# Towards a Semantic Wiki Experience –
# Desktop Integration and Interactivity in WikSAR

David Aumueller, Sören Auer

Department of Computer Science
University of Leipzig, Augustusplatz 10-11, 04103 Leipzig, Germany
`{david|auer}@informatik.uni-leipzig.de`

**Abstract.** Common Wiki systems such as MediaWiki lack semantic annotations. WikSAR (Semantic Authoring and Retrieval within a Wiki), a prototype of a semantic Wiki, offers effortless semantic authoring. Instant gratification of users is achieved by context aware means of navigation, interactive graph visualisation of the emerging ontology, as well as semantic retrieval possibilities. Embedding queries into Wiki pages creates views (as dependant collections) on the information space. Desktop integration includes accessing dates (e.g. reminders) entered in the Wiki via local calendar applications, maintaining bookmarks, and collecting web quotes within the Wiki. Approaches to reference documents on the local file system are sketched out, as well as an enhancement of the Wiki interface to suggest appropriate semantic annotations to the user.

## 1 Introduction

The ease of content authoring and publishing on the World Wide Web culminates in Wiki environments. The huge free encyclopaedia Wikipedia for example was created by an ever-growing community of contributing end-users. To manage the vast amount of content therein categories and lists of related concepts are manually maintained by certain users. Extending the Wiki syntax to allow for simple semantic annotations would cut down the effort of keeping such collections up-to-date, as also proposed in [5]. The WikSAR [1] prototype demonstrates a straightforward approach to enter semantics in a Wiki and presents a variety of features building on these annotations. The additional expenses in semantic authoring are enticed by instantaneously gratifying the users in various ways. Especially, the new interactive graph visualisation and navigation possibilities make the creation of meaningful connections between concepts attractive.

The next section explains the features of the semantic Wiki prototype – characteristics that common Wiki systems do not offer – i.e. how to enter semantics and take advantage of them. Section 3 explains the possibilities currently available in WikSAR to integrate the user's desktop and outlines potentials for further future approaches. Section 4 presents the interactive graphical navigation system novel to WikSAR.
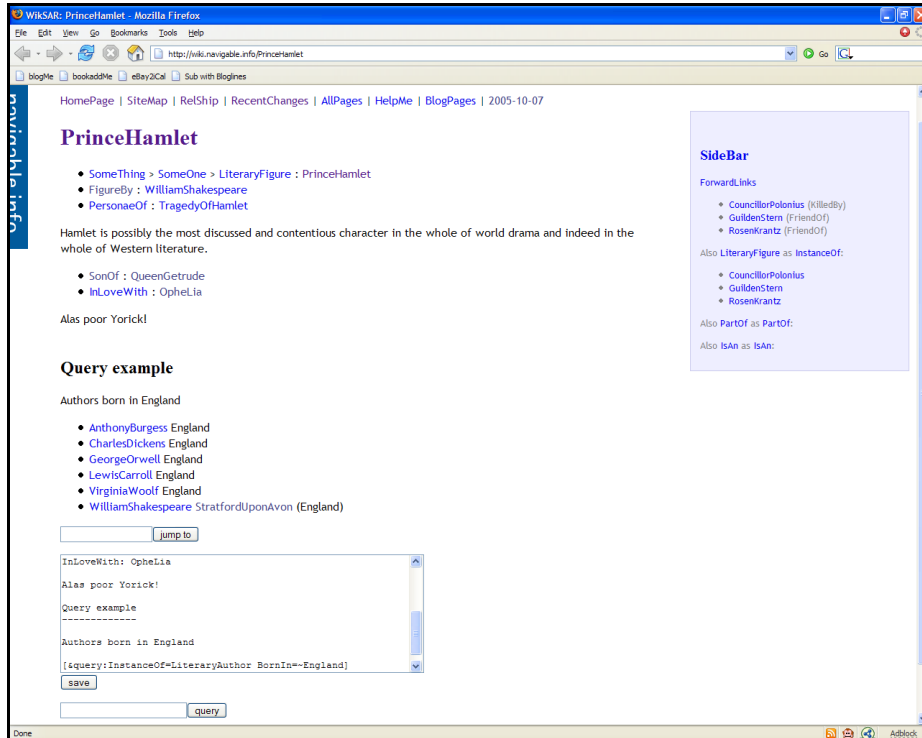
**Figure 1.** Interface of the WikSAR prototype.

