Semantic Wiki in Environmental Project Management

Jakub Talaš, Tomáš Gregar, Tomáš Pitner

Semantic Wiki in Environmental Project Management

Jakub Talaš, Tomáš Gregar, and Tomáš Pitner

Masaryk University, Faculty of Informatics,
Research Lab Software Architectures and Information Systems
Botanická 68a, 60200 Brno, Czech Republic

173016@mail.muni.cz, xgregar@fi.muni.cz, tomp@fi.muni.cz
http://lasaris.fi.muni.cz

Abstract. Advanced management of environmental data requires appropriate metadata and tooling. Moreover, contemporary environmental data is heterogenous, encompassing also data about people, environmentaly-related projects, and documents. Also geographical- and time-related information are vital for any environmental data and should be supported. To satisfy these requirements, "semantic wikis" represent suitable tools but usually requiring knowledge management skills. In this paper, we present a semantic extension to project-management tool Trac, enabling advanced but user-friendly querying in semantic data stored in the system and its visualization. As it builds upon the general concept of RDF-based semantic data, it can be freely extended into other contexts and deployed in various environmental management applications. We also discuss automatic formal visualization of semantic data. Such process significantly lowers the requirements on users in preparing the visualization in domains like environmental studies – and also helps to build *Linked Data*-aware and better human-readable interfaces of semantic data repositories.

Keywords: environmental project management, semantic wiki, RDF

1 Semantic wiki in real project management

Since many (if not a majority) of modern systems for computer-aided environmental project management require not only the baseline, generic functionality and collaborative content creation but also its sharing and publication. The idea of Semantic Web builds upon the assumption that the data should be accessible, reusable and even re-purposable. So they should be accompanied by rich metadata sets based on standardized (or commonly-agreed) ontologies wherever applicable. However, their key users are usually not experts in knowledge management, they are not capable of operating advanced semantic tools like ontology editors. They like to work with really simple tools, editing a wiki is the most "technical" activity that can be expected in this terms. On the other hand, we might need at least some support of complex functionality like semantic querying. It enables to automatically present selected data about projects, people,

their participation and progress on wiki pages. The wiki is no more a tool for fast collaborative creation of static textual content by end-users and starts to be a powerful instrument to present anything stored in a *semantic repository*.

In an environment targeted to *software projects*, these requirements are even amplified, cf.[1]. The members of a project team must share a common environment, see all relevant analytical documents, and contribute to discussions with the clients. All these activities intensively deal with content. For instant content creation, wiki systems became a de-facto "golden standard" as a mean to share, present and exchange information, and are present in nearly all project hosting and management platforms such as Sourceforge.net, Google Code, Codeplex.com, or github.org, among others. They are less demanding than online office-like suites like *Google Docs* though they still allow user to create and edit web content without a need to know languages like HTML/CSS or to have physical access to the web server.

There exist many implementations of such interface, based on different languages and storage databases[1] — like *MediaWiki* (PHP and MySQL), *Dokuwiki* (PHP), *Twiki* (Perl) etc. An integrated Wiki is also an inevitable part of the (installed) open-source project management system *Trac*[2] being used by many open-source projects as referred later.

2   Semantic support in existing wiki systems

However, the vast majority of systems run without support for semantics. Only few wikis are built with semantic capabilities [2]. Most of current semantic wiki engines are developed in academic environment. Such conditions bring some advantages (they are almost all open source and freely distributed), as well as some drawbacks (lots of them is currently abandoned, not developed at all; sometimes, they even do not ever reach a usable milestone)[3]. Existing semantic wikis can stand as the functionality model. Actively developed semantic wikis are in alphabetical order:

ACE wiki works with an interesting concept of syntax definition in natural language (definitions like "every country is an area"). It is easy and well-arranged wiki with wide capabilities in building the semantics. The statements are checked for potential semantic collisions like *"Country has one capital"* vs. *"The capital of Germany is Berlin"* and *"The capital of Germany is Bonn"*.

KiWi stands for *Knowledge In Wiki*. This system is intended for sharing and managing the knowledge.

KnowWE this system has a technique for annotating content of wiki pages similar to techniques used in our developed *Trac Semantic Extension*. It also recommends pages with semantically related topics or topics with similar

---

[1] `http://www.wiki.org/wiki.cgi?WhatIsWiki`
[2] `http://trac.edgewall.org/`
[3] `http://semanticweb.org/wiki/Tools`

attributes. It features an easy-to-use semantic search functionality ("Select sports where Flexibility = spontanious").

OntoWiki very advanced and user friendly wiki-engine.

