Semantic Web

Tools

Semantic crawler: Swoogle

Slides based on http://swoogle.umbc.edu/

About crawlers

- Also known as a Web spider or Web robot.
- Other less frequently used names for Web crawlers are ants, automatic indexers, bots, and worms.
- “ A program or automated script which browses the World Wide Web in a methodical, automated manner ” (Kobayashi and Takeda, 2000).
- The process or program used by search engines to download pages from the web for later processing by a search engine that will index the downloaded pages to provide fast searches.
- In concept a semantic web crawler differs from a traditional web crawler in only two regards: the format of the source material it is traversing, and the means of specifying links between information resources.

Crawlers

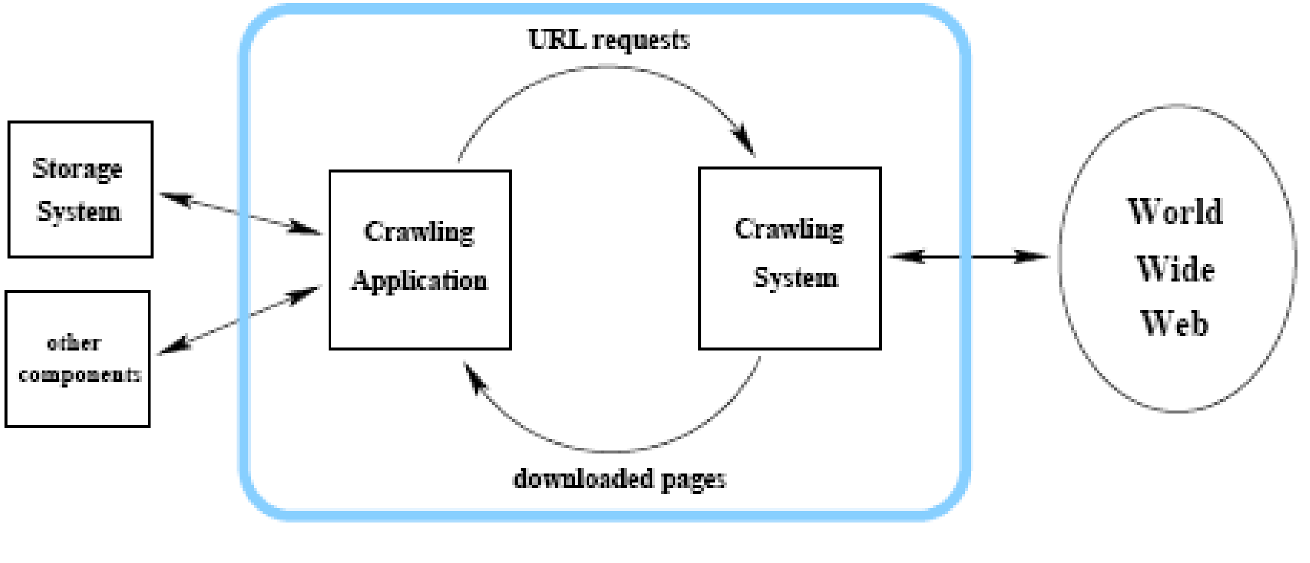
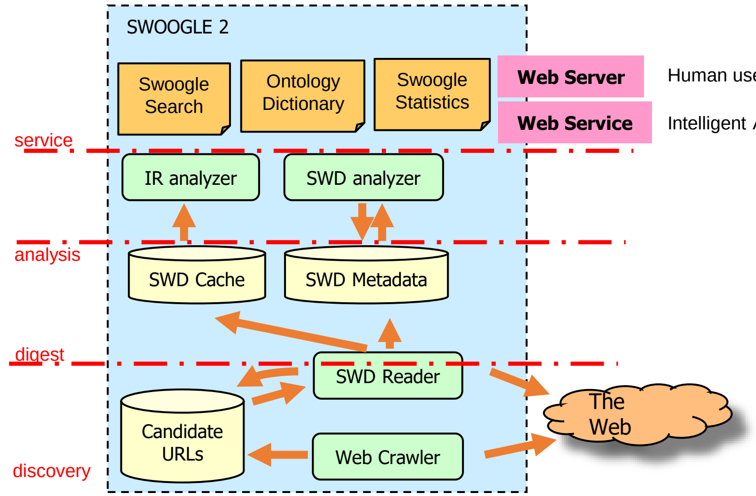


Figure 1: Basic two components of the crawler

Crawlers: Swoogle

search and metadata for the semantic web



Swoogle uses four kinds of crawlers to discover semantic web documents and several analysis agents to compute metadata and relations among documents and ontologies. Metadata is stored in a relational DBMS. Services are provided to people and agents.

http://swoogle.umbc.edu/

Swoogle provides services to people via a web interface and to agents as web services.

SWDs	336,000	**Classes**	95,000
Triples	47,000,000	**Properties**	53,000
Ontologies	4,200	**Individuals**	7,200,000

Swoogle concepts

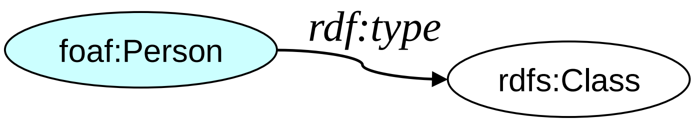
- Document
  - A Semantic Web Document (SWD) is an online document written in semantic web languages (i.e. RDF and OWL).

In swoogle, a document D is a valid SWD iff. JENA* correctly parses D and produces at least one triple.

