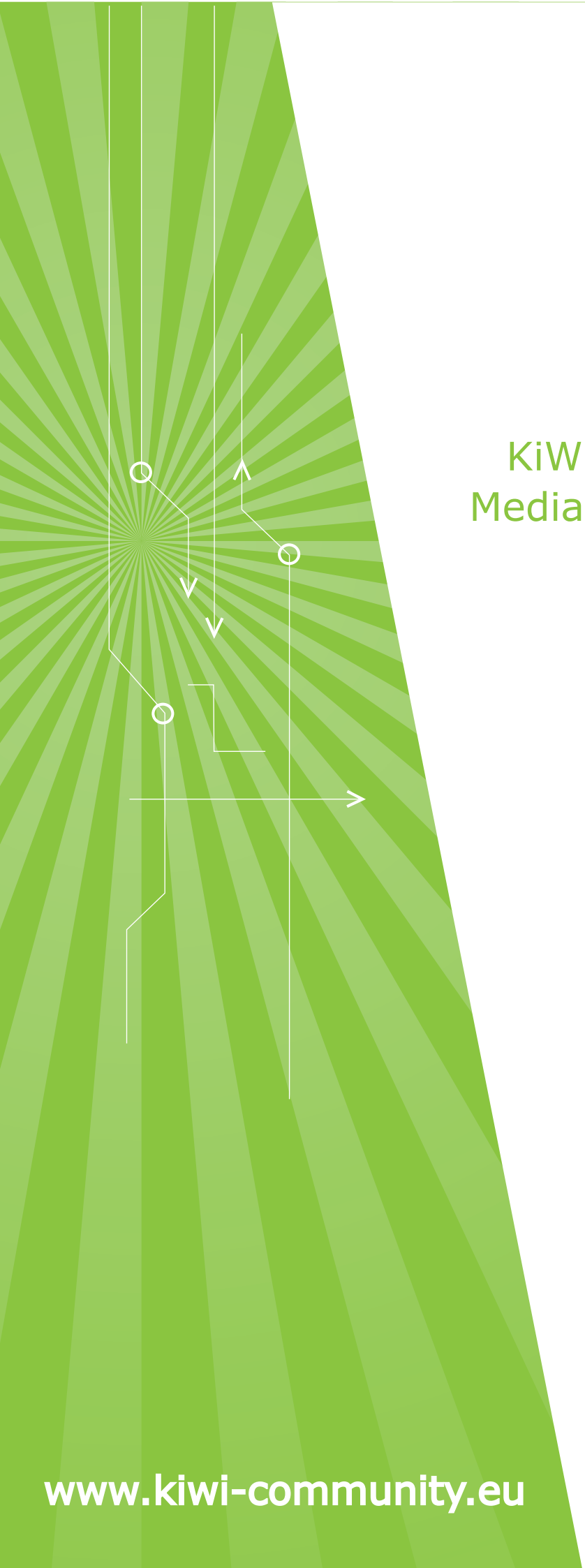
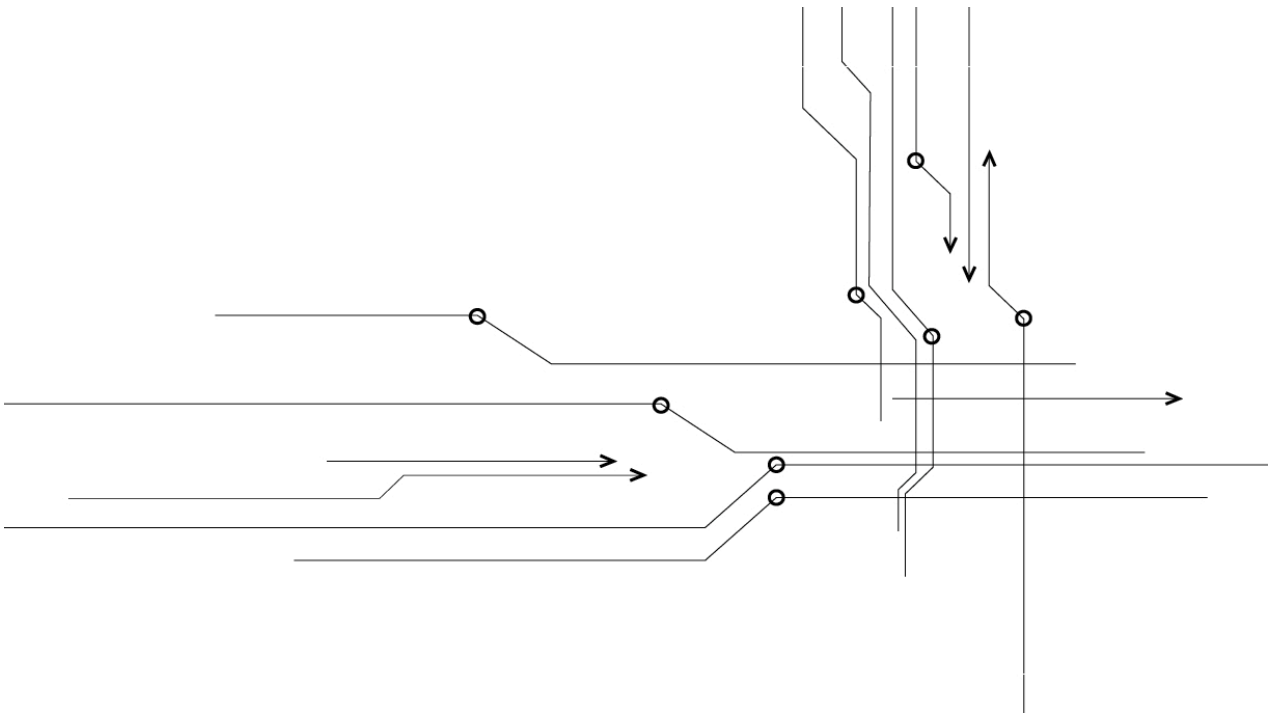


Web 3.0 Made Simple
KiWi - The Semantic Social Media Development Platform

October 2010

www.kiwi-community.eu

White Paper



“KiWi is more than a ‘semantically spiced Wiki’. It is a semantic social platform that breaks information boundaries by allowing users to connect content in new ways that go beyond the level of today’s user interfaces”

Executive Summary

KiWi is an Open Source development platform for building Semantic Social Media applications.

It offers features required for Social Media applications such as versioning, (semantic) tagging, rich text editing, easy linking, rating and commenting, as well as advanced "smart" services such as recommendations, rule-based reasoning, information extraction, intelligent search and querying, a sophisticated social reputation system, vocabulary management, and rich visualisation.

KiWi can be used both, as a platform for building custom Semantic Social Media applications, and as a Semantic Social Index, integrating content and data from a variety of different sources, e.g. Wikis, blogs and content management systems in an enterprise intranet. Third-party applications can access the KiWi System using simple-to-use web services.

