OntoWiki

a Semantic Data Wiki Enabling the Collaborative Creation and (Linked Data) Publication of RDF Knowledge Bases

Sebastian Tramp   Philipp Frischmuth   Norman Heino

{tramp|frischmuth|heino}@informatik.uni-leipzig.de

Agile Knowledge Engineering and Semantic Web (AKSW)
University of Leipzig / Institute of Computer Science
Johannisgasse 26, 04103 Leipzig, Germany

ABSTRACT

In our demonstration (with supportive poster) we want to present the Semantic Wiki application OntoWiki, which is an extensible tool for managing structured information in a collaborative, web-based environment. OntoWiki provides sophisticated means for navigating, visualising and authoring of RDF-based Knowledge Bases. It serves and consumes Linked Data and comprises a comprehensive middleware API for building custom Semantic Web applications.

We refer to it as a *Wiki*, since our focus is on simplicity, adaptability and collaboration. However, other than annotating text-based Wiki pages with a special syntax (as suggested by text-based Semantic Wiki approaches), OntoWiki uses RDF in the first place to represent information. For human users, OntoWiki allows to create different views on data, such as tabular representations or maps. For machine consumption it supports various RDF serialisations as well as RDFa, Linked Data and SPARQL interfaces. Since its introduction [1], the application has evolved into a framework for building Semantic Web applications [2] and was recently updated to support the collaboration across multiple domains and application via Semantic Pingback [3] and RDFauthor [5].

1. DEMONSTRATION CONTENT

OntoWiki can be used to author, manage and publish RDF-based knowledge bases. In particular, our demonstration will focus on the following topics[1]. We have highlighted several phrases and keywords to give the reader a better overview on the main demonstration points.

Navigation and Visualisation

OntoWiki provides a number of ways for navigating through RDF knowledge bases. These include **taxonomy and hierarchy browsing** (e.g. via SKOS taxonomies or the class hierarchy), **facet-based browsing** (with **complex filter** conditions and **attribute based tag clouds**) and **full-text search**. These different navigation facilities can also be used as a **combined navigation**, for example a user can start with a full-text search and **refine the results** using faceted-search or restrict them to a certain part of the selected hierarchy. OntoWiki translates these different navigation features into a single SPARQL query. In addition to rendering the results of such a query as a **resource list**, OntoWiki provides an **extension mechanism** for implementing dedicated views like **maps and calendars**. Once a specific entity has been selected, the **resource view** displays all available information for that entity. Besides rendering this information generically in a tabular way, OntoWiki can be extended with **domain specific resource views** (i.e. vocabulary). Specific resource views for the SKOS and FOAF vocabularies are already included. Figure 1 shows the domain-independent list and resource view of OntoWiki.

Authoring

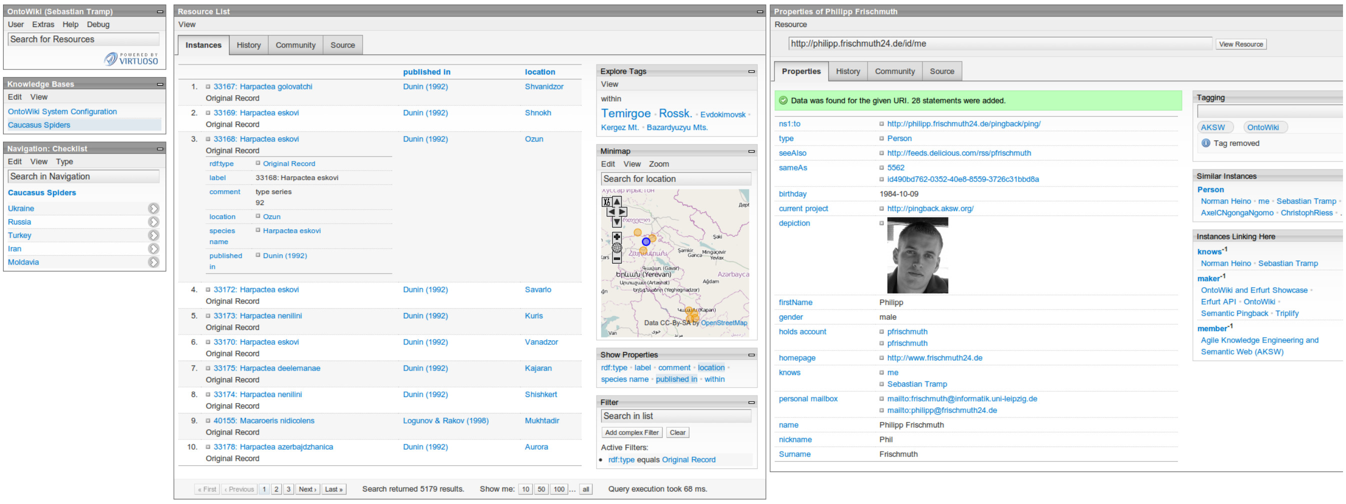
All views in OntoWiki can be equipped with corresponding **RDFa annotations**. OntoWiki employs the **RDFauthor mechanism** [4] to automatically transform these views into **editable forms**. As a consequence, all information items displayed in OntoWiki can be directly edited in place. This allows OntoWiki users to edit the knowledge base even without being acquainted with the RDF, RDF-Schema or OWL data models. Following the Wiki approach all changes are put under **version control** and can be easily **rolled back**.