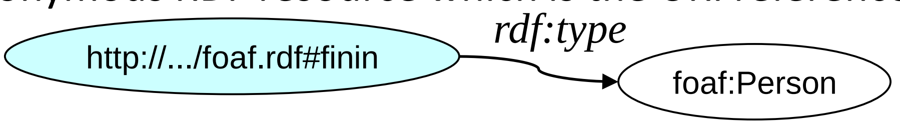
*JENA is a Java framework for writing Semantic Web applications. http://www.hpl.hp.com/semweb/jena2.htm

  - An ontology document (SWO) is a SWD that contains mostly term definition (i.e. classes and properties). It corresponds to T-Box in Description Logic.
  - An instance document (SWI or SWDB) is a SWD that contains mostly class individuals. It corresponds to A-Box in Description Logic.

- Term
  - A term is a non-anonymous RDF resource which is the URI reference of either a class or a property.



- Individual
  - An individual refers to a non-anonymous RDF resource which is the URI reference of a class member.



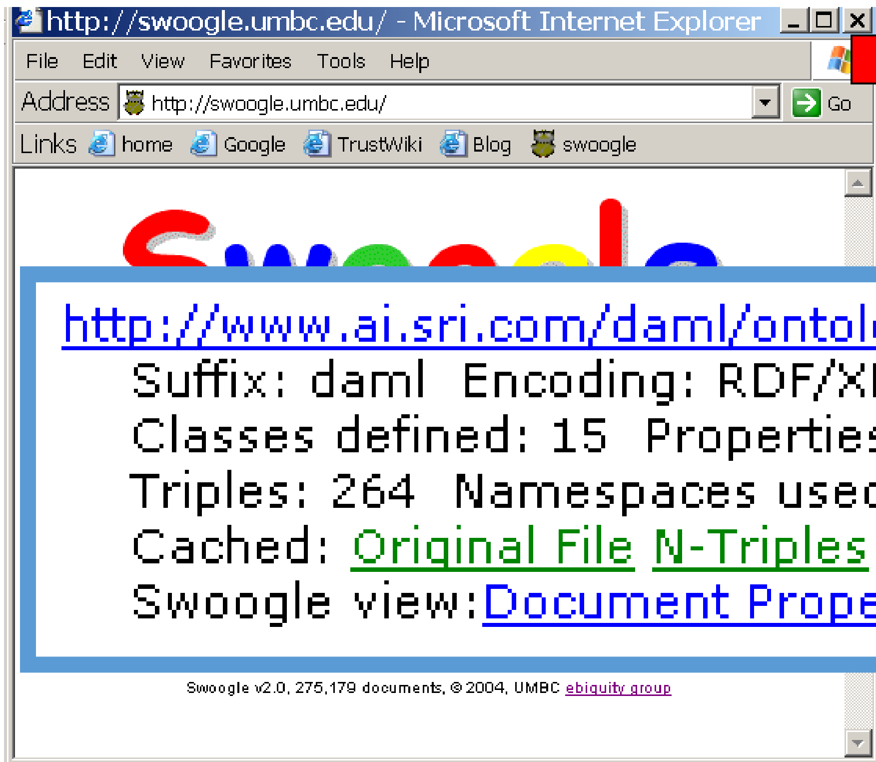
Example

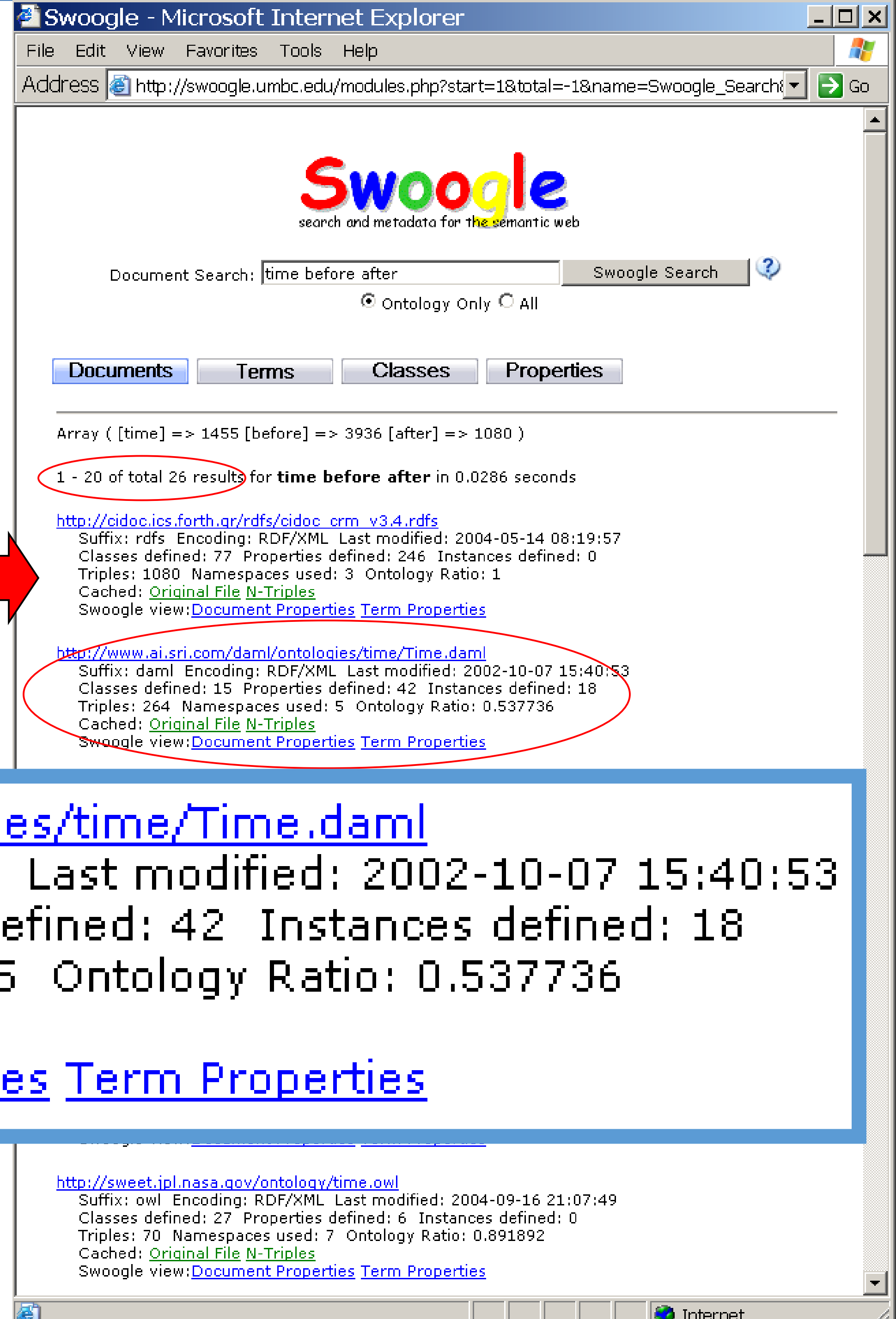
1. Find “Time” Ontology (Swoogle Search)
2. Digest “Time” Ontology
   - Document view
   - Term view
3. Find Term “Person” (Ontology Dictionary)
4. Digest Term “Person”
   - Class properties
   - (Instance) properties
5. Swoogle Statistics