Semantics made Simple

The KiWi platform allows to annotate, manage and exploit semantic metadata.

Content in KiWi can be viewed based on the user's perspective e.g. an event might be a calendar event or represented as a geo-located point. KiWi supports this by associating different "perspectives" with a content item, e.g. a normal Wiki view, a location view using a map component, and a calendar view. Perspectives usually consist of view, editing, and search preview descriptions and may make use of Semantic Forms.

The different content views is a core KiWi feature and is used not only for the presentation of information, but also for interaction with the system.

Developers and advanced users can easily utilise "Semantic Forms" to enter content in a structured way. For example, an editor view for "portrait photos" provides a field with a link to an existing profile or to immediately enter information about the depicted person.

With information extraction services users can efficiently search and browse as well as benefit from tag recommendations. Furthermore, KiWi's rule-based reasoning system makes it possible to derive implicit information from what is given explicitly.

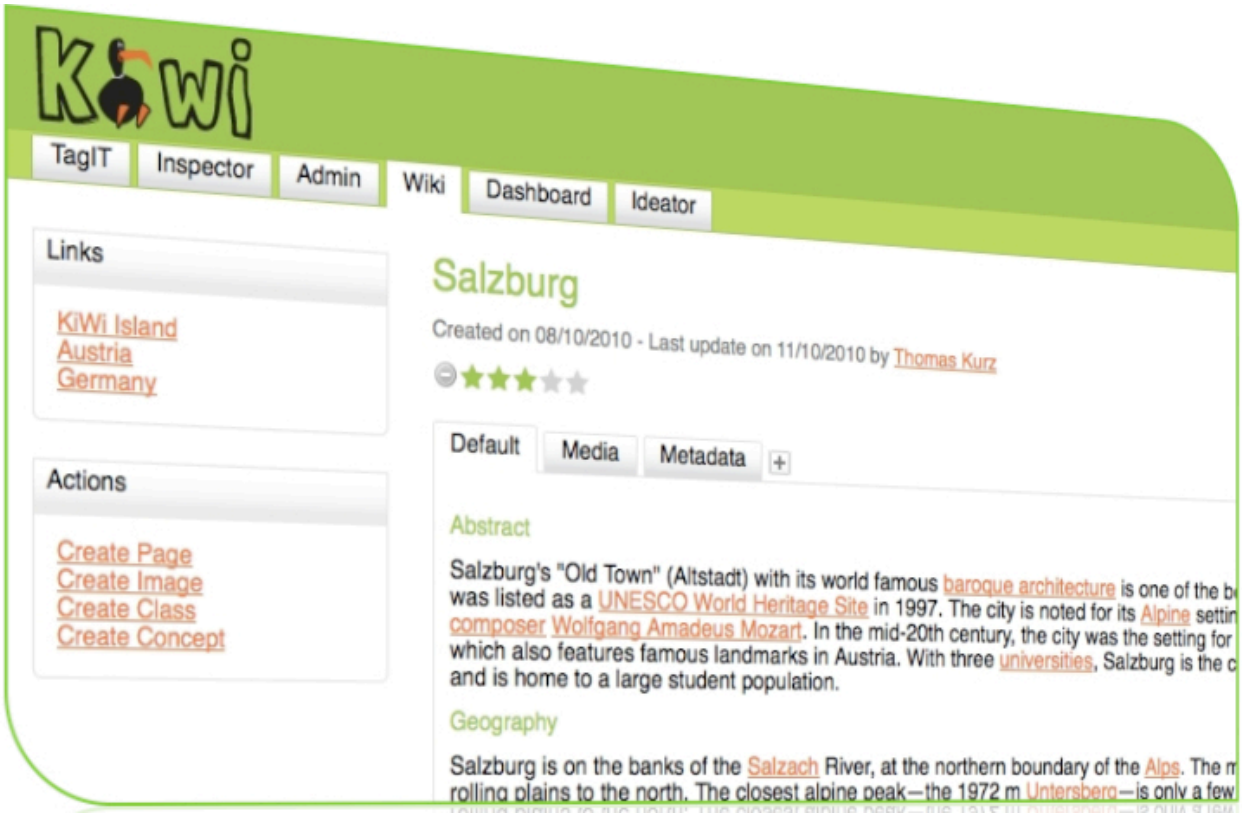
KiWi has received funding from the European Community's Seventh Framework Programme FP7(2007-2013) under the grant agreement No. 211932

Table Of Contents

What is KiWi?

KiWi serves as a platform for implementing and integrating many different kinds of semantic social software services. This new kind of semantically enhanced social software platform allows users to share and integrate knowledge more easily, naturally and tightly, and allows them to adapt content and functionalities according to their personal requirements.

At the heart of KiWi are the 17 core functionalities that enable social software developers to easily build and adapt new services as they are required, e.g. within enterprises or on public social software sites:



1. ***Content and Metadata Indexing*** The KiWi System allows indexing of the unstructured content as well as the associated metadata of textual and multimedia content for the purpose of searching.
2. ***RDF and Ontology Support***. All metadata in KiWi is represented using the standardised RDF data model and format and can be queried and reused in other applications supporting RDF. KiWi comes with a number of core ontologies that are directly supported and used by the system, e.g. Dublin Core, SIOC, FOAF, HGTAGS, SCOT, and Geo.

3. ***Semantic Search***. The KiWi core system offers a powerful semantic search engine that allows searching over content and metadata alike. The search engine offers the traditional, keyword based search as well as "facetted browsing", guiding the user through a refinement of the search results.

4. ***Transactions and Versioning***. All updates (content as well as metadata) to the KiWi Core System are wrapped in transactions and can be undone at any later time.

5. ***Activity Logging***. Every activity performed by a user in relation to content stored in KiWi is logged. The logging data is used on the one hand for calculating community equity values of information, and on the other hand for building user models for personalisation.

6. ***Community Equity***. KiWi implements Sun's Community Equity system that allows to determine the social value of content indexed by the KiWi system. Content that has a high level of social interactivity (e.g. read very often, updated very frequently, commented a lot, rated very high) has a high community equity value. Community Equity is based on a sophisticated mathematical model that takes into account that the "value" of an activity decreases over time depending on the type of content and activity.

7. ***Semantic Tagging***. The KiWi system offers a tagging service for all content indexed by the system. The tagging functionality can be used either directly in KiWi or integrated in the systems where the content originates using web services or widgets. KiWi uses semantic tagging instead of purely keyword-based tagging, i.e. tags can be disambiguated and connected e.g. in a SKOS thesaurus. Tagging metadata is represented using the HGTags and SCOT ontologies.

8. ***Commenting***. Similar to tagging, KiWi offers a commenting and rating functionality which allows to build social feedback loops for all content indexed by the system that can be used either directly or integrated into other applications using widgets or web services. Comment metadata is represented using the SIOC ontology.

