Wikidata through the Eyes of DBpedia

Ali Ismayilov [a] Dimitris Kontokostas [b] Sören Auer [a] Jens Lehmann [a] and Sebastian Hellmann [b]

[a] *University of Bonn, Enterprise Information Systems and Fraunhofer IAIS*
e-mail: alismayilov@gmail.com | auer@cs.uni-bonn.de | jens.lehmann@cs.uni-bonn.de
[b] *Universität Leipzig, Institut für Informatik, AKSW*
e-mail: {lastname}@informatik.uni-leipzig.de

Abstract. DBpedia is one of the earliest and most prominent nodes of the Linked Open Data cloud. DBpedia extracts and provides structured data for various crowd-maintained information sources such as over 100 Wikipedia language editions as well as Wikimedia Commons by employing 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 (DBw) dataset.

Keywords: DBpedia, Wikidata, RDF

1. Introduction

DBpedia [5] is one of the earliest and most prominent nodes of the Linked Open Data cloud. DBpedia extracts and provides structured data for various crowd-maintained information sources such as over 100 Wikipedia language editions as well as Wikimedia Commons [6] by employing a mature ontology and a stable and thorough Linked Data publishing lifecycle. Wikidata [7] has recently emerged as a user-curated source for structured information with an API that enables to include facts into Wikipedia articles. Both resources overlap as well as complement each other as described in the high-level overview below.

Identifiers 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.

Structure The multilingual DBpedia ontology, organizes the extracted data and integrates the different language editions while Wikidata is per design schemaless, providing only simple templates and attribute recommendations.

Curation 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 called WikiBase[1], which is based on the MediaWiki framework.

Publication DBpedia publishes a number of datasets for each language edition in a number of Linked Data ways, including datasets dumps, dereferencable URIs and SPARQL endpoints.

Coverage While DBpedia covers most Wikipedia language editions, Wikidata unifies all language editions and defines resources beyond Wikipedia. Although Wikidata may have a bigger resource coverage, there is no study yet that performs a qualitative and quantitavive comparison between Wikidata and DBpedia.

We argue that as a result of this complementarity, aligning both efforts in a loosely coupled way would produce an improved resource and 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.
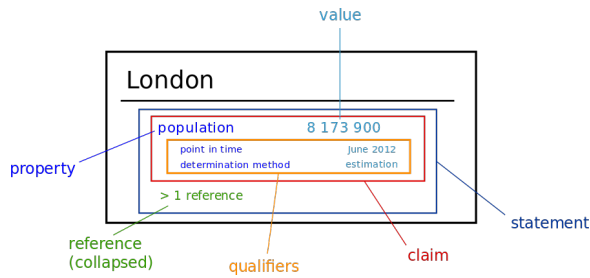
---

[1] http://wikiba.se/

Fig. 1. Wikidata statements, image taken from Commons[5]

In this article we describe the integration of Wikidata into the DBpedia Data Stack. While DBpedia has a relatively stable and commonly used ontology, people face difficulties when confronted with the young and still evolving Wikidata schema. As a result, with the DBpedia Wikidata (DBw) dataset can be queried with the same queries that are used with DBpedia.

2. Background

Wikidata [7] is community-created knowledge base to manage factual information of Wikipedia and its sister projects operated by the Wikimedia Foundation. In other words, Wikidata's goal is to be the central data management platform of Wikipedia. As of April 2016, Wikidata contains more than 20 million items and 87 million statements [2] and more than 2.7 million registered users[3]. In 2014, an RDF export of Wikidata was introduced [2] and recently a few SPARQL endpoints were made available as external contributions as well as an official one later on[4]. 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 depicted in Figure 1, each statement consists of a claim and an optional reference. Each claim consists of a property - value pair, and optional qualifiers. Values are also divided into three types: no value, unknown value and custom value. "No value" marker means that there is certainly no value for the property, "unknown value" marker means that there is some value, but exact value not known for the property and