## 2 Semantic Wiki features

The WikSAR interface consists of a Wiki page and the optional interactive graph visualisation that can be switched on and off within a separate frame. A Wiki page in WikSAR is divided into three parts (Figure 1): The form for editing the text, the rendered Wiki text above, and the sidebar on the right containing context-dependent links, constructed merely from semantic information present on other Wiki pages.

### 2.1 Semantic authoring

The WikSAR prototype uses the WikiWord or CamelCase syntax similar to Ward Cunningham's original Wiki Wiki Web [6]. However, WikiWords are not only used to create hyperlinks to other Wiki pages but are interpreted either as subject, predicate, or object in Semantic Web statements, i.e. RDF triples. The page name of a Wiki page here always denotes the subject of statements embedded in the Wiki text. Predicate and object are simply entered on an empty line within the Wiki text with WikiWords separated by a colon and space. Thus, on a page named "PrinceHamlet" the line "FigureBy: WilliamShakespeare" represents the statement *<PrinceHamlet>*

*<FigureBy> <WilliamShakespeare>*. By combining all such embedded statements a formal ontology emerges within the Wiki. An important aspect in ontology creation regarding the Semantic Web is to adhere to given schemas and to use already established vocabularies. An editing assistance in that direction can be integrated in the Wiki by suggesting already used WikiWords or vocabulary from external reference ontologies. The recently emerged notion of mixed client and server-side Web applications (e.g. AJAX[1] Asynchronous Javascript and XML) eases the creation of intuitively usable interfaces, such as Google Suggest[2] and del.icio.us[3] tag suggestions – these techniques can support the user in editing semantic statements in the Wiki.

## 2.2 Semantic navigation in context

The entered statements are used immediately to create links to related pages depending on the current context. Firstly, breadcrumbs inform the user about her position in the Wiki showing the path back to the root of the site or concept. Secondly, the sidebar shows pages or concepts related to the current concept including their type of relationship. These special *backlinks*, i.e. typed backlinks, include links to pages that contain e.g. more detailed information about mentioned concepts on the current page, often leading deeper into the site or taxonomy. E.g. on the page WilliamShakespeare there would be a typed backlink that leads to "PrinceHamlet" which is a "FigureBy" the current concept. The breadcrumbs on the other hand show the way back, up the hierarchy, saying e.g. that WilliamShakespeare is an author, which again is a person. Thus, two important questions regarding the usability of a web site get answered: "Where am I?" and "Where can I go?". In addition to these navigational means on every page, the semantic annotations are used to automatically generate a class hierarchy or a complete map of the ontology as labeled graph, i.e. a typed site map.

## 2.3 Semantic retrieval

The triples created in WikSAR are available for semantic queries. Using a triple store that supports RDF query languages, such as RDQL and SPARQL, the Wiki space can be queried to return distinct concepts (pages). The current query syntax in WikSAR allows filtering by specific predicate-object combinations, as in "predicate operator object", allowing equality, quantitative comparisons, and regular expressions as operator. Concatenating expressions by spaces implies 'and'-semantics. The query "InstanceOf=LiteraryAuthor BornIn=~England DateOfBirth between 1800 and 1900" returns a list of authors born in the 19th century in England, as demonstrated in [1].

