# An Ontology for Ontology Metrics: Creating a Shared Understanding of Measurable Attributes for Humans and Machines

Achim Reiz[a] and Kurt Sandkuhl[b]
*Rostock University, 18051 Rostock, Germany*

Keywords: Ontology Metrics, NEOntometrics, OntoMetrics, OWL, Ontology Quality.

Abstract: Measuring ontologies using metrics requires specialized software. While the past years saw various developments regarding tools and frameworks, these efforts mainly stayed isolated in their applied assessments. A paper measuring an ontology using the oQual framework is hardly comparable to one that applies the metrics from OntoQA. First, the performed calculations are often bound to the used tools, and second, the correct interpretation of ontology metrics requires a deep understanding of their measured aspects. Our research tackles these challenges by providing an ontology for ontology metrics. This artifact (A.) collects the various proposed ontology measurement frameworks with human-readable descriptions. It lets users quickly inform themselves on the assessments and aspects one can measure. (B) it formalizes the metric calculations. The framework metrics are connected to shared measurable elements, homogenizing the notations and languages. At last, (C.) the ontology is the backbone of the newly developed NEOntometrics application. The software uses the formalized metric descriptions to set up the calculations for the various frameworks. We believe our research can break the silos of different measurements, enable knowledge engineers to calculate various metrics quickly, and researchers to put new measurements into use through simple adaption of the metric ontology.

## 1 INTRODUCTION

Computational ontologies are complex interconnected graphs with description logics as underpinnings. They can capture knowledge, allow to infer implicit facts, and generate a shared understanding between human and computational actors. However, their development is far from a trivial task: there are countless ways to create an ontology. The developer has to make many modeling decisions until the artifact is completed. The first development decision is whether and which ontology shall be reused. Afterward, change assessment on the ontology as a whole gets more into focus: What kind of elements are affected in particular, is the change aligned with the overall set goals, and how does a change affect the structure?

Ontology metrics can guide these assessments. They provide an objective and reproducible way to grasp the attributes of ontologies (or ontology versions), allow the development team to set and pursue KPIs, and help the ontology engineer understand the change implications.

There are several metrics one can measure in ontologies. Selecting the proper measure for the job requires a deep understanding of the modeling goals and the logical foundations on which ontologies are built. Metric frameworks can guide making these decisions. Over the past years, several of these metric frameworks have been proposed. They set the atomic measurement points into context *(e.g., axiom/class ratio)* and often offer interpretations for the results or associate them with quality dimensions like reusability or readability.

While these frameworks aid the use and usability of metrics, one could argue that they also amplify the problem of metric selection. Now there are even more metrics to choose from. Furthermore, while a few of the frameworks build on each other, the proposed frameworks are often isolated, mainly because a study made with Framework A is not easily comparable to one made with Framework B, often

[a] https://orcid.org/0000-0003-1446-9670
[b] https://orcid.org/0000-0002-7431-8412



Reiz, A. and Sandkuhl, K.
An Ontology for Ontology Metrics: Creating a Shared Understanding of Measurable Attributes for Humans and Machines.
DOI: 10.5220/0011551500003335
In *Proceedings of the 14th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2022) - Volume 2: KEOD*, pages 193-199
ISBN: 978-989-758-614-9; ISSN: 2184-3228

due to the different terminologies.

The following research tackles this heterogeneity by proposing an ontology for ontology metrics. We collected information on (at the time of publication) seven metric frameworks, extracted their metric descriptions and interpretations (if applicable), and formalized the underlying measurements.

It allows human actors to inform themselves on the various measurable attributes in an ontology and possible interpretations and guides the selection of metrics and metric frameworks without having to read all of the underlying specifications. The aligned terminology makes the different frameworks more easily comparable.

For computational actors, the ontology provides the necessary formalization to set up an automatic calculation. New compositional metrics can be implemented by simply modeling them, thus reducing the implementation time and complexity.

