Connectivity, Value and Evolution of a Semantic Warehouse

Michalis Mountantonakis[1,2], Nikos Minadakis[1], Yannis Marketakis[1],
Pavlos Fafalios[*,1,2], Yannis Tzitzikas[1,2]
[1] Institute of Computer Science, FORTH-ICS, GREECE, and
[2] Computer Science Department, University of Crete, GREECE
{mountant,minadakn,marketak,fafalios,tzitzik}@ ics.forth.gr

ABSTRACT

In many applications one has to fetch and assemble pieces of information coming from more than one source for building a *semantic warehouse* offering more advanced query capabilities. This chapter describes the corresponding requirements and challenges, and focuses on the aspects of quality, value and evolution of the warehouse. It details various metrics (or measures) for quantifying the connectivity of a warehouse and consequently the warehouse's ability to answer complex queries. The proposed metrics allow someone to get an overview of the contribution (to the warehouse) of each source and to quantify the value of the entire warehouse. Moreover, the paper shows how the metrics can be used for monitoring a warehouse after a reconstruction, thereby reducing the cost of quality checking and understanding its evolution over time. The behaviour of these metrics is demonstrated in the context of a real and operational semantic warehouse for the marine domain. Finally, the chapter discusses novel ways to exploit such metrics in global scale and for visualization purposes.

Keywords:
Linked Data, Semantic Warehouse, Measures of Connectivity, Quality, Evolution, Data Quality, Semantic Integration, Interlinking

INTRODUCTION

An increasing number of datasets are already available as Linked Data. For exploiting this wealth of data, and building domain specific applications, in many cases there is the need for fetching and assembling pieces of information coming from more than one sources. These pieces are then used for constructing a *Semantic Warehouse*, offering thereby more complete and efficient browsing and query services (in comparison to those offered by the underlying sources). The term *Semantic Warehouse* (for short warehouse) refer to a read-only set of RDF triples fetched (and transformed) from different sources that aims at serving a particular set of query requirements. In general, there exists *domain independent* warehouses, like the Sindice (Oren, et al., 2008) and SWSE (Hogan, et al., 2011), but also *domain specific*, like TaxonConcept[1] and the MarineTLO-based warehouse (Tzitzikas, et al., 2013, November). Domain specific warehouses aim to serve particular needs, for particular communities of users, consequently their "quality" requirements are stricter. It is therefore worth elaborating on the process that can be used for building such warehouses, and on the related difficulties and

* Current affiliation: L3S Research Center, University of Hannover, Germany
[1] TaxonConcept. http://www.taxonconcept.org/

challenges.

In brief, for building such a warehouse one has to tackle various challenges and questions, e.g., how to define the objectives and its scope, how to *connect* the fetched pieces of information (common URIs or literals are not always there), how to tackle the various issues of provenance that arise, and how to keep the warehouse fresh (i.e., how to automate its reconstruction or refreshing). This chapter has focused on the following questions:

- How to measure the value and quality of the warehouse (since this is important for e-science)?
- How to monitor its quality after each reconstruction or refreshing (as the underlying sources change)?
- How to understand the evolution of the warehouse?
- How to measure the contribution of each source to the warehouse, and hence deciding which sources to keep or exclude?

These questions have been encountered in the context of a real semantic warehouse for the *marine* domain which harmonizes and connects information from different sources of marine information[2]. Most past approaches have focused on the notion of conflicts (Michelfeit & Knap, 2012), and have not paid attention to *connectivity*. The term *connectivity* express the degree up to which the contents of the warehouse form a connected graph that can serve, ideally in a correct and complete way, the query requirements of the warehouse, while making evident how each source contributes to this degree. Besides, connectivity is a notion which can be exploited in the task of dataset or endpoint selection.

To this end, this chapter summarizes the methods and metrics introduced in Tzitzikas et al. (2014, March) and Mountantonakis et al. (2016) for quantifying the connectivity of a warehouse, reports their implementation on real datasets, and discusses interesting and novel works that exploit them. What the authors call *metrics* could be also called *measures*, i.e. they should not be confused with distance functions. These metrics allow someone to get an overview of the contribution (to the warehouse) of each source (enabling the discrimination of the important from the non-important sources) and to quantify the value (benefit) of such a warehouse. In a nutshell, this chapter presents:

- An extensive report on related literature on *dataset quality* and *quality assessment frameworks*, as well as the placement of the presented work.
- A set of *connectivity metrics* for comparing pairs and sets (lattice-based) of sources.
- A set of *single-valued metrics* for evaluating the overall contribution and the value of each source as well as the quality of the entire warehouse. The former makes easier and faster the identification and inspection of pathological cases (redundant sources or sources that do not contribute new information).
- Methods that exploit the proposed metrics for *understanding* and *monitoring* the *evolution* of the warehouse.
- Novel ways for exploiting such metrics in *global scale* and for *visualization* purposes.

The rest of this chapter is organized as follows: The second section describes the main requirements, and provides the context by briefly describing the process used for constructing such warehouses. The third section describes related work and what distinguishes the current one. The fourth section introduces the quality metrics and demonstrates their use. The fifth section discusses how these metrics can be used for monitoring and understanding the evolution of the warehouse over time, while the sixth section describes novel ways for exploiting the metrics in global scale and for visualizing LOD datasets. Finally, the seventh

[2] Used in the context of the projects iMarine (FP7 Research Infrastructures, 2011-2014, http://www.i-marine.eu) and BlueBRIDGE (H2020 Research Infrastructures, 2015-2018, http://www.bluebridge-vres.eu).

section identifies future research directions, while the last section concludes the chapter and identifies directions for future research.

BACKGROUND

Context and Requirements

The spark for this work was the recently completed *iMarine* project (and the ongoing *BlueBRIDGE* project) that offers an operational distributed infrastructure that serves hundreds of scientists from the marine domain. As regards semantically structured information, the objective was to integrate information from various marine sources, specifically from:

- **WoRMS**[3]: marine registry containing taxonomic information and lists of common names and synonyms for more than 200 thousand species in various languages.
- **Ecoscope**[4]: knowledge base containing geographical data, pictures and information about marine ecosystems.
- **FishBase**[5]: global database of fish species, containing information about the taxonomy, geographical distribution, biometrics, population, genetic data and many more.
- **FLOD**[6]: is a network of marine linked data containing identification information using different code lists.
- **DBpedia** (Bizer, et al., 2009): knowledge base containing content that has been converted from Wikipedia, that by the time of writing this chapter, the English version contained more than 4.5 million resources.

For integrating the sources, *MarineTLO* top-level ontology (Tzitzikas, et al., 2013, November) was used. The integrated warehouse[7] is operational and it is exploited in various applications, including the gCube infrastructure (Candela, et al., 2010), or for enabling exploratory search services, e.g., (Fafalios & Tzitzikas, 2013, July), (Fafalios & Tzitzikas, 2014) that offers semantic post-processing of search results.

Warehouse Construction Process

Figure 1 sketches the construction process. For this, the tool *MatWare* (Tzitzikas, et al., 2014, May) can be used which *automates* the entire process. The proposed metrics are used in steps 4 and 8, and are important for *monitoring* the warehouse after a reconstruction. For example by comparing the metrics in the past and new warehouse, one can understand whether a change in the underlying sources affected positively or negatively the quality (connectivity) of the warehouse. More information about these steps can be found in (Tzitzikas, et al., 2014, May) and (Mountantonakis, et al., 2016).

[3] WoRMS - World Register of Marine Species (http://www.marinespecies.org).
[4] Ecoscope - Knowledge Base on Exploited Marine Ecosystems (http://www.ecoscopebc.ird.fr).
[5] FishBase (http://www.fishbase.org).
[6] FLOD - Fisheries Linked Open Data (http://www.fao.org/figis/flod/).
[7] The warehouse can be accessed from https://i-marine.d4science.org/.

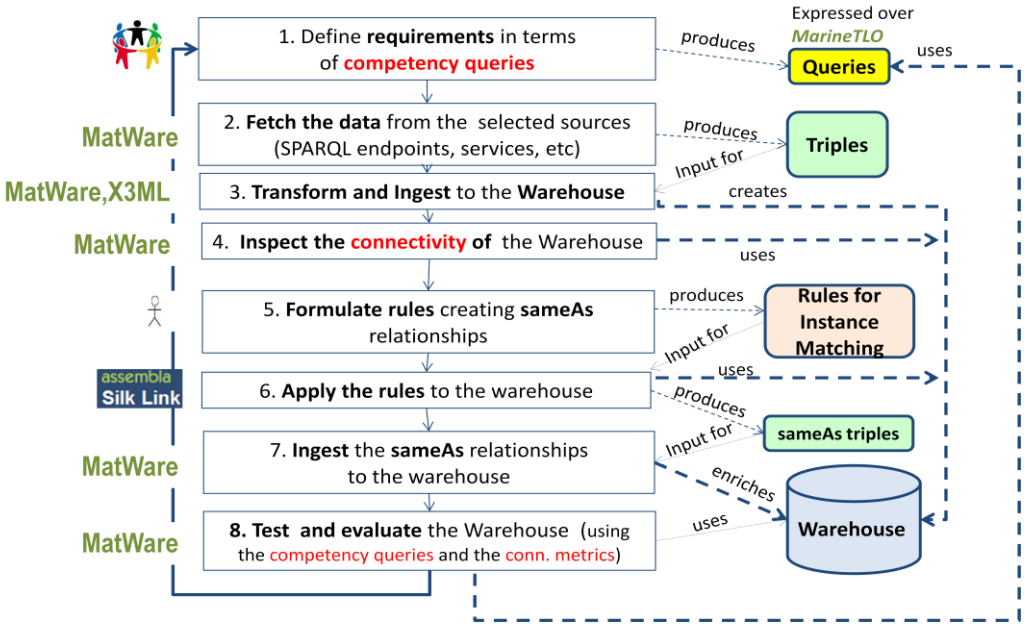

Figure 1. Warehouse Construction and Monitoring

RELATED WORK

Data quality is commonly conceived as *fitness of use* for a certain application or use case ((Knight & Burn, 2005), (Wang & Strong, 1996)). The issue of data quality, especially for the case of a *data warehouse*, is older than the RDF world, e.g., the database community has studied it in the relational world ((Ballou & Tayi, 1999), (Shanks & Darke, 1998)). *Connectivity*, as defined in the Introduction, can be considered as a dimension of data quality in the context of a Semantic Warehouse.