"custom value", which provides a known value for the property. [6]

```
1  wkdt:Q42 wkdt:P26s wkdt:Q42Sb88670f8-456b
       -3ecb-cf3d-2bca2cf7371e.
2  wkdt:Q42Sb88670f8-456b-3ecb-cf3d-2
       bca2cf7371e wkdt:P580q wkdt:
       VT74cee544.
3  wkdt:VT74cee544 rdf:type :TimeValue.;
4   :time "1991-11-25"^^xsd:date;
5   :timePrecision "11"^^xsd:int;    :
       preferredCalendar wkdt:Q1985727.
6  wkdt:Q42Sb88670f8-456b-3ecb-cf3d-2
       bca2cf7371e wkdt:P582q wkdt:
       VT162aadcb.
7  wkdt:VT162aadcb rdf:type :TimeValue;
8   :time "2001-5-11"^^xsd:date;
9   :timePrecision "11"^^xsd:int;    :
       preferredCalendar wkdt:Q1985727.
```

Listing 1: RDF serialization for the fact: Douglas Adams (Q42) spouse is Jane Belson (Q14623681) from (P580) 25 November 1991 until (P582) 11 May 2001. Extracted from [7] Figure 3

DBpedia [5] 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 [4] and allows the easier integration of different Wikipedia language editions.

3. Challenges and Design Decisions

In this section we describe the design decisions we took to shape the DBpediaWikidata (DBw) dataset while maximising compatibility, (re)usability and coherence.

New IRI minting The most important design decision we had to take was whether to re-use the existing Wikidata IRIs or minting new IRIs in the DBpedia namespace. The decision dates back to 2013, when this project originally started and after lengthy discussions we concluded that minting new URIs was the only vi-

---

[2]https://tools.wmflabs.org/wikidata-todo/stats.php
[3]http://www.wikidata.org/wiki/Special:Statistics
[4]https://query.wikidata.org/

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

able option[7]. The main reason was the impedance mismatch between Wikidata data and DBpedia as both projects have minor deviations in conventions. Thus, creating new IRIs allows DBpedia to make local assertions on Wikidata resources without raising too many concerns.

Re-publishing minted IRIs as linked data   DBpedia has a big community and there has been extensive tool development to explore and exploit DBpedia data through the DBpedia ontology. Thus, we add another bubble in the LOD cloud, which helps the semantic web and DBpedia community for an easier transition to Wikidata data. The main use case of the DBw dataset for the DBpedia association is to create a new fused version of the Wikimedia ecosystem that integrates data from all DBpedia language editions, DBpedia Commons and Wikidata. Normalizing datasets to a common ontology is the first step towards data integration and fusion but most companies (e.g. Google, Freebase, Yahoo, Bing, Samsung) keep these datasets hidden. Our approach is to keep all the DBpedia data open to the community for reuse and feedback

Ontology design, reification and querying   The DBpedia ontology is an ontology developed and maintained since 2006, has reached a stable state and has 375 different datatypes[8]. The Wikidata ontology on the other hand is quite fresh and evolving and thus not as stable. Datatype support in Wikidata started at the end of 2015 and datatypes are still limited. In addition, Wikidata did not start with RDF as a first class citizen. There were different RDF serializations of Wikidata data and in particular different reification techniques. For example the RDF we get from content negotiation[9] is still different from the RDF dumps[10] and the announced reification design [2]. For these reasons we chose to use the DBpedia ontology and simple RDF reification. Performance-wise neither reification techniques brings any great advantage [3] and switching to the Wikidata reification scheme would require to duplicate all DBpedia properties.

---

[7]http://www.mail-archive.com/
dbpedia-discussion@lists.sourceforge.net/
msg05494.html
[8]http://mappings.dbpedia.org/index.php?
title=Special:AllPages&namespace=206
[9]https://www.wikidata.org/entity/Q42.nt
[10]http://tools.wmflabs.org/wikidata-exports/
rdf/

4. 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 Wiki-Text 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 5) and the Wikidata properties and the R2RWikidataExtractor that uses a mapping-based approach to map, in real-time, Wikidata statements to the DBpedia ontology. Figure 2 depics the current DBw extraction architecture.

