Semantic Web Technologies II

SS 2009

22.04.2009

Engineering Ontologies and Semantic Applications

Dr. Sudhir Agarwal
Dr. Stephan Grimm
Dr. Peter Haase
PD Dr. Pascal Hitzler

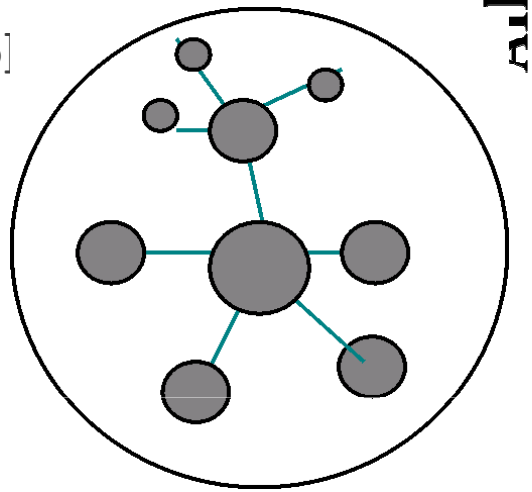
Denny Vrandecic

Topics

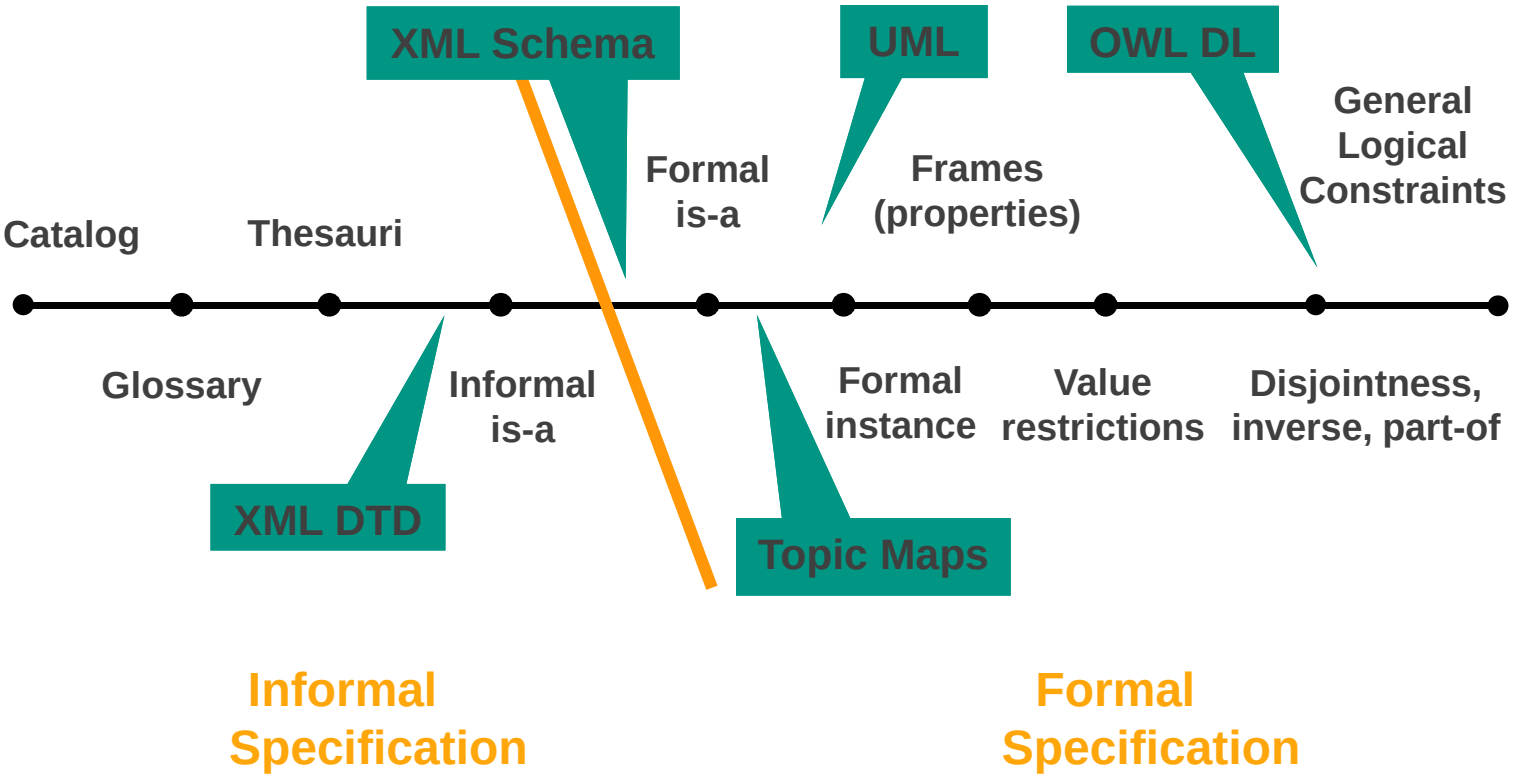
- What are ontologies?
- What are applications of ontologies in semantic applications?
- What are development processes and lifecycle activities for engineering
  1) ontologies
  2) semantic applications?
- What are supporting infrastructures/architectures for engineering ontologies and semantic applications