## 2.4 Semantic views and query chaining

WikSAR accepts a variety of proprietary commands embeddable in Wiki pages to generate and include content gathered from all available data. For example the links

---

[1] http://www.adaptivepath.com/publications/essays/archives/000385.php
[2] http://www.google.com/webhp?complete=1
[3] http://del.icio.us

in the sidebar get created by putting special operators in the template Wiki page for the SideBar. Likewise, queries in WikSAR need not produce only search results available temporarily to the querying user; queries can be embedded persistently within Wiki pages, too. Whenever a page containing a query statement gets loaded the specified query is executed and the result is embedded seamlessly within the Wiki page, thus creating a special view on the Wiki information space. That way it is possible to maintain pages containing always up-to-date query results, e.g. collections of related concepts such as EnglishAuthors or StrongBeer. Newly entered content anywhere on the Wiki will immediately be reflected in dependant collections or concepts as well – without any manual effort. We plan to use such collections further to create more complex queries by allowing to chain queries, i.e. by using the result set of one query as input for another one. Allowing any list of concepts as input (even manually created lists, e.g. lists only of personal interest) would facilitate retrieval possibilities for very specific interest. In a Wiki collecting e.g. scientific publications, a user might have lists of favourite authors and favourite subjects; using these as input to a query she could retrieve publications of her explicit needs and have the list always available as distinct Wiki page. Publishing query results or single Wiki pages as RSS-feed informs the user immediately about changes matching her special interest without the need to visit the Wiki.

## 3　Desktop Integration

Due to their flexibility Wikis are often used for Personal Information Management (PIM), as already suggested by [6]. Important in this context is to bridge the gap between the Wiki accessible only by a web browser and the local desktop environment, i.e. local applications and the file system. Both worlds have their advantages for PIM; the Wiki and the information therein residing on a web server are omnipresent, accessible from any machine with Web access. Conversely, reminders of appointments can be better triggered locally by calendar applications, and collections of user-centric files, such as office documents and PDFs, usually reside locally, as well.

WikSAR publishes dates entered in the Wiki as remote calendar entries in the iCalendar format. Such calendar files accessible by the HTTP protocol can be imported or subscribed to by desktop calendar applications (e.g. Mozilla Sunbird), which can then for example trigger reminders.

Interesting PIM applications are the maintenance of lists of web bookmarks, as well as of quotes from web pages. WikSAR uses so-called bookmarklets (some JavaScript statements that are called from within a web browser by a single user click to provide access to information on the currently open web page) to put selected pieces of text and/or the URI of the resource onto the Wiki. Such collections may be of interest to single person or a whole community. Using a Wiki to maintain bookmarks and web quotes makes them accessible not only from the web browser where they got collected but from anywhere. Further, they can be easily annotated, too.

Accessing the local file system from within a remote web application is impossible due to security reasons. We plan to be able to reference local files from within the Wiki nevertheless by either providing a link to one's desktop search engine as these

search engines (e.g. Google Desktop Search) are more and more accessible via HTTP calls on *localhost*. Another approach may utilize the magnet URI scheme, an open scheme "enabling seamless integration between websites and locally-running utilities"[4]. This is already successfully applied in the P2P-domain by providing unique keys for files that file-sharing applications can then search for. To reference emails of a PIM user, cross platform component models to email clients may provide the missing link, such as XPCOM[5] for Mozilla Thunderbird.

## 4 Interactive Graph Visualisation and Navigation

Visualisation of complex information spaces is an important area of research in itself to help understand data, e.g. [2]. In the Wiki context the link structure, i.e. the interconnections between Wiki pages, may be drawn as a graph, as e.g. done by TG Wiki-Browser[6]. Labelled graphs as available in WikSAR via the entered Semantic Web statements offer more sophisticated visualisation and filter possibilities as merely a graph of un-typed hyperlinked pages. Visualising ontologies is possible for example in Protégé[7]. WikSAR offers a novel approach by integrating an interactive graphical representation within the Wiki; navigating through the Wiki space is possible in either the Wiki or the graph, changing focus simultaneously in both views.

The graph of WikSAR pages is built upon the entered semantic triples. These are interpreted as directed labelled graph with subjects and objects being nodes, and predicates becoming labelled edges. Since an object of one triple may be the subject of another, these edges are chained to spawn a directed graph. To overcome the problems of visualising large graphs, the popular fisheye views and hyperbolic trees show the part of the graph of interest more prominently than the rest of the graph.

