

DBpedia: A Nucleus for a Web of Open Data

Sören Auer^{1,3}, Christian Bizer², Georgi Kobilarov², Jens Lehmann¹, Richard Cyganiak², and Zachary Ives³

¹ Universität Leipzig, Department of Computer Science, Johannisgasse 26,
D-04103 Leipzig, Germany,
{auer,lehmann}@informatik.uni-leipzig.de

² Freie Universität Berlin, Web-based Systems Group, Garystr. 21,
D-14195 Berlin, Germany,
chris@bizer.de, georgi.kobilarov@gmx.de richard@cyganiak.de

³ University of Pennsylvania, Department of Computer and Information Science
Philadelphia, PA 19104, USA,
auer@seas.upenn.edu, zives@cis.upenn.edu

Abstract DBpedia is a community effort to extract structured information from Wikipedia and to make this information available on the Web. DBpedia allows you to ask sophisticated queries against datasets derived from Wikipedia and to link other datasets on the Web to Wikipedia data. We describe the extraction of the DBpedia datasets, and how the resulting information is published on the Web for human- and machine-consumption. We describe some emerging applications from the DBpedia community and show how website authors can facilitate DBpedia content within their sites. Finally, we present the current status of interlinking DBpedia with other open datasets on the Web and outline how DBpedia could serve as a nucleus for an emerging Web of open data.

1 Introduction

It is now almost universally acknowledged that stitching together the world’s structured information and knowledge to answer semantically rich queries is one of the key challenges of computer science, and one that is likely to have tremendous impact on the world as a whole. This has led to almost 30 years of research into information integration [15,19] and ultimately to the Semantic Web and related technologies [1,11,13]. Such efforts have generally only gained traction in relatively small and specialized domains, where a closed ontology, vocabulary, or schema could be agreed upon. However, the broader Semantic Web vision has not yet been realized, and one of the biggest challenges facing such efforts has been how to get enough “interesting” and broadly useful information into the system to make it useful and accessible to a *general* audience.

A challenge is that the traditional “top-down” model of designing an ontology or schema *before* developing the data breaks down at the scale of the Web: both data and metadata must constantly evolve, and they must serve many different communities. Hence, there has been a recent movement to build the Semantic Web grass-roots-style, using incremental and Web 2.0-inspired collaborative

approaches [10,12,13]. Such a collaborative, grass-roots Semantic Web requires a new model of structured information representation and management: first and foremost, it must handle inconsistency, ambiguity, uncertainty, data provenance [3,6,8,7], and implicit knowledge in a uniform way.

Perhaps the most effective way of spurring synergistic research along these directions is to provide a rich corpus of diverse data. This would enable researchers to develop, compare, and evaluate different extraction, reasoning, and uncertainty management techniques, and to deploy operational systems on the Web.

The DBpedia project has derived such a data corpus from the Wikipedia encyclopedia. Wikipedia is heavily visited and under constant revision (e.g., according to alexa.com, Wikipedia was the 9th most visited website in the third quarter of 2007). Wikipedia editions are available in over 250 languages, with the English one accounting for more than 1.95 million articles. Like many other web applications, Wikipedia has the problem that its search capabilities are limited to full-text search, which only allows very limited access to this valuable knowledge base. As has been highly publicized, Wikipedia also exhibits many of the challenging properties of collaboratively edited data: it has contradictory data, inconsistent taxonomical conventions, errors, and even spam.

The DBpedia project focuses on the task of converting Wikipedia content into structured knowledge, such that Semantic Web techniques can be employed against it — asking sophisticated queries against Wikipedia, linking it to other datasets on the Web, or creating new applications or mashups. We make the following contributions:

- We develop an information extraction framework, which converts Wikipedia content to RDF. The basic components form a foundation upon which further research into information extraction, clustering, uncertainty management, and query processing may be conducted.
- We provide Wikipedia content as a large, multi-domain RDF dataset, which can be used in a variety of Semantic Web applications. The DBpedia dataset consists of 103 million RDF triples.
- We interlink the DBpedia dataset with other open datasets. This results in a large Web of data containing altogether around 2 billion RDF triples.
- We develop a series of interfaces and access modules, such that the dataset can be accessed via Web services and linked to other sites.

