STI · INNSBRUCK

Semantic Web

Lecture XIII – 25.01.2010

Tools

Dieter Fensel and Katharina Siorpaes

Today's lecture

STI · INNSBRUCK

#	Date	Title
1	12.10,2009	Introduction
2	12.10,2009	Semantic Web Architecture
3	09.11.2009	RDF and RDFs
4	09.11.2009	Web of hypertext (RDFa, Microformats) and Web of data
5	23.11.2009	Semantic Annotations
6	23.11.2009	Repositories and SPARQL
7	07.12.2009	OWL
8	07.12.2009	RIF
9	11.01.2010	Web-scale reasoning
10	11.01.2010	Social Semantic Web
11	11.01.2010	Ontologies and the Semantic Web
12	**25.01.2010**	**SWS**
13	**25.01.2010**	**Tools**
14	**25.01.2010**	**Applications**
15	TBD	Exam

Agenda

STI · INNSBRUCK

1. Motivation
2. Technical solutions and illustrations
   1. Protégé
   2. Collaborative Protégé
   3. Semantic Media Wiki
   4. Web Service Modeling Toolkit (WSMT)
3. Extensions (Overview)
4. Summary
5. References

MOTIVATION

Motivation

STI · INNSBRUCK

- Ontology editors
- Ontology alignment
- Semantic wikis (see lecture on Social Semantic Web)
- Games for semantic content creation (see lecture on Social Semantic Web)
- Reasoners
- Storage
- Semantic annotation tools for different types of content (see lecture on semantic annotation)
- Semantic Web service development tools

www.semanticweb.org

TECHNICAL SOLUTION AND ILLUSTRATIONS: TOOLS

Selection of tools

STI · INNSBRUCK

- Protege
- Semantic MediaWiki
- Web Service Modeling Toolkit

Protégé/Collaborative Protégé

Protégé-Facts

STI · INNSBRUCK

- Free, open source ontology editor and knowledge-base ramework.
- Based on Java.
- Written as a collection of plug-ins which can be replaced singly or as a whole.
- Extensible.
- Provides a plug-and-play environment.
- Can be customized in order to provide domain-friendly support.
- Available at http://protege.stanford.edu/

Protégé Facts

STI · INNSBRUCK

- Supports the creation, visulization and manipulation of ontologies.
- Supports a variety of formats like RDF(S), OWL and XML Schema.
- Enables rapid prototyping and application development.

There are two different ways to modell ontologies:

- Frame based via the Protégé-Frames editor
- In OWL via the Protégé-OWL editor

Protégé Frame-based editor

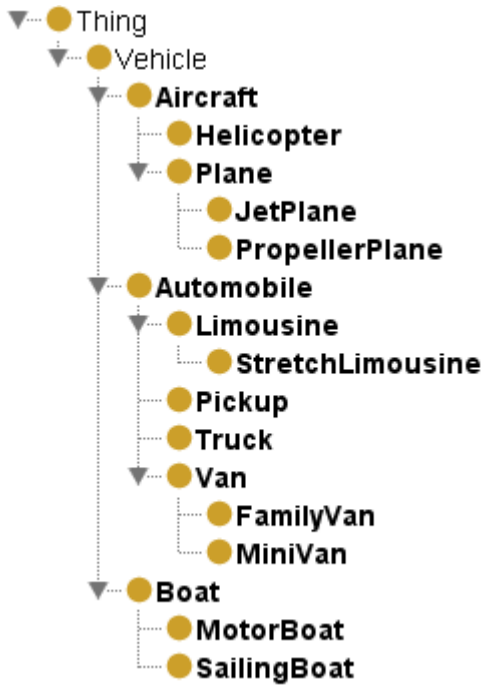
STI · INNSBRUCK

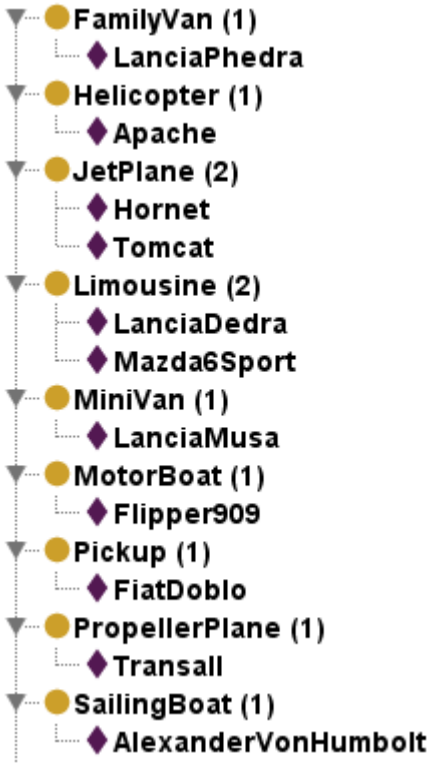
- Construction and population of ontologies that are frame-based.
- Conformant to OKBC (Open Knowledge Base Connectivity Protocol).
  - An ontology is a set of classes.
  - These are structured in a subsumption hierarchy.
  - To each class a set of slots to express properties and relationships is assigned.
  - Each class has a set of instances (individuals which hold concrete values of the properties of the respective class.

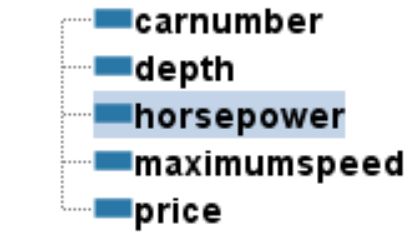
Protégé-Frame-based editor

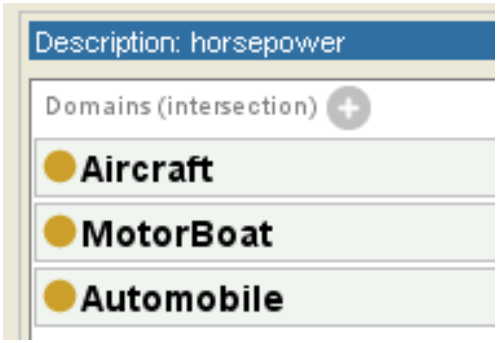
STI · INNSBRUCK

- Classes structured in a taxonomy
- Instances assigned to classes
- Properties assigned to classes









Protégé OWL editor

STI · INNSBRUCK

- Protégé-OWL editor is an extension of Protégé that supports the Web Ontology Language (OWL).
- An OWL ontology may include descriptions of classes, properties and their instances.
- OWL formal semantics specifies how to derive its logical consequences.
- Those are facts not literally present in the ontology, but entailed by the semantics.