The rest of the paper is structured as follows: Section two is concerned with the related work, followed by an overview of the modeled metric frameworks. Section four describes the newly created ontology with its relations and classes. Before the conclusion, section five describes how the NEOntometrics application uses this ontology to automatically set up and orchestrate the calculation service with its frontend, backend, and API.

There are many different approaches to evaluate an ontology based on the corpus, the given tasks, or predefined criteria. (Raad & Cruz, 2015) provides an extensive overview of available methodologies. This research, however, only considers automatically calculated criteria-based evaluation methods based solely on the ontologies' structure.

## 2 RELATED WORK

To the best of our knowledge, the idea of creating an ontology for ontology metrics is an endeavor without precedence. However, there have been other related work that either contributed to this research or researched comparable approaches. Most categorization papers developed smaller theoretical frameworks for ontology evaluation.

(Klein, 2004, p. 83) studied in his Ph.D. changes and change management in distributed ontology environments. Part of the thesis is a formalized UML meta-model of the web ontology language. Even though it is not directly linked to evolution efforts, the meta-model provides valuable information on the various (measurable) aspects of OWL-based ontologies.

In his thesis, (Vrandecic, 2010, p. 38) developed a theoretical framework for ontology evaluation. He organized evaluation methods and ontology evaluation with the concepts of ontologies, their ontology documents, and the conceptualizations that the ontologies represent.

The abstract model for ontology evaluation by (Verma, 2016) shows how ontology metrics can be categorized along the various hierarchical categories.

(Jarosław, 2018) conceptualized (in an ontology) the various methods and tools for a successful ontology evaluation process. Here, ontology metrics are one part of the evaluation process. Unfortunately, the ontology is not available for further analysis.

Further significant are papers reviewing the state-of-the-art in ontology metrics. Here are relevant the paper by (Lourdusamy & John, 2018), which is concerned with ontology metrics in general. Based on this literature review, the authors assembled 27 metrics in the categories *complexity, graph, knowledge base,* and *schema.*

(Porn et al., 2016) performed a systematic literature review on OWL-based ontology evaluation. They extracted quality criteria and categorized and organized the paper according to their evaluation technique and criteria.

(McDaniel & Storey, 2019) collected approaches specifically for domain ontologies. The authors gathered evaluation criteria for *domain/task fit, error checking, libraries, modularization*, and *metrics.*

## 3 ONTOLOGY METRIC FRAMEWORKS

At the time of the publication, the metric ontology contains information on seven measurement frameworks. As the research is open source, we invite the community to participate and add frameworks. Thus, in the future, the research presented in this section might not be exhaustive.

OntoQA, developed by (Tartir et al., 2005), proposes 17 measurements for assessing structure and population. OntoQA proposes metrics measuring the ontology as a whole and for specific classes and relations.

oQual, introduced by (Gangemi et al., 2005), is the largest of the introduced frameworks. It contains (among other criteria) 34 structural assessments measuring mostly graph-related attributes like depth, breadth, and leaf cardinality. The authors further propose some non-exhaustive quality dimensions and link them to quality metrics.

As part of a study on ontology characteristics, (Fernández et al., 2009) developed 12 measurements, assessing depth and breadth similar to oQual and the number of classes, properties, and instances.

(Yao et al., 2005) described three metrics concerning the cohesion of ontologies: The number of root classes, leaf classes, and the average depth of the inheritance tree of leaf nodes.

The evaluation of the complexity of ontologies using metrics was researched by (Zhang et al., 2006). They propose seven ontology metrics that build on one another. The paper assesses the Gene Ontology and measures the complexity through the number of subclass relations and paths.

(Orme et al., 2007) measures quality, stability, and completeness. They proposed six measurements assessing mainly graph-related attributes. Some of their metrics are similar to the cohesion metrics by (Yao et al., 2005).

OQuaRE, which transfers the SQuaRE software quality framework to ontologies, was first proposed by (Duque-Ramos et al., 2011) and has since been used by several publications, always involving the same group of authors. While implementing the framework, we discovered heterogeneities in these publications. In our ontology, we use the calculation published in the resulting homogenization effort (Reiz & Sandkuhl, 2022). OQuaRE provides linkages to quality dimensions.