Semantic MediaWiki is an extension of the popular *MediaWiki*. It adds possibility to add properties to pages and semantic searching.

TaOPis easy wiki system for project and contributors' listing.

Wikidsmart communicates with project management system *Jira*. Wiki authoring can create bug/feature/feature document in Jira (or other connected management system).

3 Projects presentation in a broader context

Typically, a particular environmental project is a part of a broader activity, being led by a public body (agency, office), an academic institution, or a commercial subject. So, relationships to other activities should be taken into account. In order to satisfy this aspect, an appropriate project management tool has to be found and deployed. Another important factor determining the selection of a right project management system are its semantic capabilities related to project management tasks [3]. The system selected to satisfy these needs, *Trac*, includes an integrated wiki for project presentation, allows synchronization with a Subversion repository, contains modules for bug tracking and issue reporting, a module for browsing the source code, and the like. It can be enhanced with the support of multiple projects and localization which is still vital, particularly in the highly international context.

Our experimental installation implemented in the research lab of the authors is called *Deep Thought* (DT), cf. [4]. A fundamental advantage of Trac/Deep Thought is a wide support for extensions. Trac is based on the *Python component architecture* — a basic module, and also each plug-in are components distributed as `.egg` packages. There exist two types of extensions — *plugins* and *macros*. Plugins can, thanks to their low-level interface to other Trac modules, extend whole system functionality. Macros are direct components of the Trac wiki-subsystem. They allow to dynamize wiki pages.

4 Semantic extension in the wiki

4.1 Key requirements

The key ideas how to combine the wiki ease of use with semantic information are:

1. Connect the wiki with a semantic repository containing info on categories like projects, people, or documents.
2. Provide instruments to easily annotate/link the wiki content with references to pieces of semantic information from the repository.
3. Offer enhancements to insert even complex semantic queries into wiki pages in order to dynamize the content.

Fig. 1. Popups with semantic information from the wiki

4. Facilitate the querying with visual appliances like context-aware drop-down menus and stored queries for repeating tasks.

   Our recent project, *Semex* (Semantic extension) experimentally implements the above ideas, see also [5]. Semex allows annotations of wiki content according to ontological definitions in the knowledge database *Sesame 2*. It also enables visual highlighting of the tagged parts of text and viewing their context. This added functionality brings the possibility of browsing pages via semantic relations; e.g. *person* — his/her *projects*; or *"show all people contributing to this project, who are older than 18 years"*. It can integrate the content with other services (e-mail), mine concept definitions from the repository or other web-service, or list other instances of the same concept [6]. It is composed of a Trac plugin assuring the administrative functions and communication with the knowledge repository and a set of wiki macros. Macros visualize semantic data instantly on the wiki pages and interact with the users; they are capable to wrap the authoring of semantic statements directly in the user interface, cf. Fig. 1.

4.2   Technology

As the Trac environment is Python-based, the extensions must also satisfy its non-functional requirements: at least to be written in Python. There exists li-

brary *RDFLib*[4] for work with RDF triples (i.e. inner representation of sources, statements, relations, ontologies) in Python. Python interface with Sesame 2 (via its REST protocol interface) could be offered by *RDF Alchemy*[5]; Sesame is just one of triple-stores the library can communicate with.

The important part of the Semex development was also the selection – to prevent reinventing the wheel – of prepared ontologies (see [7]) to work with. DT, as multi-project management system cope with information about persons (`foaf`, `http://xmlns.com/foaf/0.1/`, see [8]), projects (`doap`, `http://usefulinc.com/ns/doap#`, see [9]), documents (`dc`, `http://purl.org/dc/elements/1.1/`), wiki pages (`wiki`, `http://sw.deri.org/2005/04/wikipedia/wikiont.owl#`), in project management also with temporal events (`event`, `http://purl.org/NET/c4dm/event.owl#`), or places (`geo`, `http://www.w3.org/2003/01/geo/wgs84_pos#`). This ontological structure allows describing semantics of the wiki documents — define developers, deadlines, meetings, used literature etc. A very important feature of our approach is *generality* and *universality*. The ontologies used in the system can be extended, replaced, or combined with others which enables seamless employment of environmental ontologies like [10] as well together with the more general ones.

4.3 Functionality

The solution covers all basic use-cases required for a computer-aided project management and presentation using wiki. The administrative section allows creating, editing and deleting persons, projects, events, places and documents via a form-based web interface. The module also automatically creates wiki pages for the respective objects — names of these wiki pages start with prefix `Semex:`, therefore they are easy recognizable and can be quickly deleted in the administrative section. In the prototype, access to the sections is authenticated and authorized via *Kerberos* and the users can use their Faculty accounts credentials and do not have to bother with maintaining separate access data, namely passwords.