The DBpedia datasets can be either imported into third party applications or can be accessed online using a variety of DBpedia user interfaces. Figure 1 gives an overview about the DBpedia information extraction process and shows how extracted data is published on the Web. These main DBpedia interfaces currently use Virtuoso [9] and MySQL as storage back-ends.

The paper is structured as follows: We give an overview about the DBpedia information extraction techniques in Section 2. The resulting datasets are described in Section 3. We exhibit methods for programmatic access to the DBpedia dataset in Section 4. In Sections 5 we present our vision of how the DBpedia

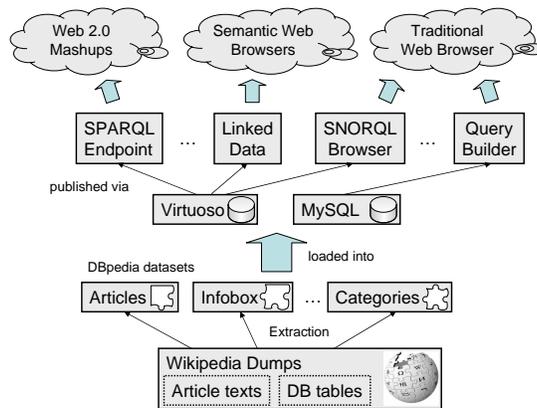


Figure 1. Overview of the DBpedia components.

datasets can be a nucleus for a Web of open data. We showcase several user interfaces for accessing DBpedia in Section 6 and finally review related work in Section 7.

2 Extracting Structured Information from Wikipedia

Wikipedia articles consist mostly of free text, but also contain different types of structured information, such as infobox templates, categorisation information, images, geo-coordinates, links to external Web pages and links across different language editions of Wikipedia.

Mediawiki⁴ is the software used to run Wikipedia. Due to the nature of this Wiki system, basically all editing, linking, annotating with meta-data is done inside article texts by adding special syntactic constructs. Hence, structured information can be obtained by parsing article texts for these syntactic constructs.

Since MediaWiki exploits some of this information itself for rendering the user interface, some information is cached in relational database tables. Dumps of the crucial relational database tables (including the ones containing the article texts) for different Wikipedia language versions are published on the Web on a regular basis⁵. Based on these database dumps, we currently use two different methods of extracting semantic relationships: (1) We map the relationships that are already stored in relational database tables onto RDF and (2) we extract additional information directly from the article texts and infobox templates within the articles.

We illustrate the extraction of semantics from article texts with an Wikipedia infobox template example. Figure 2 shows the infobox template (encoded within

⁴ <http://www.mediawiki.org>

⁵ <http://download.wikimedia.org/>

```

{{infobox City Korea|
  full_name=Busan Metropolitan City|
  image=[[Image:Haeundaebeachbusan.jpg|
    250px|Haeundae Beach, Busan]]|
  rr=Busan Gwangyeoksi|
  mr=Pusan Kwangyöksi|
  hangul=부산 광역시|
  hanja=釜山廣域市|
  short_name=Busan (Pusan; 부산; 釜山)|
  population=3,635,389 ...|
  area=763.46 km2|
  government=[[Metropolitan cities of
    South Korea|Metropolitan City]]|
  divisions=15 wards (Gu),
  <br>1 county (Gun)|
  region=[[Yeongnam]]|
  dialect=[[Gyeongsang Dialect|
    Gyeongsang]]|
  map=[[Image:Busan map.png|Map of
    South Korea highlighting the city]]|
}}

```

Busan Metropolitan City	
	
Korean name	
Revised Romanization	Busan Gwangyeoksi
McCune-Reischauer	Pusan Kwangyöksi
Hangul	부산 광역시
Hanja	釜山廣域市
Short name	Busan (Pusan; 부산; 釜山)

Figure 2. Example of a Wikipedia template and rendered output (excerpt).

a Wikipedia article) and the rendered output of the South-Korean town Busan. The infobox extraction algorithm detects such templates and recognizes their structure using pattern matching techniques. It selects significant templates, which are then parsed and transformed to RDF triples. The algorithm uses post-processing techniques to increase the quality of the extraction. MediaWiki links are recognized and transformed to suitable URIs, common units are detected and transformed to data types. Furthermore, the algorithm can detect lists of objects, which are transformed to RDF lists. Details about the infobox extraction algorithm (including issues like data type recognition, cleansing heuristics and identifier generation) can be found in [2]. All extraction algorithms are implemented using PHP and are available under an open-source license⁶.