For the Wiki we chose a special tree layout algorithm that always renders the concept in focus as root of a tree and grows the tree via breadth-first-search along the graph, as provided by the prefuse toolkit [4] as radial tree layout[8]. Here, the node in focus, i.e. the root, resides in the centre of the visualisation, with the children attached. Deeper descendants get displayed further away from the centre such that every level of the hierarchy resides on its own circle around the root. For the Wiki context this tree layout got adapted to display less information the deeper the concept resides in the current tree, i.e. the concept in focus and its immediate children are displayed with labels both for concept nodes and edge type, whereas deeper nodes merely are cherished with a circle instead of label and without labelling the edge type. By default, concepts being further away than four hops from the current one in focus do not show up at all. The visualisation will be enhanced by allowing the user to control the displayed tree depth and filter or highlight specific types of edges.

---

[4] http://magnet-uri.sourceforge.net

[5] http://xulplanet.com/references/xpcomref/group_Mail.html

[6] http://www.touchgraph.com

[7] http://protege.stanford.edu

[8] Due to patent restrictions the hyperbolic tree layout algorithm is not available in prefuse.

We believe that the interactive graph helps to understand the relationships between concepts in WikSAR by providing a playful approach to examine the information space, e.g. the relationships among the characters in a Shakespearian play (Figure 2).
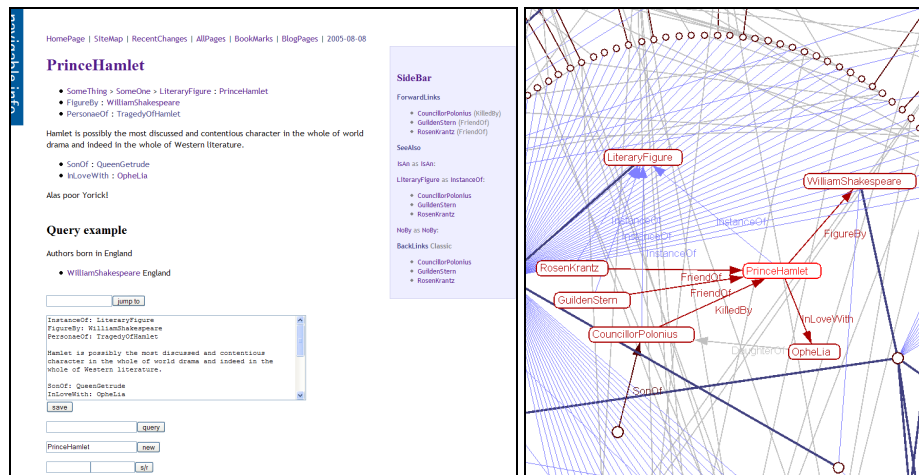


**Figure 2.** Interface of the WikSAR prototype with interactive graph.

## 5    Conclusion

With WikSAR, the presented Wiki prototype, it is straightforward to author semantically annotated content and to enjoy the benefits thereof, a challenge identified by [3]. A simple query interface offers retrieving information not only by keyword but also with respect to specific semantics represented in the Wiki. Persistently embedding queries into Wiki pages allows to create automatically updated collections of concepts and monitoring of specific changes of content collected on the site. Browsing the Wiki ontology as interactive graph representation further enhances navigation and helps to understand complex structures by giving a graspable overview. Future enhancements to bridge the gap between the Wiki application on the Web and local applications will improve personal information management.

## References

[1] Aumüller, D., Semantic authoring and retrieval within a Wiki. In Demo Session at the ESWC 2005, Heraklion, Greece (2005). Available at <http://wiksar.sf.net>.
[2] Geroimenko, V., Chen, C. (Eds.) Visualizing the Semantic Web. Springer (2003)
[3] Halevy, A. Y., *et.al.*: Crossing the Structure Chasm. Conf. on Innovative Data Systems Research, California (2003)
[4] Heer, J., Card, S. K., Landay, J. A.: prefuse: a toolkit for interactive information visualization. In CHI 2005, Human Factors in Computing Systems (2005)
[5] Krötzsch, M., Vrandečić, D., Völkel, M.: Wikipedia and the Semantic Web – The Missing Links. In Wikimania 2005, 1st Int. Wikimedia Conf., Frankfurt, Germany (2005)
[6] Leuf, B., Cunningham, W.: The Wiki way: quick collaboration on the web. AW (2001)