Wikidata through the Eyes of DBpedia

Ali Ismayilov[1], Dimitris Kontokostas[2], Sören Auer[1], Jens Lehmann[2], and Sebastian Hellmann[2]

[1] University of Bonn, Enterprise Information Systems and Fraunhofer IAIS
s6alisma@uni-bonn.de|auer@cs.uni-bonn.de
[2] Universität Leipzig, Institut für Informatik, AKSW
{lastname}@informatik.uni-leipzig.de

Abstract. DBpedia is one of the first and most prominent nodes of the Linked Open Data cloud. It provides structured data for more than 100 Wikipedia language editions as well as Wikimedia Commons, has a mature ontology and a stable and thorough Linked Data publishing lifecycle. Wikidata, on the other hand, has recently emerged as a user curated source for structured information which is included in Wikipedia. In this paper, we present how Wikidata is incorporated in the DBpedia eco-system. Enriching DBpedia with structured information from Wikidata provides added value for a number of usage scenarios. We outline those scenarios and describe the structure and conversion process of the DBpediaWikidata dataset.

Keywords: DBpedia, Wikidata, RDF

1 Introduction

DBpedia is one of the first and most prominent nodes of the Linked Open Data cloud. It provides structured data for more than 100 Wikipedia language editions as well as Wikimedia Commons, has a mature ontology and a stable and thorough Linked Data publishing lifecycle. Wikidata has recently emerged as a user curated source for structured information which is included in Wikipedia.

DBpedia uses human-readable Wikipedia article identifiers to create IRIs for concepts in each Wikipedia language edition and uses RDF and Named Graphs as its original data model. Wikidata on the other hand uses language-independent numeric identifiers and developed its own data model, which provides better means for capturing provenance information. The multilingual DBpedia ontology, organizes the extracted data and integrates the different language editions while Wikidata is rather schemaless, providing only simple templates and attribute recommendations. All DBpedia data is extracted from Wikipedia and Wikipedia authors thus unconciously also curate the DBpedia knowledge base. Wikidata on the other hand has its own data curation interface, which is also based on the MediaWiki framework. DBpedia publishes a number of datasets for each language edition in a number of Linked Data ways, including datasets dumps, dereferencable resources and SPARQL endpoints. While DBpedia covers

a very large share of Wikipedia at the expense of partially reduced quality, Wikidata covers a significantly smaller share, but due to the manual curation with higher quality and provenance information. As a result of this complementarity, aligning both efforts in a loosely coupled way would render a number of benefits for users. Wikidata would be better integrated into the network of Linked Open Datasets and Linked Data aware users had a coherent way to access Wikidata and DBpedia data. Applications and use cases have more options for choosing the right balance between coverage and quality.

In this article we describe the integration of Wikidata into the DBpedia Data Stack. People are not used to the currently very evolving Wikidata schema, while DBpedia has a relatively stable and commonly used ontology. As a result, with the DBpedia Wikidata (DBw) dataset can be queried with the same queries that are used with DBpedia.

2 Background

Wikidata [5] is community-created knowledge base to manage factual information of Wikipedia and its sister projects operated by the Wikimedia Foundation. As of March 2015, Wikidata contains more than 17.4 million items and 58 million statements. The growth of Wikidata attracted researchers in Semantic Web technologies. In 2014, an RDF export of Wikidata was introduced [2] and recently a few SPARQL endpoints were made available as external contributions. Wikidata is a collection of entity pages. There are three types of entity pages: items, property and query. Every item page contains labels, short description, aliases, statements and site links. As described in the following listing (cf. [2, Figure 3]), each statement consists of a claim and an optional reference. Each claim consists of a property - value pair, and optional qualifiers. [3]

```
% Douglas Adams (Q42) spouse is Jane Belson (Q14623681)
%   - start time (P580) 25 November 1991, end time (P582) 11 May 2001.
wkdt:Q42 wkdt:P26s wkdt:Q42Sb88670f8-456b-3ecb-cf3d-2bca2cf7371e.
wkdt:Q42Sb88670f8-456b-3ecb-cf3d-2bca2cf7371e wkdt:P580q wkdt:VT74cee544.
wkdt:VT74cee544 rdf:type :TimeValue.;
 :time "1991-11-25"^^xsd:date;
 :timePrecision "11"^^xsd:int;   :preferredCalendar wkdt:Q1985727.
wkdt:Q42Sb88670f8-456b-3ecb-cf3d-2bca2cf7371e wkdt:P582q wkdt:VT162aadcb.
wkdt:VT162aadcb rdf:type :TimeValue;
 :time "2001-5-11"^^xsd:date;
 :timePrecision "11"^^xsd:int;   :preferredCalendar wkdt:Q1985727.
```

DBpedia [4] The semantic extraction of information from Wikipedia is accomplished using the DBpedia Information Extraction Framework (DIEF). The DIEF is able to process input data from several sources provided by Wikipedia. The actual extraction is performed by a set of pluggable *Extractors*, which rely on certain *Parsers* for different data types. Since 2011, DIEF is extended to provide better knowledge coverage for internationalized content [3] and allows the easier integration of different Wikipedia language editions.