3 The DBpedia Dataset

The DBpedia dataset currently provides information about more than 1.95 million "things", including at least 80,000 persons, 70,000 places, 35,000 music albums, 12,000 films. It contains 657,000 links to images, 1,600,000 links to relevant external web pages, 180,000 external links into other RDF datasets, 207,000 Wikipedia categories and 75,000 YAGO categories [16].

DBpedia concepts are described by short and long abstracts in 13 different languages. These abstracts have been extracted from the English, German,

⁶ <http://sf.net/projects/dbpedia>

French, Spanish, Italian, Portuguese, Polish, Swedish, Dutch, Japanese, Chinese, Russian, Finnish and Norwegian versions of Wikipedia.

Altogether the DBpedia dataset consists of around 103 million RDF triples. The dataset is provided for download as a set of smaller RDF files. Table 1 gives an overview over these files.

Dataset	Description	Triples
<i>Articles</i>	Descriptions of all 1.95 million concepts within the English Wikipedia including titles, short abstracts, thumbnails and links to the corresponding articles.	7.6M
<i>Ext. Abstracts</i>	Additional, extended English abstracts.	2.1M
<i>Languages</i>	Additional titles, short abstracts and Wikipedia article links in German, French, Spanish, Italian, Portuguese, Polish, Swedish, Dutch, Japanese, Chinese, Russian, Finnish and Norwegian.	5.7M
<i>Lang. Abstracts</i>	Extended abstracts in 13 languages.	1.9M
<i>Infoboxes</i>	Data attributes for concepts that have been extracted from Wikipedia infoboxes.	15.5M
<i>External Links</i>	Links to external web pages about a concept.	1.6M
<i>Article Categories</i>	Links from concepts to categories using SKOS.	5.2M
<i>Categories</i>	Information which concept is a category and how categories are related.	1M
<i>Yago Types</i>	Dataset containing rdf:type Statements for all DBpedia instances using classification from YAGO [16].	1.9 M
<i>Persons</i>	Information about 80,000 persons (date and place of birth etc.) represented using the FOAF vocabulary.	0.5M
<i>Page Links</i>	Internal links between DBpedia instances derived from the internal pagelinks between Wikipedia articles.	62M
<i>RDF Links</i>	Links between DBpedia and Geonames, US Census, Musicbrainz, Project Gutenberg, the DBLP bibliography and the RDF Book Mashup.	180K

Table 1. The DBpedia datasets.

Some datasets (such as the *Persons* or *Infoboxes* datasets) are semantically rich in the sense that they contain very specific information. Others (such as the *PageLinks* dataset) contain meta-data (such as links between articles) without a specific semantics. However, the latter can be beneficial, e.g. for deriving measures of closeness between concepts or relevance in search results.

Each of the 1.95 million resources described in the DBpedia dataset is identified by a URI reference of the form `http://dbpedia.org/resource/Name`, where *Name* is taken from the URL of the source Wikipedia article, which has the form `http://en.wikipedia.org/wiki/Name`. Thus, each resource is tied directly to an English-language Wikipedia article. This yields certain beneficial properties to DBpedia identifiers:

- They cover a wide range of encyclopedic topics,
- They are defined by community consensus,

- There are clear policies in place for their management,
- And an extensive textual definition of the concept is available at a well-known web location (the Wikipedia page).

4 Accessing the DBpedia Dataset on the Web

We provide three access mechanisms to the DBpedia dataset: Linked Data, the SPARQL protocol, and downloadable RDF dumps. Royalty-free access to these interfaces is granted under the terms of the GNU Free Documentation License.

Linked Data. Linked Data is a method of publishing RDF data on the Web that relies on `http://` URIs as resource identifiers and the HTTP protocol to retrieve resource descriptions [4,5]. The URIs are configured to return meaningful information about the resource—typically, an RDF description containing everything that is known about it. Such a description usually mentions related resources by URI, which in turn can be accessed to yield their descriptions. This forms a dense mesh of web-accessible resource descriptions that can span server and organization boundaries. DBpedia resource identifiers, such as `http://dbpedia.org/resource/Busan`, are set up to return RDF descriptions when accessed by Semantic Web agents, and a simple HTML view of the same information to traditional web browsers (see Figure 3). HTTP content negotiation is used to deliver the appropriate format.