Find “Time” Ontology

We can use a set of keywords to search ontology. For example, “time, before, after” are basic concepts for a “Time” ontology.

1 - 20 of total 26 results





http://www.ai.sri.com/daml/ontologies/time/Time.daml
Suffix: daml Encoding: RDF/XML Last modified: 2002-10-07 15:40:53
Classes defined: 15 Properties defined: 42 Instances defined: 18
Triples: 264 Namespaces used: 5 Ontology Ratio: 0.537736
Cached: Original File N-Triples
Swoogle view: Document Properties Term Properties

Demo 4

Digest Term "Person"

Term Definition:
http://xmlns.com/foaf/0.1/Person

- Related Terms (same namespace)
- Related Terms (same local name)

Class Properties (Fused) | Instance Properties (from ontology) | Instance Properties (from instances)

Fused Definition of http://xmlns.com/foaf/0.1/Person

It is defined as **class** by 15 SWDs with 8 predicates

#	Predicate-Value	Related Ontology
1	http://www.w3.org/2003/06/sw-vocab-status/ns#term_status	
	testing	6 source
2	owl:disjointWith	
	foaf:Organization	6 source
	foaf:Document	5 source
	foaf:Project	5 source

Instance Properties (from ontologies) of http://xmlns.com/foaf/0.1/Person

By analyzing ontologies, this class is rdfs:domain of 167 different properties

#	Prope...		Related Ontology
1	http://...		6 source
			2 source
2	http://...		6 source
		rdfs:Resource	2 source
3	http://xmlns.com/foaf/0.1/knows	foaf:Person	8 source

167 different properties

Instance Properties (from instances) of http://xmlns.com/foaf/0.1/Person

By analyzing the instances, this class is domain of 562 different properties

#	property			
1	foaf:mbox_sha...			...1daf964b378ed9caf1
2	foaf:nick			
3	foaf:weblog			...ipop/
4	foaf:name	33090 sources	4	????
5	rdfs:seeAlso	31905 sources	329	http://dac.lolipop.jp/bookmarks/foaf.rdf
6	foaf:knows	29957 sources	3187	179dce4:fe896e1b77:-7fe3

562 different properties

Ontology editor: Protégé/Collaborative Protégé

Ontology editors

- Ontology editors provide an environment to build ontologies.
- As we heard in the lecture on ontologies, there are various ways of building ontologies (i.e. collaborative – community-driven, heavweight – lightweight ontologies, etc.).
- Different tools might be suitable for different purposes.
- Sometimes tools impose an ontology building methodology.
- Today:
  - Protégé
  - Collaborative Protégé
  - Also in annotation: Semantic MediaWiki

Protégé-Facts

- 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

- 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:

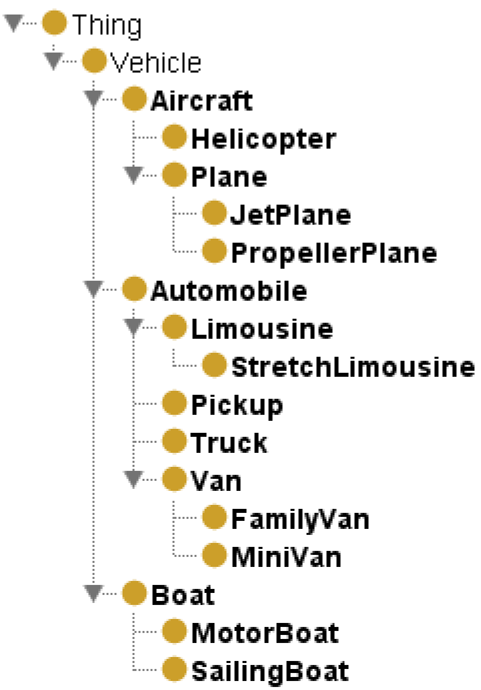
- Frame based via the Protégé-Frames editor
- In OWL via the Protégé-OWL editor

