be taken from services such as Linked Open Vocabularies.[3]

As an example, we evaluate the property `agrelon:hasChief`[4] that links an *organization* to its *chief*. The property is dereferenceable, it has labels available in four languages; however, no `rdfs:range`, `rdfs:domain` or `rdfs:comment` is specified. Based on this, we define its i_{doc} as 0.5. For measuring the frequency of usage we type the term 'chief' into the search field of the Linked Open Vocabularies service. Looking through the results, we can conclude that on the Web of Data the term 'chief' is most commonly used to model a military commander and the chosen property `agrelon:hasChief` has a low popularity metric value of 0.271. Thus, the quality index for this evaluated property is $i_{pop} * i_{doc} = 0.271 \cdot 0.5 = 0.1355$.

The next task is to find the most commonly used term for the evaluated property. The list of possible substitutes will include properties matching 'chief', 'leader', 'head', or 'boss'. For each item on the list the property with the highest i_{pop} and satisfying the following requirements, should be found:

- `rdfs:comment`, `rdfs:domain` and `rdfs:range` statements should correspond to the domain model

[3] http://lov.okfn.org.

[4] fullURI:http://d-nb.info/standards/elementset/agrelon.owl#hasChief.

- depending on the application settings, special requirements such as presence of an inverse property in the same vocabulary may need to be satisfied.

In our case, the metrics for the best candidates are represented in Sect. 3. Due to space limitations, we only take into account the first two synonyms from that list, as the two other ones do not have matches that satisfy the requirements given above. According to the investigation, the `agrelon:hasChief` property is not the best possible alternative for linking an organization to its chief. The `swpo:hasLeader` property should be used instead, as its quality index is $i_{pop} \cdot i_{doc} = 0.682 \cdot 0.8 = 0.5456$.

Table 3. Metrics for semantically identical properties to link an organization and its chief

	'chief'	**'leader'**
best candidate	agrelon:hasChief	swpo:hasLeader
i_{pop}	0.271	0.682
i_{doc}	0.5	0.8
i_{qual}	**0.1355**	**0.5456**

The last step is to compare the quality of the chosen property and the best possible alternatives. The resultant score for the chosen property is calculated as the ratio between its quality index and the quality index of the best possible alternative: $i_{qual_{chosen}}/i_{qual_{best}}$ where $i_{qual_{best}}$ is the quality index of the best alternative and $i_{qual_{chosen}}$ is the quality index of the evaluated property. In our case, choosing the `agrelon:hasChief` property adds $0.1355/0.5456 = 0.248$ points to the interoperability metric. Thus, the overall value of the *quality of reused vocabulary elements* metric is defined as the ratio between the sum of the quality indexes of all reused vocabulary elements and the number of reused vocabulary elements.

Accuracy of Reused Properties. We determine two requirements for accurate reuse of existing properties: (i) respecting the property definition and (ii) unambiguous meaning of the property. Respecting the property definition means that the domain model must be consistent with the definition of the property, namely its `rdfs:range`, `rdfs:domain` and `rdfs:comment` relations. Unambiguous meaning is important when similar relations are presented in the domain model. For example, consider the domain model of our SlideWiki OpenCourseWare authoring platform[5] and the following relations between decks (collections of slides):

```
– Deck1  sw:isTranslationOf  Deck2
– Deck2  sw:isRevisionOf     Deck3
– Deck3  sw:isVersionOf      Deck4
```

[5] http://www.slidewiki.org.

Here, `Deck2` results from *translating* `Deck1` into a different language, `Deck3` is a result of *editing* `Deck2`, and `Deck4` results from *moving* `Deck3` into another e-learning platform (for example, Moodle[6]). If we decide to reuse the property `dcterms:isVersionOf` instead of one of these three properties from SlideWiki's domain-specific vocabulary, the meaning of the property will be ambiguous. A better solution in this case is to declare all three new properties to be subproperties of `dcterms:isVersionOf`.

The index of accuracy of reused properties can be calculated as the number of accurately reused properties (i.e. the properties, that satisfy both requirements) in relation to the total number of reused properties in the dataset.

Accuracy of Reused Classes. An important aspect of reusing classes is the semantic *compatibility*. By this we mean the compatibility of the meaning of a reused class and a domain object declared to be an instance of the class. The index of accuracy of reused classes is defined as the ratio between semantically compatible reused classes and the total number of reused classes. We illustrate the issue with an example below.

Let us assume that we need to link a deck of slides to its CSS style. The property `oa:styledBy` from the *Open Annotation Data Model ontology*[7] could be chosen to model the link. However, its domain is `oa:Annotation`, which is not compatible with the `sw:Deck` class in terms of its intended semantic. If such a statement occurs within the mapping, the reused class `oa:Annotation` should be considered as incompatible for reuse.