Web agents that can access Linked Data include: 1. Semantic Web browsers like Disco⁷, Tabulator[17] (see Figure 3), or the OpenLink Data Web Browser⁸; 2. Semantic Web crawlers like SWSE⁹ and Swoogle¹⁰; 3. Semantic Web query agents like the Semantic Web Client Library¹¹ and the SemWeb client for SWI prolog¹².

SPARQL Endpoint. We provide a SPARQL endpoint for querying the DBpedia dataset. Client applications can send queries over the SPARQL protocol to this endpoint at `http://dbpedia.org/sparql`. This interface is appropriate when the client application developer knows in advance exactly what information is needed. In addition to standard SPARQL, the endpoint supports several extensions of the query language that have proved useful for developing user interfaces: full text search over selected RDF predicates, and aggregate functions, notably COUNT. To protect the service from overload, limits on query cost and result size are in place. For example, a query that asks for the store’s entire contents is rejected as too costly. SELECT results are truncated at 1000 rows. The SPARQL endpoint is hosted using Virtuoso Universal Server¹³.

⁷ <http://sites.wiwiiss.fu-berlin.de/suhl/bizer/ng4j/disco/>

⁸ <http://demo.openlinksw.com/DAV/JS/rdfbrowser/index.html>

⁹ <http://swse.org>

¹⁰ <http://swoogle.umbc.edu/>

¹¹ <http://sites.wiwiiss.fu-berlin.de/suhl/bizer/ng4j/semwebclient/>

¹² <http://moustaki.org/swic/>

¹³ <http://virtuoso.openlinksw.com>



Figure 3. <http://dbpedia.org/resource/Busan> viewed in a web browser (left) and in Tabulator (right).

RDF Dumps. N-Triple serializations of the datasets are available for download at the DBpedia website and can be used by sites that are interested in larger parts of the dataset.

5 Interlinking DBpedia with other Open Datasets

In order to enable DBpedia users to discover further information, the DBpedia dataset is interlinked with various other data sources on the Web using RDF links. RDF links enable web surfers to navigate from data within one data source to related data within other sources using a Semantic Web browser. RDF links can also be followed by the crawlers of Semantic Web search engines, which may provide sophisticated search and query capabilities over crawled data.

The DBpedia interlinking effort is part of the Linking Open Data community project¹⁴ of the W3C Semantic Web Education and Outreach (SWEO) interest group. This community project is committed to make massive datasets and ontologies, such as the US Census, Geonames, MusicBrainz, the DBLP bibliography, WordNet, Cyc and many others, interoperable on the Semantic Web. DBpedia, with its broad topic coverage, intersects with practically all these datasets and therefore makes an excellent “linking hub” for such efforts.

Figure 4 gives an overview about the datasets that are currently interlinked with DBpedia. Altogether this Web-of-Data amounts to approximately 2 billion RDF triples. Using these RDF links, surfers can for instance navigate from a computer scientist in DBpedia to her publications in the DBLP database, from a DBpedia book to reviews and sales offers for this book provided by the RDF Book Mashup, or from a band in DBpedia to a list of their songs provided by Musicbrainz or dbtune.

The example RDF link shown below connects the DBpedia URI identifying Busan with further data about the city provided by Geonames:

`<http://dbpedia.org/resource/Busan>`

¹⁴ <http://esw.w3.org/topic/SweoIG/TaskForces/CommunityProjects/LinkingOpenData>

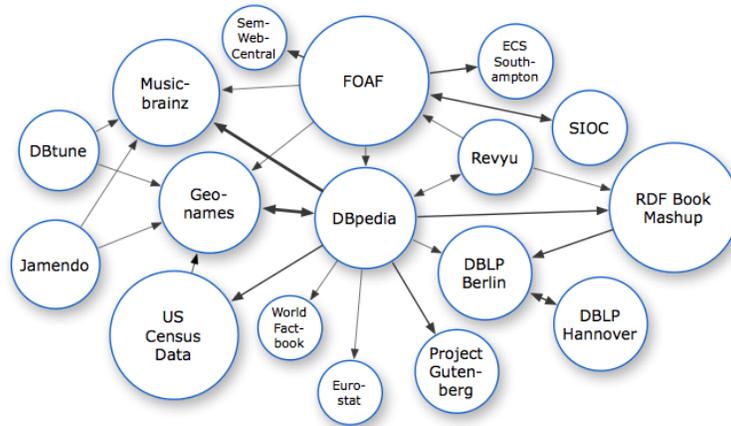


Figure 4. Datasets that are interlinked with DBpedia.

```
owl:sameAs <http://sws.geonames.org/1838524/> .
```

Agents can follow this link, retrieve RDF from the Geonames URI, and thereby get hold of additional information about Busan as published by the Geonames server, which again contains further links deeper into the Geonames data. DBpedia URIs can also be used to express personal interests, places of residence, and similar facts within personal FOAF profiles:

```
<http://richard.cyganiak.de/foaf.rdf#cygri>
  foaf:topic_interest <http://dbpedia.org/resource/Semantic_Web> ;
  foaf:based_near <http://dbpedia.org/resource/Berlin> .
```

Another use case is categorization of blog posts, news stories and other documents. The advantage of this approach is that all DBpedia URIs are backed with data and thus allow clients to retrieve more information about a topic:

```
<http://news.cnn.com/item1143>
  dc:subject <http://dbpedia.org/resource/Iraq_War> .
```

6 User Interfaces

User interfaces for DBpedia can range from a simple table within a classic web page, over browsing interfaces to different types of query interfaces. This section gives an overview about the different user interfaces that have been implemented so far.

6.1 Simple Integration of DBpedia Data into Web Pages

DBpedia is a valuable source of general-purpose data that can be used within web pages. Therefore, if you want a table containing German state capitals, African musicians, Amiga computer games or whatever on your website, you can generate this table using a SPARQL query against the DBpedia endpoint. Wikipedia is kept up-to-date by a large community and a nice feature of such tables is that they will also stay up-to-date as Wikipedia, and thus also DBpedia, changes. Such tables can either be implemented using Javascript on the client or with a scripting language like PHP on the server. Two examples of Javascript generated tables are found on the DBpedia website¹⁵.

6.2 Search DBpedia.org

Search DBpedia.org is a sample application that allows users to explore the DBpedia dataset together with information from interlinked datasets such as Geonames, the RDF Book Mashup or the DBLP bibliography. In contrast to the keyword-based full-text search commonly found on the Web, search over structured data offers the opportunity to make productive use of the relations in the data, enabling stepwise narrowing of search results in different dimensions. This adds a browsing component to the search task and may reduce the common “keyword-hit-or-not-hit” problem.

A *Search DBpedia.org* session starts with a keyword search. A first set of results is computed by direct keyword matches. Related matches are added, using the relations between entities up to a depth of two nodes. Thus, a search for the keyword “Scorsese” will include the director Martin Scorsese, as well as all of his films, and the actors of these films.

The next step is result ranking. Our experiments showed that important articles receive more incoming page links from other articles. We use a combination of incoming link count, relevance of the link’s source, and relation depth to calculate a relevance ranking.

After entering a search term, the user is presented with a list of ranked results, and with a tag cloud built from the classes found in the results, using a combination of the DBpedia and YAGO [16] classifications. Each class weight is calculated from the sum of associated result weights and the frequency of occurrence. The tag cloud enables the user to narrow the results to a specific type of entities, such as “Actor”, even though a simple keyword search may not have brought up any actors.

When a resource from the results is selected, the user is presented with a detailed view of all data that is known about the resource. Label, image and description are shown on top. Single-valued and multi-valued properties are shown separately. Data from interlinked datasets is automatically retrieved by following RDF links within the dataset and retrieved data from interlinked datasets is shown together with the DBpedia data.