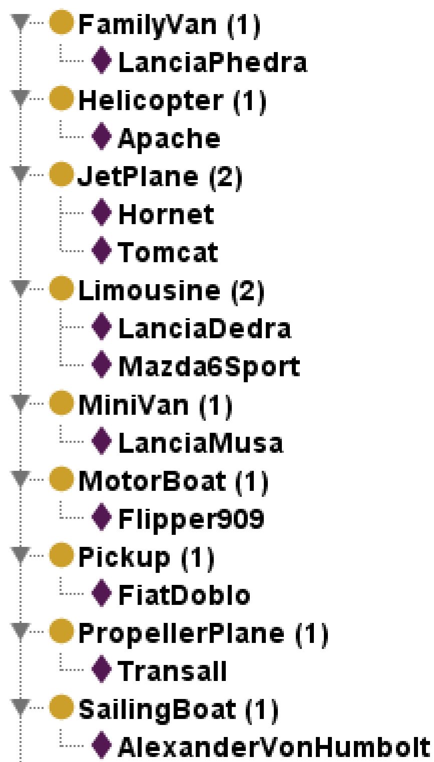
Protégé Frame-based editor

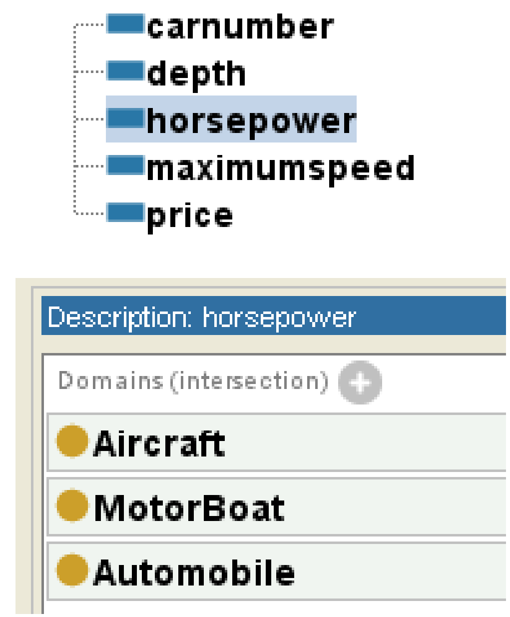
- Construction and population of ontologies that are frame-based.
- Conformant to OKBC (Open Knowledge Base Connectivity Protocol).
  - An ontology is a set of classes.
  - These are structured in a subsumption hierarchy.
  - To each class a set of slots to express properties and relationships is assigned.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

Protégé-Frame-based editor

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







Protégé OWL editor

- 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.

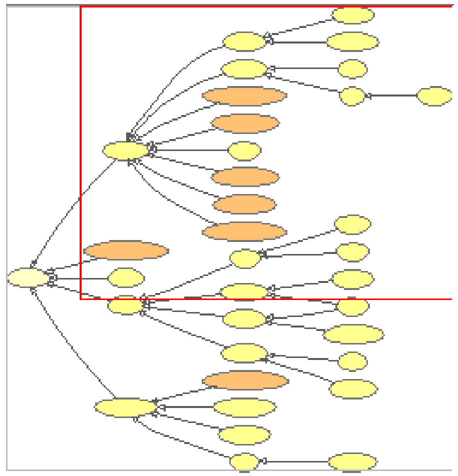
Protégé-OWL editor

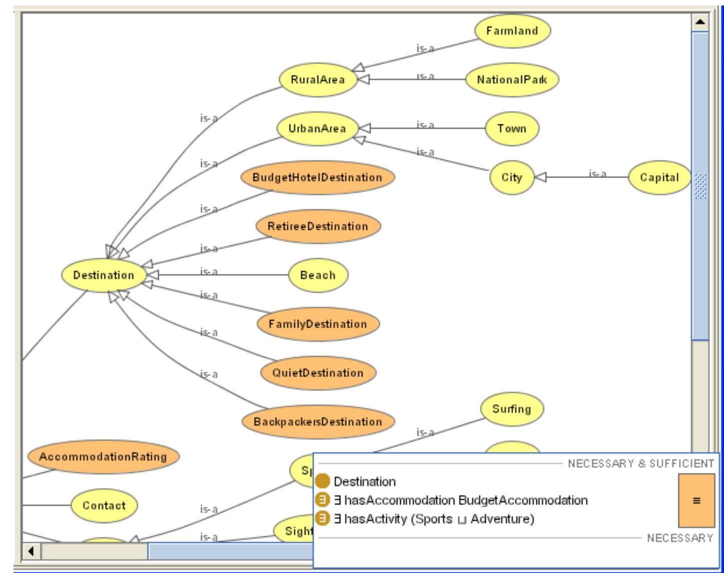
The Protégé-OWL editor enables users to:

- 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

- 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 rules.