4.1. 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*.

4.1.1. Schema Mappings
The DBpedia mappings wiki[11] is a community effort to map Wikipedia infoboxes to the DBpedia ontology and at the same time crowd-source 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`.[12] Although Wikidata does not define classes 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`.[13]

---

[11]http://mappings.dbpedia.org
[12]http://mappings.dbpedia.org/index.php/
OntologyProperty:BirthDate
[13]http://mappings.dbpedia.org/index.php/
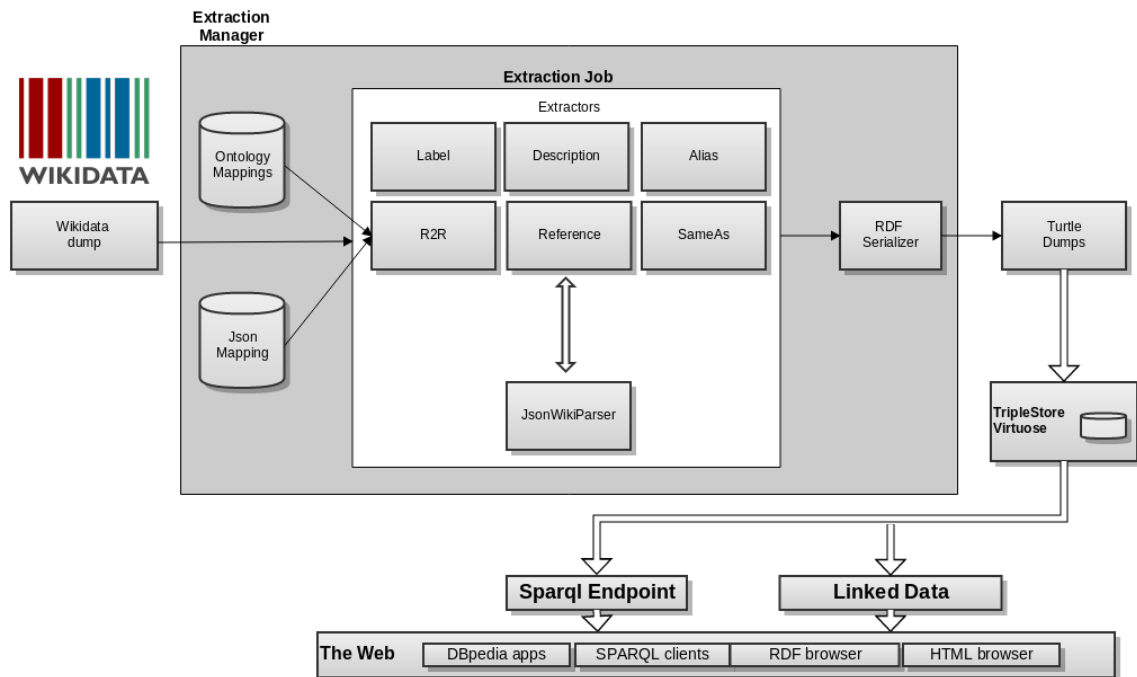OntologyClass:Person

Fig. 2. DBw extraction architecture

4.1.2. Value Transformations

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. [14] 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"}`.

$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 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:

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

Listing 2: Geographical DBw mappings

```
1  DW:Q64 rdf:type geo:SpatialThing ;
2    geo:lat "52.51667"^^xsd:float ;
3    geo:long "13.38333"^^xsd:float ;
```

```
4 ‖   geo:point "52.51667 13.38333" .
```

Listing 3: Resulting RDF from applied mappings for Wikidata item Q64

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.

4.2. 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.

4.3. 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`[15].