¹⁵ <http://dbpedia.org>

The image shows two side-by-side screenshots of the Search DBpedia.org interface. The left screenshot shows the search results for the query 'Busan', with 'city' being the primary result. The right screenshot shows the detailed view for 'Busan', which includes a map, a list of properties such as 'Population: 3,657,840 (as of 12 31 2005)', 'Full name: Busan Metropolitan City', and 'Description'. The description states that Busan is the second largest metropolis in South Korea, situated in the narrow Nakdonggang valley. It also lists various related entities like 'Busan Subway' and 'Busan Lotte Tower'.

Figure 5. Search results and details view for Busan.

6.3 Querying DBpedia Data

Compared to most of the other Semantic Web knowledge bases currently available, for the RDF extracted from Wikipedia we have to deal with a different type of knowledge structure – we have a very large information schema and a considerable amount of data adhering to this schema. Existing tools unfortunately mostly focus on either one of both parts of a knowledge base being large, schema *or* data.

If we have a large data set and large data schema, elaborated RDF stores with integrated query engines alone are not very helpful. Due to the large data schema, users can hardly know which properties and identifiers are used in the knowledge base and hence can be used for querying. Consequently, users have to be guided when building queries and reasonable alternatives should be suggested.

We specifically developed a graph pattern builder for querying the extracted Wikipedia content. Users query the knowledge base by means of a graph pattern consisting of multiple triple patterns. For each triple pattern three form fields capture variables, identifiers or filters for subject, predicate and object of a triple. While users type identifier names into one of the form fields, a look-ahead search proposes suitable options. These are obtained not just by looking for matching identifiers but by executing the currently built query using a variable for the currently edited identifier and filtering the results returned for this variable for matches starting with the search string the user supplied. This method ensures, that the identifier proposed is really used in conjunction with the graph pattern under construction and that the query actually returns results. In addition, the identifier search results are ordered by usage number, showing commonly used identifiers first. All this is executed in the background, using the Web 2.0 AJAX technology and hence completely transparent for the user. Figure 6 shows a screenshot of the graph pattern builder.

UNIVERSITÄT LEIPZIG pedia

Query Wikipedia

This semantic database contains over 10 million statements extracted from the English Wikipedia.

[search for queries](#) | [Most popular](#) | [Upcoming](#)

[Tennis players from Moscow](#)

[Sitcoms set in NYC](#)

[Soccer player with tricot nr. 11, playing for a club having a stadium with >40.000 seats, born in a country with >10M inhabitants](#)

[People influenced by Friedrich Nietzsche](#)

[Films longer than 5 hours](#)

[Space Missions](#)

[Film music composer born 1965](#)

[People being 1.80m tall](#)

[List of Web browser software](#)

[Mayors of US cities higher than 1000m](#)

[Pictures of American guitarists](#)

[Battles in Saxony](#)

[What connects Innsbruck and Leipzig](#)

[Hip hop CDs from Texas Artists](#)

[Scientists and their doctoral advisors](#)

<< >>

Soccer player with tricot nr. 11, playing for a club having a stadium with >40.000 seats, born in a country with >10M inhabitants

Subject	Predicate	Object
?player	currentclub	?club
?player	clubnumber	11
?player	countryofbirth	?country
?club	capacity	>40000
?country		>10000000
[+]	GDP_PPP (11)	
	population_estimate (10)	
	population_census (9)	
	established_date2 (8)	
	established_date1 (6)	
	established_date3 (5)	
	GDP_nominal (3)	
	accessionEUdate (2)	

Click on a column header to filter this page. Results:

10 results found in 0.00

Nr.	?player	?club	?country	>40000	>10000000
1	Cicinho	Kearif Marito	Brazil	80354	187560000
2	Gonzalo Fierro	Colo-Colo	Chile	62000	16432674
3	Lukas Podolski	FC Bayern Munich	Poland	69901	38536869
4	Mark González	Liverpool F.C.	South Africa	45362	47432000
5	Michael Thürk	Eintracht Frankfurt	Germany	52000	82438000
6	Ramón Morales	Chivas de Guadalajara	Mexico	72480	107784179
7	Robin van Persie	Arsenal F.C.	Netherlands	60432	16336346
8	Stefano Mauri	S.S. Lazio	Italy	82656	58751711

Figure 6. Form based query builder.

