IkeWiki and KiWi

Semantic Wikis for Collaborative Knowledge Management

Ontolog Mini-Series

November 2008

Dr. Sebastian Schaffert
Salzburg Research Forschungsgesellschaft

Dr. Peter Dolog
Aalborg University

http://www.kiwi-project.eu

http://planet.kiwi-project.eu

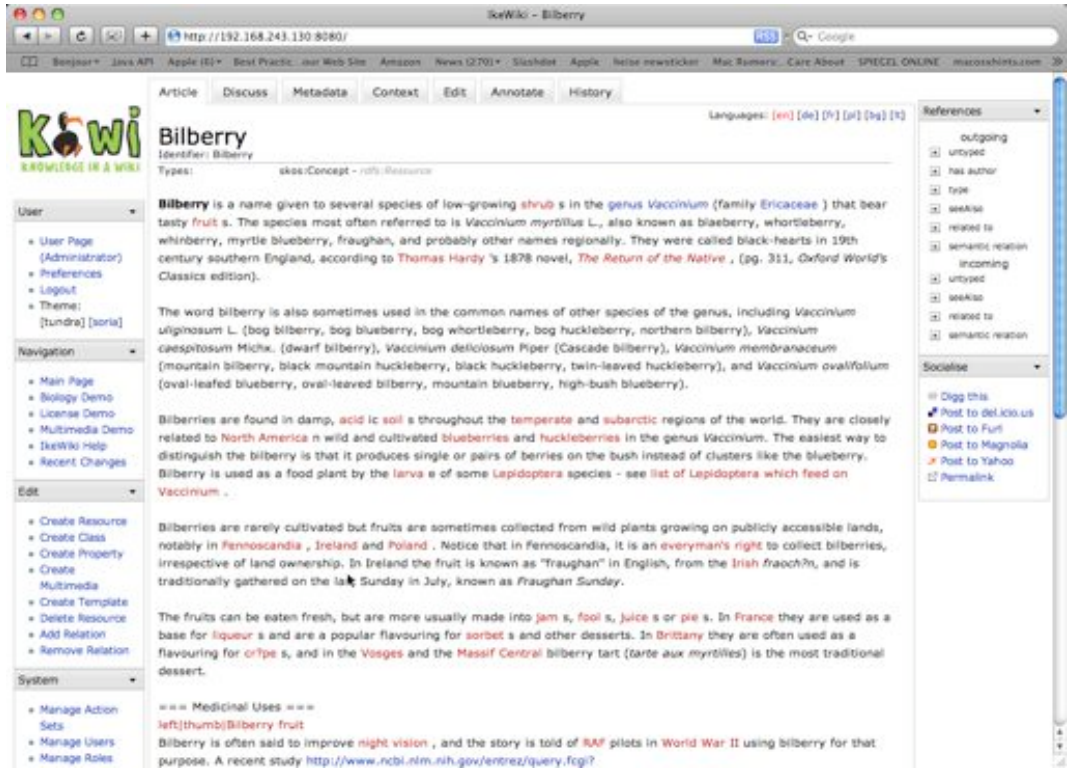
Outline

1. IkeWiki Interface
   - Wiki Interface
   - What to do with Semantic Annotations
   - How to do Semantic Annotations
2. IkeWiki Architecture
   - Storing Pages and Metadata
   - Rendering Pipeline
   - Transformation
3. From IkeWiki to KiWi

IkeWiki Interface

IkeWiki Interface

- „normal“ wiki interface for viewing/editing „normal“ content
- somewhat resembling Wikipedia/MediaWiki



IkeWiki Interface – What to do with Semantic Annotations?