Collaborative Protégé

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

Collaborative Protégé

Multi-user mode:

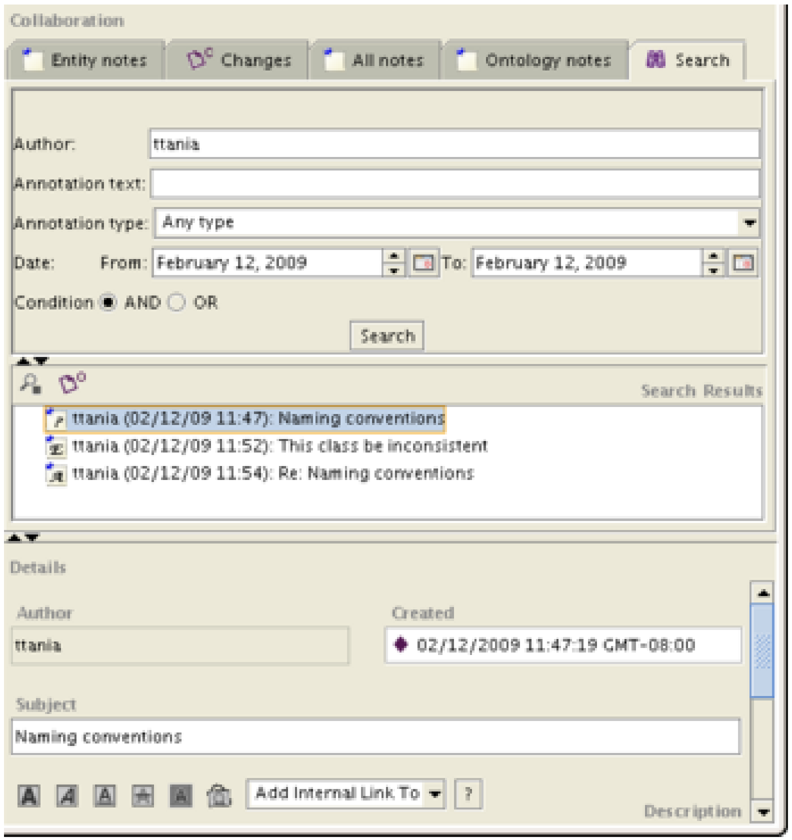
- Ontology is hosted on server.
- Multiple clients can edit ontology simultaneously.
- Changes introduced by one client become visible to the others immediately.
- Preferred mode Collaborative Protégé should be run in.

Standalone mode:

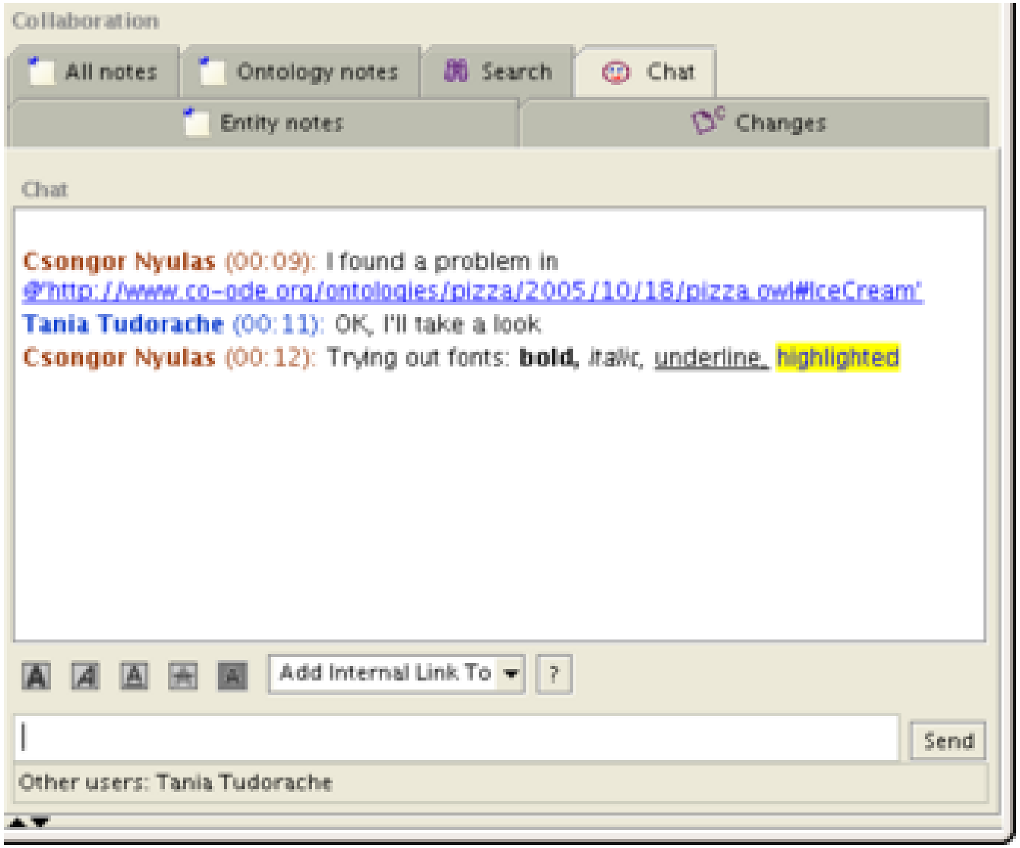
- Multiple users access one ontology in succession.
- Ontologies are stored on a shared drive.
- Users access the same project files.
- Parallel access is not possible.

Collaborative Protégé con't

- Searching notes from other users based on certain criteria.



- Chating with other users while working on one ontology.