6.4 Third Party User Interfaces

The DBpedia project aims at providing a hotbed for applications and mashups based on information from Wikipedia. Although DBpedia was just recently launched, there is already a number of third party applications using the dataset. Examples include:

- A SemanticMediaWiki [14,18] installation run by the University of Karlsruhe, which has imported the DBpedia dataset together with the English edition of Wikipedia.
- WikiStory (see Figure 7) which enables users to browse Wikipedia articles about people on a large timeline.
- The Objectsheet JavaScript visual data environment, which allows spreadsheet calculations based on DBpedia data¹⁶.

7 Related Work

A second project that also works on extracting structured information from Wikipedia is the YAGO project [16]. YAGO extracts only 14 relationship types, such as *subclassOf*, *type*, *familyNameOf*, *locatedIn* from different sources of information in Wikipedia. One source is the Wikipedia category system (for *subclassOf*, *locatedIn*, *diedInYear*, *bornInYear*), and another one are Wikipedia redirects. YAGO does not perform an infobox extraction as in our approach. For determining (sub-)class relationships, YAGO does not use the full Wikipedia category hierarchy, but links leaf categories to the WordNet hierarchy.

¹⁶ http://richk.net/objectsheet/osc.html?file=sparql_query1.os

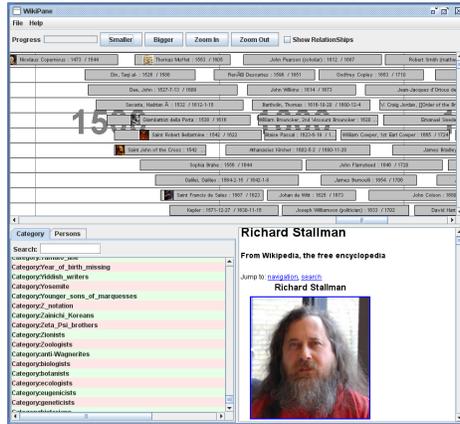


Figure 7. WikiStory allows timeline browsing of biographies in Wikipedia.

The Semantic MediaWiki project [14,18] also aims at enabling the reuse of information within Wikis as well as at enhancing search and browse facilities. Semantic MediaWiki is an extension of the MediaWiki software, which allows you to add structured data into Wikis using a specific syntax. Ultimately, the DBpedia and Semantic MediaWiki have similar goals. Both want to deliver the benefits of structured information in Wikipedia to the users, but use different approaches to achieve this aim. Semantic MediaWiki requires authors to deal with a new syntax and covering all structured information within Wikipedia would require to convert all information into this syntax. DBpedia exploits the structure that already exists within Wikipedia and hence does not require deep technical or methodological changes. However, DBpedia is not as tightly integrated into Wikipedia as is planned for Semantic MediaWiki and thus is limited in constraining Wikipedia authors towards syntactical and structural consistency and homogeneity.

Another interesting approach is followed by Freebase¹⁷. The project aims at building a huge online database which users can edit in a similar fashion as they edit Wikipedia articles today. The DBpedia community cooperates with Metaweb and we will interlink data from both sources once Freebase is public.

8 Future Work and Conclusions

As future work, we will first concentrate on improving the quality of the DBpedia dataset. We will further automate the data extraction process in order to increase the currency of the DBpedia dataset and synchronize it with changes in Wikipedia. In parallel, we will keep on exploring different types of user interfaces and use cases for the DBpedia datasets. Within the W3C Linking Open

¹⁷ <http://www.freebase.com>

Data community project¹⁸, we will interlink the DBpedia dataset with further datasets as they get published as Linked Data on the Web. We also plan to exploit synergies between Wikipedia versions in different languages in order to further increase DBpedia coverage and provide quality assurance tools to the Wikipedia community. Such a tool could for instance notify a Wikipedia author about contradictions between the content of infoboxes contained in the different language versions of an article. Interlinking DBpedia with other knowledge bases such as Cyc (and their use as back-ground knowledge) could lead to further methods for (semi-) automatic consistency checks for Wikipedia content.

DBpedia is a major source of open, royalty-free data on the Web. We hope that by interlinking DBpedia with further data sources, it could serve as a nucleus for the emerging Web of Data.

Acknowledgments

We are grateful to the members of the growing DBpedia community, who are actively contributing to the project. In particular we would like to thank Jörg Schüppel and the OpenLink team around Kingsley Idehen and Orri Erling.