To measure the requirement we propose to calculate the index of the accuracy of reused classes. It can be calculated as the number of reused classes semantically compatible with domain objects in relation to the total number of reused classes.

Quality of Declared Classes/Properties. In order to obtain high quality, the classes/properties whose *definition* is *introduced* by the mapping should meet the requirements of documentation quality analogically to reused vocabulary elements (see Sect. 3.4). Thus, the proposed index of quality of declared vocabulary elements is computed separately for class and properties and accumulates the documentation quality score in relation to the total number of properties/classes declared.

4 Evaluation

In order to evaluate how well the proposed methodology agrees with real-life common practices, we conducted a survey using a questionnaire.[8] The survey was formed of statements, one per requirement from Sect. 3. Each requirement was phrased in the style of a practical guideline, in that complying with this

[6] http://moodle.org.

[7] http://www.w3.org/ns/oa.

[8] http://slidewiki.org/application/questionnaire.php.

guideline when designing a mapping would help the mapping to obtain the highest possible score for the corresponding metric. For example, the requirement "accuracy of reused properties" (Sect. 3.4), evaluated by combination of two metrics, was divided into two statements: "The properties chosen for reuse should be available, dereferenceable and have unambiguous meaning in the application domain." and "When choosing the properties for reuse, *rdfs:domain*, *rdfs:range* and *rdfs:comment* must be taken into account." For each statement the experts had to express to what extent they agreed with it. Thus, the agreement of a majority of experts with a statement would prove the relevancy of both the requirement and the proposed way of its measuring. We did not include requirements for *output data quality*, *coverage* and *accuracy of data* in the survey, as their metrics had been proposed by prior research. In addition to the opinions about the statements we collected open feedback.

Table 4. Results of survey on degree of agreement with the key concepts of the proposed methodology. Numbers represent the degree of agreement from *absolutely agree* (5) to *absolutely disagree* (1). The colors indicate the self-estimated level of expertise in the RDB2RDF domain from *expert* (darkest) to *experienced* (lightest).

Requirement	**P1**	**P2**	**P3**	**P4**	**P5**	**P6**	**P7**	**P8**	**P9**	**P10**	**P11**	**P12**	**P13**	**Mode**
Data accessibility	4	2	5	5	4	5	5	4	4	2	4	2	5	5, 4
Representation	2	2	4	5	1	5	4	4	3	4	3	2	5	4
Incorporation of domain semantics	4	3	3	4	3	3	3	4	2	5	5	3	5	3
Simplicity	3	3	3	4	3	3	4	4	5	2	1	4	4	4, 3
Data integration	4	3	4	3	5	2	5	4	3	4	2	4	3	4
Reuse of existing ontologies	5	3	4	3	5	4	5	4	5	2	2	4	2	5
Quality of reused properties (frequency)	4	4	4	3	4	3	4	4	3	2	2	4	2	4
Quality of reused properties (dereferenceability)	5	1	3	4	5	3	5	4	5	4	4	4	4	4
Accuracy of reused properties	4	4	3	4	5	4	5	4	4	4	3	4	5	4
Accuracy of reused classes	5	4	4	3	4	4	4	3	4	4	3	4	5	4
Quality of declared classes/properties	4	3	3	4	2	4	5	4	5	2	2	4	4	4

The survey was announced to the RDB2RDF community via mailing lists and personal e-mails. We received 13 individual responses. All participants assessed their level of RDB2RDF expertise as either *experienced* or *expert* (the two highest out of four possible levels). They could decide either to stay incognito or to fill in their name, affiliation and institution. Section 4 shows the results.

Due to the high level of participants' experience in the domain, we chose not to leverage the individual opinions by calculating their means or medians. Instead, we base our analysis on the *mode value*. We did not calculate separate mode values for responses from participants with different experience levels ("experienced" vs. "expert"), as we do not consider the difference between these two groups sufficiently significant to influence the responses (however, we indicate the level of experience in the result table).

As Sect. 4 shows, the experts approved most of the requirements. The ones that the experts were doubtful about were the *incorporation of domain semantics* and *simplicity* requirements. This outcome can, however, be explained by the circumstance that these two requirements are difficult to explain in a short statement. On the other hand, there were experts who accepted both these requirements, even with a value of *absolutely agree*. Thus, we consider the low overall agreement with these two requirements to result from incomplete understanding of the statements, as the requirements and their metrics are not trivial. Finally, within the open feedback no participant suggested further objective requirements (besides ones aggregated in the data quality requirement), which provides evidence of the completeness of our system, at least in view of today's state-of-the-art.