# 4 THE METRIC ONTOLOGY

The metric ontology is available online[1] as part of the NEOntometrics repository. It has three interconnected main parts: Two classes (including their subclass elements) *Elemental Metrics* and *Quality Frameworks,* and individuals.

The *Elemental Metrics* contain the atomic, directly measurable elements of the ontology, like the number of axioms, individuals, and sub-class declarations. Examples of these elements are *Axioms* (the number of defined axioms in the ontology) and the *Maximum Depth* (the depth of the inheritance tree).

All subclasses of this category contain human-readable annotations through the custom annotation properties *metricDescription, metricDefinition,* and *metricInterpretation*. These annotations are created by the ontology authors and further explain the measured attribute with human-readable information.

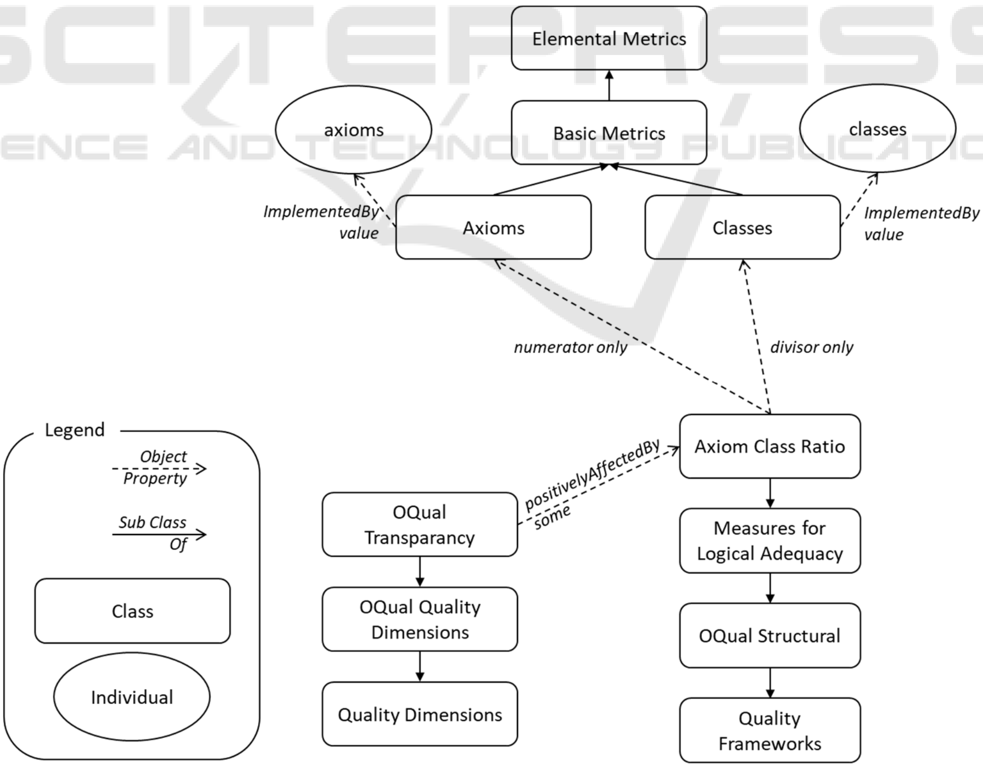


Figure 1: Excerpt of the metric ontology with the example of the metric axiom class ratio (without annotation properties).

[1] http://ontology.neontometrics.com

The metrics are connected to individuals using the object property *implementedBy*. The connected individuals represent calculated metric values in the database of NEOntometrics. Not all *Elemental Metrics* have an implementation in the application. Thus, not all *Elemental Metrics* are connected to individuals — more on the automatic setup in the upcoming section. The separation of implementation (*individuals*) and definition (*Elemental Metrics*) allows for a comprehensive knowledge base, even though not all metrics have been implemented yet.