Annotation:
Semantic Media Wiki

Slides based on presentation by Völkl et al., University Karlsruhe

Semantic Annotation

- Linking content to ontologies in order to make data machine-understandable and allow machines to interpret data.
- Different ways of annotation:
  - Manual
  - Semi-automatic (usually with training sets)
  - Automatic
- Manual approach: Semantic MediaWiki (annotation embedded in the workflow of content creation)
- Automatic approach: KIM (large knowledge base in the background is matched to content)

Semantic Media Wiki Facts

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 Benefits

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

- 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 Editing

- 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]]}}

Editing Category:Limousine

Warning: You are not logged in. Your IP address will be recorded in this page's edit history.

```
A Limousine is a nice type of an Automobile.
[[Category:Automobile]]
```

Editing LanciaDedra

Warning: You are not logged in. Your IP address will be recorded in this page's edit history.

```
A LanciDedra is a Limousine<br>
max speed:[[speed::200|200kmh]]
length:[[length::4|4 meters]]
hight:[[hight::150|150 centimeters]]
[[Category:Limousine]]
```

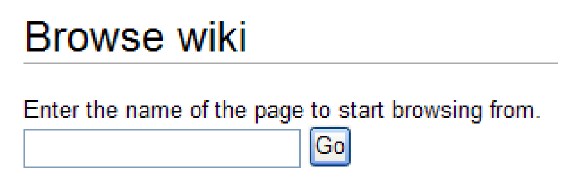
Editing Concept:BundeslandOesterreich

Warning: You are not logged in. Your IP address will be recorded in this page's edit history.

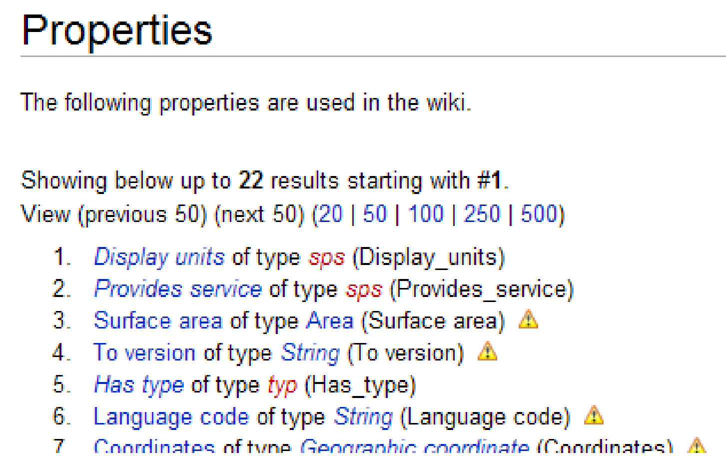
```
{{#concept:  [[Vorarlberg||Tirol||Salzburg||Kaernten
||Oberoesterreich||Niederoesterreich||Wien||Burgenland
||Steiermark]]| Mein Text}}
```

Semantic Media Wiki Browsing

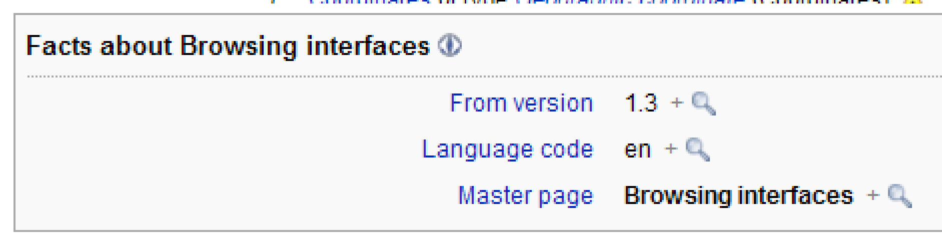
- Semantic browsing via Special:Browse interface.



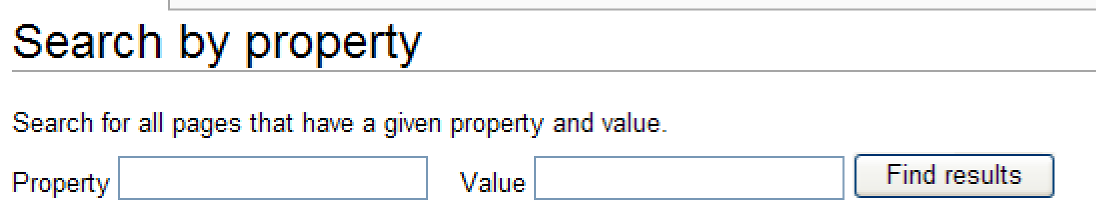
- Viewing all properties, types and values via Special:Properties (not only for properties but many more).



- The factbox summarizes the semantic data of each page.



- Simple search interfaces for different types of searches.



Semantic Media Wiki Searching

- 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)

Repository and Reasoner: Sesame and OWLIM

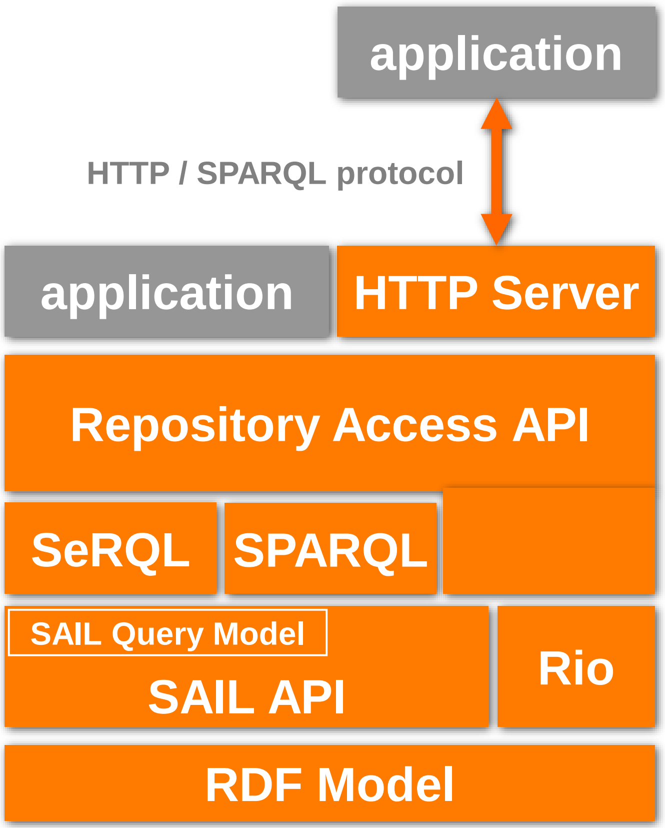
What is Sesame?