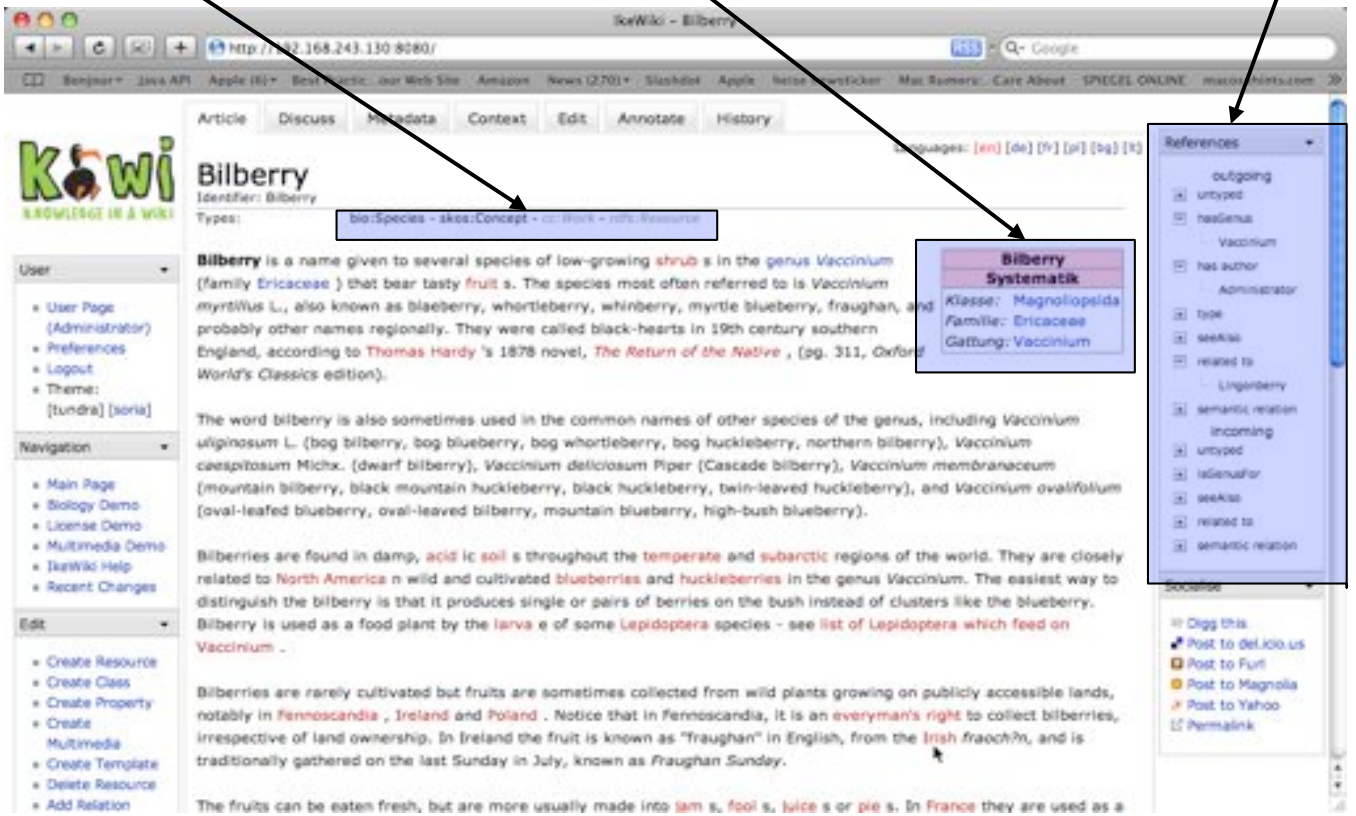
- people will only use metadata when they see a benefit in it!
- possible uses of metadata (for authors):
  - support in editing (e.g. avoiding redundancy of data)
  - interoperability and exchange between systems
- possible uses of metadata (for users):
  - improved search and navigation
  - improved page presentation

IkeWiki Interface – What to do with Semantic Annotations?

