Analysing and promoting ontology interoperability in Wikibase

Daniil Dobriy, Axel Polleres

Vienna University of Economics and Business, Vienna, AT

Abstract
Wikibase, the open-source software behind Wikidata, increasingly gains popularity among third-party Linked Data publishers. However, the platform's unique data model decreases the degree of inter-operability with existing Semantic Web standards and tools that underlie Linked Data as codified by Linked Data principles. In particular, this unique data model of Wikibase also undermines the direct reuse of ontologies and vocabularies, in a manner compliant with Semantic Web standards and Linked Data principles. To this end, firstly, we compare the Wikibase data model to the established RDF data model. Secondly, we enumerate a series of challenges for importing existing ontologies into Wikibase. Thirdly, we present practical solutions to these challenges and introduce a tool for importing and re-using ontologies within Wikibase. Thus, the paper aims to promote ontology interoperability in Wikibase and by doing so hopes to contribute to higher degree of inter-linkage of Wikibase instances with Linked Open Data.

1. Introduction

Many openly published datasets that conform to the Linked Data principles are collected in the manually curated LOD Cloud (Linked Open Data Cloud) [1]. However, studies analysing Linked Data have shown that in reality Linked Data datasets have significant availability issues [2]. Furthermore, in previous studies we found that links between datasets in the LOD Cloud are very sparse on the level of individuals and although ontology reuse was common, many links on the level of ontologies were broken [3, 4]. In particular, in our previous work we have also shown that ontology links are very sparse in Wikidata [5].

Herein, we would like to argue that the differences between Wikidata's (and fundamentally Wikibase's) data model and plain RDF are a major factor in this mismatch and non-reuse of vocabularies and ontologies: although Wikibase (WB), the open-source software behind Wikidata, is intended to and is increasingly used by third parties to manage more specialized Knowledge Graphs outside of scope of Wikidata (WD), it lacks built-in functionality to import RDF vocabularies[1] which makes ontology re-use nontrivial. Generally, the standardized nature of RDF, the ability to conveniently join RDF datasets and the usage of URIs as global identifiers for both properties and classes promote ontology re-use in Linked Data. However, the WB data model is unique and quite distinct from plain RDF in a number of ways (discussed in Section 2) and can merely be *mapped* to RDF for the purpose of querying the data of an instance

[1]further referred to as *ontologies*
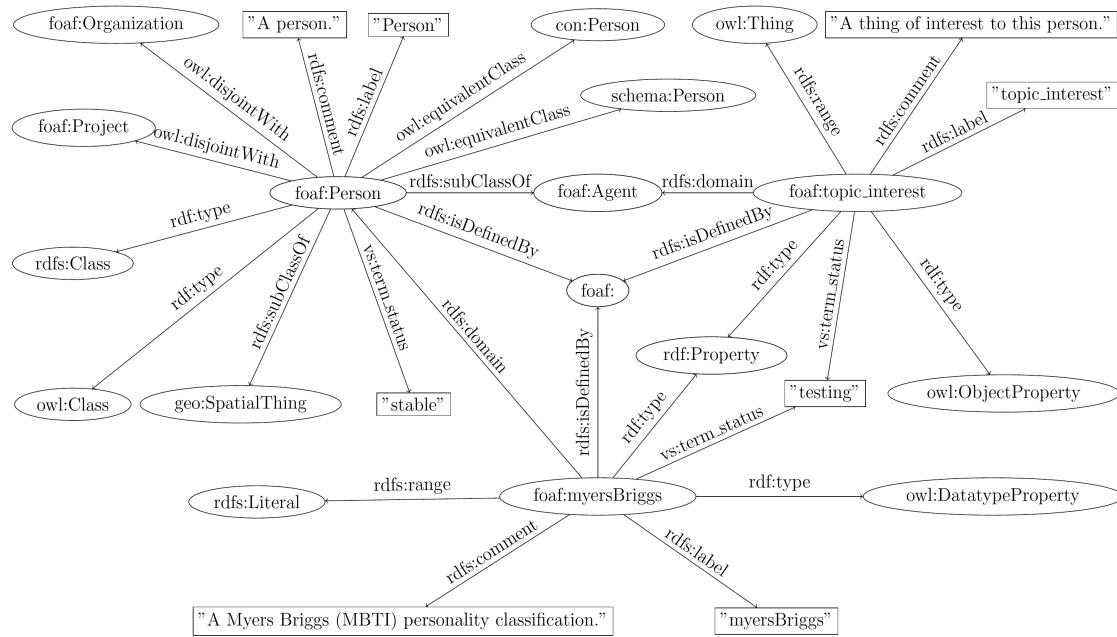
Figure 1: An exemplary fragment of the FOAF ontology.

with SPARQL or publishing it as an RDF dump. With the aim of appropriately representing ontologies in Wikibase and therefore allowing their reuse through facilitating their import, management and export in Wikibase, this paper poses the following research questions:

R1 Which features distinguish the WB and plain RDF data models?
R2 How can RDF ontologies be represented in the WB data model?
R3 How can ontologies be imported, managed and exported in WB?