[3] @prefix wkdt: $< http://wikidata.org/entity/ >$.

3 Conversion Process

The DBpedia Information Extraction Framework observed major changes to accommodate the extraction of data in Wikidata. The major difference between Wikidata and the other Wikimedia projects DBpedia extracts is that Wikidata uses JSON instead of WikiText to store items.

In addition to some DBpedia provenance extractors that can be used in any MediaWiki export dump, we defined 10 additional Wikidata extractors to export as much knowledge as possible out of Wikidata. These extractors can get labels, aliases, descriptions, different types of sitelinks, references, statements and qualifiers. For statements we define a RawWikidataExtractor that extracts all available information but uses our reification scheme (cf. Section 4) and the Wikidata properties and the R2RWikidataExtractor that uses a mapping-based approach to map, in real-time, Wikidata statements to the DBpedia ontology.

Wikidata Property Mappings In the same way the DBpedia mappings wiki defines infobox to ontology mappings, in the context of this work we define Wikidata property to ontology mappings. Wikidata property mappings can be defined both as *Schema Mappings* and as *Value Transformation Mappings*.

Schema Mappings The DBpedia mappings wiki[4] is a community effort to map Wikipedia infoboxes to the DBpedia ontology and at the same time crowdsource the DBpedia ontology. Mappings between DBpedia properties and Wikidata properties are expressed as `owl:equivalentProperty` links in the property definition pages, e.g. `dbo:birthPlace` is equivalent to `wkdt:P569`.[5] Although Wikidata does not define class in terms of `RDFS` or `OWL` we use `OWL` punning to define `owl:equivalentClass` links between the DBpedia classes and the related Wikidata items, e.g. `dbo:Person` is equivalent to `wkdt:Q5`.[6]

Value Transformations At the time of writing, the value transformation takes the form of a JSON structure that binds a Wikidata property to one or more value transformation strings. A complete list of the existing value transformation mappings can be found in the DIEF. [7] The value transformation strings that may contain special placeholders in the form of a '$' sign as functions. If no '$' placeholder is found, the mapping is considered constant. e.g. `"P625": {"rdf:type": "geo:SpatialThing"}`. In addition to constant mappings, one can define the following functions:

$1 replaces the placeholder with the raw Wikidata value. e.g.
`"P1566": {"owl:sameAs": "http://sws.geonames.org/$1/"}`.

$2 replaces the placeholder with a space the wiki-title value, used when the value is a Wikipedia title and needs proper whitespace escaping. e.g.
`"P154": {"logo": "http://commons.wikimedia.org/wiki/Special:FilePath/$2"},".`

[4] `http://mappings.dbpedia.org`
[5] `http://mappings.dbpedia.org/index.php/OntologyProperty:BirthDate`
[6] `http://mappings.dbpedia.org/index.php/OntologyClass:Person`
[7] `https://github.com/dbpedia/extraction-framework/blob/2ab6a15d8ecd5fc9dc6ef971a71a19ad4f608ff8/dump/config.json`

$getDBpediaClass Using the schema class mappings, tries to map the current value to a DBpedia class. This function is used to extract `rdf:type` and `rdfs:subClassOf` statement from the respective Wikidata properties. e.g

```
"P31": {"rdf:type": "$getDBpediaClass"}
"P279":{"rdfs:subClassOf": "$getDBpediaClass"}
```

$getLatitude, $getLongitude & $getLongitude Geo-related functions to extract coordinates from values. Following is a complete geo mapping that the extracts geo coordinates similar to the DBpedia coordinates dataset. For every occurrence of the property P625, four triples - one for every mapping - are generated:

```
"P625":[{"rdf:type":"geoSpatialThing"},
 {"geo:lat": "$getLatitude" },
 {"geo:long": "$getLongitude"},
 {"georss:point": "$getGeoRss"}]
```

```
DW:Q64 rdf:type geo:SpatialThing ;
 geo:lat "52.51667"^^xsd:float ;
 geo:long "13.38333"^^xsd:float ;
 geo:point "52.51667 13.38333" .
```

Mappings Application The R2RWikidataExtractor merges the schema & value transformation property mappings and for every statement or qualifier it encounters, if mappings for the current Wikidata property exist, it tries to apply them and emit the mapped triples.