Ontologies - Definition

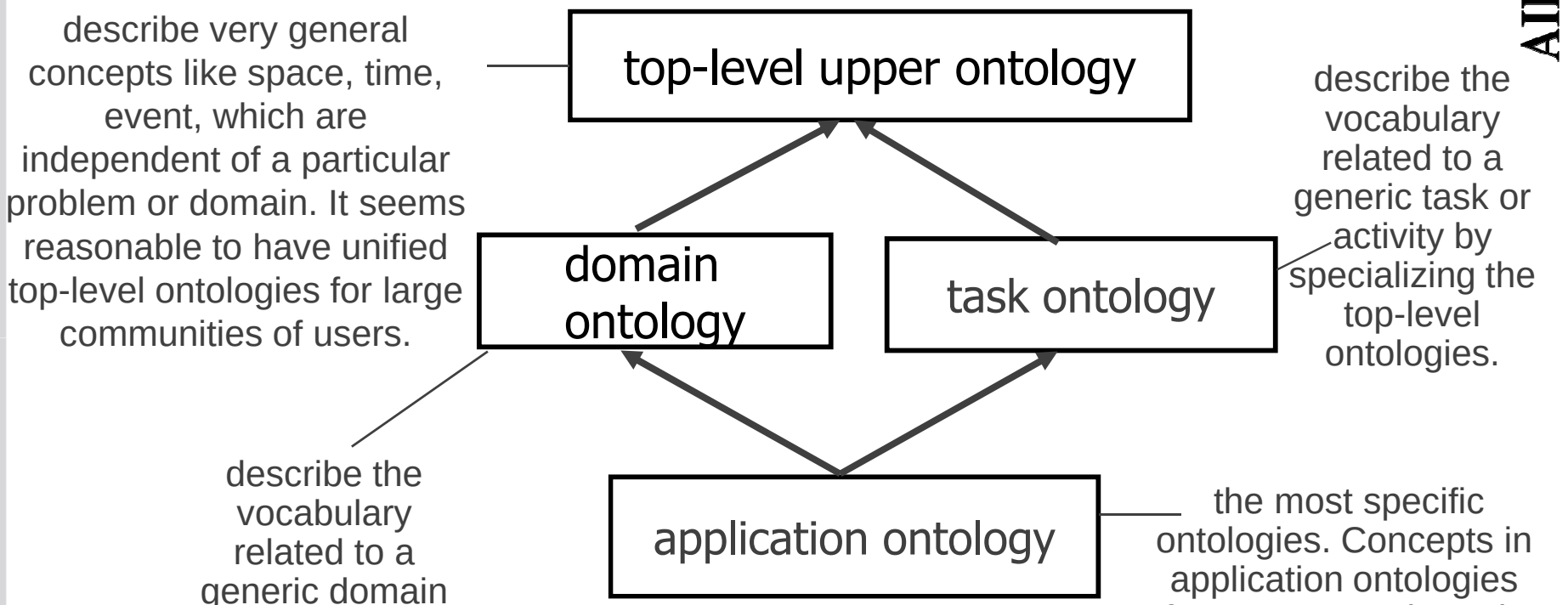
- "An ontology is a **shared understanding** of some **domain of interest**."[Uschold, Gruninger96]
- "An ontology is an **explicit specification** of a **shared conceptualisation**." [Gruber93]
- Ontologies
  - Describe a common vocabulary of terms and their relations
  - Define the meaning of this vocabulary
- Semantics based on logical languages guarantees well-defined interpretation
- Can be shared as resources