9. ***Perspectives***. A content item may represent several different concepts, e.g. an event might be a calendar event as well as a geo-located point. KiWi supports this by associating different "perspectives" with a content item, e.g. a normal Wiki view, a location view using a map component, and a calendar view. Perspectives usually consist of view, editing, and search preview descriptions and may make use of Semantic Forms.

10. ***Content Annotation (RDFa)***. KiWi allows the annotation of unstructured content using the RDFa format. KiWi supports RDFa for displaying content in the browser, and offers a special RDFa editor that allows marking up and annotating of the content in the KiWi system.

11. ***Semantic Forms (JSF+RDFa)***. Advanced users can develop custom tailored forms for editing content and metadata in the KiWi system. These "Semantic Forms" also use the RDFa format, but extend it to be used in forms.

12. ***Vocabulary Management***. For updating SKOS thesauruses, the KiWi system offers a vocabulary management tool that allows

associating concepts with each other and creating new sub-concepts. The vocabulary management tool also recognises "free" tags and can convert them into "semantic" tags by turning them into thesaurus concepts.

13. ***Metadata Editing***. In addition to the "user-friendly" ways of adding metadata (tagging, annotation, information extraction, vocabulary manager), the KiWi system also offers the possibility to directly edit the metadata associated with content in the system so that advanced users can change the metadata as desired.

14. ***Rule-based Reasoning***. In addition to the ordinary RDF support, KiWi contains a rule-based reasoner that can reason over the RDF metadata contained in the index. KiWi already contains a set of simple rules for RDFS reasoning, which can be extended by developers by arbitrary functionalities, e.g. for implementing specific support for a certain schema.

15. ***Information Extraction***. The KiWi core system offers several methods for advanced information extraction from the textual content indexed by the KiWi system. Information extraction is on the one hand used as a base for tag recommendations and for recommendations of related content, and on the other hand for semi-automatic annotation of the content in the editor.

16. ***Recommendations and Personalisation***. In addition to the recommendations based on information extraction, the KiWi system also offers personalised recommendations and personalised search results based on the user model.

17. ***Advanced Structured Querying***. The KiWi system offers advanced structured querying using a variety of formalisms, e.g. SPARQL. A particularly salient feature of KiWi is the query language KWQL that combines the simplicity of search with the power of structured querying.

What makes KiWi different?

The KiWi approach is what we call *"Content Versatility"*. The principle is that every piece of information is a combination of human-readable content and associated metadata, and that the same piece of information can be presented to the user in many different forms: as a Wiki page, as a blog post, as a comment to a blog, as a photo, or even in a bubble in a map-based application. The decision how the information is displayed is taken based on the metadata of the content, and the context of the content and the user (e.g. metadata, user preference, different applications). Metadata is represented using RDF and thus does not require a-priori schema definitions, so the data model of the system can be extended even at runtime.

KiWi makes it possible to tailor the presentation and functionality of the platform to the information needs and experience of the user, and to make the use of the platform as easy as possible. In KiWi language this is called Content Versatility, meaning that the same content can be (re-) used in many different contexts.

The simplest form of Content Versatility in KiWi is implemented as so-called Perspectives. A perspective is a certain viewpoint on some content: the photo can be seen as a photo, as a location, as a generic content item, as a list of the metadata properties associated with it, etc. Most perspectives consist of at least a view and edit definition specifying how the content is displayed and modified when taking the chosen perspective.

KiWi comes with a number of default perspectives:

- ***Default***: the ordinary Wiki perspective on content items. Provides a page view and Wiki-style editor with all the KiWi editor functionalities.
- ***Media***: a perspective for displaying multimedia content stored in the KiWi system. Provides a media view and a media editor that allows you to upload a new version end edit the media description.
- ***Location***: displays the geo-location of the content item on a map and allows to re-locate it using drag and drop on the map or entering the address.
- ***Person***: allows displaying and editing the content item as a person, e.g. first/last name, photo, ...
- ***Class***: displays the content item as a RDF class, showing additional class-related information like sub- and superclasses, instances, etc.
- ***Property***: displays the content item as a RDF property, showing additional property-related information like range, domain, inverse, etc.

- ***Concept***: similar to the class perspective but tailored towards SKOS concepts; allows displaying and editing SKOS properties like alternative labels, narrower/broader, etc.
- ***Tag***: allows displaying end editing the content item as a tag; the tag can be associated with a SKOS thesaurus and the tagged content items can be displayed
- ***Metadata***: allows displaying and editing of the RDF data type and annotation properties associated with the content item
- ***Equity***: calculates the social value of content, people and metadata items based on the level of activity over time.

Of course, this is just a selection. Developers can easily add additional perspectives tailored towards their application domains.

"The metadata foundation for Enterprise 2.0 applications"

“Automated categorisation of content blended with user driven tagging and personalised discovery enables much better knowledge management and expertise discovery for large enterprises”

How can Enterprises benefit from KiWi?

One important objective of the KiWi project is to build functions and services which can be used and integrated into existing software architectures. The KiWi platform and functionality is validated through real live use cases from companies like Logica and Sun Microsystems.

Knowledge Management and Expertise Discovery

Sun (Oracle) was one of the first large companies that implemented an internal Enterprise 2.0 platform (SunSpace) which was used by over 30'000 employees. In 2008 Sun (Oracle) joined the KiWi project to explore how Semantic Technologies could be leveraged to improve the effectiveness and efficiency of the existing E 2.0 knowledge model around Apply, Manage and Exploit Metadata.

Apply Metadata

Business challenge: How to improve the quality and quantity of metadata?

Current State

Tagging represents the only way of relating content, people and communities to each other. Users are allowed to assign any keywords (tags) which they think is appropriate to the individual content items, like documents, people and communities. Tagging policies are defined but not enforced.

Reuse of tags which already exist in the system, is done on the UI level, by the tag-suggesting functionality which offers users with tags based on the simple string similarity match.

How Kiwi can help with - Tagging, Tag recommending and Information Extraction?

The basic tag-recommender functionality provided by the KiWi system will be customized to implement:

- **Support for both free (folksonomy) and controlled (taxonomy) tags and concepts.** One of the main purposes of any tag recommender is to enforce reuse of tags which were already entered into the system by other users. This functionality becomes broader in the context of Sun/Oracle knowledge management platforms with the introduction of controlled concepts grouped in vocabularies/taxonomies/thesauruses. These controlled concepts will have higher priority among free tags in the suggestions delivered to the user. It is expected that this will in general result in a cleaner and better structured tag-base. It is also necessary

to provide users with new interaction models for seamless navigation through structured concept models (i.e. taxonomies), thus allowing them to choose one or more controlled tags which they think are appropriate for the particular content item.