categories/types

context-dependent presentation

navigation

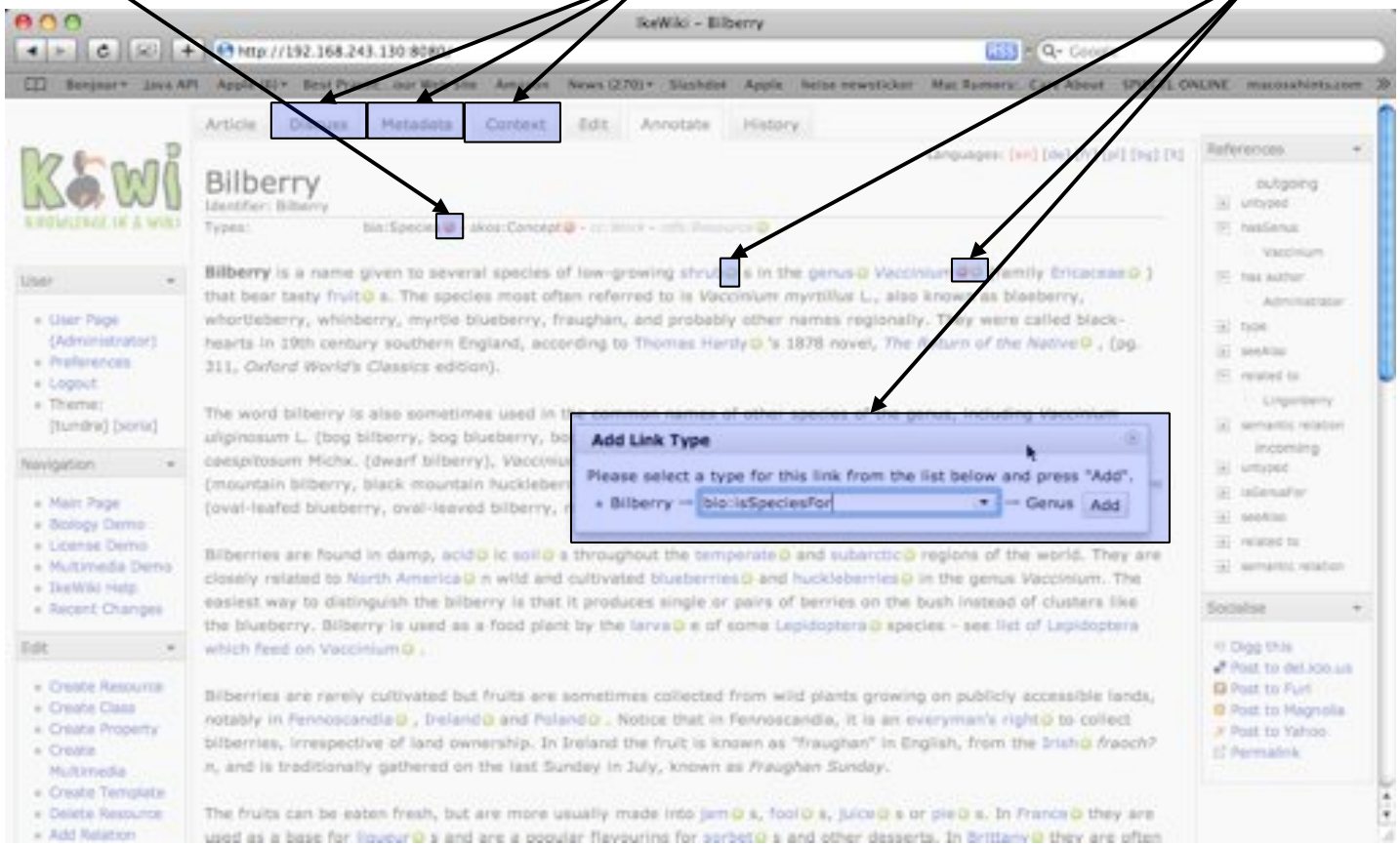


IkeWiki Interface – How to do Semantic Annotations?

- people will only use metadata if it is easy!
- lower the technical barrier for metadata creation
  - provide an easy to use interface for adding annotations (AJAX-based adding of link and page types)
  - support the user by reasonable suggestions where possible (link and page type suggestions based on reasoning)
  - support different levels of experience and hide unnecessary complexity (showing advanced features only to advanced users)
  - allow domain experts and knowledge engineers to collaborate
  - immediate exploitation of semantic annotations (instant reward)
  - supporting different levels of formalisation (evolving knowledge models)