The next three sections sequentially discuss each of the posed research questions and the final two sections discuss related works and draw conclusions. A fragment of the FOAF ontology[2] is used to exemplify the approach presented in the paper. Figure 1 illustrates the mentioned fragment that includes a definition of the class `foaf:Person` and of two properties: the `foaf:topic_interest` (of type `owl:ObjectProperty`) and the `foaf:myersBriggs` (of type `owl:DatatypeProperty`).

2. Data model comparison

The RDF data model [6] describes factual knowledge through subject-predicate-object Triples. IRIs or Blank Nodes take the position of subjects representing described things and concepts, IRIs take the predicate position representing relations, whereas objects can be IRIs, Blank Nodes, or (Typed) Literals.

---

[2]see `https://xmlns.com/foaf/spec/|`

The current realization of WB's data model is documented in [7]. "Technical Support" is one of the design goals of the data model and posits that the WB data model should allow adequate representation of data in existing data formats like RDF/OWL. The basic structure to describe factual knowledge in Wikibase is, similarly to RDF, by subject-predicate-object statements or *Snaks*. These statements may have additional qualifying property-value pairs that further describe the underlying Snak (Qualifiers), a *ReferenceList* that indicates sources for the overall content of the structure and a *Rank* for various presentation and administration purposes.

Items, *Properties* or *Datatypes*, jointly referred to as *Entities*, take the position of subjects in a Snak, Properties take the position of predicates and Entities as well as *DataValues* take up the position of objects. Entities are identified by IRIs which most notably follow the IRI schemas `https://www.wikidata.org/entity/Qnnn` and `https://www.wikidata.org/entity/Pnnn` for Items and Properties, resp. Datatypes and their IRIs are pre-defined by the Wikibase software. Each Property has an allowed Datatype, and although data model does not require strict typing for Properties, the Wikibase UI and API nevertheless only permit this required Datatype in the object position. Basic lexical information about Items, Properties and Datatypes as their *Labels*, *Descriptions* and *Aliases* is treated specially by the data model and defined as *EntityDescriptions* rather than as DataValues objects of Snaks.

Indeed, the two data models have key similarities. Snaks representing factual data resemble plain Triples in RDF, things, concepts and possible relation types are uniquely identified by IRIs, DataValues used in object position of Snaks resemble Typed Literals in object position of plain RDF Triples.

However, the WB data model also shows key restrictions that do not allow direct translation of data from plain RDF data model into the WB data model:

- Wheres WB only allows Entities identified by IRIs to be subjects, RDF also permits Blank Nodes which stand as placeholders for things and concepts: WB does not have Blank Nodes.
- WB makes an additional distinction between Items and Properties (and Datatypes) identified by IRIs and prohibits Items to be used in predicate position whereas RDF allows any IRI to be reused in any Triple position.
- The UI and API of WB imposes a de-facto strict typing of Properties on data entry which in contrast with RDF requires the definition of Property Datatypes and objects used in Snaks with there Properties to be of the expected Datatype.

3. Ontology Representation Requirements

When considering importing existing RDF ontologies in WB we define the following requirements:

- imported classes (e.g. `foaf:Person`) and properties (e.g. `foaf:topic_interest`) should be usable inside WB as Items and Properties;
- the imported ontology should be editable in WB with minimal restrictions;
- roundtripping: an (edited) ontology from WB should be exportable in the same form as it was imported;

In order to achieve the roundtripping requierement, we need to define a meta-ontology within WD to identify Snaks related to imported/edited ontologies and distinguish them from "regular" Snaks in order to be able to export ontologies or include new items into them. Such a meta-ontology is defined in RDF and then represented in WD according to the representation rules applicable to any ontology and discussed in the next section. The meta-ontology is accessible at and defines its classes and properties in the following namespace: `https://data.wu.ac.at/ns/o2wb#` .

Therefore, an ontology representation must sufficiently represent the features of the RDF data model not to lose any information necessary for ontology export but must ideally not introduce any excessive modeling components that could be ambiguously translated back on ontology export. Every RDF ontology is fully represented by the set of its Triples (all vertices in Figure 1). Since both data models rely on s-p-o representation of factual statements and factual statements are represented in Snaks in WB, our approach is based on faithful recreation of every ontology Triple as a Snak in WD.