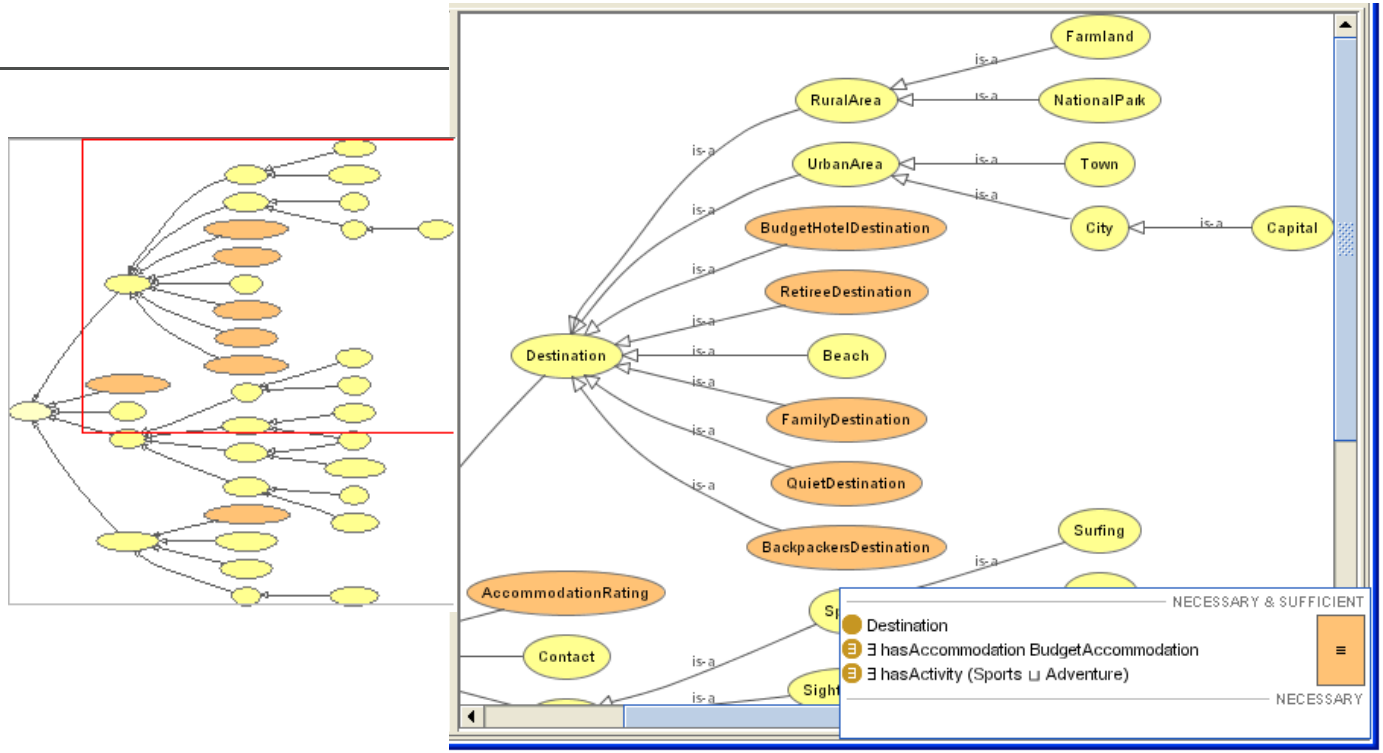
The Protégé-OWL editor enables users to:

- Load and save OWL and RDF ontologies.
- Edit and visualize classes, properties, and SWRL rules.
- Define logical class characteristics as OWL expressions.
- Execute reasoners such as description logic classifiers.
- Edit OWL individuals for Semantic Web markup.

Protégé-OWL editor

STI · INNSBRUCK

- Graphical representation of taxonomy together with axioms.



SWRL Rules

Name	Expression
Def-hasAunt	hasParent(?x, ?y) ∧ hasSister(?y, ?z) → hasAunt(?x, ?z)
Def-hasBrother	hasSibling(?x, ?y) ∧ Man(?y) → hasBrother(?x, ?y)
Def-hasDaughter	hasChild(?x, ?y) ∧ Woman(?x) → hasDaughter(?x, ?y)
Def-hasFather	hasParent(?x, ?y) ∧ Man(?y) → hasFather(?x, ?y)
Def-hasMother	hasParent(?x, ?y) ∧ Woman(?y) → hasMother(?x, ?y)
Def-hasNephew	hasSibling(?x, ?y) ∧ hasSon(?y, ?z) → hasNephew(?x, ?z)
Def-hasNiece	hasSibling(?x, ?y) ∧ hasDaughter(?y, ?z) → hasNiece(?x, ?z)
Def-hasParent	hasConsort(?y, ?z) ∧ hasParent(?x, ?y) → hasParent(?x, ?z)
Def-hasSibling	hasChild(?x, ?y) ∧ hasChild(?z, ?y) ∧ differentFrom(?x, ?z) → hasSibling(?x, ?z)
Def-hasSister	hasSibling(?x, ?y) ∧ Woman(?y) → hasSister(?x, ?y)
Def-hasSon	hasChild(?x, ?y) ∧ Man(?x) → hasSon(?x, ?y)
Def-hasUncle	hasParent(?x, ?y) ∧ hasBrother(?y, ?z) → hasUncle(?x, ?z)

- Definition of SWRL rules.

Collaborative Protégé

- is an extension to Protégé.
- supports collaborative ontology editing.
- supports annotation of ontologies and ontology changes.
- supports searching and filtering of annotations.
- supports a voting mechanisms for changes.
- provides two different ways to enable collaborative ontology editing.
  - Multi-user mode
  - Standalone mode

STI · INNSBRUCK

Multi-user mode:

- Ontology is hosted on server.
- Multiple clients can edit ontology simultaneously.
- Changes introduced by one client become visible to the others immediately.
- Preferred mode Collaborative Protégé should be run in.