Additions and Post Processing Steps Besides the basic extraction phase, additional processing steps are added in the workflow.

Type Inferencing In a similar way DBpedia calculates transitive types for every resource, the DBpedia Information Extraction Framework was extended to generate these triples directly at extraction time. As soon as an `rdf:type` triple is detected from the mappings, we try to identify the related DBpedia class. If a DBpedia class is found, all super types are assigned to a resource.

Transitive Redirects DBpedia has already scripts in place to identify, extract and resolve redirects. After the redirects are extracted, a transitive redirect closure (excluding cycles) is calculated and applied in all generated datasets by replacing the redirected IRIs to the final ones.

Validation The DBpedia extraction framework already takes care of the correctness of the extracted datatypes during extraction. We provide two additional steps of validation. The first step is performed in real-time during extraction and checks if the property mappings has a compatible `rdfs:range` (literal or IRI) with the current value. The rejected triples are stored for feedback to the DBpedia mapping community. The second step is performed in a post-processing step and validates if the type of the object IRI is disjoint with the `rdfs:range` of the property. These errors, although they are excluded from the SPARQL endpoint and the Linked Data interface, are offered for download.

IRI Schemes As mentioned earlier, we decided to generate the RDF datasets under the `wikidata.dbpedia.org` domain. For example, `wkdt:Q42` will be transformed to `dw:Q42`[8].

[8] `@prefix dw: <http://wikidata.dbpedia.org/resource/>` .

Title	Triples	Description
Provenance	17,771,394	PageIDs & revisions
Redirects	434,094	Explicit & transitive redirects
Aliases	4,922,617	Resource aliases with dbo:alias
Labels	61,668,295	Labels with rdfs:label
Descriptions	95,667,863	Descriptions with dbo:description
Sitelinks	41,543,058	DBpedia inter-language links
Wikidata links	17,562,043	Links to original Wikidata URIs
Mapped facts	90,882,327	Aggregated mapped facts
- Types	8,579,745	Direct types from the DBpedia ontology
- Transitive Types	48,932,447	Transitive types from the DBpedia ontology
- Coordinates	6,415,120	Geo coordinates
- Images	1,519,368	Depictions using foaf:depiction & dbo:thumbnail
- mappings	22,270,694	Wikidata statements with DBpedia ontology
- External links	3,164,953	sameAs links to external databases
Mapped facts (R)	138,936,782	Mapped statements reified (all)
Mapped facts (RQ)	626,648	Mapped qualifiers
Raw facts	59,458,835	Raw simple statements (not mapped)
Raw facts (R)	237,836,221	Raw statements reified
Raw facts (RQ)	1,161,294	Raw qualifiers
References	34,181,399	Reified statements references with dbo:reference
Mapping Errors	2,711,114	Facts from incorrect mappings
Ontology Errors	3,398	Facts excluded due to ontology inconsistencies

Table 1. Description of the DBw datasets. (R) stands for a reified dataset and (Q) for a qualifiers dataset

Reification In contrast to Wikidata, simple RDF reification was chosen for the representation of qualifiers. This lead to a simpler design and further reuse of the DBpedia properties. The IRI schemes for the `rdf:Statement` IRIs follow the same verbose approach from DBpedia to make them easily writable manually by following a specific pattern. When the value is an IRI (Wikidata Item) then for a subject IRI Qs, a property Px and a value IRI Qv the reified statement IRI has the form `dw:Qs_Px_Qv`. When the value is a Literal then for a subject IRI Qs, a property Px and a Literal value Lv the reified statement IRI has the form `dw:Qs_Px_H(Lv,5)`, where H() is a hash function that takes as argument a string (Lv) and a number to limit the size of the returned hash (5). The use of the hash function in the case of literals guarantees the IRI uniqueness and the value '5' is safe enough to avoid collisions and keep it short at the same time. The equivalent representation of the Wikidata example in Section 2 is: [9]

```
1 dw:Q42_P26_Q14623681 a rdf:Statement ;
2  rdf:subject dw:Q42 ;
3  rdf:predicate dbo:spouse ;
4  rdf:object dw:Q14623681 ;
5  dbo:startDate "1991-11-25"^^xsd:date ;
6  dbo:endDate "2001-5-11"^^xsd:date ;
```

4 Dataset Description

A statistical overview of the DBw dataset is provided in Table 1. We extract provenance information, e.g. the MediaWiki page and revision IDs as well as redirects. Aliases labels and descriptions are extracted from the related Wikidata item section and are similar to the RDF data Wikidata provides. A difference to Wikidata are the properties we chose to associate aliases and description.