IRIs in the subject position of a Triple are represented as Items or Properties. Since the RDF data model does not impose restrictions as Item/Property distinction and strict typing, the faithful representation of IRIs would allow them to be used in any capacity in accordance to the second requirement of allowing ontology management with minimal restrictions. In order to allow the same "freedom" within WB, any given RDF IRI has to be defined both as an Item and as (multiple) Properties (spanning every possible WB Datatype). Although the representation of an IRI as multiple Properties spanning every Datatype in WB would effectively already be sufficient for it to be used in any statement position (subject, predicate, object as it is possible in the RDF data model) thus rendering representation of an IRI as an Item unnecessary, the over-use of Properties in subject position of Snaks might lead to confusion as Items would normally be expected in this position and therefore we choose to also represent IRIs as Items.

The foaf:Person IRI would be represented as a Item with a distinct Q-ID and a multiple of Properties with different P-IDs and Datatypes (e.g. String, Monolingual Text, Item etc.[3])

It is important to note that the Datatypes used in WB do not directly or approximately correspond to (e.g. XSD) datatypes commonly used in Typed Literals, since they represent more complex data structures (e.g. geographical shapes or coordinates). Literals in plain RDF consist of a Unicode string, datatype IRI and (only if the datatype IRI is `rdf:langString`) a non-empty language tag [8]. We propose mapping RDF Literals occurring in Triples to WB data model by representing their Turtle serialization [9] as String Datatype of WB. Although this approach simplifies the variety of possible mappings of various datatype IRIs to WB Datatypes, it proposes a unified procedure that guarantees full freedom to modify Literal values represented in WB data model as they could be modified in the RDF data model. Further work may focus on establishing a more fine-grained mapping approach.

Next, Blank Nodes do not exist in the data model of WB and therefore have to be represented as skolemized IRIs with the same representation in WB as previously discussed IRIs. They are not representing any original IRIs (are therefore not used with o2wb:representsIRI relation) and for clarity are instances of class o2wb:BlankNode.

Finally, since the original IRIs (and skolemized Blank Node IRIs) are replaced by different Item

---

[3]for all WB Datatypes see: `https://www.wikidata.org/wiki/Help:Data_type|`

and Property Q-ID and P-ID IRIs within WB, in order to maintain ontological interoperability, equivalence to the original IRIs has to be established again for every representation of an IRI in WB. The meta-ontology therefore defines the `o2wb:representsIRI` property that is used to relate the IRI representations in WB to the original IRIs. In the data model of WB a Snak is created relating the Entity representing the IRI through the `o2wb:representsIRI` Property with Datatype String to the String representing the original IRI.

Additional meta-ontological concepts that have to be defined are the `o2wb:Ontology` class whose representation in WB is used to indicate that a specified Item represents an imported ontology and `o2wb:memberOfAnOntology` represented as Property with Datatype Item and used in a qualifier to indicate that a given Snak is a representation of a Triple from a given ontology. For each ontology import, an ontology individual of class `o2wb:Ontology` has to be established and represented in WD for all the related Snaks to be set in `o2wb:memberOfAnOntology` relation with it.

Finally, Triples are represented as Snaks. IRIs or Blank Nodes in subject position of a Triple are represented as the corresponding equivalent WB (`o2wb:representsIRI`) Items in subject position of a Snak. IRIs in predicate position of a Triple are represented as Properties with either Datatype Item or String in predicate position of a Snak depending on whether the object of a Triple is either an IRI/Blank Node, or a Literal. IRIs or Blank Nodes in object position of a Triple are represented as Items and Literals are represented as DataValues with Datatype String in the object position of a Snak.

Practically, the triple `foaf:Person rdfs:subClassOf foaf:Agent .` would be translated into a Snak that has the Item representing `foaf:Person` in its subject position, the Property with Datatype Item representing `rdfs:subClassOf` in predicate position and the Item representing `foaf:Agent` in object position. The triple `foaf:myersBriggs vs:term_status "testing" .` would be translated into a Snak that has the Item representing `foaf:myersBriggs`, the Property with Datatype String representing `vs:term_status` and a String "testing" for the Literal.

4. Ontology import, management, export

We propose o2wb, a novel tool that enables ontology import into WB by consequently implementing the approach described in the previous sections. For each ontology to be imported, o2wb creates an Item in WB representing the ontology and iterates over each Triple of an ontology creating, if they do not exist yet, appropriate representations of encountered IRIs, Blank Nodes and Literals, instantiating Snaks that correspond to the Triples using the newly created representations with correct Datatypes and including qualifiers to indicate that the Snaks belongs to the imported ontology. The tool is available under the following link: `https://git.ai.wu.ac.at/dobriy/o2wb`.