- A framework for *storage*, *querying* and *inferencing* of RDF and RDF Schema
- A Java Library for handling RDF
- A Database Server for (remote) access to *repositories* of RDF data

Sesame features

- Light-weight yet powerful Java API
- Highly expressive query and transformation languages
  - SeRQL, SPARQL
- High scalability (O(10^7) triples on desktop hardware)
- Various backends
  - Native Store
  - RDBMS (MySQL, Oracle 10, DB2, PostgreSQL)
  - main memory
- Reasoning support
  - RDF Schema reasoner
  - OWL DLP (OWLIM)
  - domain reasoning (custom rule engine)
- Transactional support
- Context support
- Rio Toolkit: parsers and writers for different RDF syntaxes:
  - RDF/XML, Turtle, N3, N-Triples

Sesame architecture



The SAIL API

- Storage And Inferencing Layer
- Abstraction from physical storage
  - allows other Sesame components to function on any type of store
  - can be used as a *wrapper* layer for a particular data source
- System Internal API
  - application developers typically do not use it directly

The Repository Access API

- A single Java object representation for a Sesame database, offering methods for
  - evaluating a query and retrieving the result
  - adding RDF data from local file, from the web, as a text string, etc.
  - adding/removing (sets of) RDF statements
  - starting/stopping transactions

Querying RDF

- RDF is a labeled, directed graph of semistructured data
  - no rigid schema
- An RDF query language needs to be able to address this:
  - graph path expressions
  - dealing with semistructured nature of RDF
  - flexible querying of both data and schema

SeRQL vs. SPARQL

- Both: expressive query and transformation language for RDF
  - SELECT and CONSTRUCT
  - optional path expressions
  - support for context/named graphs
- SeRQL (“circle”)
  - nested queries (IN, EXISTS operators)
  - user-friendly syntax (a matter of taste of course)
  - efficient Sesame implementation
- SPARQL (“sparkle”)
  - W3C Standard (in progress)
    - tool interoperability: Jena, Redland, 3Store, Sesame, …

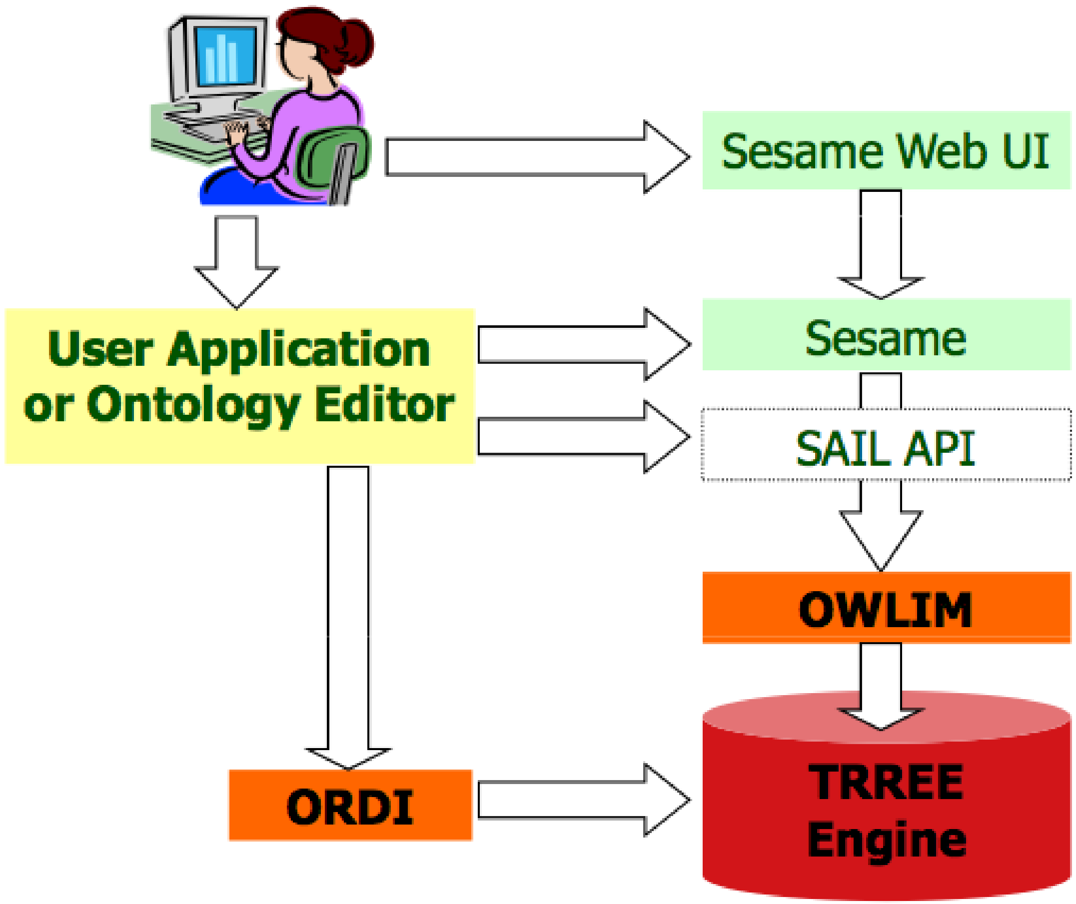
Reasoning for OWL

- OWLIM plugin support (by OntoText)
  - inductive, scalable reasoning over a pragmatic subset of OWL
- Custom reasoner
  - rule-based reasoner with user-defined rules
  - can be used to capture (part of) the semantics of OWL Lite / DL.