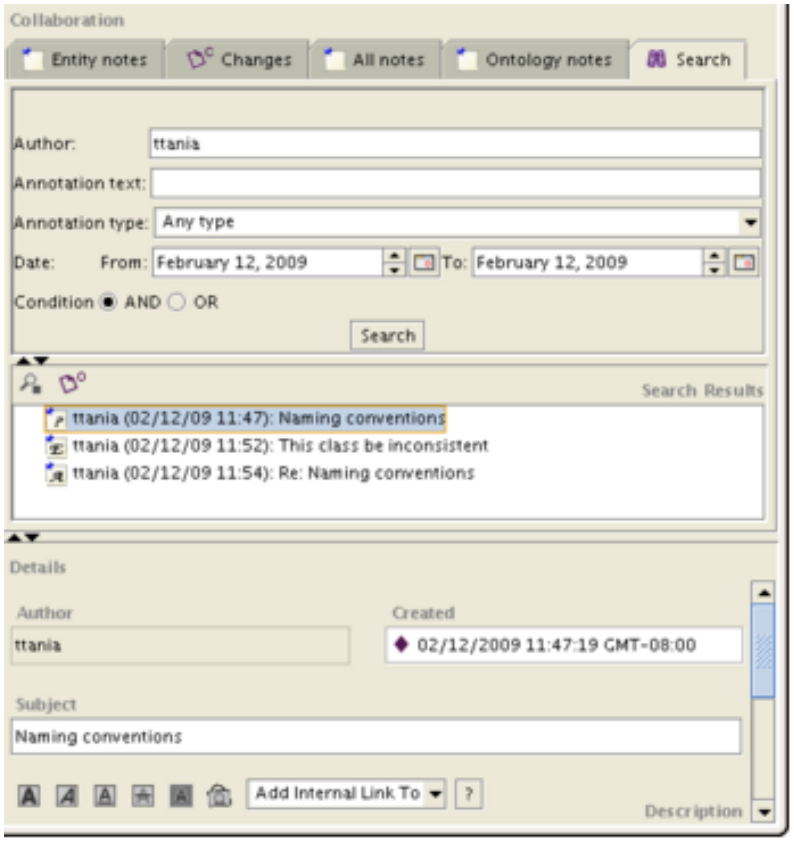
Standalone mode:

- Multiple users access one ontology in succession.
- Ontologies are stored on a shared drive.
- Users access the same project files.
- Parallel access is not possible.

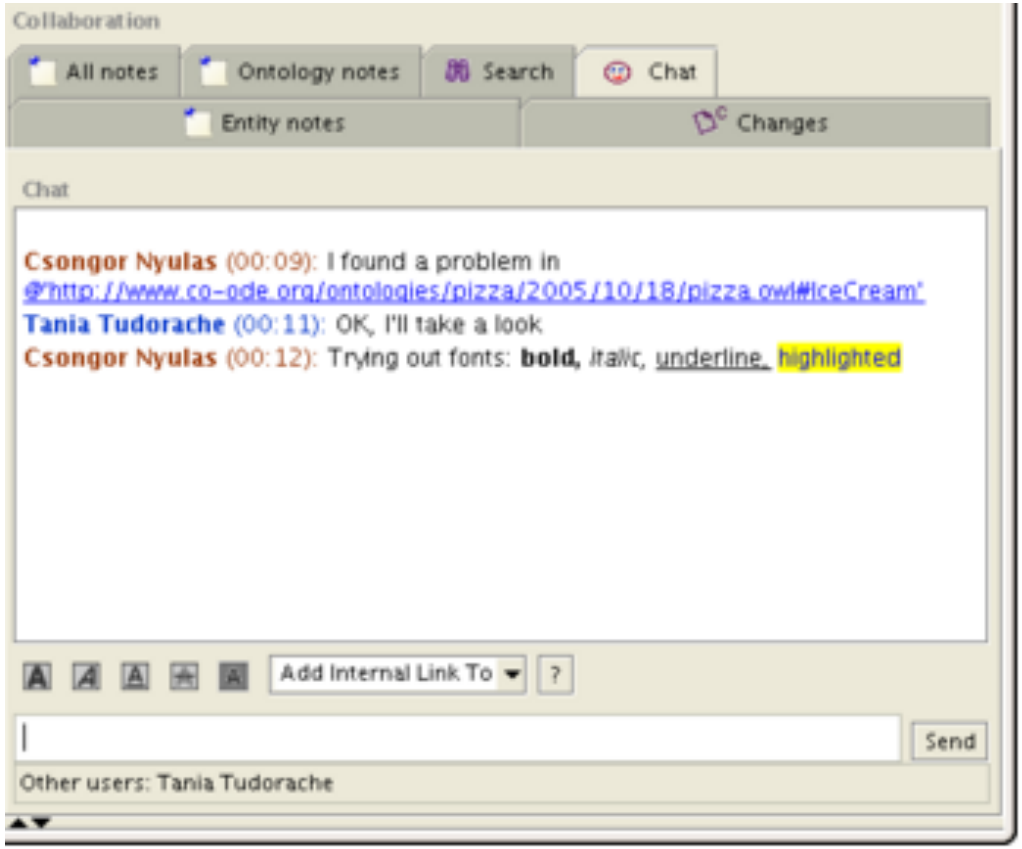
Collaborative Protégé con't

STI · INNSBRUCK

- Searching notes from other users based on certain criteria.



- Chating with other users while working on one ontology.



Semantic Media Wiki

STI · INNSBRUCK

Semantic Media Wiki

- Extension of Media Wiki (Wikipedia).
- Tool for semantic annotation of Wiki content
- Search, organise, tag, browse, evaluate and share content.
- Adding semantic annotations to the traditional Media Wiki.
- Enables machines to understand and evaluate texts.
- Available at http://semantic-mediawiki.org/wiki/Semantic_MediaWiki

Semantic Media Wiki provides:

- Autmatically-generated lists: manually updated lists are error prone, computationally created lists are always up-to-date and can be customized easily.