Fürber and Hepp (2010, May) investigated data quality problems for RDF data originating from relational databases, while a systematic review of approaches for assessing the data quality of Linked Data is presented by (Zaveri, et al., 2016). In that work, the authors surveyed 21 approaches and extracted 26 data quality dimensions (such as *completeness*, *provenance*, *interlinking*, *reputation*, *accessibility*, and others) along with the corresponding metrics.

Below, the authors first discuss some quality aspects that are especially useful for the case of a Semantic Warehouse, they report approaches that have tried to address them and they place their work in the literature. Then they compare (according to various perspectives) several frameworks and systems that automate quality assessment for the RDF world.

Quality Aspects

Completeness. *Completeness* refers to the degree up to which all required information is presented in a particular dataset. In the RDF world, completeness can be classified (according to (Zaveri, et al., 2016)) as: *schema completeness* (degree to which the classes and properties of an ontology are represented), *property completeness* (measure of the missing values for a specific property), *population completeness* (percentage of all real-world objects of a particular type that are represented in the datasets), and *interlinking completeness* (degree to which instances in the dataset are interlinked).

The problem of assessing completeness of Linked Data sources was discussed by Harth and Speiser (2012, July). Darari et al. (2013) introduce a formal framework for the declarative

specification of *completeness statements* about RDF data sources and underline how the framework can complement existing initiatives like VoID (Keith Alexander, et al., 2011). They also show how to assess completeness of query answering over plain and RDF/S data sources augmented with completeness statements, and they present an extension of the completeness framework for federated data sources.

Provenance. *Provenance* focuses on how to represent, manage and use information about the origin of the source to enable trust, assess authenticity and allow reproducibility (Zaveri, et al., 2016). Hartig (2009) presents a provenance model for Web data which handles both data creation and data access. The author also describes options to obtain provenance information and analyzes vocabularies to express such information. Hartig and Zhao (2009) propose an approach of using provenance information about the data on the Web to assess their quality and trustworthiness. Specifically, the authors use the provenance model described by (Hartig, 2009) and propose an assessment method that can be adapted for specific quality criteria (such as accuracy and timeliness). This work also deals with missing provenance information by associating certainty values with calculated quality values. The same authors (Hartig & Zhao, 2010) introduce a vocabulary to describe provenance of Web data as metadata and discuss possibilities to make such provenance metadata accessible as part of the Web of Data. Furthermore, they describe how this metadata can be queried and consumed to identify outdated information. Given the need to address provenance, the W3C community has standardised the PROV Model[8], a core provenance data model for building representations of the entities, people and processes involved in producing a piece of data or thing in the world. The PROV Family of Documents [9] (Misser, et al., 2013, March) defines the model, corresponding serializations and other supporting definitions to enable the inter-operable interchange of provenance information in heterogeneous environments such as the Web.

Amount-of-data. *Amount-of-data* is defined as the extent to which the volume of data is appropriate for the task at hand according to (Bizer, 2007) and (Zaveri, et al., 2016). This dimension can be measured in terms of general dataset statistics like number of triples, instances per class, internal and external links, but also coverage (scope and level of detail) and metadata "richness". Tsiflidou and Manouselis(2013) carried out an analysis of tools that can be used for the valid assessment of metadata records in a repository. More specifically, three different tools are studied and used for the assessment of metadata quality in terms of statistical analysis. However, such works do not consider the characteristics of RDF and Linked Data. Auer et al. (2012) describe LODStats, a statement-stream-based approach for gathering comprehensive statistics (like classes/properties usage, distinct entities and literals, class hierarchy depth, etc.) about RDF datasets. To represent the statistics, they use VoID and the RDF Data Cube Vocabulary. The RDF Data Cube Vocabulary[10] (Cyganiak, et al., 2010) provides a means to publish multi-dimensional data (such as statistics of a repository) on the Web in such a way that it can be linked to related datasets and concepts. Hogan et al. (2012) performed analysis in order to quantify the conformance of Linked Data with respect to Linked Data guidelines (e.g., use external URIs, keep URIs stable). They found that in most datasets, publishers followed some specific guidelines, such as using HTTP URIs, whereas in other cases, such as providing human readable metadata, the result were disappointing since only a few publishers created metadata for their datasets.

[8]http://www.w3.org/TR/2013/NOTE-prov-primer-20130430/
[9]http://www.w3.org/TR/prov-overview/
[10]http://www.w3.org/TR/2013/PR-vocab-data-cube-20131217/

Accuracy. *Accuracy* is defined as the extent to which data is correct, that is, the degree up to which it correctly represents the real world facts and is also free of errors (Zaveri, et al., 2016). Accuracy can be measured by detecting outliers, conflicts, semantically incorrect values or poor attributes that do not contain useful values for the data entries. Fürber and Hepp (2011) categorize accuracy into semantic and syntactic accuracy. Semantic accuracy checks whether the data value represents the correct state of an object, whereas syntactic accuracy checks if a specific value violates syntactical rules. For measuring accuracy, the authors used three rules and four formulas, whereas the results were evaluated by using precision and recall measures. They managed to detect syntactic and semantic errors such as invalid country combinations, rules for phone numbers and so forth. *ODCleanStore* (Knap, et al., 2012), (Michelfeit & Knap, 2012) names *conflicts* the cases where two different quads (e.g., triples from different sources) have different object values for a certain subject and predicate. To such cases conflict resolution rules are offered that either select one or more of these conflicting values (e.g., ANY, MAX, ALL), or compute a new value (e.g., AVG). Finally, Knap and Michelfeit (2012) describe various quality metrics for scoring each source based on conflicts, as well for assessing the overall outcome. In (Liu, et al., 2017), the authors computed the joint distribution of variables on a network called Source-Object network. This network captures three different types of correlations in order to trust data derived from reliable sources and it takes into account the sharing between different datasets.

Relevancy. *Relevancy* refers to the provision of information which is accordant with the task at hand and suitable to the users' query (Zaveri, et al., 2016). The existence of irrelevant data can have negative consequences for the query performance, while it will be difficult for the user to explore this data, since the user expects to receive the correct information. Zaveri et al. (2013, September) divide relevancy (for DBpedia) into the following sub-categories: (i) extraction of attributes containing layout information, (ii) image related information, (iii) redundant attribute values, and finally (iv) irrelevant information. The existence of a number of different properties for a specific subject-object pair is an example of redundant information.

Dynamics / Evolution. *Dynamics* quantifies the evolution of a dataset over a specific period of time and takes into consideration the changes occurring in this period. Dividino et. al (2014) lists probably all works related to the dynamics of LOD datasets. A related quality perspective, identified by Tzitzikas et al. (2014), is that of the *specificity* of the ontology-based descriptions under ontology evolution, an issue that is raised when ontologies and vocabularies evolve over time.

Interlinking. *Interlinking* refers to the degree to which entities that represent the same concept are linked to each other (Zaveri, et al., 2016). This can be evaluated by measuring the existence of **sameAs** links and chains, the interlinking degree, etc. Zaveri et al. (2013, September) classify interlinking into two different categories: (i) external websites (checking whether there are links among sources which are not available), and (ii) interlinks with other datasets (trying to detect incorrect mappings and links which do not provide useful information). The authors of (Nentwig, et al., 2014) created a portal for Link Discovery, called LinkLion, which contains mappings between pairs of 462 datasets. In that repository, one can find relationships between any pair of datasets. Furthermore, in (Gimenez-Garcia, et al., 2016) the authors computed the Pagerank for 319 datasets for providing a trust measure based on dataset interlinking.

Our Placement: Connectivity

The term *connectivity* express the degree up to which the contents of the semantic warehouse form a connected graph that can serve, ideally in a correct and complete way, the query

requirements of the semantic warehouse, while making evident how each source contributes to that degree. The proposed connectivity metrics reflect the query capabilities of a warehouse as a whole (so they are important for evaluating its value), but also quantify the contribution of the underlying sources allowing evaluating the importance of each source for the warehouse at hand. Connectivity is important in warehouses whose schema is not small and consequently the queries contain paths. The longer such paths are, the more the query capabilities of the warehouse are determined by the connectivity.

In the related literature, the aspect of *connectivity* is not covered sufficiently and regards mainly the existence of **sameAs** links and chains (Zaveri, et al., 2013, September). Regarding the association of *connectivity* with existing quality dimensions, it is predominantly *interlinking* and secondly *relevancy* and *amount-of-data*. Of course, it can be exploited together with approaches that focus on *completeness* (Darari, et al., 2013), *provenance* (Hartig & Zhao, 2009), *accuracy* (Knap & Michelfeit, 2012), etc. Regarding *relevancy*, a Semantic Warehouse as created by the proposed process (see Figure 1) does not contain irrelevant data since the data has been fetched based on the requirements defined in terms of competency queries. Furthermore, the proposed metrics can even detect redundant sources and sources containing data which are not connected with data found in the other sources. Compared to existing approaches on *amount-of-data* (like **LODStats** (Auer, et al., 2012)), the proposed connectivity metrics can be used to gather statistics that regard more than one source (like common URIs, common literals, etc.) Finally, as regards *dynamics/evolution*, existing works (Dividino, et al., 2014) concern atomic datasets, not warehouses comprising parts of many datasets.