- ***Support for business rules for applying metadata to the content.*** Sun/Oracle has defined various business rules for applying metadata to the content stored in their knowledge management systems. These business rules define a minimal required set of metadata, which have to be assigned to every content item. Such rules in turn add to better structure and better consistency of the overall knowledge base. KiWi will provide support for simple management of these business rules, including their definition on the server side as well as their presentation to the system users in the form of various required fields within the tag-suggester UI itself.
- ***Integration of enhanced information extraction functionality.*** Information extraction extracts facts and terms from content created by users, which are in turn delivered to the user in the form of suggestions. It is expected that the usage of semantic-based information extraction technologies provided by the appropriate KiWi module(s) will result in better tag-recommender functionality provided to the system users.

It is required that KiWi's information extraction functionality will be capable of handling both the free as well as controlled tags and vocabularies (which were described before).

Manage Metadata

Business challenge: How to implement and integrate a corporate Ontology Management model and at the same time leverage the power of Tagging/ Folksonomy?

This use case is concerned with managing corporate concept models (vocabularies, thesauruses and taxonomies and related foksonomies).

Current State

There have been several attempts done in the past at Sun in enabling concept model governance. They have originally started in the form of a top-down governance model, where concepts were maintained by a group of dedicated experts. This model has shown itself as inflexible, slow and thus ineffective in reflecting changes to the knowledge base, which were often required by fast-changing business environment (e.g. changes in the company organizational structure, product lines etc.).

The current knowledge management venues which were in use at Sun (and mainly the SunSpace system) utilize tagging, folksonomies and basic automated content classification for building taxonomies for classification of people, communities and related content.

How Kiwi can help?

KiWi services will help to adopt a new governance model, which combines the top-down and bottom-up approaches, where concepts are maintained and shared in an open manner, by particular communities of practice and interest, while being supervised by subject matter experts. External thesauri management functionality provided by the KiWi platform and PoolParty will be used for managing both controlled (expert) and uncontrolled (folksonomy) tags.

Exploit/Utilise Metadata

Business challenge: How to improve enterprise search, expertise discovery and personalised recommendations ?

Current State

The current implementation of the SunSpace system allows users to search for content using a commercial search engine product. Enhancements have been made to improve the search ranking, show related content, communities and potential experts by leveraging the pre-calculated social values based on Community Equity. A community browser has been implemented which allows to browse communities by region, product and industry using the respective corporate taxonomies.

How KiWi can help?

It is expected that the new KiWi-based search utilising semantic technologies will provide better search results. There are in general three basic types of content handled by the social semantic knowledge management system - content, people and communities. All these types of content are described by various types of metadata, which are also handled by the system. Implemented search functionality will be aware of these individual content types, making it easier for the user search and find relevant expertise present in the knowledge base.

Searching is in general an iterative process, where users repeat the search with different queries, until they find what they are looking for. Provided search functionality will thus be extended by the ability to further refine the search query.

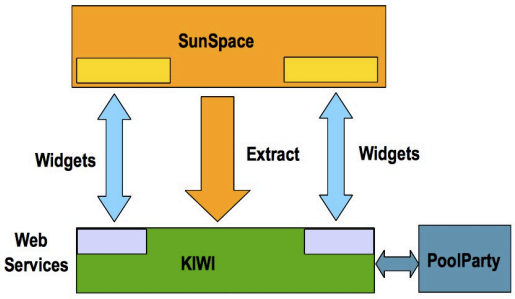
The search implementation will also be capable of handling values of particular content items, as provided by the appropriate social value system (like Community Equity). This will allow structuring the search results based not just on how much they correspond to the actual search query, but also based on their actual value and relevancy.

The search will also be context-aware providing personalized recommendations presented to the user as an addition to the actual search results. These personalized recommendations will be based on user preferences which are inferred from various types of user-related information, like the content he/she authored, viewed, discussed in the past, the tags he/she used in the past, or the information about his/her social graph.

Integration

It is envisioned to integrate the Kiwi services into the existing application using standard webservices and widget components.

The findings of the Sun Use Cases will be published in December 2010.



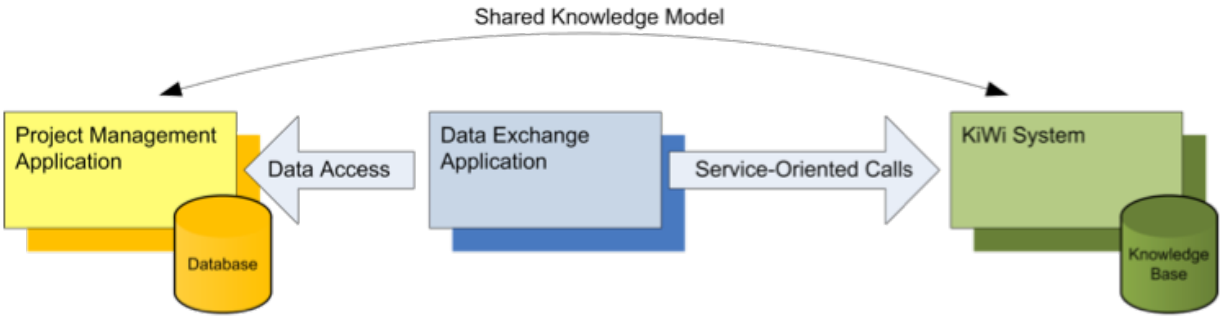
How to Manage Process Knowledge with KiWi?

Logica Use Case

In software development projects there is an ongoing struggle with keeping track of knowledge, with exploring and exploiting the knowledge we already have, and with creating and validating new knowledge gained through software development projects. The ambition for Logica is first that much more knowledge will find its way into the system by increasing the affordance for all involved software managers and developers. Second, the complex relationships between pages and contents will be used to increase the awareness of changes, dependencies and approvals. Any relevant change to a page should either propagate directly to the related pages or the writers should be alerted that further changes are necessary and the approval process and status.

KiWi Setup to Support Collaborative Project Planning

In the Logica Project Management Use Case the KiWi System was connected to the Data Exchange Agent (DxA) and the Logica Project Management Application (LPMA). According to the requirement analysis of knowledge management for Logica, an ERP system is vital for the company. Logica has to follow strict processes to fulfil standards in order to win project bids with customers. This is de facto an industry standard and companies all around the world have the same demand. To take this into account we created an architecture of systems (see figure below) that combines the KiWi platform with an ERP system.



Logica Project Management Application (LPMA)

The LPMA is an ERP system. With this tool, project managers or process designers do their work. It mainly consists of an interface to insert or edit data in forms. It contains a complex data schema so that all needed data can be stored. Additionally the LMPA contains functionality to support the planning process.

Data Exchange Agent (DxA)

To publish data into KiWi - the client DxA was developed. It accesses KiWi and checks for available templates. The user can choose and publish datasets. Being the source for the data, the LPMA is accessed directly at the database level.