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: [16]

```
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 ;
```

Listing 4: Simple RDF reification example

IRI Splitting　The wikidata data model allows the same values as objects of different statements. In those, not so common, cases the simple IRI uniqeness fails and to solve this problem we append an additional hash on the IRI. We additionally add links from the simple IRI to the new ones using the `dbo:WikidataSplitIri` property[17]. At the time of writing, there are 69,662 IRI splitted triples.

5. Dataset Description

A statistical overview of the DBw dataset is provided in Table 2. We extract provenance information,

---

Title	Triples	Description
Provenance	20.3M	PageIDs & revisions
Redirects	855K	Explicit & transitive redirects
Aliases	5.3M	Resource aliases with dbo:alias
Labels	87.1M	Labels with rdfs:label
Descriptions	137M	Descriptions with dbo:description
Sitelinks	271M	DBpedia inter-language links
Wikidata links	19.4M	Links to original Wikidata URIs
Mapped facts	320M	Aggregated mapped facts
- Types	7.9M	Direct types from the DBPedia ontology
- Transitive Types	52M	Transitive types from the DBpedia ontology
- SubClassOf	332K	Wikidata SubClasOf DBpedia ontology
- Coordinates	9.5M	Geo coordinates
- Images	2.3M	Depictions using foaf:depiction & dbo:thumbnail
- Literals	4.7M	Wikidata literals with DBpedia ontology
- Other	27.5M	Wikidata statements DBpedia ontology
- External links	4.3M	sameAs links to external databases
Mapped facts (R)	210M	Mapped statements reified (all)
Mapped facts (RQ)	998K	Mapped qualifiers
Raw facts	82M	Raw simple statements (not mapped)
Raw facts (R)	328M	Raw statements reified
Raw facts (RQ)	2.2M	Raw qualifiers
References	56.8M	Reified statements references with dbo:reference
Mapping Errors	2.9M	Facts from incorrect mappings
Ontology Errors	42K	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

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), 2) owl:sameAs links between DBw IRIs and

Title	Before	After
Types	7,457,860	7,911,916
Transitive Types	49,438,753	52,042,144
SubClassOf	237,943	331,551

Table 2

Number of triples comparison before and after automatic class mappings extracted from Wikidata SubClassOf relations

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 4.1). 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 4).

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 4 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 4.2.

6. Statistics and Evaluation

The statistics we present are based on the Wikidata XML dump from January 2016. We managed to generate a total of 1.4B triples with 188,818,326 unique

Class	Count
dbo:Agent	3.43M
dbo:Person	3.08M
geo:spatialThing	2.39M
dbo:TopicalConcept	2.12M
dbo:Taxon	2.12M

Table 3

Top classes

Property	Count
owl:sameAs	320.8M
rdf:type	192.7M
dbo:description	136.9M
rdfs:label	87.2M
rdfs:seeAlso	10.1M

Table 4

Top properties

Property	Count
dbo:date	380K
dbo:startDate	265K
dbo:endDate	116K
dbo:country	40K
geo:point	31K

Table 5

Top mapped qualifiers

Property	Count
wd:P31	15,3M
wd:P17	3,9M
wd:P21	2,8M
wd:P131	2,7M
wd:P625	2,4M

Table 6

Top properties in Wikidata

Fig. 3. Number of daily visitors in http://wikidata.dbpedia.org

There were more than 10 million requests to wikidata.dbpedia.org since May 2015 from 28,557 unique IPs as of February 2016 and the daily visitors range from 300 to 2.7K. The access logs were analysed by using WebLog Expert[18]. The full report can be found on our website [19].

resources. In Table 2 we provide the number of triples per combined datasets.

Class & property statistics   We provide the 5 most popular DBw classes in Table 3. We managed to extract a total of 7.9M typed Things with Agents and Person as the most frequent types. The 5 most frequent mapped properties in simple statements are provided in Table 4 and the most popular mapped properties in qualifiers in Table 5. Wikidata does not have a complete range of value types and date paoperties are the most frefuent at the moment.