Frameworks/Systems for Quality Assessment

Here, the authors discuss frameworks/systems that automate quality assessment. At first they give a brief description of each framework/system and what quality aspects it can handle, and then they compare them regarding several aspects.

ODCleanStore (Michelfeit & Knap, 2012) is a tool that can download content (RDF graphs) and offers various transformations for cleaning it (deduplication, conflict resolution), and linking it to existing resources, plus assessing the quality of the outcome in terms of *accuracy*, *consistency*, *conciseness* and *completeness*.

Sieve (Mendes, et al., 2012, March) is part of the Linked Data Integration Framework (LDIF) [11] and proposes metrics for assessing the dimensions in terms of *schema completeness*, *conciseness* and *consistency*. The role of this tool is to assess the quality by deciding which values to keep, discard or transform according to a number of metrics and functions which are configurable via a declarative specification language.

RDFUnit (Kontokostas, et al., 2014, April) measures the *accuracy* and the *consistency* of a dataset containing Linked Data. More specifically, it checks the correct usage of vocabularies according to a number of constraints (e.g., cardinality restriction on a property). One can use some custom SPARQL queries to quantify the quality of a specific dataset for the aforementioned aspects.

LinkQA (Guéret, et al., 2012) uses a number of metrics to assess the quality of Linked Data mappings regarding the dimensions of *interlinking* and *completeness*. This tool can be used for detected pathological cases, such as bad quality links, before they are published.

Luzzu (Debattista, et al., 2015) is a framework for assessing the quality of Linked data for 10 different dimensions, such as *availability*, *provenance*, *consistency* and so forth. In particular, by using this tool one can perform quality evaluation either by using some of the 25 available metrics or by defining his own metrics.

[11] Linked Data Integration Framework (LDIF) - http://www4.wiwiss.fu-berlin.de/bizer/ldif/

SWIQA (Fürber & Hepp, 2011) is a framework for the validation of the values of semantic resources based on a set of rules. This framework allows the calculation of quality scores for various dimensions, such as *completeness*, *timeliness* and *accuracy*, in order to identify possible problems with the data values. Moreover, it is also applicable on top of relational databases with the support of wrapping technologies (i.e., D2RQ).

SeaStar (Sarasua, et al., 2017) is a framework that analyzes the available links or a set of sources according to the set of principles of data interlinking. Its goal is to assess the quality and the accuracy of existing links, and to understand the gain of the connectivity for a source dataset when it is connected to a target dataset.

Finally, *MatWare* (Tzitzikas, et al., 2014, May) is a tool that automates the process of constructing semantic warehouses by fetching and transforming RDF triples from different sources. To this end MatWare exploits several external tools like SILK framework (Volz, Bizer, Gaedke, & Kobilarov, 2009), X3ML Engine (Marketakis, et al., 2016), and computes and visualizes the connectivity metrics described in this chapter.

Figure 2 illustrates the dimensions that the aforementioned approaches measure, while Table 1 provides a categorization according to several aspects.

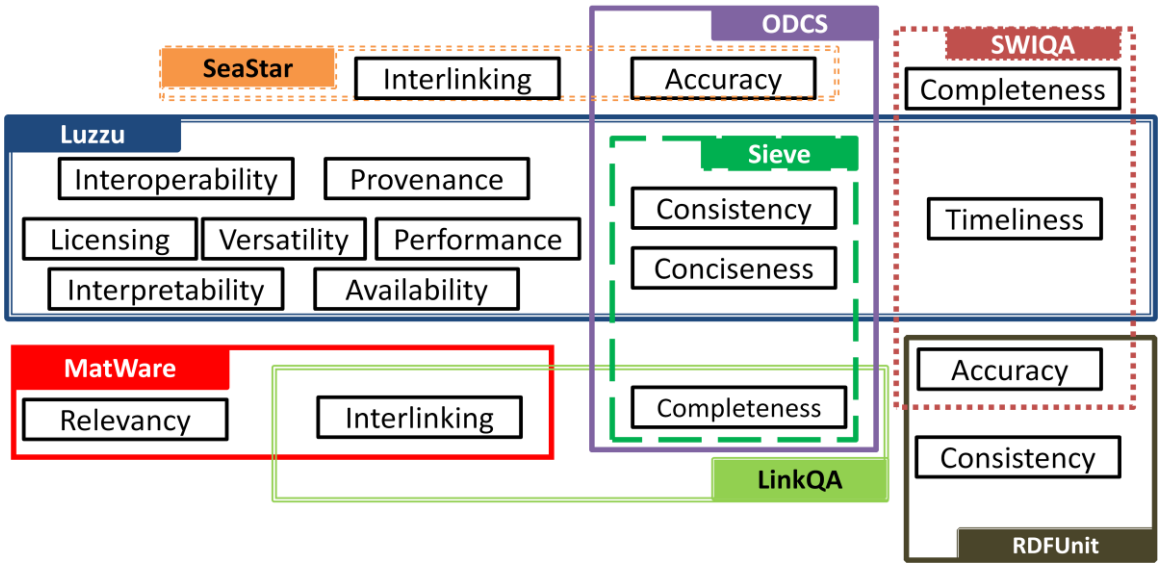

Figure 2. Existing frameworks and the dimensions they measure

	LinkQA	**Luzzu**	**ODCleanStore**	**Sieve**	**SWIQA**	**RDFUnit**	**SeaStar**	**MatWare**
Input	RDF/ XML	RDF/ XML	RDF/ XML	RDF/ XML	RDF/ XML	RDF/ XML	RDF/ XML	RDF/XML
Number Of Sources	Set of Mappings	One Source	Collection of Quads	One (integrated) Source	One Source	One Source	One or Two Sources	Set of Sources
Output Kind	Numeric Values	Numeric Values	Numeric Values	Numeric Values	Numeric Values	Numeric Values	Numeric Values	Numeric Values, 3D
Output Format	HTML	RDF	RDF, HTML	Quads	HTML	RDF, HTML	HTML	RDF, 3D, HTML
Computability	JAVA	JAVA	JAVA	JAVA	SPARQL	SPARQL	JAVA	SPARQL, JAVA
Extensible	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

Table 1. Categorizing existing frameworks.

CONNECTIVITY METRICS

Def. 1. The term *connectivity metric* (or *connectivity measure*) refer to a measurable quantity that expresses the degree up to which the contents of the semantic warehouse form a connected graph that can serve, ideally in a correct and complete way, the query requirements of the semantic warehouse, while making evident how each constituent source contributes to that degree. ⋄

They include measures of similarity between two sources in the form of percentages (e.g. regarding common URIs), natural numbers (e.g. cardinalities of intersections), matrices of measures, means of other measures, as well as relative measures (e.g. increase of average degrees, unique contribution and others). Such measures can assist humans on assessing in concrete terms the quality and the value offered by the warehouse. In addition they provide a summary of the contents of the warehouse which can be exploited by external applications in the context of distributed query answering.

To aid understanding, after defining each metric the authors show the values of these metrics as computed over the **MarineTLO**-based warehouse which is built using data from five marine-related sources (FLOD, WoRMS, Ecoscope, DBpedia, FishBase). Since the warehouse is real and operational, this way of presentation also allows the reader to see how the metrics behave in a real setting.

This section is organized as follows: At first, it introduces notations and discusses ways for comparing URIs. Then it introduces metrics for comparing *pairs of* sources and metrics for comparing a *set* of sources. Subsequently it introduces metrics for quantifying the value of the entire warehouse as well as metrics for quantifying the value of one source (in the context of one warehouse).

Notations and Ways to compare URIs

At first some required notations are introduced. Let $S = S_1, \ldots S_k$ be the set of underlying sources. Each contributes to the warehouse a set of triples (i.e., a set of **subject-predicate-object** statements), denoted by $triples(S_i)$. This is not the set of all triples of the source. It is the subset that is contributed to the warehouse (fetched mainly by running SPARQL queries). U_i denotes the URIs that appear in $triples(S_i)$. Hereafter, only those URIs that appear as *subjects* or *objects* in a triple are considered. The URIs of the properties are not included because they concern the schema and this integration aspect is already tackled by the top level schema. Let W denote triples of all sources of the warehouse. In general, the set of all triples of the warehouse, say W_{All}, is superset of W (i.e., $W_{All} \supset W = \bigcup_{i=1}^{k} triples(S_i)$) because the warehouse apart from the triples from the sources, contains also the triples representing the top-level ontology, the schema mappings, the **sameAs** relationships, etc.

On Comparing URIs. For computing the metrics that are defined next, methods are needed to compare URIs coming from different sources. There are more than one method, or policy, for doing so. Below three main policies are distinguished:

i. *Exact String Equality.* Two URIs u_1 and u_2 are equal, denoted by $u_1 \equiv u_2$, if $u_1 = u_2$ (i.e., strings equality).

ii. *Suffix Canonicalization.* $u_1 \equiv u_2$ if $last(u_1) = last(u_2)$ where $last(u)$ is the string obtained by a) getting the substring after the last "/" or "#", and b) turning the letters of

the picked substring to lowercase and deleting the underscore letters as well as space and special characters that might exist. According to this policy:
http://www.dbpedia.com/Thunnus_Albacares≡
http://www.ecoscope.com/thunnus_albacares
since their canonical suffix is the same, i.e., **thunnusalbacares.** Another example of equivalent URIs:
http://www.s1.com/entity#thunnus_albacares≡
http://www.s2.org/entity/thunnusAlbacares

iii. *Entity Matching.* $u_1 \equiv u_2$ if u_1 sameAs u_2 according to the entity matching rules that are (or will be eventually) used for the warehouse. In general such rules create **sameAs** relationships between URIs. Here, SILK framework is used for formulating and applying such rules.