Wiki macros are the "visible" part of the module, cf. Fig. 2. A common visitor of Deep Thought sees only them. General Trac macros could be written in wiki page as short tag enclosed in doubled square brackets (with optional parameters in parentheses).

The following "bootstrap" macros have been developed for Semex so far:

Info it shows popup menu with options and information about selected object. It also allows user to see all statements of this object.
Query returns a list of RDF sources according to entered SPARQL query. Suitable for automatic lists.
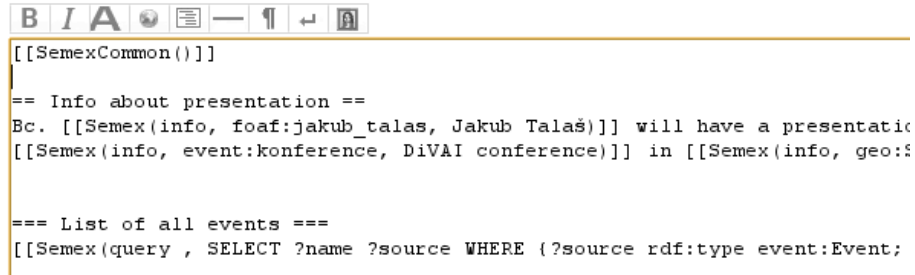**Calendar** prints out simple calendar with upcoming events.
Allabout prints out all information about a source and similar ones.

---

[4] `http://www.rdflib.net/`
[5] `http://www.openvest.com/trac/wiki/RDFAlchemy`

SemexCommon this macro is the prerequisition for other semantic macros (includes semantic methods, defines GUI forms used in macros), and hence has to be included in every wiki page.

Editing info

```
[[SemexCommon()]]

== Info about presentation ==
Bc. [[Semex(info, foaf:jakub_talas, Jakub Talaš)]] will have a presentatio
[[Semex(info, event:konference, DiVAI conference)]] in [[Semex(info, geo:S


=== List of all events ===
[[Semex(query , SELECT ?name ?source WHERE {?source rdf:type event:Event;
```

Fig. 2. Writing complex queries in wiki

The GUI of macros is created via the *jQuery 1.3* library. It creates popup menus, and also user-centered simple wizards helping common user to incorporate macros which makes them accessible even for users with only a minimal notion of the formal semantic. However, the semantics in the wiki can be explored to its full power using queries in fully-fledged languages like *SPARQL*, as depicted on the last figure.

5   System integration

One of the usage patterns of the semantically enhanced project management system Deep Though is the university teaching. Choosing the right pattern of integration into the whole study process and administration is crucial, it was necessary to integrate the extension module and the whole system with the *University's study administration information system* (IS) in order to ensure smooth application in software development oriented courses. Students in these courses have to *build a team*, sign up for some *project assignment* and then *work on it*. This integration allows to get needed data from the IS, automatically create an environment for student projects defined in IS containing basic wiki pages, bug-tracker module, or source-code browser. Metadata about project is also imported into the semantic repository and utilized via the semantic module of Deep Thought.
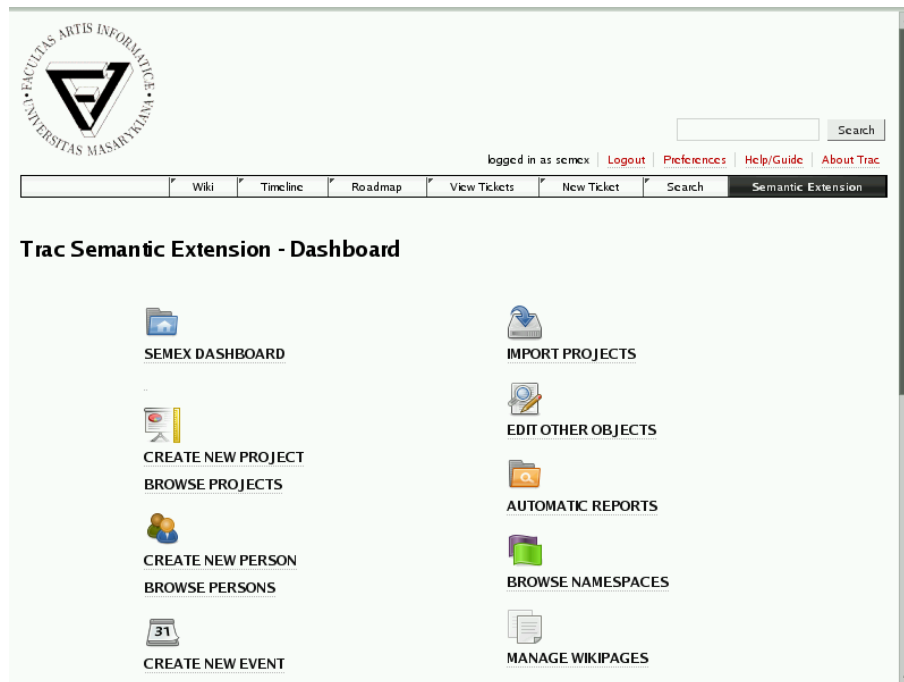
The most visible features and contributions of this semantically-enhanced project management system in comparison with tradition PM systems is a common knowledge base consisting of concepts (entities) maintained centrally and reusable in all projects wherever it is relevant, see Fig. 3. A clear advantage of this approach is in its flexibility — once the DOAP ontology schema is replaced