Importing ontologies with the help of this approach ensures that the ontology can be effectively edited though the WB UI and extracted back. The edit history is then preserved as page edit logs for respective Entities. However, representing IRIs and Blank Nodes through multiple Entities in the WB data model creates a challenge for exporting when a given representation is changed through WB UI. There is a number of solutions to this obstacle: changes can be implemented to all equivalent representations through bots whenever there is a change to one,

user could be asked to indicate the changes that should be exported back or users could be notified of the inconsistency and asked to consequently solve it. This considerations are subject of future work which includes enhancing the tool to be able to export ontologies back.

5. Related works

Although some work has described importing various data into a Wikibase instance [10] [11] [12] , we are unaware of any work specifically targeting ontology import in order to systematically increase ontology interoperability in the Wikibase ecosystem. Similar tools have been developed [13] for Semantic MediaWiki. However, the design choices are strongly dependent on the underlying data model of the system and we are unaware of any work discussing the differences and translation of WB data model and RDF.

6. Conclusion

It is possible to faithfully import and maintain an ontology in Wikibase in a way that will also allow its export at a later stage. Wikibase data model possesses a number of unique features (effectively strict property typing and hard distinction between Items and Properties) that decrease its ontological interoperability. To ensure a faithful representation of ontologies in the Wikibase data model, IRIs are represented as both Items and multiple Properties spanning all Datatypes, Blank Nodes are skolemized before being represented as IRIs and a supporting ontology is established to identify ontology contents during its editing in Wikibase. Finally, the o2wb tool is introduced as an implementation of the discussed approach to reusing ontologies in Wikibase.

References

[1] J. McCrae, et al., The linked open data cloud, since 2007. URL: https://lod-cloud.net/.

[2] A. Polleres, M. R. Kamdar, J. D. Fernández, T. Tudorache, M. A. Musen, A more decentralized vision for linked data, in: R. Verborgh, T. Kuhn, T. Berners-Lee (Eds.), Proceedings of the 2nd Workshop on Decentralizing the Semantic Web co-located with the 17th International Semantic Web Conference, DeSemWeb@ISWC 2018, Monterey, California, USA, October 8, 2018, volume 2165 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2018.

[3] S. Neumaier, A. Polleres, S. Steyskal, J. Umbrich, Data integration for open data on the web, in: Reasoning Web International Summer School, Springer, 2017, pp. 1–28.

[4] A. Haller, J. D. Fernández, M. R. Kamdar, A. Polleres, What are links in linked open data? a characterization and evaluation of links between knowledge graphs on the web, ACM Journal of Data and Information Quality (JDIQ) 2 (2020) 1––34. doi:10.1145/3369875.

[5] A. Haller, A. Polleres, D. Dobriy, N. Ferranti, S. J. Rodríguez Méndez, An analysis of links in wikidata, in: European Semantic Web Conference, Springer, 2022, pp. 21–38.

[6] F. Manola, E. Miller, B. McBride, et al., Rdf primer, W3C recommendation 10 (2004) 6.

[7] V. contributors, Wikibase/datamodel, 2014. URL: https://www.mediawiki.org/w/index.php?title=Wikibase/DataModel.

[8] R. Cyganiak, D. Wood, M. Lanthaler, Rdf 1.1. concepts and abstract syntax: W3c recommendation 25 february 2014, 2014. URL: https://www.w3.org/TR/2014/REC-rdf11-concepts-20140225.

[9] E. Prud'hommeaux, G. Carothers, Rdf 1.1 turtle-terse rdf triple language, w3c recommendation 25 february 2014. w3c recommendation, World Wide Web Consortium (W3C). http://www. w3. org/TR/2014/REC-turtle-20140225 (2014).

[10] T. Pellissier Tanon, D. Vrandečić, S. Schaffert, T. Steiner, L. Pintscher, From freebase to wikidata: The great migration, in: Proceedings of the 25th international conference on world wide web, 2016, pp. 1419–1428.

[11] R. Shigapov, J. Mechnich, I. Schumm, Raisewikibase: Fast inserts into the berd instance, in: ESWC2021 Poster and Demo Track, 2021.

[12] R. Shigapov, P. Zumstein, J. Kamlah, L. Oberländer, J. Mechnich, I. Schumm, bbw: Matching csv to wikidata via meta-lookup, in: CEUR Workshop Proceedings, volume 2775, RWTH, 2020, pp. 17–26.

[13] A. Castro, P. Strömert, Importing RDF ontologies to SMW with ontology2smw and the Academic Event Ontology, 2020. URL: https://www.semantic-mediawiki.org/w/images/2/24/Ontology2smw_smwcon.pdf.