- Visual display of information: additionally to lists SMW provides much richer views like calendars, timelines, graphs, maps and others.

- Improved data structure: reduces complexity by using queries to structure data, provides templates to create structure and forms which facilitate the addition of semantic information.

Semantic Media Wiki Benefits

STI · INNSBRUCK

- Searching information: users can access information through the formulation of their own queries.
- Inter-language consistency: redundant data distributed over different languages can be expressed semantically. That ensures consistency among the used languages and enables the reuse of information.
- External reuse: SMW can serve as a source of data for certain applications by providing the means to export content in formats like CSV, JSON and RDF.

Semantic Media Wiki Namespaces

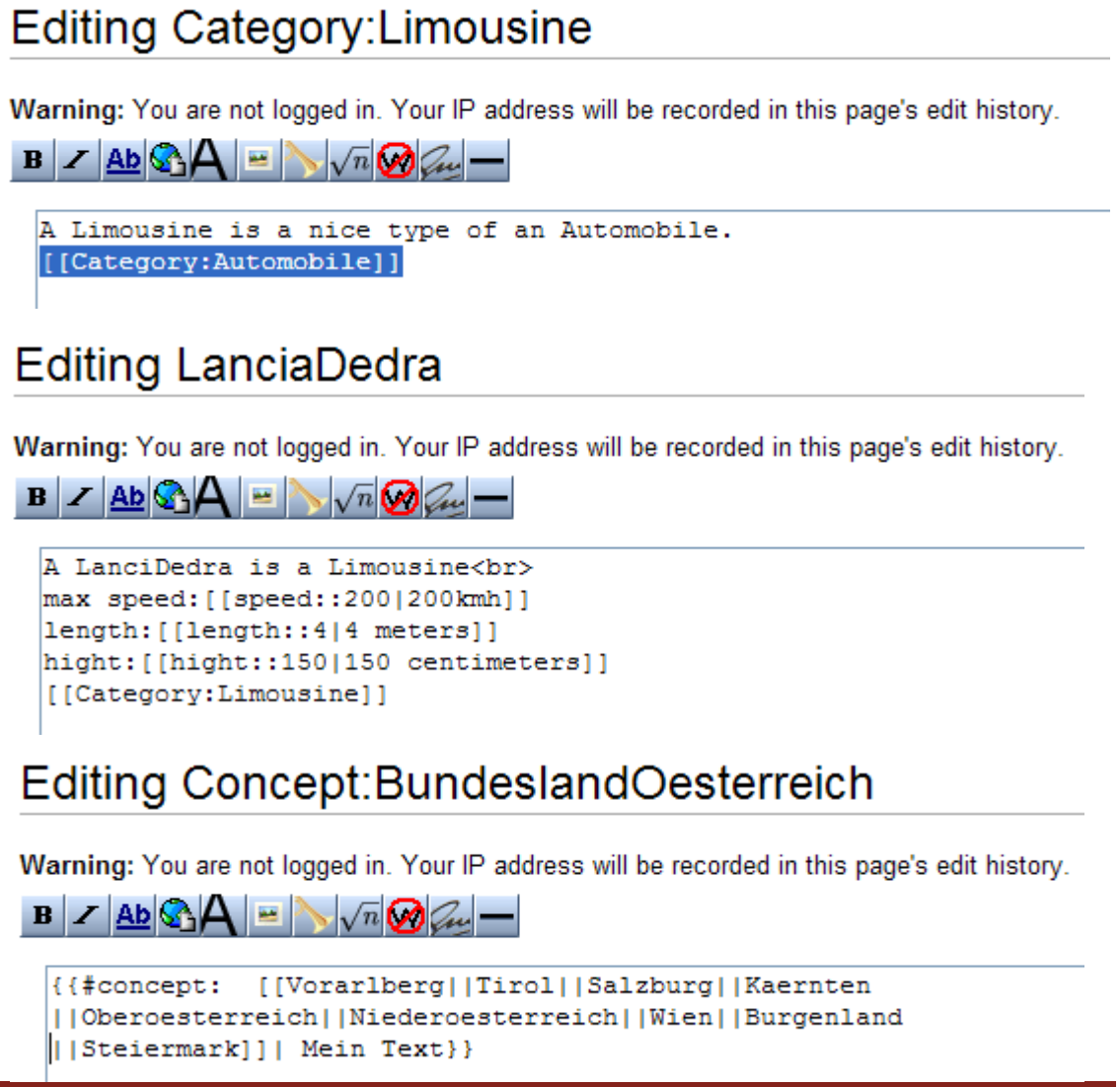
STI · INNSBRUCK

- In SMW, content is divided by using namespaces.
- Those are e.g. „Category:“, „Property:“, „Special:“, „Help:“, „User:“.
- Some of them are editable by every user, some cannot be edited at all and some other need authentication before being displayed.
- When a page is edited, the old version is not deleted.
- This enables recovery in case of vandalism or spam.
- Each page can contain text, images, files, internal and external links...

Semantic Media Wiki Editing

STI · INNSBRUCK

- Creating a taxonomy of categories via [[Category:Supercategory]]
- Typing of an element via [[Category:CategoryXYZ]]
- Assigning property/value pairs via [[PropertyXYZ::Value]]
- Creating concepts for automatic list generation via {{#concept: [[List elements]]}}