by a newer standard for project description, the transition will be significantly easier that in a classical database system where the relational or object/class schema must be modified — with all the undesired consequencies it might have.

6   Conclusion and Further work

Let us summarize the achievements reached so far. The key requirements showed to be realistic and implementable. The presented semantic extension to a project management tool Trac has proven as a working combination of

- Advanced, popular project management system for hosting public-, community-, or internal projects including commercial ones;
- Integrated semantic repository containing semantically precise and well-treatable data about environmental projects, people, places, events, and documents that represent a substantial part of all data typically processed;
- Tools for advanced semantic querying and dynamic wiki content;
- Visualization facility to user-centered query formulation and content browsing.

Fig. 3. Entities common for all projects in Deep Though

So, all the main ideas have resulted in a succesfully implemented working prototype. Further development will include both general and domain-specific directions that should be tested on a larger-scale projects and data sets and also to compare/combine the approach with tagging tools developed within the *TaToo Project* [11].

References

1. Kadenbach, D., Kleiner, C.: Benefits and Challenges of Using Collaborative Development Environments with Social Software in Higher Computer Science Education, In Online Communities and Social Computing Lecture Notes in Computer Science, Vol. 5621, pp. 479–487, Berlin/Heidelberg: Springer Verlag. (2009)
2. Völkel, M., Krötzsch, M., Vrandecic, D., Haller, H., and Studer, R.: Semantic Wikipedia,In WWW '06: Proceedings of the 15th international conference on World Wide Web, New York, NY, USA: ACM, pp. 585–594 (2006)
3. Landaeta R. E., Pinto, C.A., Kotnour, T.: Assessing faulty knowledge management systems in project-based organisations, International Journal of Knowledge and Learning, Vol. 5, No.2 pp. 122–143 (2009)
4. Gregar, T., Pospíšilová, R., Pitner, T.: Deep Thought : Web based System for Managing and Presentation of Research and Student Projects. In CSEDU 2009 — Proceedings of the first International Conference on Computer Supported Education. Lisboa, Portugal : INSTICC — Institute for Systems and Technologies of Information, Control and Comunication, p. 5. (2009)
5. Talaš, J. a Gregar, T. a Pitner, T.: *Semantically Enriched Tools for the Knowledge Society: Case of Project Management and Presentation.* In *Knowledge Management, Information Systems, E-Learning, and Sustainability Research*, Third World Summit on the Knowledge Society, Springer (2010)
6. Geurts, J. et al.: Towards Ontology-driven Discourse: From Semantic Graphs to Multimedia Presentations. In 2nd International Semantic Web Conference, 1. Sundial Resort, Sanibel Island, Florida, USA, p. 16 (2003)
7. Gómez-Pérez, A. a Corcho, O.: Ontology Languages for the Semantic Web, IEEE Inteligent Systems (17), p. 54-60 (2002)
8. Brickley, D., Miller, L.: FOAF vocabulary specification. available online `http://xmlns.com/foaf/spec/`. (2007)
9. Dumbill, E.: Description of a Project (DOAP). available online `http://trac.usefulinc.com/doap`. (2004)
10. Morrison, N.: The Environment Ontology – Linking Environmental Data. Proceedings of the *Towards the eEnvironment*, Prague, Czech Republic. (2009)
11. Rizzoli, A. E., Schimak G., Donatelli M., Hřebíček J., Avellino G., Mon J. L., et al. TaToo: tagging environmental resources on the web by semantic annotations. Proceedings of the *International Congress on Environmental Modelling and Software Modelling for Environment's Sake.* (2010)