Varying levels of formality and expressivity



Coverage/Specify of Ontologies



[Guarino, 98]

Specific ontologies

- Domain-oriented
  - Domain-specific
    - Medicine => cardiology => rhythm disorders
  - Domain generalizations
    - components, organs, documents, gene function
- Task-oriented
  - task specific
    - configuration design, instruction, planning, annotation analysis
  - task generalizations
    - problem solving methods

Upper Ontologies

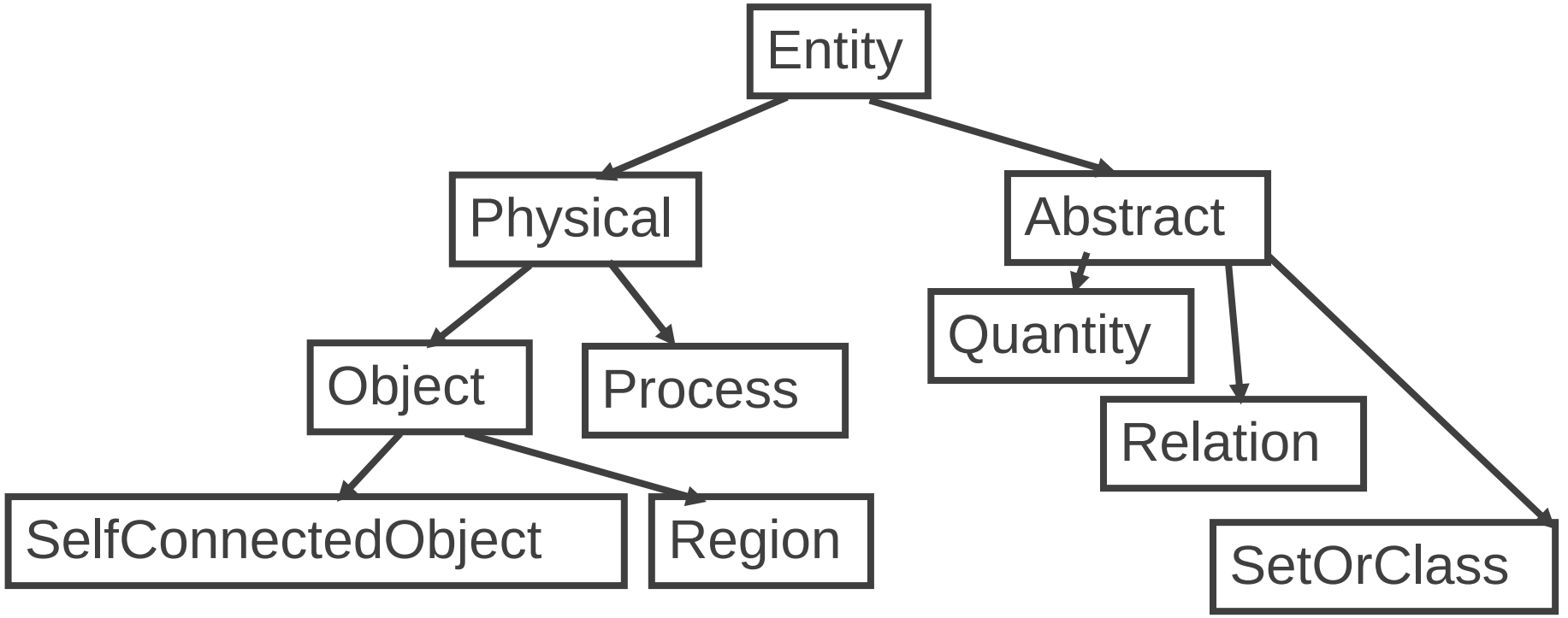
- A.k.a. top level, core, generic or reference
- An attempt to capture the most general and reusable terms and definitions
- “Physical”, “Abstract”, “Structure”, “Substance”
- Useful for ontology re-use
- Important when generating or analysing natural language expressions

- **Examples of Top Level ontologies**
- SUMO
- DOLCE
- CyC
- WordNet
- EuroWordNet

Suggested Upper Merged Ontology (SUMO)