The subclasses of the class *Quality Frameworks* capture the various metric proposals of the frameworks listed in the previous section. Here, all presented information mirrors the contents of the corresponding papers (linked with *rdf:seeAlso* annotations). This content mirroring has the effect that some elements have thorough descriptions while others lack them.

At first, the metrics proposed by the *Quality Frameworks* look rather diverse. The metrics are heavily different in their naming conventions or design and description of the metrics. At their core, however, they often use the same building blocks but name them differently. An example is the depth of the graph, which is named *Maximal Depth* by (Gangemi et al., 2005), while (Zhang et al., 2006) describe it as *Max Path Length*. Connecting the *Quality Frameworks* to the *Elemental Metrics* solves the challenge of differently named elements.

Some of the *Quality Frameworks* directly use attributes, For example, the *Maximum Depth* metrics. These attributes are connected to the elemental metrics using the object property *directlyUsesMetric*. The resulting relationship thus is o*Qual_MaximalDepth subClassOf directlyUsesMetric only MaximumDepth*.

Other metrics, however, calculate ratios. E.g., the *oQual Axiom Class Ratio* divides the number of *Axioms* by the number of *Classes*. We set up object properties for this relation, capturing the mathematical coherence. The object property *division* has the sub-properties *divisor* and *numerator*. Thus, the *oQual_AxiomClassRatio* is connected to the elemental metrics by making it a *subClassOf (divisor only Classes) and (numerator only Axioms)*. More object properties are available for the primary arithmetic operations like addition, subtraction, and multiplication.

Some metric frameworks not only propose metrics but link them to abstract quality dimensions like *Transparency*, *Organizational Fitness* (oQual), *Reusability, or Readability* (OQuaRE). These rather abstract quality implications are separated from the metrics using the class *Quality Dimensions.* They are connected to the metrics via the relations *negativeAffectedBy* and *positiveAffectedBy*. An example is the dimension *Transparency* of the oQual framework, which is (among others) *positiveAffectedBy some AnonymousClassesRatio*.

## 5 NEOntometrics

NEOntometrics is the primary ontology consumer. It provides a public endpoint to evaluate ontologies and inform on the over 160 available ontology metrics. The software is open source[2] and available online[3]. The following section states how NEOntometrics uses the ontology for setting up its API for metric retrieval. Further information on the application, especially the calculation engine, is available on its web pages.

The sequence diagram in Figure 2 outlines how the ontology augments the start-up procedure. In the beginning, the Django[4]-based backend first runs an initial SPARQL query, retrieving the information on available metrics and corresponding calculation information (1-2). It is followed by a second query (3-4) for retrieving the structure for the help page.

In the next step (5), the system transforms the results of the first query from the tabular query result structure into a list of python dictionaries[5], consisting of the basic information on each metric, like the descriptions and the corresponding metric category. If the metrics are implemented in the system (thus, if individuals are attached to the given *Elemental Metrics*), a string containing the calculation function is generated (6). Taking the example of the *Axiom Class Ratio*, the function field contains "*axioms/classes*", with *axioms* and *classes* named consistently with the corresponding database fields.

After this step, an internal python object contains all the information necessary for the automated metric calculation. As a next step, this object is integrated into a hierarchical structure for the help pages using the results of the second query, which requested the superclasses of metric elements. The superclasses allow the building of a tree-like structure, which the Django framework serializes to a nested JSON representation (7) for consumption in the frontend, primarily the metric explorer (cf. Figure 4).

[2] https://github.com/achiminator/NEOntometrics
[3] http://neontometrics.com
[4] https://www.django-rest-framework.org/
[5] A python dictionary stores key-value pairs.