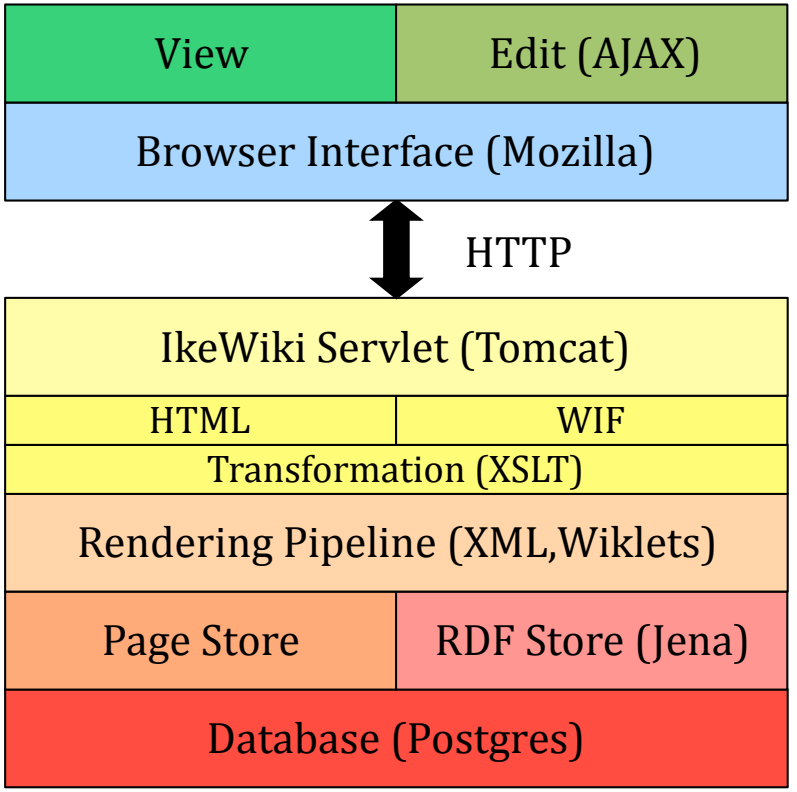
IkeWiki Interface – How to do Semantic Annotations?

page types advanced features link types



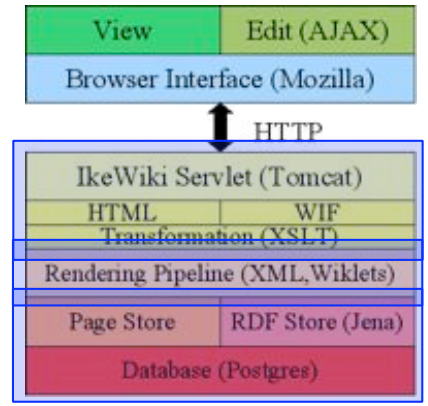
IkeWiki Architecture

IkeWiki Architecture



- Page Store: XML content
- RDF Store: RDF metadata
- Rendering Pipeline: combination of content and metadata
- transformation into HTML and other formats
- AJAX editing and viewing in Mozilla/Firefox

IkeWiki Architecture
Storing Content and Metadata



- page content and metadata stored separately
  - *page content*: PostgreSQL database
  - *metadata*: Jena RDF memory model with OWL-DL reasoning, backed by a database model for persistent storage
- rendering pipeline combines page content with metadata
  - „wiklets" enrich page content with information from the knowledge model
- XSLT transformation transforms „enriched" page content to different formats
  - HTML for presentation
  - HTML for tooltip
  - XML/WIF for exchange