OWLIM

- OWLIM is a high-performance OWL repository
- Storage and Inference Layer (SAIL) for Sesame RDF database
- OWLIM performs OWL DLP reasoning
- It is uses the IRRE (Inductive Rule Reasoning Engine) for forward-chaining and “total materialization”
- In-memory reasoning and query evaluation
- OWLIM provides a reliable persistence, based on RDF N-Triples
- OWLIM can manage millions of statements on desktop hardware
- Extremely fast upload and query evaluation even for huge ontologies and knowledge bases

Overview – Sesame and OWLIM

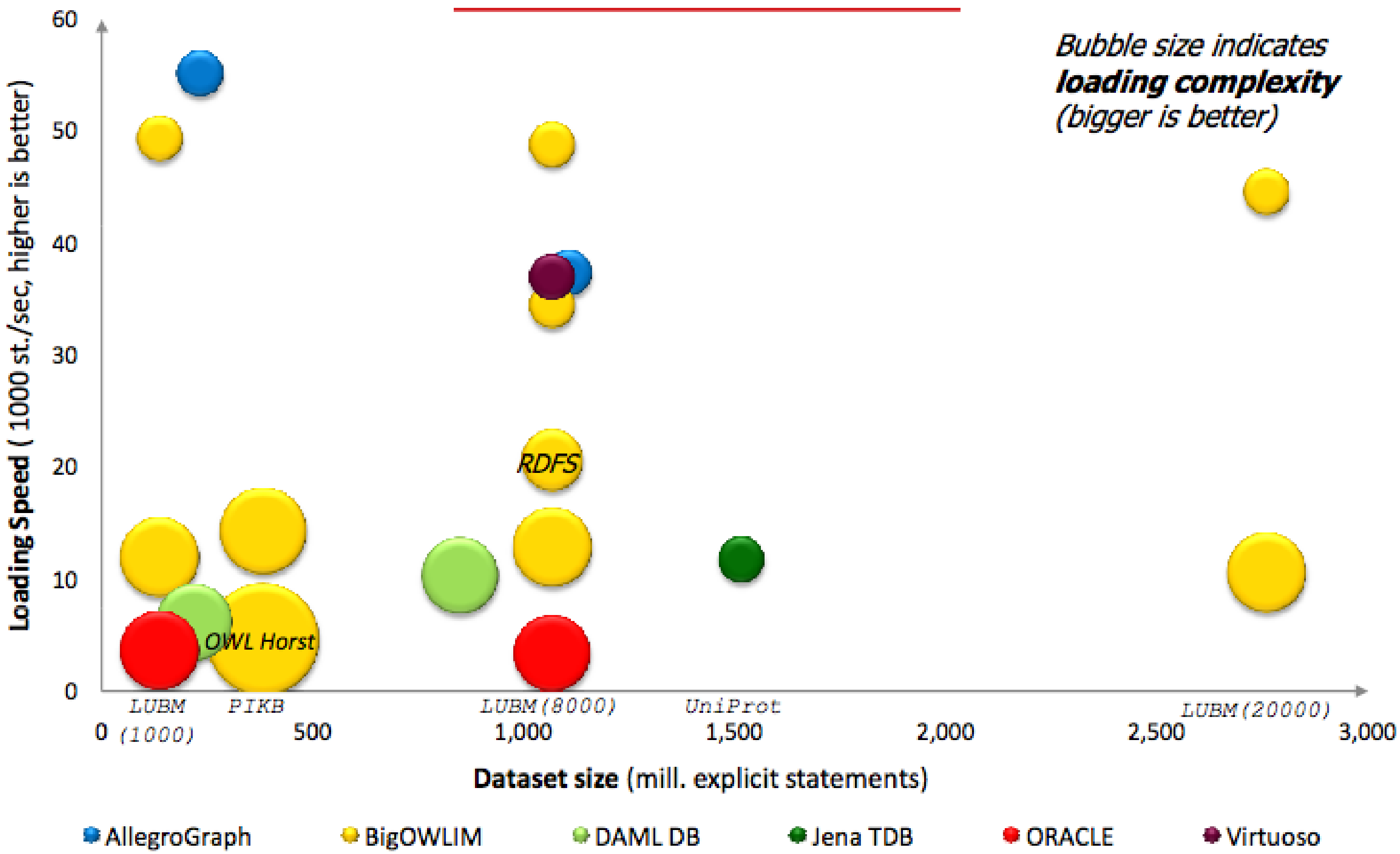


SwiftOWLIM and BigOWLIM

- 2 main species of OWLIM

	SwiftOWLIM	BigOWLIM
Scale (Mill. of explicit statem.)	10 MSt, using 1.6 GB RAM **100 MSt**, using 16 GB RAM	130 MSt, using 1.6GB **1068 MSt**, using 8GB
Processing speed (load+infer+store)	30 KSt/s on notebook **200 KSt/s** on server	5 KSt/s on notebook **60 KSt/s** on server
Query optimization	No	Yes
Persistence	Back-up in N-Triples	Binary data files and indices
Licence and Availability	Open-source under LGPL; Uses SwiftTRREE that is free, but not open-source	Commercial. Research and evaluation copies provided for free

Scalable inference map



EXTENSIONS

Ontology editors (Extensions)

- 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/

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)

- 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/

Questions?