Wikidata sitelinks are processed to provide three datasets: 1) `owl:sameAs` links between DBw IRIs and Wikidata IRIs (e.g. `dw:Q42 owl:sameAs wkdt:Q42`),

[9] DBw does not provide precision. Property definitions exist in the DBpedia ontology

Class	Count
dbo:Agent	2,884,505
dbo:Person	2,777,306
geo:spatialThing	2,153,258
dbo:TopicalConcept	1,907,203
dbo:Taxon	1,906,747

Table 2. Top classes

Property	Count
owl:sameAs	232,890,848
rdf:type	145,407,453
dbo:description	95,667,863
rdfs:label	61,704,172
rdfs:seeAlso	5,125,945

Table 3. Top properties

Property	Count
dbo:date	301,085
dbo:startDate	158,947
geo:point	108,526
dbo:endDate	50,058
dbo:country	33,698

Table 4. Top mapped qualifiers

Property	Count
wd:P31	13,070,656
wd:P17	3,051,166
wd:P21	2,604,741
wd:P131	2,372,237
wd:P625	2,167,100

Table 5. Top properties in Wikidata

2) `owl:sameAs` links between DBw IRIs and sitelinks converted to DBpedia IRIs (e.g. `dw:Q42 owl:sameAs db-en:Douglas_Adams`) and 3) for every language in the mappings wiki we generate `owl:sameAs` links to all other languages (e.g. `db-en:Douglas_Adams owl:sameAs db-de:Douglas_Adams`). The latter is used for the DBpedia releases in order to provide links between the different DBpedia language editions.

Mapped facts are generated from the *Wikidata property mappings* (cf. Section 3). Based on a combination of the predicate and object value of a triple they are split in different datasets. Types, transitive types, geo coordinates, depictions and external `owl:sameAs` links are separated. The rest of the mapped facts are in the *mappings* dataset. The reified mapped facts (R) contains all the mapped facts as reified statements and the mapped qualifiers for these statements (RQ) are provided separate (cf. Listing 3).

Raw facts consist of three datasets that generate triples with DBw IRIs and the original Wikidata properties. The first dataset (raw facts) provides triples for simple statements. The same statements are reified in the second dataset (R) and in the third dataset (RQ) we provide qualifiers linked in the reified statements. Example of the raw datasets can be seen in Listing 3 by replacing the DBpedia properties with the original Wikidata properties. These datasets provide full coverage and, except from the reification design and different namespace, can be seen as equivalent with the WikidataRDF dumps.

Wikidata statement references are extracted in the *references* dataset using the reified statement resource IRI as subject and the `dbo:reference` property. Finally, in the mapping and ontology errors datasets we provide triples rejected according to Section 3.

5 DBpedia WikiData In Use

Statistics and Evaluation The statistics we present are based on the Wikidata XML dump from March 2015. We managed to generate a total of 1B triples with 131,926,822 unique resources. In Table 1 we provide the number of triples per combined datasets.

Class & property statistics We provide the 5 most popular DBw classes in Table 2. We managed to extract a total of 6.5M typed Things with Agents and

SpatialThing as the most frequent types. The 5 most frequent mapped properties in simple statements are provided in Table 3 and the most popular mapped properties in qualifiers in Table 4. Wikidata does not have a complete range of value types and date paoperties are the most frefuent at the moment.
Mapping statistics In total, 270 value transformation mappings were defined along with 163 `owl:equivalentProperty` and 318 `owl:equivalentClass` schema mappings. Wikidata has 1465 properties defined with a total of 60,119,911 occurrences. With the existing mappings we covered 77.2% of the occurrences.
Redirects In the current dataset we generated 434,094 redirects – including transitive. When these redirects were applied to the extracted datasets, they replaced 1,161,291 triples to the redirected destination. The number of redirects in Wikidata is small compared to the project size but is is also a relatively new project. As the project matures in time the number of redirects will increase and resolving them will have an impact on the resulting data.
Validation According to Table 1, a total of 2.7M errors originated from schema mappings and 3,398 triples did not pass the ontology validation (cf. Section 3).