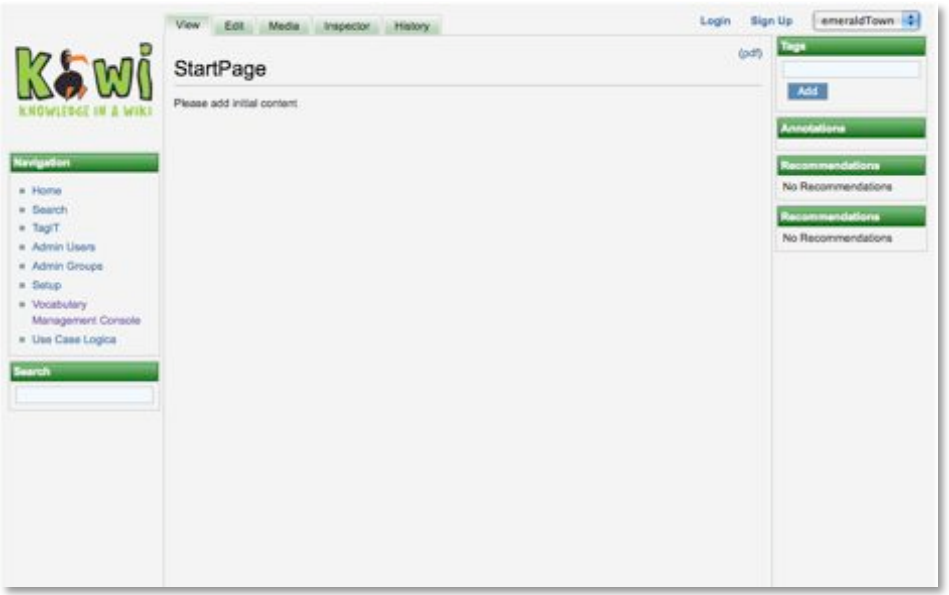
From IkeWiki to KiWi

KiWi: A Flexible Platform for Semantic Social Software

- most Social Software systems build on the same principles like Wikis: everyone can participate, easy linking, easy editing, versioning
- KiWi uses the Semantic Wiki technology as a platform for many similar kinds of Semantic Social Software systems (see next slides)
- KiWi provides the technological infrastructure in the areas of content access (data + metadata), transactions, versioning, reasoning, information extraction and personalisation
- different Social Software applications can be implemented easily as "perspectives" (or "view plugins") on top of this infrastructure

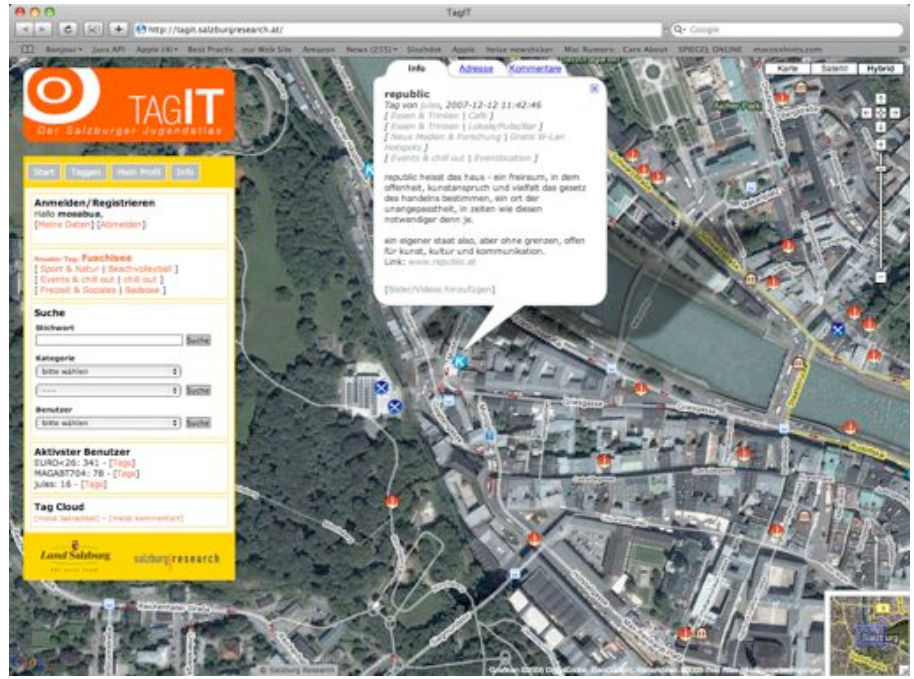
Example Perspective: Wiki

- ordinary Wiki interface (View, Edit, History, …)
- similar functionality like IkeWiki (Semantic Annotations)
- implemented as a generic perspective on KiWi



Example Perspective: TagIT

- tagIT: map based interface to describe locations ("youth atlas of Salzburg")
- youths can „tag" locations on a map, add descriptions, photos, comments, etc.
- search by various different „navigation paths":
  map, full-text, topic, user, rating, ...



KiWi Technology

- component-based Java EE/JBoss Seam application
- core components:
  - KiWiEntityManager: access to content and meta-data, transactions, revisions
  - Facades: different (Java) perspectives on content and meta-data (similar to Elmo but more powerful)
  - Services (stateless EJB): provide common functionality (e.g. recommendations, tag clouds, …)
  - ActionBeans (stateful EJB): provide user interaction functionality for the user interface
  - JSF Views: specify different user interfaces

Project Facts

- EU FP7 research project
  - started in March 2008
  - duration 36 months
- Partners: Salzburg Research, University of Aalborg, University of Munich, Technical University of Brno, Sun Microsystems, Semantic Web School, WM-data

Contact

| Dr. Sebastian Schaffert
| Salzburg Research Forschungsgesellschaft
| Jakob Haringer Str. 5/II
| A-5020 Salzburg

| sebastian.schaffert@salzburgresearch.at
| http://www.kiwi-project.eu
| http://planet.kiwi-project.eu