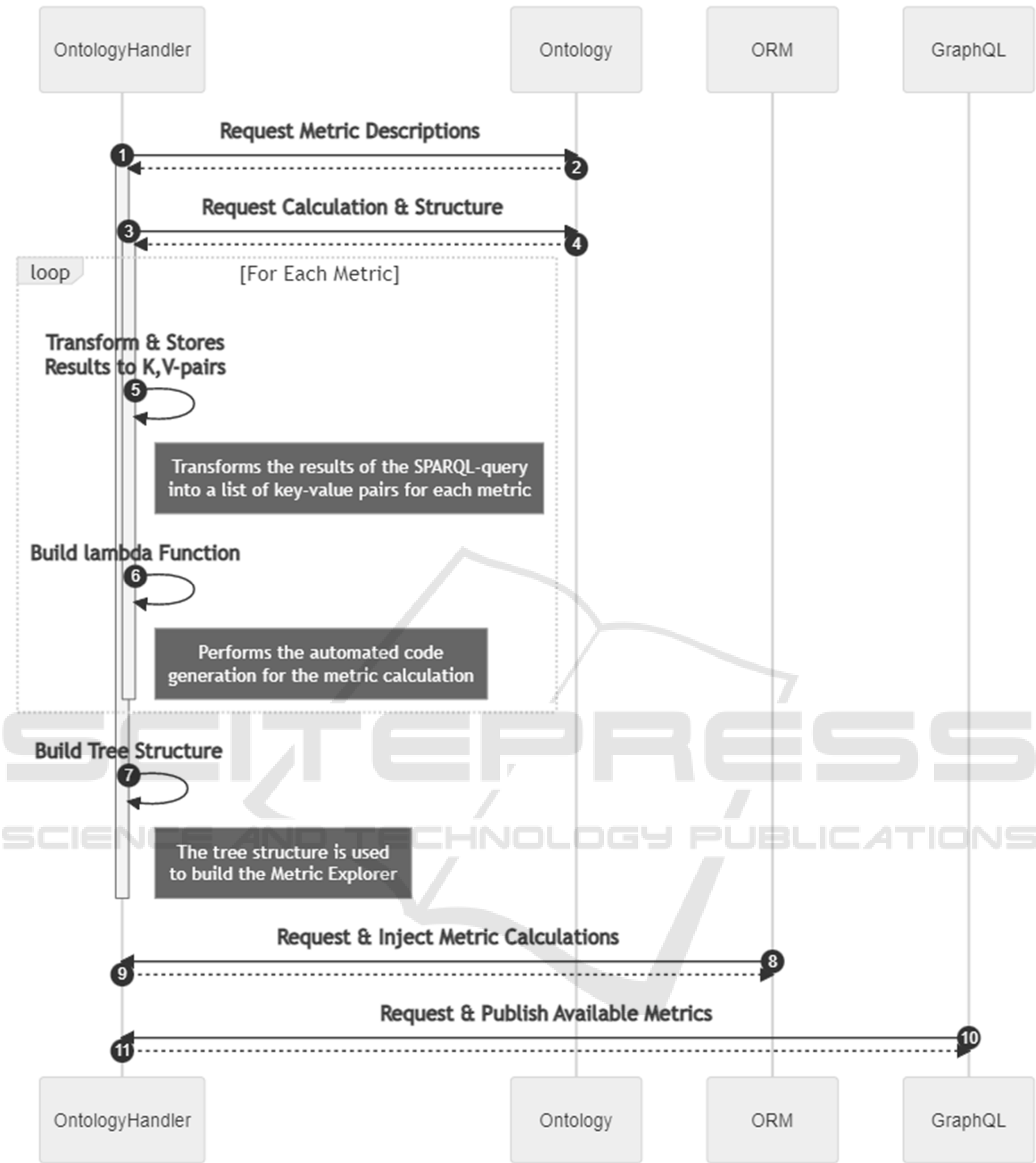


Figure 2: The start-up process of NEOntometrics. The ontology augments the database model and provides human-readable information for the frontend.

After the application has retrieved all the required ontology data and converted it into a more usable data representation, it starts with the augmentation of the calculation service. First, the object-relational mapping (ORM) of the Django framework is extended with the elements of the *Quality Frameworks* (8-9). The naming of elements in the previously created calculation function is equivalent to the database objects, and the created function strings are injected into the database model. The newly created elements (like *OQual_AxiomClassRatio*) behave like ordinary database objects, except that they are read-only.