The presented metrics are defined and computed assuming policy [ii], i.e., whenever there exists a set operation, equivalence according to policy [ii] is assumed (e.g., $A \cap B$ means $\{ a \in A \mid \exists\, b \in B \text{ s.t. } a \equiv_{[ii]} b \}$). Then, after applying the entity matching rules, the metrics are computed according to policy [iii], which actually characterizes the query behaviour of the final and operational warehouse.

Metrics for Comparing two Sources

Matrix of Percentages of Common URIs

The number of *common URIs* between two sources S_i and S_j, is given by $\left|U_i \cap U_j\right|$. The *percentage of common URIs* (a value ranging $[0,1]$) is defined as follows:

$$curi_{i,j} = \frac{\left|U_i \cap U_j\right|}{\min\left(\left|U_i\right|, \left|U_j\right|\right)} \quad (1)$$

In the denominator $\min\left(\left|U_i\right|, \left|U_j\right|\right)$ is used although one could use $\left|U_i \cap U_j\right|$ that is used in the Jaccard similarity. With Jaccard similarity the integration of a small triple set with a big one would always give small values, even if the small set contains many URIs that exist in the big set, while the Jaccard similarity reveals the overall contribution of a source. Now, the above metric is extended and consider *all pairs of* sources aiming at giving an overview of the warehouse. Specifically, a $k \times k$ matrix is computed where $c_{i,j} = curi_{i,j}$. The higher values this matrix contains, the more glued its "components" are.

For the warehouse at hand, Table 2 shows the matrix of the common URIs (together with the corresponding percentages). One can notice that the percentages range from 0.3% to 27.39%, while in some cases one can observe a significant percentage of common URIs between the different sources. The biggest intersection is between FishBase and DBpedia.

S_i / S_j	FLOD	WoRMS	Ecoscope	DBpedia	FishBase
FLOD	173,929 (100%)	239 (0.3%)	523 (8.98%)	631 (0.9%)	887 (2.54%)
WoRMS		80,485 (100%)	200 (3.43%)	1,714 (2.44%)	3,596 (10.28%)
Ecoscope			5,824 (100%)	192 (3.3%)	225 (3.86%)
DBpedia				70,246 (100%)	9,578 (27.39%)
FishBase					34,974 (100%)

Table 2. Matrix of common URIs (with their percentages) using Policy [ii].

Measurements after Adding the Rule-derived 'sameAs' Relationships and Applying the Transformation Rules

So far in the computation of the above metrics policy [ii] was used (suffix canonicalized URIs) when comparing URIs. Here the authors show the results from computing again these metrics using policy [iii] and after adding the triples as derived from the transformation rules described earlier. Moreover, extra URIs have been produced due to transformation rules (e.g., in order to assign a URI to a species name). As a result, now when comparing URIs, the authors consider the **sameAs** relationships that have been produced by the entity matching rules of the warehouse. In the current warehouse 11 SILK rules were used. An indicative SILK rule is the following: *"If the value of the attribute "preflabel" of an Ecoscope individual (e.g., Thunnus albacares) in lower case is the same with the attribute "label" in latin of a FLOD individual (e.g., 'thunnus albacares'@la), then these two individuals are the same (create a sameAs link between them)"*. It is worth noting that policy [ii] considers the triples as they are fetched from the sources. Computing the metrics using policy [iii], not only allows evaluating the gain achieved by these relationships, but it also reflects better the value of the warehouse since query answering considers the **sameAs** relationships.

Table 3 shows the matrix of the common URIs after the rule-derived relationships and the execution of the transformation rules (together with the corresponding percentages). One can see that, compared to the results of Table 2, after considering the **sameAs** relationships the number of common URIs between the different sources is significantly increased (more than 7 times in some cases).

Furthermore, Table 4 shows the Jaccard similarity between the pairs of sources. By comparing the results between these two tables, one can see that the percentages when using Jaccard similarity table have been reduced remarkably.

S_i / S_j	FLOD	WoRMS	Ecoscope	DBpedia	FishBase
FLOD	190,749 (100%)	1,738 (2.64%)	869 (11.2%)	4,127 (5.46%)	6,053 (17.31%)
WoRMS		65,789 (100%)	809 (10.43%)	1,807 (2.75%)	4,373 (12.5%)
Ecoscope			7,759 (100%)	1,117 (14.4%)	2,171 (27.98%)
DBpedia				75,518 (100%)	10,388 (29.7%)
FishBase					34,973 (100%)

Table 3. Matrix of common URIs (and their percentages) using Policy [iii].

S_i / S_j	FLOD	WoRMS	Ecoscope	DBpedia	FishBase
FLOD	1	0.68%	0.44%	1.56%	2.69%
WoRMS		1	1.11%	1.29%	4.5%
Ecoscope			1	1.36%	5.35%
DBpedia				1	10.31%
FishBase					1

Table 4. Matrix of percentages of common URIs using Policy [iii] and Jaccard Similarity.

Matrix of Percentages of Common Literals between two Sources

The *percentage of common literals*, between two sources S_i and S_j can be computed by:

$$colit_{i,j} = \frac{\left|Lit_i \cap Lit_j\right|}{\min\left(\left|Lit_i\right|, \left|Lit_j\right|\right)} \quad (2)$$

To compare two literals coming from different sources, the authors convert them to lower case, to avoid cases like comparing "Thunnus" from one source and "thunnus" from another. Additionally, they ignore the language tags (e.g., "$salmon"@en \equiv$ "$salmon"@de$). Table 5 shows the matrix of the common literals (together with the corresponding percentages). One can see that, as regards the literals, the percentages of similarity are even smaller than the ones regarding common URIs. The percentages range from 2.71% to 12.37%.

S_i / S_j	FLOD	WoRMS	Ecoscope	DBpedia	FishBase
FLOD	111,164 (100%)	3,624 (7.1%)	1,745 (12.37%)	5,668 (5.1%)	9,505 (8.55%)
WoRMS		51,076 (100%)	382 (2.71%)	2,429 (4.76%)	4,773 (9.34%)
Ecoscope			14,102 (100%)	389 (2.76%)	422 (2.99%)
DBpedia				123,887 (100%)	14,038 (11.33%)
FishBase					138,275 (100%)

Table 5. Matrix of common Literals (and their percentages).

Matrix of the Harmonic Mean of Common URIs and Literals

By combining the previous two metrics, a single metric is now defined which actually corresponds to their harmonic mean:

$$cUrisLit_{i,j} = \frac{2 * curi_{i,j} * colit_{i,j}}{curi_{i,j} + colit_{i,j}} \quad (3)$$

Table 6 presents the results of this metric.

S_i / S_j	FLOD	WoRMS	Ecoscope	DBpedia	FishBase
FLOD	1	1.61%	10.44%	1.53%	3.93%
WoRMS		1	4.45%	11.02%	21.86%
Ecoscope			1	3.08%	3.82%
DBpedia				1	15.85%
FishBase					1

Table 6. Harmonic Mean of Common URIs and Literals.

Metrics for Comparing a Set of Sources (Lattice-based)

The measurements described earlier are measurements between pairs of sources. However, one can generalize the present metrics between *any subset* of the sources of the warehouse, e.g., the number of common literals in 4 sources.

The idea is to provide measurements for each *subset* of the set of sources, i.e., for every element of $P(S)$ where $P(S)$ denotes the powerset of S. For visualizing (and understanding) these measurements, the authors propose a partial set-like visualization. Specifically, they propose constructing and showing the measurements in a way that resembles the *Hasse Diagram* of the *poset* (partially ordered set) $(S, \subseteq)$.

Let R be any nonempty subset of S (i.e., $R \subseteq S$). It is not hard to generalize the aforementioned metrics for every such subset. For example, consider the metric common triples. The nodes at the lower level of the Hasse Diagram (corresponding to the singletons of S), actually show the number of triples of each source in S, while the nodes in the level above correspond to pairs of sources, i.e., they correspond to what the matrices show. At the topmost level, one can see the intersection between all sources in S (i.e., the value $|U_1 \cap \ldots \cap U_k|$).

Figure 3 presents the lattice concerning the common URIs according to policy [ii]. One can see that the number of common URIs of DBpedia, FishBase, and Ecoscope are more than the number of common URIs among the subsets of the same level, while 74 common URIs are included in all sources.

This approach can be used for all metrics, e.g., for common URIs, for common literals, as well for the metrics that will be presented later for the entire warehouse. The diagram contains $2^{|S|}$ nodes.