Mapping statistics   In total, 269 value transformation mappings were defined along with 185 `owl:equivalentProperty` and 323 `owl:equivalentClass` schema mappings. Wikidata has 1935 properties (as of January 2016) defined with a total of 81,998,766 occurrences. With the existing mappings we covered 74.21 % of the occurrences.

Redirects   In the current dataset we generated 854,578 redirects – including transitive. 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 2, a total of 2.9M errors originated from schema mappings and 42,541 triples did not pass the ontology validation (cf. Section 4.2).
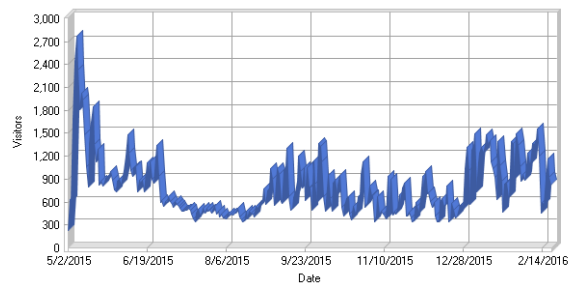
7. Access and Sustainability

This dataset is part of the official DBpedia knowledge infrastructure and is published through the regular releases of DBpedia, along with the rest of the DBpedia language editions. The first DBpedia release that included this dataset is DBpedia release 2015-04. 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.[20]

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[21] and provides machine readable metadata as void[22] and DataID[23] [1]. Since the project is now part of the offi-

---

[18]`https://www.weblogexpert.com/`
[19]`http://wikidata.dbpedia.org/report/report.html`
[20]`http://wikidata.dbpedia.org/downloads`
[21]`http://datahub.io/dataset/dbpedia-wikidata`
[22]`http://wikidata.dbpedia.org/downloads/void.ttl`
[23]`http://wikidata.dbpedia.org/downloads/20150330/dataid.ttl`

cial 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[24] for submitting bugs.

8. 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 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 4.3) 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`:[25]

```
1   #DBw
2   SELECT * WHERE
3   {
4     ?place a dbo:Place ;
5            dbo:country dw:Q183.
6     ?place rdfs:label ?label .
7     FILTER (LANG(?label)='en')
8   }
9
10  #wikidata
11  SELECT ?place ?placeLabel WHERE
12  {
13      ?place wdt:P31/wdt:P279 wd:Q486972;
14           wdt:P17 wd:Q183.
15
16    SERVICE wikibase:label {
17      bd:serviceParam wikibase:language "en".
18    }
```

```
19  }
```

Listing 5: Queries with simple statement.

```
1   #DBw
2   SELECT DISTINCT ?person WHERE
3   {
4     ?statementUri rdf:statement ?person ;
5                   rdf:predicate dbo:spouse ;
6                   dbo:startDate ?date.
7     FILTER (?date < "2000-01-01"^^xsd:date)
8   }
9
10  #wikidata
11  SELECT DISTINCT ?person WHERE
12  {
13    ?person p:P26 ?spouse.
14    ?spouse pq:P580 ?marriage_date.
15    FILTER (year(?marriage_date)<2000)
16
17  }
```

Listing 6: Queries with reified statements

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.

9. 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 1.4 billion RDF triples. According to the web server statistics the daily number DBw visitors range from 300 to 2,700 and we counted almost 30,000 unique IPs since the start of the project, which indicates that this dataset is heavily used. 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. HernÃąndez, A. Hogan, and M. KrÃűtzsch. Reifying rdf: What works well with wikidata? In T. Liebig and A. Fokoue, editors, *SSWS@ISWC*, volume 1457 of *CEUR Workshop Proceedings*, pages 32–47. CEUR-WS.org, 2015.

[4] 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.

[5] 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.

[6] G. Vaidya, D. Kontokostas, M. Knuth, J. Lehmann, and S. Hellmann. DBpedia Commons: Structured Multimedia Metadata from the Wikimedia Commons. In *Proceedings of the 14th International Semantic Web Conference*, Oct. 2015.

[7] D. Vrandečić and M. Krötzsch. Wikidata: A free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85, Sept. 2014.