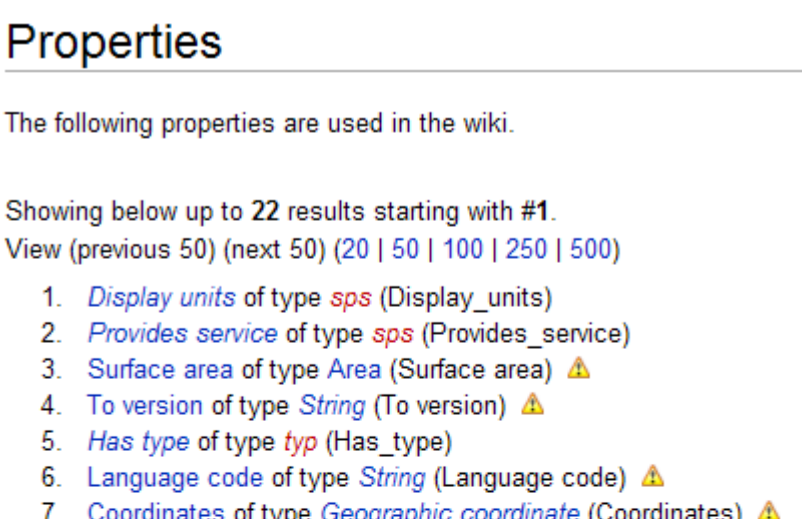


Semantic Media Wiki Browsing

STI · INNSBRUCK

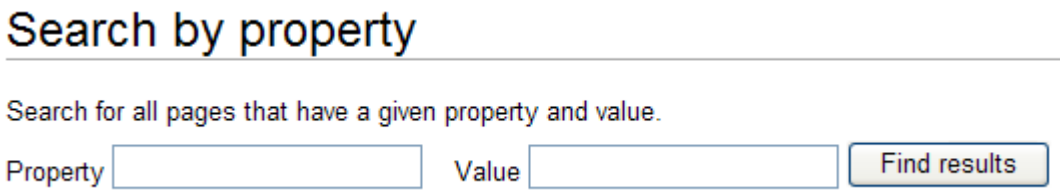
- Semantic browsing via Special:Browse interface.

- Viewing all properties, types and values via Special:Properties (not only for properties but many more).



- The factbox summarizes the semantic data of each page.

- Simple search interfaces for different types of searches.



Semantic Media Wiki Searching

STI · INNSBRUCK

- Inline queries dynamically include query results into pages. A query created by one user can then be used by many others.

```
{{#ask: [[Category:City]] [[located in::Germany]]
| ?population
| ?area#km² = Size in km²
}}
```

	Population	Size in km²
Berlin	3,391,407	891.69 km²
Hannover	515,772	
Munich	1,259,677	310.46 km²
Stuttgart	595,452	207.458 km²

- Concepts store queries on pages which can be viewed as dynamic categories. Concepts are computationally created collections of pages.

```
{{#concept:  [[Category:Event]] [[start date::> Jan 1 2008]] [[start date::< Dec 31 2008]]
| Events in the year 2008 that have been announced on semanticweb.org.
  To add more events, go to the page "Events" on semanticweb.org.
}}
```

- The Special:Ask page uses a query and additional options to display information in a structured, however not persistent manner.

Query

```
[[Located in::Germany]]
```

Additional printouts (optional)

```
?Category
```

[Add sorting condition]

Find results | Hide query | Querying help

Previous **Results 1– 5** Next (**20** | 50 | 100 | 250 | 500)

	Category
Baden-Württemberg	Category:Sample pages
Berlin	Category:City Category:Sample pages
Hannover	Category:City Category:Sample pages
Munich	Category:City Category:Sample pages
Stuttgart	Category:City Category:Sample pages

Previous **Results 1– 5** Next (**20** | 50 | 100 | 250 | 500)

Web Service Modeling Toolkit

WSMT Scope and Functionality

- The WSMT is an Integrated Development Environment (IDE) for the development of Semantic Web Services
- Aims to support the engineer through the Software Development Cycle (SDC) of Semantic Web Services
  - Improve Engineer Productivity
  - Aid in adoption of WSMO, WSML, SEE
  - High quality tools
  - Eclipse based
- Development of WSMO Semantic Descriptions through WSML
  - Ontologies
  - Goals
  - Web Services
  - Mediators
- Interfacing with Semantic Execution Environments
  - WSMX
  - IRSIII
- Creation of Mediation Mappings between Ontologies
  - Abstract Mapping Language (AML)