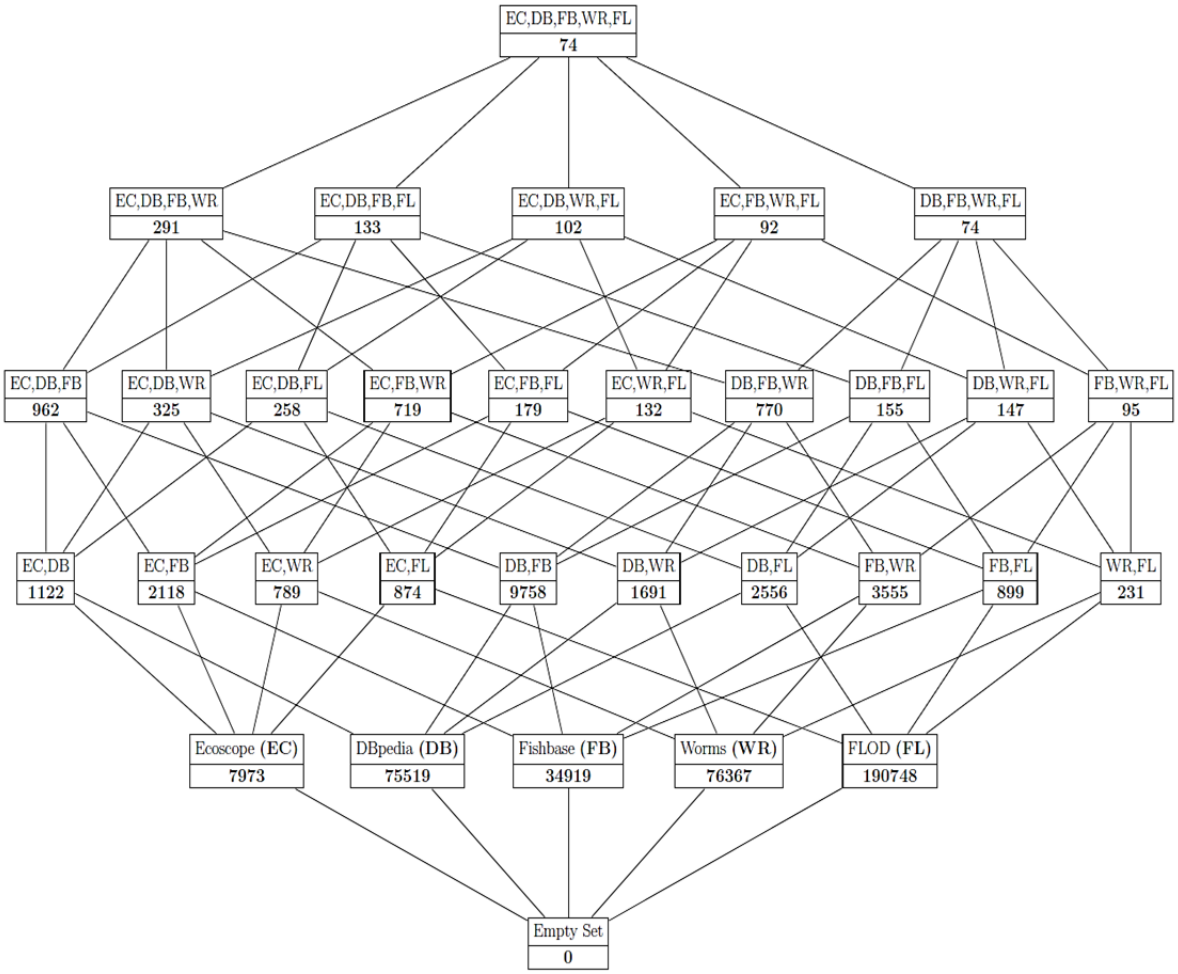

Figure 3. Common URIs Lattice using Policy [ii].

Metrics for Evaluating the Entire Warehouse

Here the authors introduce metrics that measure the quality of the entire warehouse.

Increase in the Average Degree

Now, another metric for expressing the degree of a set of nodes is presented, where a node can be either a URI or a blank node (in our case the size of blanks nodes is much bigger (about twice) than the size of unique triples). Let E be the entities of interest (or the union of all URIs and blank nodes).

If T is a set of triples, then the *degree* of an entity e in T is defined as: $deg_T(e) = |\{(s,p,o) \in T \mid s = e \text{ or } o = e\}|$ while for a set of entities E their average degree in T is defined as $deg_T(E) = avg_{e \in E}(deg_T(e))$. Now for each source S_i one can compute the average degree of the elements in E considering $triples(S_i)$. If the sources of the warehouse contain common elements of E, then if one compute the degrees in the graph of W (i.e., $deg_W(e)$ and $deg_W(E)$), they will get higher values. So the increase in the degree is a way to quantify the gain, in terms of connectivity, that the warehouse offers. Furthermore, a normalized metric for average degree increment can be defined i.e., a metric whose value

approaches 1 in the best case, and 0 in the worst. To this end, the authors define:

$$DegIncr(S_i, W) = \frac{deg_W(U_i) - deg_{S_i}(U_i)}{deg_W(U_i)} \quad (4)$$

For each source S_i, Table 7 shows the average degree of its URIs and blank nodes, and the average degree of the same URIs and blank nodes in the warehouse graph. It also reports the increment percentage, and the normalized metric for the average degree increment. The last row of the table shows the average values of each column. One can observe that the average degree is increased from 9.39 to 15.29.

S_i	avg $deg_{S_i}(U_i)$	avg $deg_W(U_i)$	Increase %	$DegIncr(S_i, W)$
FLOD	5.82	6.8	16.84%	0.14
WoRMS	4.14	4.24	2.46%	0.02
Ecoscope	21.37	47.56	122.52%	0.55
DBpedia	6.9	6.99	1.36%	0.01
FishBase	8.75	10.89	24.46%	0.19
AVERAGE	**9.39**	**15.29**	**62.89%**	**0.38**

Table 7. Average degrees in sources and in the warehouse using policy [ii].

Measurements after Adding the Rule-derived 'sameAs' Relationships and Applying the Transformation Rules

Table 8 shows the average degree of the URIs and blank nodes of each source S_i, and the average degree of the same URIs and blank nodes in the warehouse graph, when policy [iii] is used. The last row of the table shows the average values of each column. One can see that the average degree, of all sources, after the inclusion of the **sameAs** relationships is significantly bigger than before. In comparison to Table 7, the increase is from 2 to 9 times bigger. This means that a great increase was achieved in terms of the connectivity of the information in the warehouse.

S_i	avg $deg_{S_i}(U_i)$	avg $deg_W(U_i)$	Increase %	$DegIncr(S_i, W)$
FLOD	5.82	52.01	739.64%	0.89
WoRMS	4.14	8.19	97.94%	0.49
Ecoscope	21.37	90.52	323.51%	0.76
DBpedia	6.9	42.97	523.23%	0.84
FishBase	8.71	18.99	117.19%	0.54
AVERAGE	**9.39**	**42.53**	**353%**	**0.78**

Table 8. Average degrees in sources and in the warehouse using Policy [iii].

Unique Triples Contribution

The authors now define metrics for quantifying the complementarity of the sources. The "contribution" of each source S_i can be quantified by counting the triples it has provided to the warehouse, i.e., by $|triples(S_i)|$. Moreover, its "unique contribution" can be computed by excluding from $|triples(S_i)|$ those belonging to the triples returned by the other sources. Formally, for the k sources of the warehouse, it is defined:

$$triplesUnique(S_i) = triples(S_i) \setminus \left(\cup_{1 \le j \le k, j \ne i} triples(S_j) \right) \quad (5)$$

It follows that if a source S_i provides triples which are also provided by other sources, then $triplesUnique(S_i) = \varnothing$. Consequently, and for quantifying the contribution of each source to the warehouse, one can compute and report the number of its triples $|triples(S_i)|$, the number of the unique triples $|triplesUnique(S_i)|$, the unique contribution of each source as:

$$UniqueTContrib(S_i) = \frac{|triplesUnique(S_i)|}{|triples(S_i)|} \quad (6)$$

Obviously, it becomes 0 in the worst value and 1 in the best value. To count the unique triples of each source, for each triple of that source the authors perform suffix canonicalization on its URIs, convert its literals to lower case, and then check if the resulting (canonical) triple exists in the canonical triples of a different source. If not, they count this triple as unique. Let *triplesUnique* be the union of the unique triples of all sources, i.e., $triplesUnique = \cup_i triplesUnique(S_i)$ This set can be proper subset of *W* (i.e., $triplesUnique \subset W$), since it does not contain triples which have been contributed by two or more sources.

Table 9 shows for each source the number of its triples ($|triples(S_i)|$), the number of unique triples ($|triplesUnique(S_i)|$), and the unique triples contribution of that source ($UniqueTContrib(S_i)$). One can see that every source contains a very high ($> 99\%$) percentage of unique triples, so all sources are important.

S_i	$\|triples(S_i)\|$	$\|triplesUnique(S_i)\|$	$UniqueTContrib(S_i)$
FLOD	665,456	664,703	99.89%
WoRMS	461,230	460,741	99.89%
Ecoscope	54,027	53,641	99.29%
DBpedia	450,429	449,851	99.87%
FishBase	1,425,283	1,424,713	99.96%

Table 9. (Unique) triple contributions of the sources using policy [ii].

Measurements after Adding the Rule-derived 'sameAs' Relationships and Applying the Transformation Rules

As regards the unique contribution of each source using Policy [iii], the values in the first column are increased in comparison to Table 9. This is because of the execution of the transformation rules after the ingestion of the data to the warehouse, which results to the creation of new triples for the majority of sources. Finally one can observe that, in general, the unique triples contribution of each source is decreased. This happens because the transformation rules and the same-as relationships have turned previously different triples, the same.

S_i	$\lvert triples(S_i)\rvert$	$\lvert triplesUnique(S_i)\rvert$	$UniqueTContrib(S_i)$
FLOD	810,301	798,048	98.49%
WoRMS	528,009	527,358	99.88%
Ecoscope	138,324	52,936	38.27%
DBpedia	526,016	517,242	98.33%
FishBase	1,425,283	1,340,968	94.08%

Table 10. (Unique) triple contributions of the sources using policy [iii].

Complementarity of Sources

The authors now define another metric for quantifying the value of the warehouse for the entities of interest. With the term "entity" they mean any literal or URI that contains a specific string representing a named entity, like the name of a fish or country. The set of triples containing information about the entity of interest can be defined as $triples_W(e) = |\{\langle s,p,o\rangle \in W \mid s = e \text{ or } o = e\}|$. Specifically they define the *complementarity factor* for an entity e, denoted by $cf(e)$, as the percentage of sources that provide *unique* material about e. It can be defined declaratively as:

$$cf(e) = \frac{|\{i \mid triples_W(e) \cap triplesUnique(S_i) \neq \emptyset\}|}{|S|} \quad (7)$$

where S is the set of underlying sources.

Note that if $|S| = 1$ (i.e., there exists only one source), then for every entity e, $(e) = 1.0$. If $|S| = 2$, i.e., then there exists the following cases:

– $cf(e) = 0.0$, if both sources have provided the same triple (or triples) about e or no source has provided any triple about e,

– $cf(e) = 0.5$, if the triples provided by the one source (for e) are subset of the triples provided by the other, or if only one source provide triple(s) about e,

– $cf(e) = 1.0$, if each source has provided at least one different triple for e (of course they can also have contributed common triples). Consequently for the entities of interest one can compute and report the average *complementarity factor* as a way to quantify the value of the warehouse for these entities.