The KiWi System

With the KiWi System we refer to the combination of the KiWi platform, the LPMA and the connecting DxA. The whole system shares the knowledge model. While it is a data schema in the LPMA it is a translated version to RDF part of the KiWi platform. Thus the DxA is able to take data from one tool and insert it into the other. With the successful publishing a resource is created in KiWi that directly relates to data in the LPMA. From that moment on, data can be updated in both directions.

Every page that contains data does so by applying the semantic web functionality. A page is a resource of a certain kind from the knowledge model and its data is described through properties. The KiWi system then takes the data for the page and inserts it where the template definitions are placed. Thus the viewed page always shows the current data set and it can be edited without actually editing the page.

Project Planning Scenario

In the KiWi platform the project participants analyze, review, comment and add detailed information about the various products and requirements that they are responsible for. They might also enter ideas about how to fulfil the requirements, design ideas, and the like for later use. During this process they identify issues that need to be clarified, and they challenge and review information about product size and complexity, as this information will be used during estimation. To do so the participants search information entered by other projects about how they have dealt with similar requirements and products. Pages containing relevant information

are recommended by the KiWi platform. The participants also use process descriptions and checklists relevant for the tasks, e.g., checklists that highlight important issues to remember when reviewing requirements.

The project group reviews the risk analysis and especially the resolution strategies proposed by the project manager in the KiWi platform. The resulting changes and comments are documented on the pages containing the risk analysis.

Project participants get the opportunity to review, comment and adjust the estimates, based on their experience. When doing so they can search information from similar projects/activities (how was a similar task estimated in a previous project?), various checklists as well as information about products, requirements and risks for this specific project are recommended.

When the project manager uses the system, she can easily get an overview over changes and comments on various pages, and provide comments to the comments. The participants (and other stakeholders) can comment on the overall schedule as well as their own personal schedule describing when they are supposed to start and finish activities. When finished the information is feed back to the LPMA using the DxA.

The project manager finalizes the schedule, and checks consistency using the LPMA, the consistency check (e.g. is expected future productivity for remaining activities realistic compared to past productivity?) becomes a part of the status report. A final project plan, schedule etc. is published in the KiWi platform.

Summary of User Results

Despite the usual technical problems of a working in prototype modus the users were able to evaluate the concepts of the KiWi system. The test persons were of the opinion that the KiWi concepts clearly made it through to the prototype. They further agreed that a system like this would be highly beneficial in the different settings and that it solves issues that current systems show. Examples for these:

- With the current solution it is actually more difficult for project managers to violate the company's rules than to set up a correct plan.
- The tight communication between process designer and implementers shortens the feedback circle and thus likely to increase the process quality.
- Through the recommendations and the search users are able to spot potentially useful and interesting content and their authors.

"We see the findings very positive. Even though it was not a production ready system, we gathered insights into how such a system could look like and could endorse the use case for project knowledge management. The evaluation proved that our thoughts and ideas were correct and heading in the right direction. We were able to prove the principles of our approach for a successful knowledge management in project management and process design"

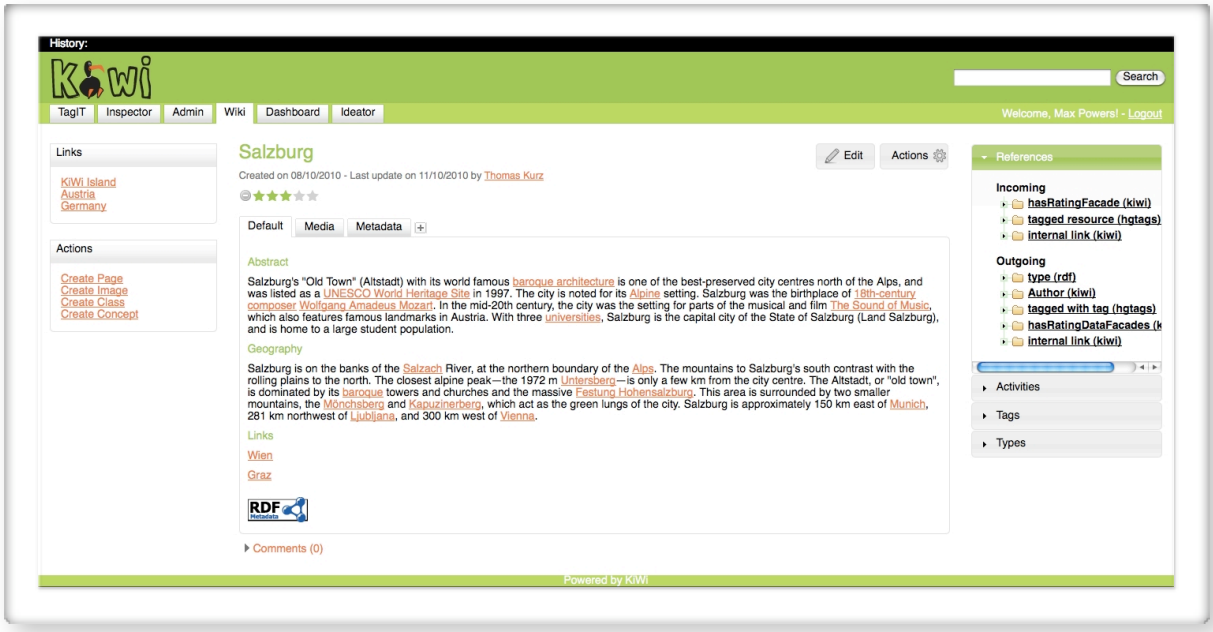
Remarks from Logica Use Case Users

KiWi Core Applications

On top of the KiWi platform, we have implemented a number of exemplary Social Media applications. Here we showcase the Semantic Wiki that illustrates the functionalities offered by the different KiWi services and TagIT, a map-based application for browsing online news and user generated content. KiWi can be used in many different scenarios, e.g. for building public Social Media sites, for knowledge management in the Enterprise 2.0, and for content integration. We also introduce you to the Dashboard, Inspector and provide an overview of the search options.

The Semantic Wiki

The primary and most generic interface to the KiWi system is a Semantic Wiki. The layout and functionality of the KiWi Wiki is inspired by its predecessor IkeWiki: the left column offers navigation functionality, the centre column contains the main (human-readable) Wiki content, and the right column contains dynamic widgets that display additional information about the content item based on its metadata (e.g. a map or incoming and outgoing links).