WSMT Perspective

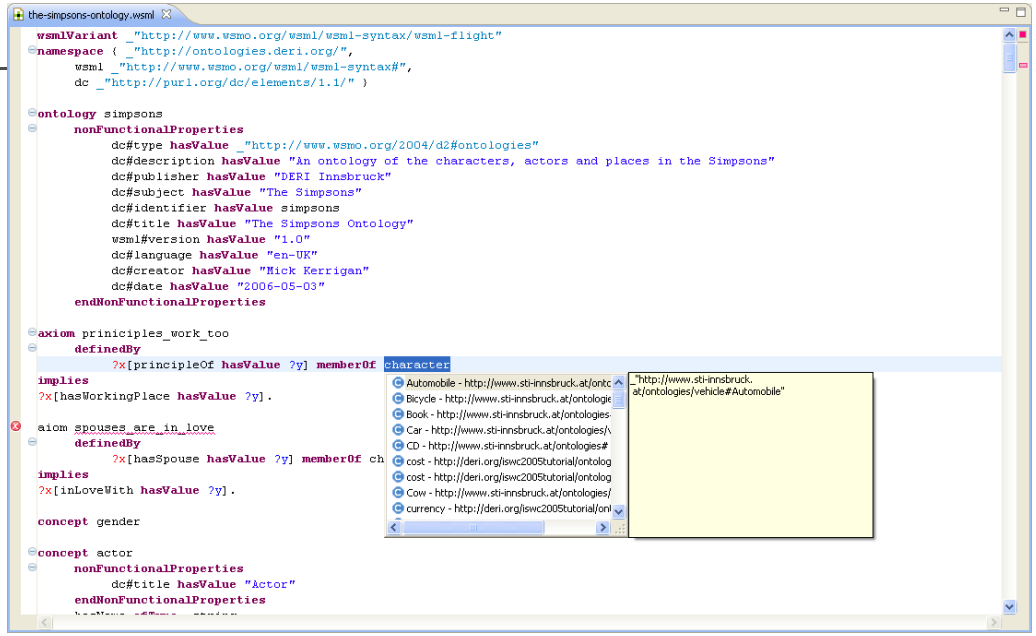
STI · INNSBRUCK

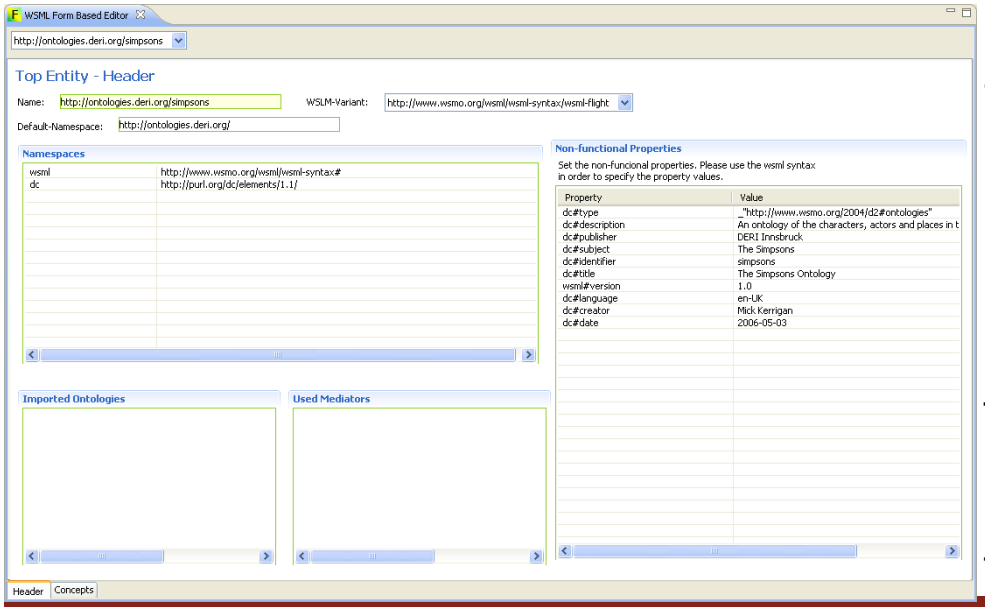
- Semantic Execution Environments need Ontologies, Goals, Web Services, and Mediators in order to function
- Provide support to the engineer in creating these descriptions
- Provide mechanisms for browsing semantic descriptions to aid in developer understanding
- Abstract the developer from the underlying syntax
- Assist in the validation and testing of semantic descriptions

WSMT Text Editor and Form Based Editor

STI · INNSBRUCK

- Abstracting from syntax is good but…
- Existing developers familiar with the syntax
- Certain tasks are just easier with a textual representation
- WSML Human Readable Syntax is designed to be light
- Must support the more experienced developer