Table 11 shows (indicatively) the *complementarity factors* for a few entities which are important for the problem at hand. One can see that for the entities "Thunnus" and "Shark" each source provides unique information ($cf = 1.0$). For the entity "Greece" and "Astrapogon" unique information is obtained from three sources ($cf = 3/5 = 0.6$). The fact that the complementarity factor is big means that the warehouse provides unique information about each entity from many/all sources. Moreover, Table 12 shows the average complementarity factor of the species that are native to Greece. One can observe that there are no species with very small complementarity factor, which means that at least 2 sources provide unique information for each species $cf(e) \geq 0.4$. Indeed, exactly 2 sources provide unique information for 116 species, while for 35 species unique data is returned from all the sources. In general the average complementarity factor for all species that are native in Greece is approximately 0.63 (3.15/5) (meaning that at least 3 sources contain unique information for such species).

Kind of Entity	$cf(\cdot)$
Thunnus	1.0 (5/5)
Greece	0.6 (3/5)
Shark	1.0 (5/5)
Astrapogon	0.6 (3/5)

Table 11. Complementarity factor (cf) of some entities.

$cf(\cdot)$	No. of Species
0.2 (1/5)	0
0.4 (2/5)	116
0.6 (3/5)	180
0.8 (4/5))	113
1.0 (5/5)	35
Average: 0.63 (3.15/5)	Sum: 444

Table 12. cf of species that are native to Greece.

Metrics for Evaluating a Single Source (in the context of a warehouse)

In this section the authors focus on metrics for quantifying the value that a source brings to the warehouse. Such metrics should also allow identifying pathological cases (e.g., redundant or irrelevant sources). In particular, at first they provide examples of such cases and introduce rules for identifying them. Finally they introduce a single-valued metric based on these rules for aiding their identification by a human.

Detecting Redundancies or other Pathological Cases

The metrics can be used also for detecting various pathological cases, e.g., sources that do not have any common URI or literal, or "redundant sources". To test this the authors created three artificial sources, let us call them *Airports*, *CloneSource* and *AstrapogonSource*. The *Airports* source contains triples about airports which were fetched from the DBpedia public SPARQL endpoint, the *CloneSource* is a subset of Ecoscope's and DBpedia's triples as they are stored in the warehouse, and the *AstrapogonSource* contains only 1 URI and 4 triples for the entity *Astrapogon*. In the sequel, the metrics were computed for 8 sources.

Table 13 shows in the first column the unique triples contribution and in the second column the degree increase. The metrics were calculated according to policy [iii]. As regards *Airports*, one can see that its unique contribution is 1 (all the contents of that source are unique). As regards *CloneSource,* its unique contributions is zero (as expected, since it was composed from triples of existing sources). Finally, concerning the *AstrapogonSource*, although the number of its triples contribution is very low, all its triples are unique.

Rules for Detecting Pathological Cases. It follows that one can detect pathological cases using two rules: (a) if the average increase of the degree of the entities of a source is low, then this means that its contents are not connected with the contents of the rest of the sources (this is the case of *Airports* where there was only 0.1% increase), (b) if the unique contribution of a source is very low (resp. zero), then this means that it does not contribute significantly (resp. at all) to the warehouse (this is the case of *CloneSource* where the unique contribution was zero).

A Single Metric for Quantifying the Value of a Source

To further ease the inspection of pathological cases (and the quantification of the contribution of each source), the authors define a single (and single-valued) measure. One method is to use the *harmonic mean* of the unique contribution, and the increment in the average degree (the harmonic mean takes a high value if both values are high). Therefore, one can measure the harmonic mean of the above two metrics and define the value of a source S_i, denoted by $value_0(S_i, W)$, as:

$$value_0(S_i, W) = \frac{2 * UniqueTContrib(S_i) * DegIncr(S_i, W)}{UniqueTContrib(S_i) + DegIncr(S_i, W)} \quad (8)$$

Table 13 shows these values for all sources of the warehouse, including the artificial ones, in decreasing order. One can see that the problematic sources have a value less than 0.04 while the good ones receive a value greater than 0.2. However, *AstrapogonSource* has the highest score although it contains only 4 triples. The two reasons why this source seems the best according to this metric are that all the triples are unique and the only instance that it contains has a lot of properties in other sources. Therefore, the degree increment of this source is almost 1. Consequently this metric makes evident the contribution of each source to the warehouse.

S_i	$UniqueTContrib(S_i)$	$DegIncr(S_i, W)$	$value_0(S_i, W)$
AstrapogonSource	1	0.93	**0.9637**
FLOD	0.9849	0.89	**0.935**
DBpedia	0.96	0.84	**0.896**
FishBase	0.9408	0.54	**0.686**
WoRMS	0.9988	0.49	**0.6575**
Ecoscope	0.129	0.76	**0.2206**
Airports	1	0.001	**0.02**
CloneSource	0	0.89	**0**

Table 13. The value of a source in the Warehouse (using $value_0(S_i, W)$).

Although the above metric is good for discriminating the good from the not as good (or useless) sources, it ignores the number of triples that each source contributes. This is evident from Table 1 where AstrapogonSource gets the highest score. In general, a source with a small number of triples can have big values in the above two metrics.

For tackling this issue, there is a need of an analogous metric for the size of a specific source in the warehouse, specifically the authors define $S_i SizeInW(S_i, W) = \frac{|triples(S_i)|}{|triples(W)|}$. Then, they compute the harmonic mean of these three metrics and define the value of a source S_i, denoted by $value_1(S_i, W)$, as

$$value_1(S_i, W) = \frac{3}{\frac{1}{UniqueTContrib(S_i)} + \frac{1}{DegIncr(S_i, W)} + \frac{1}{S_i SizeInW(S_i, W)}} \quad (9)$$

Table 14 shows these values for all sources of the warehouse, including the artificial one, in decreasing order. Now one can see that FishBase is the most useful source, and the score of AstrapogonSource is very low (almost 0).

Consequently, the first metric can be used for deciding whether to include or not a source in the warehouse, while second for inspecting the importance of source for the warehouse. In case of adding a huge out-of-domain source in our warehouse while there exist a lot of useful sources which are much smaller, the values of the useful sources will remain almost stable for the first metric. On the contrary, their values will be decreased for the second metric. Regarding, the value of the out-of-domain source, it will be low in both metrics, since the increase of the average degree for this source will be almost 0. Therefore, both metrics will show that the new source should be removed from the warehouse, however, the second metric will not show the real value for each of the remaining sources in this case.

S_i	$UniqueTContrib(S_i)$	$DegIncr(S_i, W)$	$S_i SizeInW(S_i, W)$	$value_1(S_i, W)$
FishBase	0.9408	0.54	0.405	**0.5572**
FLOD	0.9849	0.89	0.2304	**0.463**
DBpedia	0.96	0.84	0.1496	**0.3364**
WoRMS	0.9988	0.49	0.1501	**0.3091**
Ecoscope	0.129	0.76	0.0393	**0.0869**
Airports	1	0.001	0.0089	**0.0027**
AstrapogonSource	1	0.93	0.000001	**0.000001**
CloneSource	0	0.89	0.0089	**0**

Table 14. The value of a source in the warehouse (using $value_1(S_i, W)$).

WAREHOUSE EVOLUTION

The objective here is to investigate how one can understand the evolution of the warehouse and how can detect problematic cases (due to changes in the remote sources, mistakes in the equivalence rules, addition of a redundant or a "useless" source etc). Let v denote a version of the warehouse and v' denote a new version of the warehouse. A number of questions arise:

- Is the new version of the warehouse better than the previous one? From what aspects, the new warehouse is better than the previous one, and from what aspects it is worse?
- Can the comparison of the metrics of v and v', aid us in detecting problems in the new warehouse, e.g., a change in an underlying source that affected negatively the new warehouse?

It is also useful to compare a series of versions for:

- understanding the evolution of the entire warehouse over time
- understanding the evolution of the contribution of a source in the warehouse over time

To tackle these questions, the authors first describe the used datasets and then focus on how to inspect a *sequence* of versions.

Datasets Used

To understand the evolution several series of warehouse versions are needed. The authors used 3 real versions of the **MarineTLO**-based warehouse, specifically:

- MarineTLO-based warehouse version 2 (July 2013): 1,483,972 triples
- MarineTLO-based warehouse version 3 (December 2013): 3,785,249 triples
- MarineTLO-based warehouse version 4 (June 2014): 5,513,348 triples

Inspecting a Sequence of Versions

Suppose that there exists n versions, $v_1, v_2, ..., v_n$. One can get the big picture by various plots each having in the X axis one point for each warehouse version. Below there are described several useful plots.

i. For each v_i, $|triples(W_{vi})|$ is computed. Figure 4 shows the resulting plot for the datasets.

ii. For each v_i, $|U_{Wvi}|$ and $|Lit_{Wvi}|$ are plotted, where U_{Wvi} is the set of all URIs and Lit_{Wvi} is the set of all Literals in the warehouse of that version. Figure 5 shows the resulting plot for the datasets.

iii. For each v_i, the average degree of the URIs of the warehouse $deg_W(U_{Wvi})$ are plotted, as well as the average degree of the blank nodes and URIs of the warehouse

$deg_W\left(U_{Wvi} \cup BN_{Wvi}\right)$. Figure 6 shows the resulting plot for the datasets.

iv. For each v_i and for each source S_j, $value_1\left(S_i, W\right)$ is plotted (one diagram with k plots one for each of the k sources). Figure 7 shows how the contribution of the sources in the warehouse evolves, for the datasets.