Getting Started

If you explore KiWi for the first time you might perhaps want to follow this guide, in order to discover KiWi's full potential. Depending on the permissions that have been assigned to the anonymous user you will either see the content of the Start Page and maybe also be able to

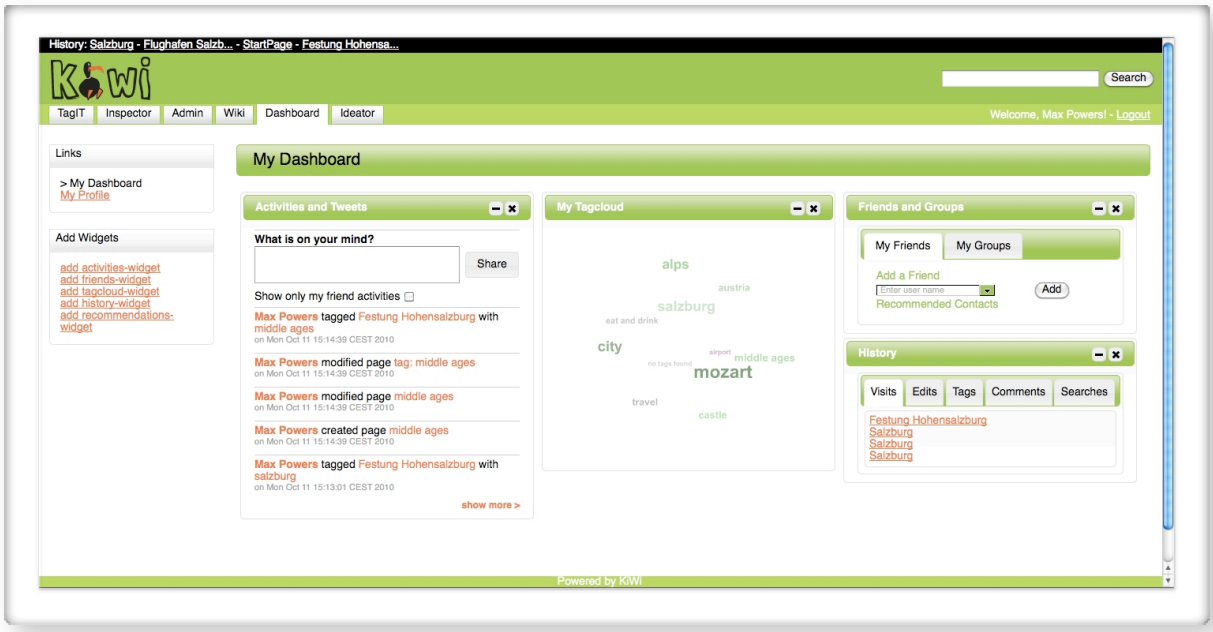
edit it, or you will get an information text that hints that you do not have enough permissions to view and/or edit the content.

- Register yourself on the KiWi platform by clicking on 'Sign Up' in the upper right corner (or login if you already have an account). You will receive registration information to your provided email address. Depending on the permission configuration, you will have more rights than before. Contact your system administrator, if you still have restricted access.
- Alternatively to the sign up form, you may signup/login via your Facebook Account or with your FOAF-SSL client certificate. These two alternatives to the standard login are available on the login site.
- You should be logged in by now and able to try things. Try out to edit, tag and annotate Wiki content. You may upload pictures or semantically link content resources. All these activities are available under the 'actions' list in the Wiki view. Your changes to content and metadata can be looked up and reverted in the history view.
- New content items can be created in the Wiki style: By creating new, empty links. If you've created a new content item, you will get owner rights and be able to restrict view and modification activities to certain users or groups in the 'Permission' section.
- All content item specific activities will be listed on the right-hand side. Every visit, edit, comment, tag and annotate action is recorded and presented together with the content.
- The incoming and outgoing semantic relations linking to and from the current content item are displayed above the list of activities in a tree-structure. The top-nodes represent the RDF properties (e.g. 'type') and the bottom-nodes represent the RDF subjects in incoming relations and RDF objects in outgoing relations. Clicking on a node redirects to the resource.
- Tagging of content can either be done by choosing the actions entry, or by clicking the button below the article title. Available and well-fitting tags will be suggested by the auto-completion process.

The Dashboard

When you login you will get redirected to the Dashboard front page.

The Dashboard is a user’s personal(ised) start page in the KiWi system. It follows the same general layout as the Wiki, i.e. the left column provides generic navigation and the centre and right columns contain the actual content. While the look of the Dashboard is freely customisable by the user, the KiWi core system by default provides the following information:

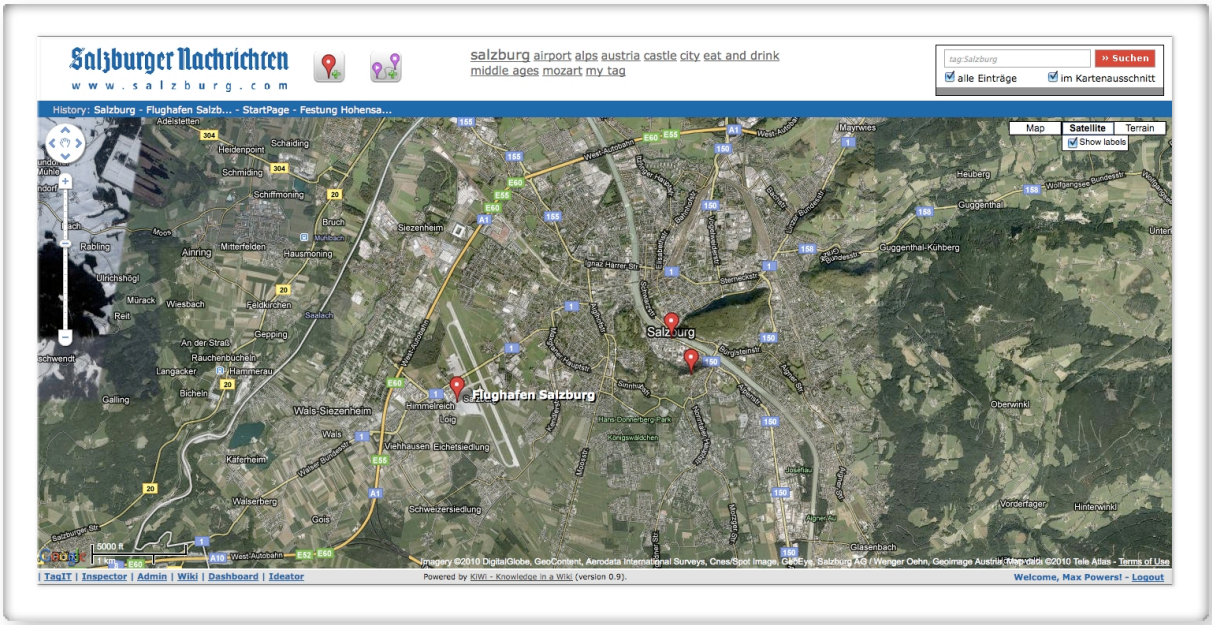


- The Stream of Activities is the most important part of the Dashboard: it contains an aggregated list of activities that happened in the user's "universe", i.e. updates to content items that are either explicitly watched by the user or implicitly added to the user's item set e.g. because they have been edited by the user, because they have been edited or rated "good" by one of the contacts of the user, or because they have been recommended to the user based on previous activities by the personalisation component of KiWi
- The Recommendations widget provides a list of additional content items that might be relevant to the user; different recommendation algorithms are investigated as part of the KiWi project. Moreover, the activity history and the user's personal tag cloud are presented on the front page.
- Besides the main view, the Dashboard is also the place where the user manages his own profile. Behind the 'My Preferences' tab the user will be able to setup his visual presentation of the Dashboard. One may also want to import RSS-feeds from other sources. Therefore, the 'My Subscriptions' link is used to subscribe for RSS-feeds. To view and/or modify the profile click on 'My Profile'. Modifications to personal information like name, location, IM-accounts, email address and so on can be managed in that view. Contacts, groups and the personal content item watch list can be administrated under the respective links, as well.
- The 'My Tags' tab allows to put meaning behind the tags that the user created or defined. This is done by describing the relation of the tag to the tagged item. The list of available meanings is currently narrowed to 'relates to', 'identifies who owns', 'identifies what is',