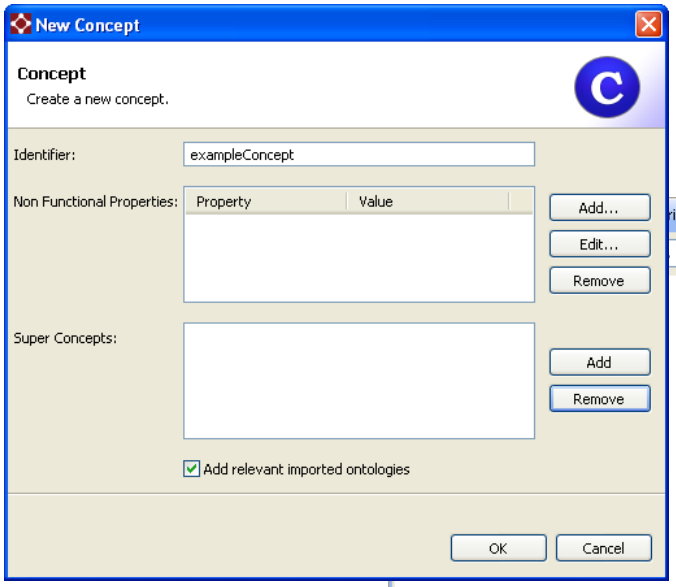


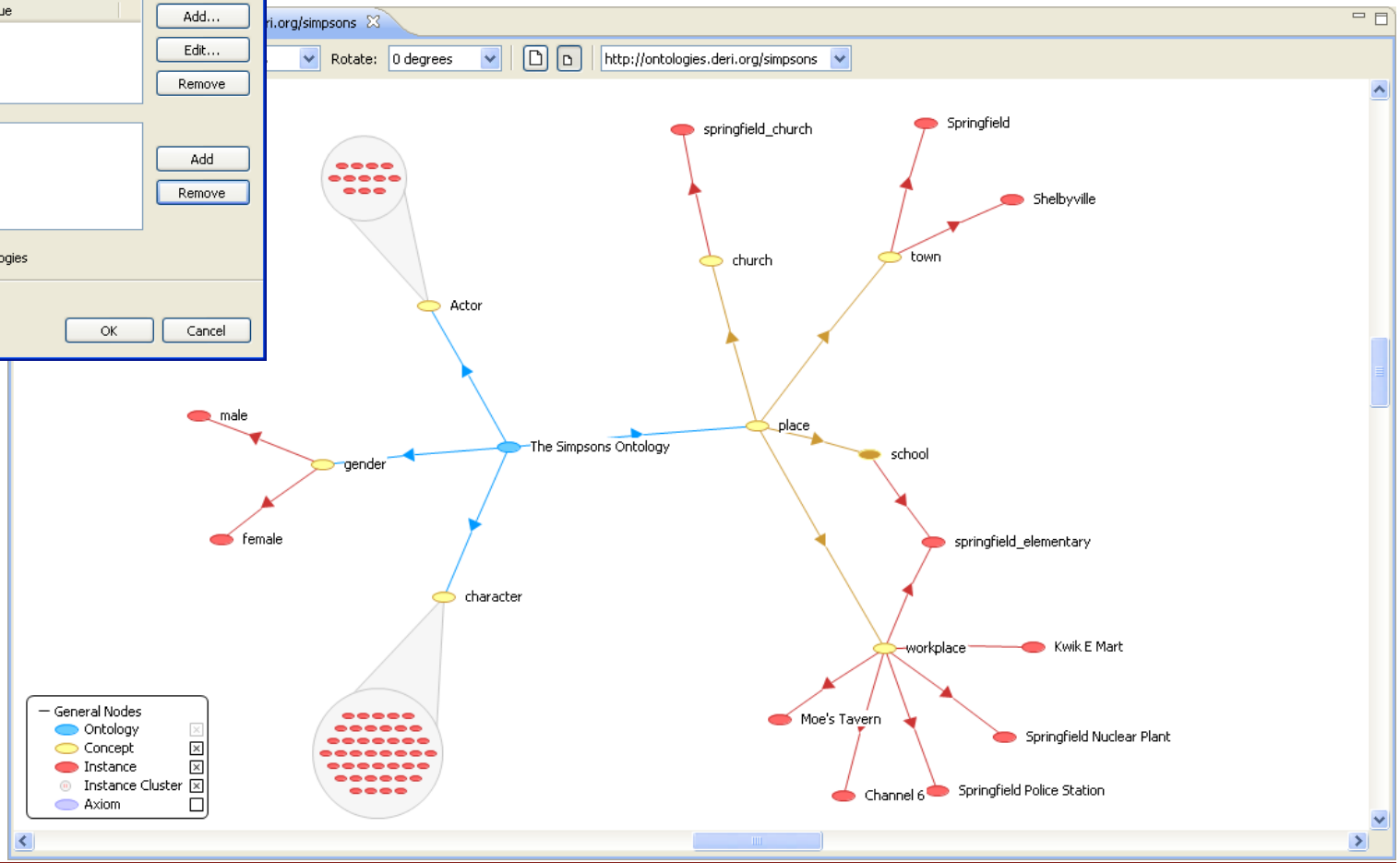


- Abstracts developers from the WSML syntax allowing them to focus on the modeling task at hand
  - Improved Developer focus
  - Reduced Errors in semantic descriptions
  - Less keystrokes improves speed of creation
- Descriptions are broken up into tabs to keep the forms small
- Forms consist of Text fields, combo boxes and tables

WSMT Entity creation and WSMT Navigator

- Entity creation
  - WSMT Navigator





WSMT Interoperability and Validation

- WSML ontologies can be transformed to OWL and RDFS to enable Web compliance
- WSMT can perform this transformation within the WSML Navigator
- Existing RDFS or OWL files can be transformed into a WSML representation
- WSML can be transformed in the RDFS or OWL depending on the WSML variant

WSMT Interoperability and Validation

STI · INNSBRUCK

- WSMO4J parser used to validate syntax
- WSMO4J validator used to validate semantics
  - Ensures features within the semantic description match that of the specified WSML Variant (Errors)
  - Checks for unrecommended usage of WSML Features (Warnings)
- Discovery engine used to validate Web Services and Goals
  - Compliance to structure of discovery approaches
- All files automatically checked as they are changed
- Immediate feedback to the user in each editor

WSMT-Testing

STI · INNSBRUCK

- Testing software usually involves deploying it and ensuring that it functions as expected
- Involves a costly Deploy-Test-Redeploy cycle
- Support within an IDE for testing software in its natural habitat can vastly reduce this iterative process
  - Reduces the cost of development
  - Improves developer productivity
  - Reduced developers involvement in tedious tasks
- Correctness of a semantic description is more than just having a valid description

WSMT Testing Ontologies, Web Services and Goals

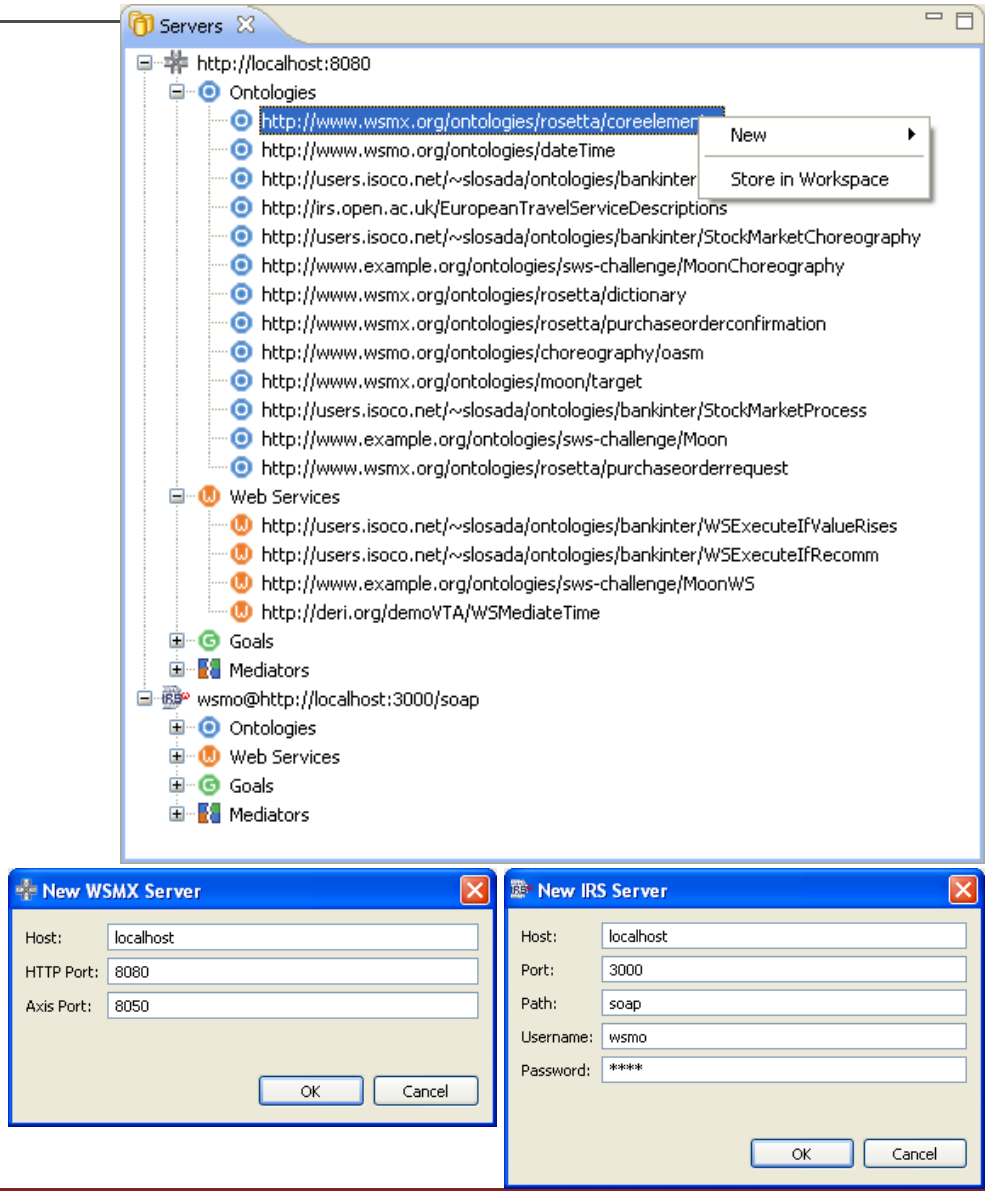
- Ontologies underlie every other semantic description in WSML
- The developer needs to be sure that each ontology behaves as expected when used in a reasoner
  - Is the ontology consistent?
  - Does is answer queries in the manner expected?
- Access to reasoners for each of the WSML Variants is thus required within the WSMT
- Allow users to perform reasoning operations over the ontology currently being edited