The first three (i-iii) concern the warehouse per se, while (iv) shows how the contribution of the source in the warehouse evolves. Finally, Table 15 shows the average degree increment of the URIs and blank nodes of each source for the 3 different versions of the real datasets. In (Mountantonakis, et al., 2016) one can find more experiments for both real and synthetic datasets, where various aspects are tested, e.g., source enlargements, increased or reduced number of sameAs relationships, addition of new sources (either relevant or irrelevant to the domain), addition of erroneous data, etc.

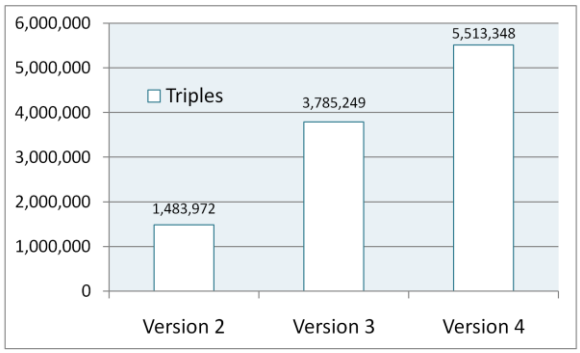

Figure 4. Triples of each version

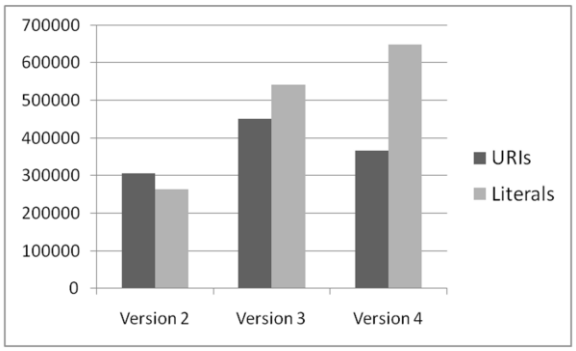

Figure 5. URIs and Literals

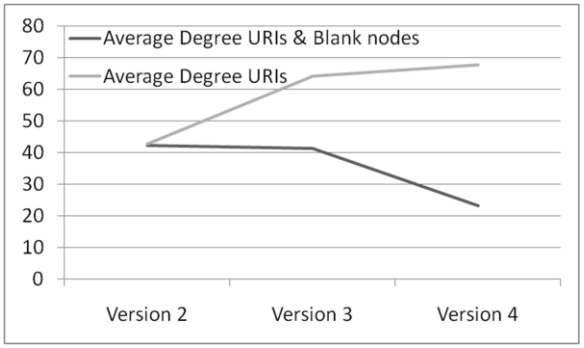

Figure 6. Average degree of warehouse

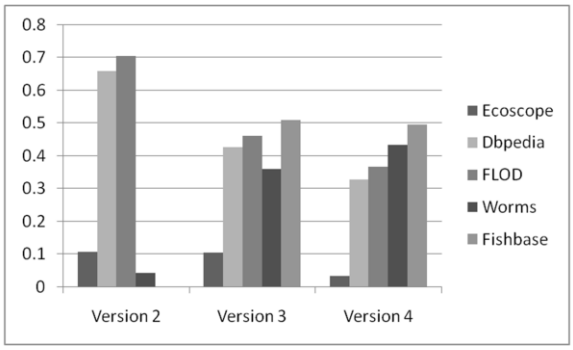

Figure 7. Value for each Source in every version.

S_i / v_i	Version2	Version 3	Version 4
FLOD	465.59%	793.64%	797.61%
WoRMS	548.67%	97.82%	103.61%
Ecoscope	108.97%	325.58%	396.84%
DBpedia	271.84%	522.75%	505.65%
FishBase	—-	117.02%	58.43%

Table 15. Average degree increment percentages for the URIs and blanks nodes of each source in every version.

EXPLOITATION OF METRICS AND NOVEL CONTRIBUTIONS

In (Mountantonakis, et al., 2016) one can find details about how the metrics can be implemented, how the results can be published by exploiting VoIDwh vocabulary, and how the results can be exploited for several real world tasks. The work presented in this chapter was the spark for extending the proposed methods and services in order to apply them in LOD scale, i.e., for a large number of datasets of any domain and to create services that concern the whole LOD cloud. Below two state of the art approaches and services are described in brief: a) Lattice-based measurements, b) 3D visualization of LOD cloud datasets.

As regards a), in (Mountantonakis & Tzitzikas, 2016), the authors introduced indexes and measurements about the commonalities of a large number of Linked Datasets. In particular, they measured the number of common real world objects between any set of datasets, that is they computed the number of classes of equivalence of URIs after having computed the symmetric and transitive closure of the set of *owl:sameAs* relationships from all datasets. The corresponding research prototype, called LODsyndesis[12], offers query services over 300 real Linked Datasets and these services can be exploited in several tasks including object co-reference, dataset discovery, visualization, and connectivity assessment and monitoring.

Concerning b), an ***interactive 3D visualization*** prototype, called 3DLOD[13], has been developed which adopts the metaphor of an urban area: each dataset is visualized as a building, whose features (e.g. volume) reflect various dataset's features (e.g. number of triples), while roads/bridges and the proximity of the buildings indicate the commonalities of the datasets (i.e., it exploits the presented metrics). Figure 8 shows such a 3D visualization for 287 real world datasets.

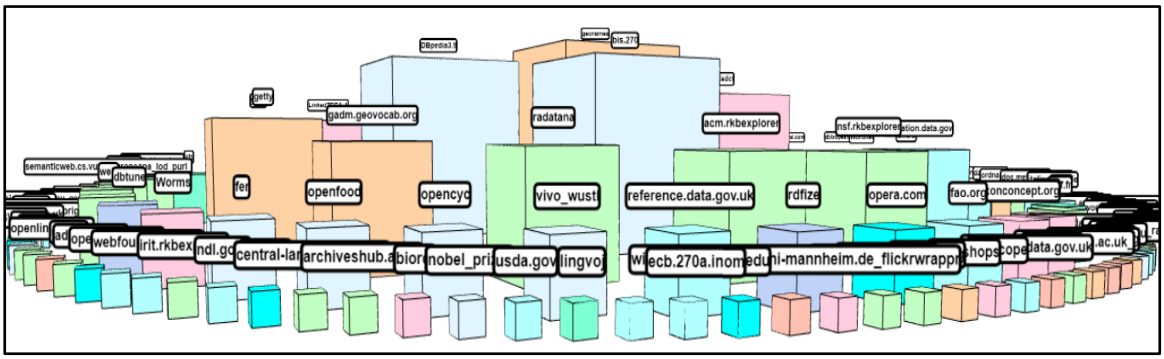

Figure 8. A 3D visualization of 287 RDF Datasets

FUTURE RESEARCH DIRECTIONS

There are various directions that are worth further research. One is to extend the lattice-based metrics for quantifying other kinds of commonalities (e.g. literals, triples, etc.) and to introduce metrics for aiding other tasks for improving the quality of data like veracity estimation and cleaning. Another direction is to apply big data management techniques for enhancing the scalability of the approach, i.e., for speeding up the computation of these metrics over large in number and big datasets. Finally, it would be interesting to introduce methods for measuring and improving schema connectivity (and thus query interoperability).

[12] http://www.ics.forth.gr/isl/LODsyndesis
[13] http://ww.ics.forth.gr/isl/3DLod/

CONCLUSION

In many applications one has to fetch and assemble pieces of information coming from more than one source. This chapter describes the main requirements and challenges, based also on an experience in building an operational semantic warehouse for marine resources. The chapter first describes the process for constructing such warehouses and then presents metrics for quantifying the *connectivity* of the outcome.

By inspecting the proposed metrics-based matrices one can very quickly get an overview of the contribution of each source and the tangible benefits of the warehouse. The main metrics are: (a) the matrix of percentages of the common URIs and/or literals, (b) the complementarity factor of the entities of interest, (c) the table with the increments in the average degree of each source, (d) the unique triple contribution of each source, and (e) a single-valued metric for quantifying the value of a source. The values of (a), (b), and (c) allow valuating the warehouse, while (c), (d) and (e) mainly concern each particular source. For instance, by combining the unique triples contribution and the increment of the average degrees, one can understand that not only one get unique information from *all* sources, but also *how much* the average degree of the entities of the sources has been increased in the warehouse. Moreover, redundant sources can be spotted through their low unique contribution, while unconnected sources through their low average increase of the degree of their entities. In addition, the chapter presents metrics and plots suitable for monitoring the evolution of a warehouse. More specifically, these metrics are exploited for understanding how the warehouse evolves and how the contribution of each source changes over time. To this end, a set of plots that allow someone to quickly spot anomalies are provided, while novel ways for exploiting such metrics in global scale and for visualization purposes are briefly presented.

The ability to assess the quality of a semantic warehouse, using methods like those presented in this chapter and those in the literature, is very important also for dataset and endpoint selection, as well as judging whether the warehouse can be used in e-Science. In the long run the authors expect that datasets and warehouses will be peer-reviewed, evaluated and cited, and this in turn will justify actions for their future maintenance and preservation.

Acknowledgement

This work was partially supported by the projects: *iMarine* (FP7 Research Infrastructures, 2011-2014), BlueBRIDGE (H2020 Research Infrastructures, 2015-2018) and *LifeWatch Greece* (National Strategic Reference Framework, 2012-2015).

REFERENCES

Auer, S., Jan, D., Martin, M., & Lehmann, J. (2012). LODStats - an Extensible Framework for High-Performance Dataset Analytics. (pp. 353-362). Berlin Heidelberg: Springer.

Ballou, D., & Tayi, G. (1999). Enhancing data quality in data warehouse environments. *Communications of the ACM, 42*(1), pp. 73-78.

Bizer, C. (2007). *Quality-Driven Information Filtering in the Context of Web-Based Information Systems.* Berlin: Freie Universität.

Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C. C., & Hellmann, S. (2009). DBpedia-A crystallization point for the Web of Data. *Web Semantics: science, services and agents on the world wide web, 7*(3), pp. 154-165.

Candela, L., Castelli, D., & Pagano, P. (2010). Making Virtual Research Environments in the Cloud a Reality: the gCube Approach. *ERCIM News, 2010*(83), p. 32.

Cyganiak, R., Field, S., Gregory, A., Halb, W., & Tennison, J. (2010). Semantic Statistics:

Bringing Together SDMX and SCOVO. *In WWW Workshop on Linked Data on the web.*

Darari, F., Nutt, W., Pirrò, G., & Razniewski, S. (2013). Completeness statements about RDF data sources and their use for query answering. *The Semantic Web--ISWC 2013* (pp. 66-83). Berlin Heidelberg: Springer.

Debattista, J., Lange, C., & Auer, S. (2015). Luzzu Quality Metric Language--A DSL for Linked Data Quality Assessment. *arXiv preprint arXiv:1412.3750*.

Dividino, R. Q., Gottron, T., Scherp, A., & Gröner, G. (2014). From Changes to Dynamics: Dynamics Analysis of Linked Open Data Sources. *In 1st International Workshop on Dataset Profiling & Federated Search for Linked Data (PROFILES'14).* Anissaras,Crete.

Fafalios, P., & Tzitzikas, Y. (2013, July). X-ENS: semantic enrichment of web search results at real-time. *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval* (pp. 1089-1090). ACM.

Fafalios, P., & Tzitzikas, Y. (2014). Exploratory Professional Search through Semantic Post-Analysis of Search Results. In *Professional Search in the Modern World, Lecture Notes in Computer Science.* Springer .

Fürber, C., & Hepp, M. (2010, May). Using sparql and spin for data quality management on the semantic web. Berlin Heidelberg: Springer.

Fürber, C., & Hepp, M. (2011). Swiqa-a semantic web information quality assessment framework. In *ECIS* (Vol. 15, p. 19).

Gimenez-Garcia, J., Thakkar, H., & Zimmermann, A. (2016). Assessing trust with PageRank in the Web of Data. In *International Semantic Web Conference* (pp. 293-307). Springer.

Guéret, C., Groth, P., Stadler, C., & Lehmann, J. (2012). Assessing linked data mappings using network measures. In *The Semantic Web: Research and Applications* (pp. 87-102). Berlin Heidelberg: Springer.

Harth, A., & Speiser, S. (2012, July). On Completeness Classes for Query Evaluation on Linked Data. In *AAAI.*

Hartig, O. (2009). Provenance Information in the Web of Data. *LDOW, 538*.

Hartig, O., & Zhao, J. (2009). Using web data provenance for quality assessment. *CEUR Workshop.*

Hartig, O., & Zhao, J. (2010). Publishing and consuming provenance metadata on the web of linked data. In *Provenance and annotation of data and processes* (pp. 78-90). Berlin Heidelberg: Springer.

Hogan, A., Harth, A., Umbrich, J., Kinsella, S., Polleres, A., & Decker, S. (2011). 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.

Hogan, A., Umbrich, J., Harth, A., Cyganiak, R., Polleres, A., & Decker, S. (2012). An empirical survey of linked data conformance. *Web Semantics: Science, Services and Agents on the World Wide Web, 14*, 14-44.

Keith Alexander, M., Cyganiak, R., Hausenblas, M., & Zhao, J. (2011). Describing linked datasets with the void vocabulary.

Knap, T., & Michelfeit, J. (2012). *Linked Data Aggregation Algorithm: Increasing Completeness and Consistency of Data.* Provided by Charles University.

Knap, T., Michelfeit, J., Daniel, J., Jerman, P., Rychnovský, D., Soukup, T., & Nečaský, M. (2012). ODCleanStore: a framework for managing and providing integrated linked data on the web. In *In Web Information Systems Engineering-WISE* (pp. 815-816). Berlin Heidelberg: Springer .

Knight, S. A., & Burn, J. M. (2005). Developing a framework for assessing information quality

on the World Wide Web. *Informing Science: International Journal of an Emerging Transdiscipline*, *8*(5), 159-172.

Kontokostas, D., Westphal, P., Auer, S., Hellmann, S., Lehmann, J., Cornelissen, R., & Zaveri, A. (2014, April). Test-driven evaluation of linked data quality. In *Proceedings of the 23rd international conference on World Wide Web* (pp. 747-758). ACM.

Liu, W., Liu, J., Duan, H., Hu, W., & Wei, B. (2017). Exploiting Source-Object Networks to Resolve Object Conflicts in Linked Data. Springer.

Marketakis, Y., Minadakis, N., Kondylakis, H., Konsolaki, K., Samaritakis, G., Theodoridou, M., . . . Doerr, M. (2016). X3ML Mapping Framework for Information Integration in Cultural Heritage and beyond. *International Journal on Digital Libraries*, 1-19.

Mendes, P. N., Mühleisen, H., & Bizer, C. (2012, March). Sieve: linked data quality assessment and fusion. In *Proceedings of the 2012 Joint EDBT/ICDT Workshops* (pp. 116-123). ACM.

Michelfeit, J., & Knap, T. (2012). Linked Data Fusion in ODCleanStore⋆. In *11th International Semantic Web Conference ISWC* (p. 45).

Missier, P., Belhajjame, K., & Cheney, J. (2013, March). The W3C PROV family of specifications for modelling provenance metadata. In *Proceedings of the 16th International Conference on Extending Database Technology* (pp. 773-776). ACM.

Mountantonakis, M., & Tzitzikas, Y. (2016). On measuring the lattice of commonalities among several linked datasets. *Proceedings of the VLDB Endowment*, pp. 1101-1112.

Mountantonakis, M., Minadakis, N., Marketakis, Y., Fafalios, P., & Tzitzikas, Y. (2016). Quantifying the connectivity of a semantic warehouse and understanding its evolution over time. *International Journal on Semantic Web and Information Systems (IJSWIS)*, pp. 27-78.

Nentwig, M., Soru, T., Ngomo, A.-C. N., & Rahm, E. (2014). Linklion: A link repository for the web of data. In *European Semantic Web Conference* (pp. 439-443). Springer.

Oren, E., Delbru, R., Catasta, M., Cyganiak, R., Stenzhorn, H., & Tummarello, G. (2008). Sindice. com: a document-oriented lookup index for open linked data. *International Journal of Metadata, Semantics and Ontologies*, *3*(1), 37-52.

Sarasua, C., Staab, S., & Thimm, M. (n.d.). Methods for Intrinsic Evaluation of Links in the Web of Data. In *European Semantic Web Conference* (pp. 68-84). Springer.

Shanks, G. G., & Darke, P. (1998). Understanding Data Quality and Data Warehousing: A Semiotic Approach. In *IQ* (pp. 292-309).

Tsiflidou, E., & Manouselis, N. (2013). Tools and Techniques for Assessing Metadata Quality. In *Metadata and Semantics Research* (pp. 99-110). Springer.

Tzitzikas, Y., Allocca, C., Bekiari, C., Marketakis, Y., Fafalios, P., Doerr, M., . . . Candela, L. (2013, November). Integrating heterogeneous and distributed information about marine species through a top level ontology. In *Metadata and Semantics Research* (pp. 289-301). Springer.

Tzitzikas, Y., Kampouraki, M., & Analyti, A. (2014). Curating the Specificity of Ontological. *Journal on Data Semantics, pages*, *3*(2), 75-106.

Tzitzikas, Y., Minadakis, N., Marketakis, Y., Fafalios, P., Allocca, C., & Mountantonakis, M. (2014, March). Quantifying the Connectivity of a Semantic Warehouse. In *EDBT/ICDT Workshops* (pp. 249-256).

Tzitzikas, Y., Minadakis, N., Marketakis, Y., Fafalios, P., Allocca, C., Mountantonakis, M., & Zidianaki, I. (2014, May). Matware: Constructing and exploiting domain specific warehouses by aggregating semantic data. In *The Semantic Web: Trends and Challenges* (pp. 721-736). Springer.

Volz, J., Bizer, C., Gaedke, M., & Kobilarov, G. (2009). Silk-A Link Discovery Framework for the Web of Data. In *Proceedings of the WWW'09 Workshop on Linked Data on the*

Web.
Wang, R. Y., & Strong, D. M. (1996). Beyond accuracy: What data quality means to data. *Journal of management information systems, 12*(4), 5-33.
Zaveri, A., Kontokostas, D., Sherif, M. A., Bühmann, L., Morsey, M., Auer, S., & Lehmann, J. (2013, September). User-driven quality evaluation of dbpedia. In *Proceedings of the 9th International Conference on Semantic Systems* (pp. 97-104). ACM.
Zaveri, A., Rula, A., Maurino, A., Pietrobon, R., Lehmann, J., Auer, S., & Hitzler, P. (2016). Quality assessment for linked data: A survey. *Semantic Web*, pp. 63-93.

KEY TERMS AND DEFINITIONS

Data quality: fitness of use for a certain application or use case.

Interlinking: the degree to which entities that represent the same concept are linked to each other.

Linked Data: a method of publishing structured data so that it can be interlinked and become more useful through semantic queries.

Semantic Integration: the process of interrelating information from diverse, heterogeneous data sources, which may conflict not only by structure but also context or value.

Semantic Warehouse: a read-only set of RDF triples fetched (and transformed) from different sources that aims at serving a particular set of query requirements.

Semantic Warehouse's Connectivity: the degree up to which the contents of the semantic warehouse form a connected graph that can serve, ideally in a correct and complete way, the query requirements of the semantic warehouse, while making evident how each source contributes to that degree.

Warehouse Evolution: the evolution of a warehouse over a specific period of time by taking into consideration the changes occurring in this period.