'identifies what is about', 'qualifies', 'categorizes', 'reminds', 'self references' and 'conceptualizes'. Specifying tag relations help to refine recommendations and improve the personalization services.

TagIT

In TagIT, users browse through a map (based on Google Maps), where they can "tag" locations with descriptions and provide interesting information about them, e.g. cafés, bars, sports parks, hiking tours, etc. Such tags can be associated with categories from a SKOS thesaurus and with additional multimedia data like photos or videos. Other users can then browse or search for interesting locations using the same interface.



Users are able to add tags to locations or routes by selecting the specific button on the top of the TagIT frame. A popup window appears that allows to insert textual and multimedia content and to attach it to an address or route on the map.

In addition to the web-based platform, TagIT also offers a mobile client that can run on a GPS-enabled mobile phone. When visiting an interesting location, users can then start the TagIT application, take a picture of the location, add a short description and directly upload the "tag". The tag is automatically geolocated and immediately available for other users.

Although quite different on the user interface and in the way it is used, TagIT actually closely follows the Wiki principles: everyone can add and edit tags, the system is easy to use, tags can be linked, tags are versioned, and different kinds of content are supported. On top of KiWi, tags are realised as content items and can thus be displayed in both the TagIT user interface and the

previously described KiWi Wiki (in which case a small map widget is displayed in the right column showing the position).

The Inspector

The KiWi Inspector is an application developed for advanced users and developers. It provides a more technical insight into the current context. The currently available functionalities are:

- Content Item Inspector : Displays the RDF data associated with the current content item as RDF/XML; the shown RDF data is the same as would be displayed to a Linked Open Data client when accessing the KiWi system
- Tag Inspector : displays a list of all tagging actions of the current content item and the RDF data generated by them as RDF/XML
- User Inspector : displays the RDF data associated with the currently logged in user as RDF/XML
- Revision Inspector: displays the transaction-ally versioned content and metadata
- Base and Inferred Triple Inspector: lists details about the base and inferred RDF triples and allows explanation for inferred triples
- Inspect Community Equity Information: displays information about the change of the community equity value for the current user over time

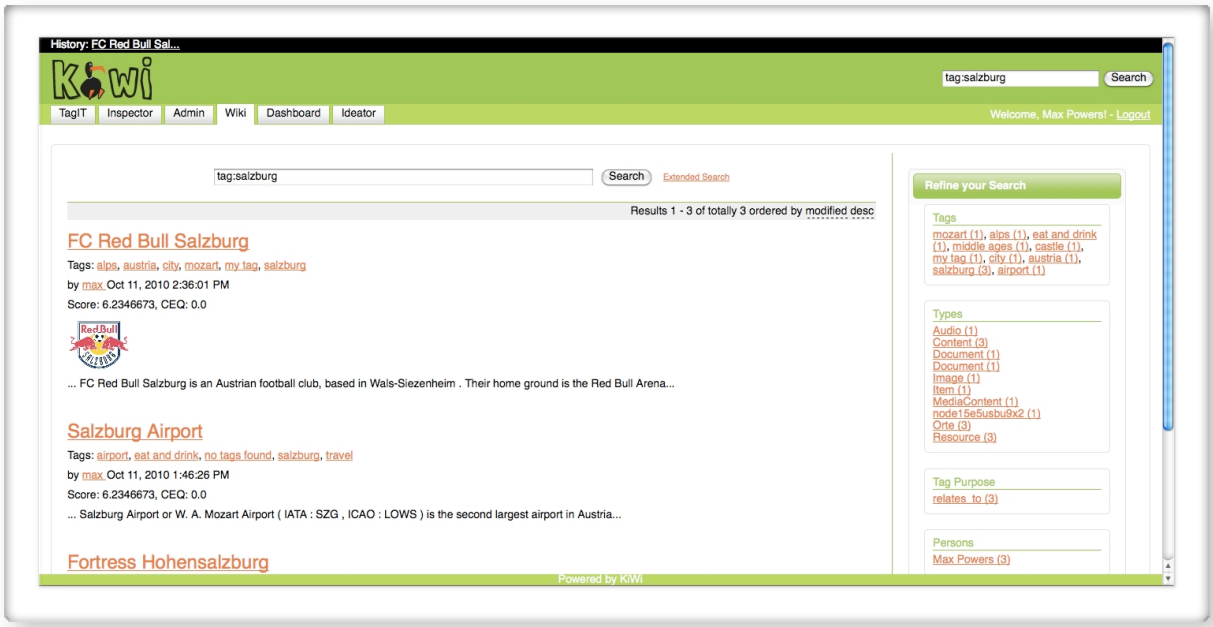
Search

The KiWi core system also provides a generic search functionality accessible from within all KiWi applications. When a user switches to search and selects a content item, he is redirected back to the previous application afterwards. KiWi currently allows a combination of full-text search, metadata search (tags, types, persons), and database search (date, title).

The KiWi search interface implements a so-called “facetted search” based on Apache SOLR: the user starts with a keyword search, resulting in a list of content items ordered by relevance or time. In case the user is not satisfied with the results, he has the option to refine his search using one or more of the facets offered in the right column of the search result box. Currently, the KiWi system offers the facets “tags”, “types”, and “persons”, which we have identified as the core facets needed in any system. For each facet, only the criteria occurring in the currently displayed search results are listed, together with a count of the content items matching the criterion. Selecting one of the criteria narrows down the search.