Access and Sustainability This dataset will be part of the official DBpedia knowledge infrastructure and be published through the regular releases of DBpedia, along with the rest of the DBpedia language editions. The first DBpedia release that will include this dataset is due on April - May 2015 (2015A). DBpedia is a pioneer in adopting and creating best practices for Linked Data and RDF publishing. Thus, being incorporated into the DBpedia publishing workflow guarantees: a) long-term availability through the DBpedia Association and b) agility in following best-practices as part of the DBpedia Information Extraction Framework. In addition to the regular and stable releases of DBpedia we provide more frequent dataset updates from the project website.[10]

Besides the stable dump availability we created `http://wikidata.dbpedia.org` for the provision of a Linked Data interface and a SPARQL Endpoint. The dataset is registered in DataHub[11] and provides machine readable metadata as void[12] and DataID[13] [1]. Since the project is now part of the official DBpedia Information Extraction Framework, our dataset reuses the existing user and developer support infrastructure. DBpedia has a general discussion and developer list as well as an issue tracker[14] for submitting bugs.

Use Cases Although it is early to identify all possible use cases for DBw, our main motivation was a) ease of use, b) vertical integration with the existing DBpedia infrastructure and c) data integration and fusion. Following we list SPARQL query examples for simple and reified statements. Since DBpedia provides transitive types directly, queries where e.g. someone asks for all 'places' in Germany can be formulated easier. Moreover, `dbo:country` can be more intuitive than `wkdt:P17c`. Finally, the DBpedia queries can, in most cases directly or

[10] `http://wikidata.dbpedia.org/downloads`
[11] `http://datahub.io/dataset/dbpedia-wikidata`
[12] `http://wikidata.dbpedia.org/downloads/void.ttl`
[13] `http://wikidata.dbpedia.org/downloads/20150330/dataid.ttl`
[14] `https://github.com/dbpedia/extraction-framework/issues`

with minor adjustments, run on all DBpedia language endpoints. When someone is working with reified statents, the DBpedia IRIs encode all possible information to visually identify the resources and items involved (cf. Section 3) in the statement while Wikidata uses a hash string. In addition, querying for reified statement in Wikidata needs to properly suffix the Wikidata property with `c/s/q`:

```
#Queries with simple statement
select * WHERE {              | select * WHERE {
 ?place a dbo:Place ;         |  ?place wkdt:P31c/wkdt:P279c* wkdt:Q2221906 ;
        dbo:country dw:Q183.} |         wkdt:P17c wkdt:Q183. }

#Queries with reified statements
select ?person where {                     | SELECT ?person WHERE {
 ?statementUri rdf:statement ?person ;     |  ?person wkdt:P26s ?spouse.
               rdf:predicate dbo:spouse ;  |  ?spouse wkdt:P580q ?marriageValue.
               dbo:startDate ?date.        |  ?marriageValue wo:time ?date.
 FILTER (?date < "2000-01-01"^^xsd:date) } |  FILTER (?date < "2000-01-01"^^xsd:date)   }
```

An additional important use case is data integration. Converting a dataset to a more used and well-known schema, it makes it easier to integrate the data. The fact the datasets are split according to the information they contain makes data consumption easier when someone needs a specific subset, e.g. coordinates. The DBw dataset is also planned to be used as an enrichment dataset on top of DBpedia and fill in semi-structured data that are being moved to Wikidata. It is also part of short-term plan to fuse all DBpedia data into a single namespace and the DBw dataset will have a prominent role in this effort.

6 Conclusions and Future Work

We present an effort to provide an alternative RDF representation of Wikidata. Our work involved the creation of 10 new DBpedia extractors, a Wikidata2DBpedia mapping language and additional post-processing & validation steps. With the current mapping status we managed to generate over 1 billion RDF triples. In the future we plan to extend the mapping coverage as well as extend the language with new mapping functions and more advanced mapping definitions.

References

1. M. Brümmer, C. Baron, I. Ermilov, M. Freudenberg, D. Kontokostas, and S. Hellmann. DataID: Towards semantically rich metadata for complex datasets. In *Proc. of the 10th International Conference on Semantic Systems*, pages 84–91. ACM, 2014.
2. F. Erxleben, M. Günther, M. Krötzsch, J. Mendez, and D. Vrandečić. Introducing Wikidata to the linked data web. In *ISWC'14*, LNCS. Springer, 2014.
3. D. Kontokostas, C. Bratsas, S. Auer, S. Hellmann, I. Antoniou, and G. Metakides. Internationalization of linked data: The case of the greek dbpedia edition. *Web Semantics:Science,Services & Agents on the World Wide Web*, 15(0):51 – 61, 2012.
4. J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P. N. Mendes, S. Hellmann, M. Morsey, P. van Kleef, S. Auer, and C. Bizer. DBpedia - a large-scale, multilingual knowledge base extracted from wikipedia. *SWJ*, 6(2):167–195, 2015.
5. D. Vrandečić and M. Krötzsch. Wikidata: A free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85, Sept. 2014.