5 Conclusion

In this paper, we proposed a methodology to evaluate the quality of mappings between relational and linked data. In particular, we proposed a set of 14 requirements for mapping quality and described ways of measuring them. We believe that the most of the measures can be done (semi-)automatically and will attempt to prove that in our future work. As we show in the paper, the proposed system can not only be used to evaluate the mappings, but also as a guideline to increase the quality of the mapping. Additionally, we evaluated the relevance of our proposed set of requirements by conducting a survey. A total of 13 experienced individuals, mainly from the RDB2RDF community, participated in our survey. The analysis of their responses allows us to claim that the community accepts our requirements and their metrics.

Acknowledgments. We would like to thank the anonymous experts as well as Richard Cyganiak (INSIGHT@NUI Galway), Ravi Kumar (TCS), Mr. Spenser Kao (Bureau of Meteorology, Australia), Dr. Nikolaos Konstantinou (National Technical University of Athens), Freddy Priyatna (Ontology Engineering Group), Juan Sequeda (University of Texas at Austin), Tim Lebo (Tetherless World/RPI), Oscar Corcho (UPM), Dr. Pedro Szekely and Jose Luis Ambite (University of Southern California). We are grateful to Mr. Nicolas Van Peteghem for providing us with one of the test databases.

References

1. Arenas, M., et al.: A Direct Mapping of Relational Data to RDF. Working Draft. W3C (2012). http://www.w3.org/TR/rdb-direct-mapping/
2. Auer, S., et al.: Use Cases and Requirements for Mapping Relational Databases to RDF (2010). http://www.w3.org/2001/sw/rdb2rdf/use-cases/reqscore (visited on June 12, 2013)
3. Bertails, A., Prud'hommeaux, E.G.: Interpreting relational databases in the RDF domain. In: K-CAP, pp. 129–136 (2011)

4. Bizer, C.: Quality Driven Information Filtering: In the Context ofWeb Based Information Systems. PhD thesis. Free University of Berlin (2007)
5. Das, S., Sundara, S., Cyganiak, R.: R2RML: RDB to RDF Mapping Language. Recommendation. W3C (2012). http://www.w3.org/TR/rdb-direct-mapping/
6. Dean, M. et al.: OWL Web Ontology Language Reference. Recommendation. W3C (2004)
7. Erling, O.: Requirements for Relational to RDF Mapping (2008). http://www.w3.org/wiki/Rdb2RdfXG/ReqForMappingByOErling (visited on June 12, 2013)
8. Euzenat, J., Shvaiko, P.: Ontology alignment. Springer (2007)
9. Flemming, A.: Quality Characteristics of Linked Data Publishing Datasources. MA thesis. Humboldt-Universität of Berlin (2010)
10. Hillairet, G., Bertrand, F., Lafaye, J.-Y.: MDE for publishing data on the semantic web. In: TWOMDE, pp. 32–46. http://ceur-ws.org/Vol-395/
11. Hogan, A., et al.: An empirical survey of linked data conformance. Web Semantics **14**, 14–44 (2012)
12. Hogan, A., et al.: Weaving the pedanticweb. In: LDOW (2010). http://ceurws.org/Vol-628/
13. Liu, C., Li, J.: Designing quality XML schemas from E-R diagrams. In: Yu, J.X., Kitsuregawa, M., Leong, H.-V. (eds.) WAIM 2006. LNCS, vol. 4016, pp. 508–519. Springer, Heidelberg (2006)
14. Sahoo, S.S., et al.: A Survey of Current Approaches for Mapping of Relational Databases to RDF. RDB2RDF Incubator Group Report. W3C (2009)
15. Sahuguet, A.: Everything you ever wanted to know about DTDs, but were afraid to ask (extended abstract). In: Suciu, D., Vossen, G. (eds.) WebDB 2000. LNCS, vol. 1997, pp. 171–183. Springer, Heidelberg (2001)
16. Schmidt, A., et al.: XMark: a benchmark for XML data management. In: VLDB, pp. 974–985 (2002)
17. Shanmugasundaram, J., et al.: Efficiently publishing relational data as XML documents. The VLDB Journal **10**(2–3), 133–154 (2001)
18. Shanmugasundaram, J., et al.: Relational databases for querying xml documents: limitations and opportunities. In: VLDB, pp. 302–314. Morgan Kaufmann (1999)
19. Spanos, D.-E., Stavrou, P., Mitrou, N.: Bringing relational databases into the semantic web: A survey. Semantic Web **3**(2), 169–209 (2012)
20. Van Rijsbergen, C.: Information Retrieval (1979)
21. Zaveri, A., et al.: Quality Assessment Methodologies for Linked Open Data. Semantic Web Journal, major revision (2013). http://www.semantic-web-journal.net/content/quality-assessment-linked-open-datasurvey