In addition to this, the integration of RDFauthor as our main authoring mechanism led to two additional usage options:

- By using OntoWiki's RDFauthor **bookmarklet**, users can **collect data from different web pages** and import it directly into their personal knowledge bases (e.g. contacts or events).

- OntoWiki can be used as a service for hosting **editable mash-ups**. These mash-ups use data from OntoWiki's **SPARQL endpoint** and other data sources (which are interpreted as named graphs) and provide a merged view on this data. The bookmarklet is able to distinguish the statements from these named graphs and allow the user to edit these data directly inside the mash-up as well as to **propagate the changes back**

[1] The feature list is far away from being complete, so please refer `http://ontowiki.net` (for end-user centred functionality and screen-casts) and `http://code.google.com/p/ontowiki/` (for administration features and extensibility as well as development information) for a more complete view on our application.


OntoWiki (Sebastian Tramp)
User Extras Help Debug
Search for Resources
POWERED BY VIRTUOSO
Knowledge Bases
Edit View
OntoWiki System Configuration
Caucasus Spiders
Navigation: Checklist
Edit View Type
Search in Navigation
Caucasus Spiders
Ukraine
Russia
Turkey
Iran
Moldavia
Resource List
View
Instances History Community Source
Explore Tags
View
within
Temirgoe · Rossk. · Evdokimovsk · Kergez Mt. · Bazardyuzyu Mts.
Minimap
Edit View Zoom
Search for location
Data CC-By-SA by OpenStreetMap
Show Properties
rdf:type · label · comment · location · species name · published in · within
Filter
Search in list
Add complex Filter Clear
« First ‹ Previous 1 2 3 Next › Last »
Show me: 10 50 100 ... all
Properties of Philipp Frischmuth
Resource
View Resource
Properties History Community Source
Tagging
AKSW OntoWiki
Tag removed
Similar Instances
Person
Norman Heino · me · Sebastian Tramp · AxelCNgongaNgomo · ChristophRiess · ..
Instances Linking Here
knows-1
Norman Heino · Sebastian Tramp
maker-1
OntoWiki and Erfurt Showcase · Erfurt API · OntoWiki · Semantic Pingback · Triplify
member-1
Agile Knowledge Engineering and Semantic Web (AKSW)

		published in	location
1.	33167: Harpactea golovatchi Original Record	Dunin (1992)	Shvanidzor
2.	33169: Harpactea eskovi Original Record	Dunin (1992)	Shnokh
3.	33168: Harpactea eskovi Original Record	Dunin (1992)	Ozun
4.	33172: Harpactea eskovi Original Record	Dunin (1992)	Savarlo
5.	33173: Harpactea nenilini Original Record	Dunin (1992)	Kuris
6.	33170: Harpactea eskovi Original Record	Dunin (1992)	Vanadzor
7.	33175: Harpactea deelemanae Original Record	Dunin (1992)	Kajaran
8.	33174: Harpactea nenilini Original Record	Dunin (1992)	Shishkert
9.	40155: Macaroeris nidicolens Original Record	Logunov & Rakov (1998)	Mukhtadir
10.	33178: Harpactea azerbajdzhanica Original Record	Dunin (1992)	Aurora

rdf:type	Original Record
label	33168: Harpactea eskovi
comment	type series 92
location	Ozun
species name	Harpactea eskovi
published in	Dunin (1992)

Search returned 5179 results.