After the ORM mapping is extended, its elements are registered in the GraphQL API (10-11). The API uses the *metricDefinition*, or if it is not available, the *metricDescription* annotation for describing a resource. This operation completes the initial start-up, and the NEOntometrics application is ready to receive a request from either the frontend or other applications.

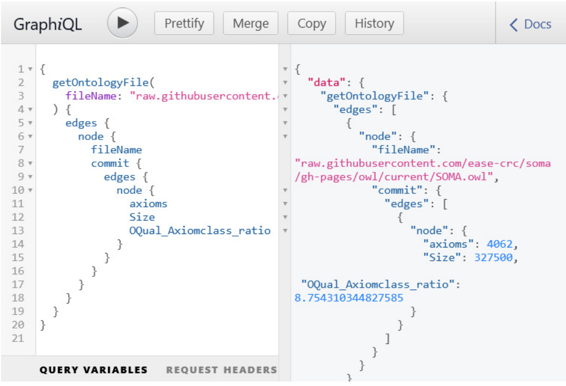


Figure 3: GraphQL-endpoint with the automatically configured metrics.

The web application queries the tree-based metric manual when opening the webpage. This information fills the *Metric Explorer* as shown in Figure 4 and the settings page *Calculation Engine*. The latter allows the selection of precisely the metrics that are needed. The selection is then translated into a GraphQL request and returns the requested data.

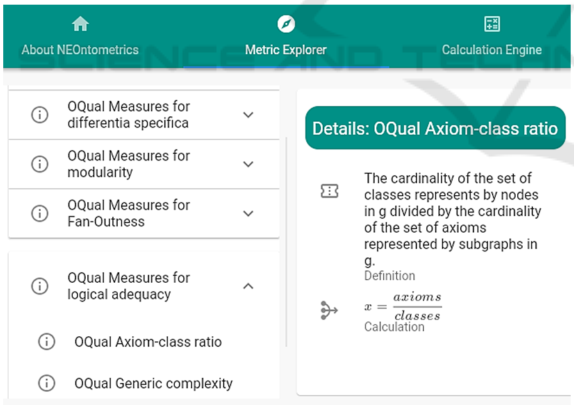


Figure 4: The Metric Explorer. The interactive ontology metrics manual builds on the metric ontology.

## 6 CONCLUSION

Various academic and business disciplines use ontologies with different requirements and ideas on how an ideal ontology should look. This diversity is also present in the proposed evaluation methods. Even though the technology behind ontologies is standardized, the calculation frameworks vary widely in their vocabulary, descriptions, and syntax.

This research collected and formalized various metric calculation approaches into a shared representation. By providing a common, interactive one-stop resource on ontology metrics, we hope it helps knowledge engineers to select the correct measurements for their use cases. It entangles the heterogeneous metric names of the various proposals by linking them to common underlying vocabulary and breaking up formerly isolated frameworks: If one ontology is calculated using the *oQual* framework, it can be downloaded for all other modeled frameworks as well.

The underlying ontology not only stores knowledge on existing approaches. It also allows the quick implementation of additional metrics. Thus, organizations can collect and build their individual set of ontology metrics without having to alter the underlying calculation code.

As part of the bigger picture, we believe that this metric ontology has the potential to increase understanding of the different observable attributes in an ontology and can strengthen the use of ontology metrics.

## REFERENCES

Duque-Ramos, A., Fernández-Breis, J. T., Stevens, R., & Aussenac-Gilles, N. (2011). OQuaRE: A square-based approach for evaluating the quality of ontologies. *Journal of Research and Practice in Information Technology*, *43*(2), 159–176. https://www.scopus.com/inward/record.uri?eid=2-s2.0-84860428632&partnerID=40&md5=dcb393aa78ee79eca9bfe365b38ed0f1

Fernández, M., Overbeeke, C., Sabou, M., & Motta, E. (2009). What Makes a Good Ontology? A Case-Study in Fine-Grained Knowledge Reuse. In A. Gómez-Pérez, Y. Yu, & Y. Ding (Eds.), *Lecture notes in computer science: Vol. 5926, The semantic web: Fourth Asian conference, ASWC 2009, Shanghai, China, December 6-9, 2009; proceedings* (pp. 61–75). Springer. https://doi.org/10.1007/978-3-642-10871-6_5

Gangemi, A., Catenacci, C., Ciaramita, M., Lehmann, J., Gil, R., Bolici, F., & Strignano Onofrio. (2005). *Ontology evaluation and validation: An integrated formal model for the quality diagnostic task*. Trentino, Italy.