WSMT Testing Ontologies, Web Services and Goals

- A Semantic Web Service that does not match the Goals it is expected to match could result in the loss of a lot of money
- Developers need to ensure that the Web Service descriptions that create match Goals as expected
- Tool support reduces the number of interactions with a testing SEE
- Quite likely that provider will issue sample Goals with their Web Service descriptions.
- Ensuring your Web Service descriptions are found by your competitors sample Goals could provide a competitive advantage.

WSMT-Interfacing with a SEE

- In order for a SEE to correctly function the necessary Ontologies, Goals, Web Services and Mediators need to be available to it
- Manually deploying descriptions to a SEE or manually retrieving them in order perform maintenance is a tiresome and lengthy process
- Automated tools for interfacing with the Web Services exposed by a SEE enable these actions to be reduced to one or two clicks of a mouse.
- The SEE perspective contains all the functionality necessary to deliver this tool support to the developer



EXTENSIONS

Ontology editors (Extensions)

STI · INNSBRUCK

- Protege (today) http://protege.stanford.edu
- Neon Toolkit: www.**neon-toolkit**.org
- myOntology: www.myontology.org
- Semantic Media Wiki
  - HALO extension http://www.mediawiki.org/wiki/Extension:Halo_Extension
  - Ontology editor extension http://smw-active.sti-innsbruck.at
- DOGMA Modeler http://starlab.vub.ac.be/website/node/47
- OntoStudio http://www.ontoprise.de/
- TopBraid Composer http://www.topbraidcomposer.com/

Ontology alignment (Extensions)

STI · INNSBRUCK

- Various algorithms and methods
- AUTOMS
- FOAM
- ... Many more
- Alignment API http://alignapi.gforge.inria.fr/
  - Allows using various algorithms for alignment
- http://ontologymatching.org

Reasoners (Extensions)

- AllegroGraph http://agraph.franz.com/
- Fact http://www.cs.man.ac.uk/%7Ehorrocks/FaCT/
- Pellet http://clarkparsia.com/pellet
- Racer http://www.racer-systems.com/
- IRIS http://www.sti-innsbruck.at/
- OWLIM http://http//ontotext.com/owlim/
- KAON http://kaon2.semanticweb.org/

Storage (Extensions)

STI · INNSBRUCK

- OWLIM http://http//ontotext.com/owlim/
- Sesame http://openrdf.org/
- YARS http://sw.deri.org/2004/06/yars/
- Allegrograph http://agraph.franz.com/
- Jena http://jena.sourceforge.net/
- Virtuoso http://virtuoso.openlinksw.com/
- Redland http://librdf.org/

SUMMARY

Summary

STI · INNSBRUCK

- Tools addressing different areas of semantic technologies:
  - Ontology editors
  - Ontology alignment
  - Semantic wikis (see lecture on Social Semantic Web)
  - Games for semantic content creation (see lecture on Social Semantic Web)
  - Reasoners
  - Storage
  - Semantic annotation tools for different types of content (see lecture on semantic annotation)
  - Semantic Web service development tools
- An up-to-date overview: http://semanticweb.org
- Today‘s selection:
  - Protege
  - Semantic MediaWiki
  - WSMT

References

STI · INNSBRUCK

- http://semanticweb.org
- Protege (today) http://protege.stanford.edu
- Neon Toolkit: www.**neon-toolkit**.org
- myOntology: www.myontology.org
- Semantic Media Wiki
  - HALO extension http://www.mediawiki.org/wiki/Extension:Halo_Extension
  - Ontology editor extension http://smw-active.sti-innsbruck.at
- DOGMA Modeler http://starlab.vub.ac.be/website/node/47
- OntoStudio http://www.ontoprise.de/
- TopBraid Composer http://www.topbraidcomposer.com/
- http://ontologymatching.org
-

References (cont'd)

STI · INNSBRUCK

- AllegroGraph http://agraph.franz.com/
- Fact http://www.cs.man.ac.uk/%7Ehorrocks/FaCT/
- Pellet http://clarkparsia.com/pellet
- Racer http://www.racer-systems.com/
- IRIS http://www.sti-innsbruck.at/
- OWLIM http://http//ontotext.com/owlim/
- KAON http://kaon2.semanticweb.org/
- OWLIM http://http//ontotext.com/owlim/
- Sesame http://openrdf.org/
- YARS http://sw.deri.org/2004/06/yars/
- Allegrograph http://agraph.franz.com/
- Jena http://jena.sourceforge.net/
- Virtuoso http://virtuoso.openlinksw.com/
- Redland http://librdf.org/

Next Lecture

STI · INNSBRUCK

#	Date	Title
1	12.10,2009	Introduction
2	12.10,2009	Semantic Web architecture
3	09.11.2009	RDF and RDFs
4	09.11.2009	Web of hypertext (RDFa, Microformats) and Web of data
5	23.11.2009	Semantic annotations
6	23.11.2009	Repositories and SPARQL
7	07.12.2009	OWL
8	07.12.2009	RIF
9	11.01.2010	Web-scale reasoning
10	11.01.2010	Social Semantic Web
11	11.01.2010	Ontologies and the Semantic Web
12	25.01.2010	SWS
13	25.01.2010	Tools
14	25.01.2010	Applications
15	**TBD**	**Exam**

Questions?

STI · INNSBRUCK