Query execution took 68 ms.

http://philipp.frischmuth24.de/id/me

Data was found for the given URI. 28 statements were added.

ns1:to	http://philipp.frischmuth24.de/pingback/ping/
type	Person
seeAlso	http://feeds.delicious.com/rss/pfrischmuth
sameAs	5562 id490bd762-0352-40e8-8559-3726c31bbd8a
birthday	1984-10-09
current project	http://pingback.aksw.org/
depiction	
firstName	Philipp
gender	male
holds account	pfrischmuth pfrischmuth
homepage	http://www.frischmuth24.de
knows	me Sebastian Tramp
personal mailbox	mailto:frischmuth@informatik.uni-leipzig.de mailto:philipp@frischmuth24.de
name	Philipp Frischmuth
nickname	Phil
Surname	Frischmuth

Figure 1: Generic OntoWiki views – resource list on the left (with two selected properties, an applied filter, map display and an attribute cloud) vs. resource view on the right (with similar and linking instances as well as tagging module).

to the different sources (please refer to [4] for more information).

Knowledge Base Evolution

OntoWiki tries to support the creation of RDF **Knowledge Bases from the scratch** (rather than using predefined ontologies only) in a wiki way. This means that our users often follow the learning by doing path, where requirements and priorities change over time. Our **evolution framework** allow OntoWiki users to apply predefined **evolution pattern** from a **pattern repository** to their data, as well as to create their own pattern in order **maintain the data quality** of their Knowledge Base with little effort. Examples for such evolution patterns include the **splitting of classes by using their property values** or the **transition from a data property to an object property** (e.g. the transition from literal based tags to resource based tags).

Linked Data & Semantic Pingback

OntoWiki acts as a **Linked Data server** as well as an **Linked Data client**. Entities that use identifiers being governed by the OntoWiki installation are automatically served as Linked Data. OntoWiki supports **access control** on Linked Data employing the **FOAF+SSL protocol** as well as traditional local account based authentication. References to entities using foreign identifiers can be dereferenced (i.e. consumed by OntoWiki) in order to **obtain additional information** from the original source. Even non-linked data resources can be consumed by utilising **specific data-wrapper**, that allow the **generation of RDF data** (e.g. from images, videos or web services).

Often it is beneficial for the owner of a Linked Data resource to be informed about the usage of their resources. When **publishing a FOAF profile**, for example, it is useful to get notified when others establish a `foaf:knows` relation. We therefore developed the **Semantic Pingback** protocol. By employing Semantic Pingback, OntoWiki users can communicate RDF statements across domains and applications.

2. REFERENCES

[1] Sören Auer, Sebastian Dietzold, and Thomas Riechert. Ontowiki - a tool for social, semantic collaboration. In *The Semantic Web - ISWC 2006, 5th International Semantic Web Conference, ISWC 2006, Athens, GA, USA, November 5-9, 2006, Proceedings*, volume 4273 of *LNCS*, pages 736–749. Springer, 2006.

[2] Norman Heino, Sebastian Dietzold, Michael Martin, and Sören Auer. Developing semantic web applications with the ontowiki framework. In *Networked Knowledge - Networked Media*, volume 221 of *Studies in Computational Intelligence*, pages 61–77. Springer, 2009.

[3] Sebastian Tramp, Philipp Frischmuth, Timofey Ermilov, and Sören Auer. Weaving a social data web with semantic pingback. In P. Cimiano and H.S. Pinto, editors, *Proceedings of the EKAW 2010 - Knowledge Engineering and Knowledge Management by the Masses; 11th October-15th October 2010 - Lisbon, Portugal*, volume 6317 of *Lecture Notes in Artificial Intelligence (LNAI)*, pages 135–149, Berlin / Heidelberg, October 2010. Springer.

[4] Sebastian Tramp, Norman Heino, Sören Auer, and Philipp Frischmuth. Making the semantic data web easily writeable with rdfauthor. In *Proceedings of 7th Extended Semantic Web Conference (ESWC 2010), 30 May – 3 June 2010, Heraklion, Greece*, volume 6089 of *LNCS*, pages 436—440. Springer, 2010.

[5] Sebastian Tramp, Norman Heino, Sören Auer, and Philipp Frischmuth. Rdfauthor: Employing rdfa for collaborative knowledge engineering. In P. Cimiano and H.S. Pinto, editors, *Proceedings of the EKAW 2010 - Knowledge Engineering and Knowledge Management by the Masses; 11th October-15th October 2010 - Lisbon, Portugal*, volume 6317 of *Lecture Notes in Artificial Intelligence (LNAI)*, pages 90–104, Berlin / Heidelberg, October 2010. Springer.