Jarosław, W. (2018). An Attempt to Knowledge Conceptualization of Methods and Tools Supporting Ontology Evaluation Process. *Procedia Computer Science*, *126*, 2238–2247. https://doi.org/10.1016/j.procs.2018.07.225

Klein, M. (2004). *Change Management for Distributed Ontologies* [Ph.D. Thesis, Vrije Universiteit Amsterdam, Amsterdam, NL]. BibTeX.

Lourdusamy, R., & John, A. (2018). A review on metrics for ontology evaluation. In *Proceedings of the 2nd International Conference on Inventive Systems and Control (ICISC)* (pp. 1415–1421). IEEE. https://doi.org/10.1109/ICISC.2018.8399041

McDaniel, M., & Storey, V. C. (2019). Evaluating Domain Ontologies. *ACM Computing Surveys*, *52*(4), 1–44. https://doi.org/10.1145/3329124

Orme, A. M., Yao, H., & Etzkorn, L. H. (2007). Indicating ontology data quality, stability, and completeness throughout ontology evolution. *Journal of Software Maintenance and Evolution*, *19*(1), 49–75. https://doi.org/10.1002/smr.341

Porn, A. M., Huve, C. A. G., Peres, L. M., & Direne, A. I. (2016). A systematic Literature Review of OWL Ontology Evaluation. In P. Isaías & L. Rodrigues (Eds.), *Proceedings of the 15th International Conference WWW/Internet 2016: Mannheim, Germany, October 28-30, 2016.* IADIS Press.

Raad, J., & Cruz, C. (2015). A Survey on Ontology Evaluation Methods. In A. Fred (Ed.), *Proceedings of the 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management: Lisbon, Portugal, November 12 - 14, 2015* (pp. 179–186). SciTePress. https://doi.org/10.5220/0005591001790186

Reiz, A., & Sandkuhl, K. (2022). Harmonizing the OQuaRE Quality Framework. In *Proceedings of the 24th International Conference on Enterprise Information Systems,* online.

Tartir, S., Arpinar, I. B., Moore, M., Sheth, A. P., & Aleman-Meza, B. (2005). OntoQA: Metric-Based Ontology Quality Analysis. In D. Caragea, V. Honavar, I. Muslea, & R. Ramakrishnan (Chairs), *IEEE Workshop on Knowledge Acquisition from Distributed, Autonomous, Semantically Heterogeneous Data and Knowledge Sources,* Houston.

Verma, A. (2016). An abstract framework for ontology evaluation. In G. S. Kumar & K. P. Jacob (Eds.), *Proceedings, 2016 International Conference on Data Science & Engineering (ICDSE): August 23-25, 2016, Cochin University of Science and Technology* (pp. 1–6). IEEE. https://doi.org/10.1109/ICDSE.2016.7823945

Vrandecic, D. (2010). *Ontology Evaluation* [Dr. rer. pol.]. Karlsruhe Institute of Technology, Karlsuhe.

Yao, H., Orme, A. M., & Etzkorn, L. (2005). Cohesion Metrics for Ontology Design and Application. *Journal of Computer Science*, *1*(1), 107–113. https://doi.org/10.3844/jcssp.2005.107.113

Zhang, D., Ye, C., & Yang, Z. (2006). An Evaluation Method for Ontology Complexity Analysis in Ontology Evolution. In S. Staab & V. Svátek (Chairs), *International Conference on Knowledge Engineering and Knowledge Management,* Poděbrady, Czech Republic.