References

1. Karl Aberer, Philippe Cudré-Mauroux, and Manfred Hauswirth. The chatty web: Emergent semantics through gossiping. In *12th World Wide Web Conference*, 2003.
2. Sören Auer and Jens Lehmann. What have innsbruck and leipzig in common? extracting semantics from wiki content. In Enrico Franconi, Michael Kifer, and Wolfgang May, editors, *ESWC*, volume 4519 of *Lecture Notes in Computer Science*, pages 503–517. Springer, 2007.
3. Omar Benjelloun, Anish Das Sarma, Alon Y. Halevy, and Jennifer Widom. Uldbs: Databases with uncertainty and lineage. In *VLDB*, 2006.
4. Tim Berners-Lee. Linked data, 2006. <http://www.w3.org/DesignIssues/LinkedData.html>.
5. Christian Bizer, Richard Cyganiak, and Tom Heath. How to publish linked data on the web, 2007. <http://sites.wiwiiss.fu-berlin.de/suhl/bizer/pub/LinkedDataTutorial/>.
6. Peter Buneman, Sanjeev Khanna, and Wang Chiew Tan. Why and where: A characterization of data provenance. In *ICDT*, volume 1973 of *Lecture Notes in Computer Science*, 2001.
7. Christian Bizer. *Quality-Driven Information Filtering in the Context of Web-Based Information Systems*. PhD thesis, Freie Universität Berlin, 2007.
8. Yingwei Cui. *Lineage Tracing in Data Warehouses*. PhD thesis, Stanford University, 2001.
9. Orri Erling and Ivan Mikhailov. RDF support in the Virtuoso DBMS. volume P-113 of *GI-Edition - Lecture Notes in Informatics (LNI)*, ISSN 1617-5468. Bonner Köllen Verlag, September 2007.

¹⁸ <http://esw.w3.org/topic/SweoIG/TaskForces/CommunityProjects/LinkingOpenData>

10. Alon Halevy, Oren Etzioni, AnHai Doan, Zachary Ives, Jayant Madhavan, and Luke McDowell. Crossing the structure chasm. In *CIDR*, 2003.
11. Alon Y. Halevy, Zachary G. Ives, Dan Suciu, and Igor Tatarinov. Schema mediation in peer data management systems. In *ICDE*, March 2003.
12. Zachary Ives, Nitin Khandelwal, Aneesh Kapur, and Murat Cakir. ORCHESTRA: Rapid, collaborative sharing of dynamic data. In *CIDR*, January 2005.
13. Anastasios Kementsietsidis, Marcelo Arenas, and Renée J. Miller. Mapping data in peer-to-peer systems: Semantics and algorithmic issues. In *SIGMOD*, June 2003.
14. Markus Krötzsch, Denny Vrandečić, and Max Völkel. Wikipedia and the Semantic Web - The Missing Links. In Jakob Voss and Andrew Lih, editors, *Proceedings of Wikimania 2005, Frankfurt, Germany*, 2005.
15. John Miles Smith, Philip A. Bernstein, Umeshwar Dayal, Nathan Goodman, Terry Landers, Ken W.T. Lin, and Eugene Wong. MULTIBASE – integrating heterogeneous distributed database systems. In *Proceedings of 1981 National Computer Conference*, 1981.
16. Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. Yago: A Core of Semantic Knowledge. In *16th international World Wide Web conference (WWW 2007)*, New York, NY, USA, 2007. ACM Press.
17. Tim Berners-Lee et al. Tabulator: Exploring and analyzing linked data on the semantic web. In *Proceedings of the 3rd International Semantic Web User Interaction Workshop*, 2006. <http://swui.semanticweb.org/swui06/papers/Berners-Lee/Berners-Lee.pdf>.
18. Max Völkel, Markus Krötzsch, Denny Vrandečić, Heiko Haller, and Rudi Studer. Semantic wikipedia. In Les Carr, David De Roure, Arun Iyengar, Carole A. Goble, and Michael Dahlin, editors, *Proceedings of the 15th international conference on World Wide Web, WWW 2006*, pages 585–594. ACM, 2006.
19. Gio Wiederhold. Intelligent integration of information. In *SIGMOD*, 1993.