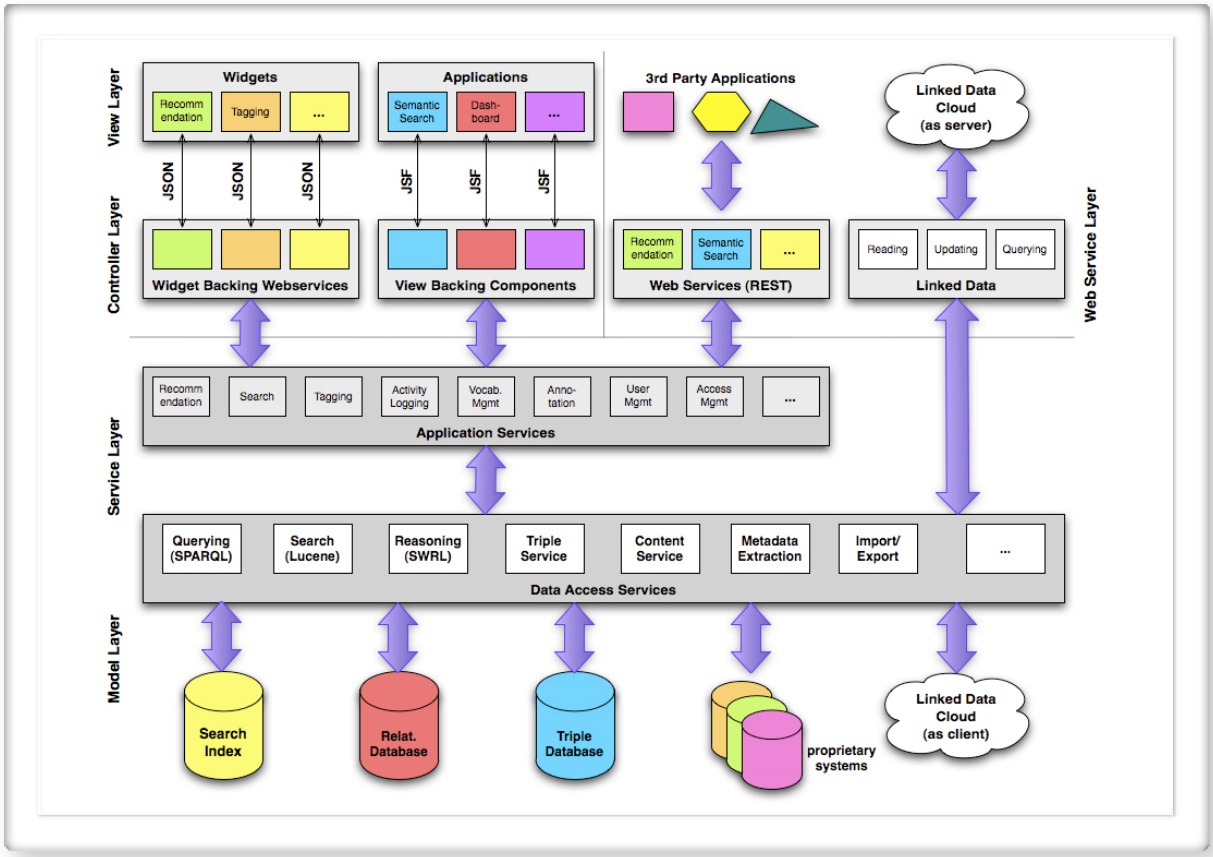
All search facets are included in the full-text search box; this decision has been made to provide all search functionality in one place without confusing the user and to allow advanced users to

directly search using the text field. Also, it makes it much simpler to bookmark searches or include them in a user's personal stream of activities on the Dashboard.



KiWi Architecture

KiWi Applications are built on top of the KiWi Platform, which provides all the core functionalities needed by most Semantic Social Software applications. In the following, we briefly describe the architecture. The KiWi system is implemented on top of JBoss Seam5 and Java Enterprise Edition (Java EE 5), and thus follows a component- and service-oriented architecture. The Figure below depicts the overall structure of the KiWi system:



Model Layer

The model layer comprises the KiWi data model and is represented in a relational database, a triple store, and a full-text index. Entities are persisted using the Hibernate framework, which maps Java objects to relational tables. The KiWi triple store is a custom implementation also based on the relational database, because existing triple store implementations provide insufficient support for features like versioning and additional metadata about triples that are needed by KiWi. The full-text index is implemented using Hibernate Search and currently allows to search over title, textual content, tags, authors, and RDF literals.

Service Layer

The service layer provides services to other components in the KiWi system. Of central importance is the KiWi Entity Manager, which provides unified access to content items and RDF metadata. Further core services are the revision service -- taking care of versioning, and the transaction service, allowing the management of all updates to KiWi content in reliable transactions. Both services are heavily used internally by the KiWi Entity Manager and usually not used by further components. Besides these core services, the service layer may contain additional services that offer certain common functionalities. For example, the KiWi system currently offers an “ontology service” that provides convenient access to the triple store using higher-level concepts like “classes” and “properties”, and a “content item service” that allows to easily access all functionalities associated with content items (creating, loading, updating).

Controller Layer

The controller layer consists of action components that implement a specific functionality in the KiWi system. For example, the Semantic Wiki application contains a “view action”, a “edit action”, a “annotation action”, and a “history action”, and the TagIT application contains a “explorer action” and a “tagging action”. Action components are usually very close to some functionality offered in the user interface, and they make use of service components to access the content in the KiWi system.

Viewer Layer

The view layer is implemented using Java Server Faces (JSF), which are used to generate the HTML presentation of the KiWi user interface and the user interaction with the system. JSF pages are linked with action components in the controller layer. Also part of the view layer are the web services offered by KiWi. Currently, there are web services for accessing the triple store and SKOS thesauruses, and there is a “linked open data” service offering the content of the KiWi system to linked open data clients.

About

KiWi – Knowledge in a Wiki is an EU-funded project (No 211932) combining the Wiki philosophy with methods of the Semantic Web, aiming to develop a new approach to knowledge management.

The KiWi consortium brings together leading research groups (Salzburg Research, Aalborg University, Brno University of Technology, LMU Munich) in the areas of semantic Wikis, reasoning, information extraction, personalisation, and knowledge management for software processes. These are matched by two large international corporations in knowledge intensive areas (Sun Microsystems/ORACLE and Logica) that offer use cases demonstrating a clear need for the advanced knowledge management we envision in the project, and by a SME (Semantic Web Company) specialised in the dissemination of semantic technologies to the industry.

Project Facts

KIWI (Knowledge in a Wiki)

Project Number: 211932, **Project Type**: Small or medium-scale focused research project (STREP), **Project Budget**: 3,8 Million Euro, **EC Contribution**: 2,69 Million Euro, **Funded By**: EU 7th Framework Programme, Theme ICT, **Start Date**: 2008-03-01 -- **End Date**: 2011-02-28,

Project Website: www.kiwi-project.eu - **Community Website**: www.kiwi-community.eu

Partners

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7(2007-2013) under grant agreement No. 211932

Contact

John Pereira
Knowledge & Media Technologies
Salzburg Research Forschungsgesellschaft m.b.H
T +43 662 2288 246
john.pereira@salzburgresearch.at
www.kiwi-community.eu

www.kiwi-community.eu