- 1000 terms, 4000 axioms, 750 rules
- Written in FOL
- Development since 2000
- Associated domain ontologies totaling 20,000 terms and 70,000 axioms
- Free
  - SUMO is owned by IEEE but basically public domain
  - Domain ontologies are released under GNU
  - www.ontologyportal.org

SUMO – Top Level Concepts



Cyc

- enCYClopedia
- Douglas Lenat at Cycorp
- Development since 1984
- general knowledge and common-sense reasoning
- Ontology – 100,000's of terms
- Millions of assertions
  - "Water is wet"
  - "Everyone has a mother"
  - "When you let go of things they usually fall."
- Open version available – opencyc.com

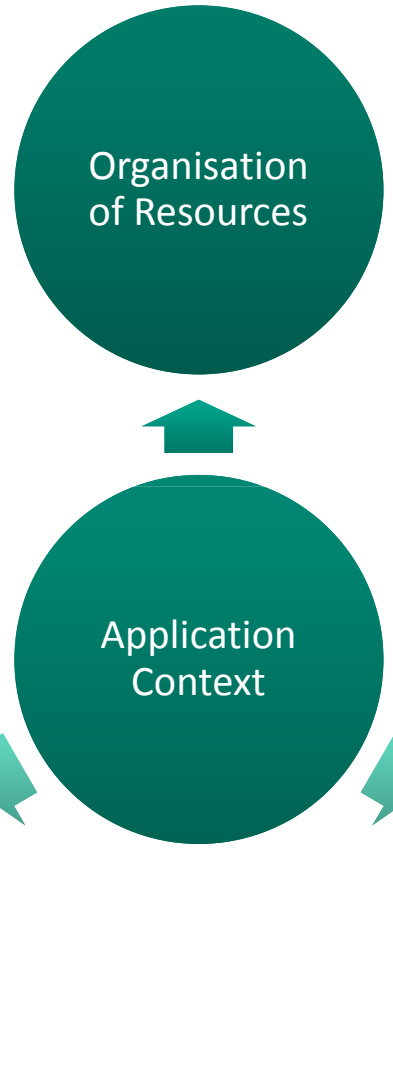
WordNet

- "Lexical ontology"
- 100,000 word senses – synsets
- Created by George Miller's group at Princeton
- Free
- De facto standard in the linguistics world

news item IS A KIND OF ...
1 sense of news item

Sense 1
news item -- (an item in a newspaper)
=> **item, point** -- (a distinct part that can be specified separately in a group of things that could be enumerated on a list; "he noticed an item in the New York Times"; "she had several items on her shopping list"; "the main point on the agenda was taken up first")
=> **part, portion, component part, component** -- (something determined in relation to something that includes it; "he wanted to feel a part of something bigger than himself"; "I read a portion of the manuscript"; "the smaller component is hard to reach")
=> **relation** -- (an abstraction belonging to or characteristic of two entities or parts together)
=> **abstraction** -- (a general concept formed by extracting common features from specific examples)

Application Context Drives Ontology Design



Scenarios: retrieval & browsing

Core paradigm: concept hierarchies

Formality: low

Scenarios: Model (meta) data, integrate (heterogeneous) metadata

Core paradigm: attributes/relations

Formality: medium-high

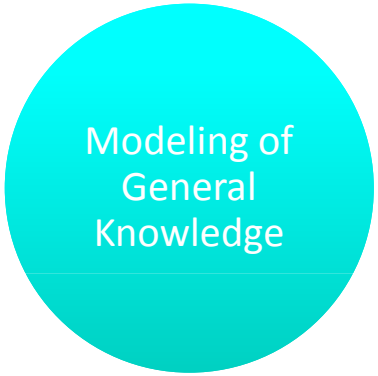
Scenarios: "Expert Systems"

Core paradigm: logical axiomatization

Formality: very high

Karlsruhe Institute of Technology

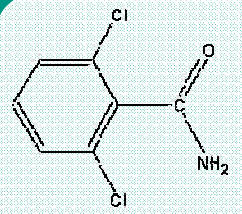
AIFB



APPLICATION SCENARIO: MODELING SCIENTIFIC KNOWLEDGE

Background

Goal:
A "Digital Aristotle" to solve scientific problems



Case study:
Chemistry Knowledge

PROJECT HALO

Project:
HALO – multi phase project funded by Vulcan Inc

Karlsruhe Institute of Technology

AIFB

Karlsruhe Institute of Technology

ontoprise
know how to use Know-how

Motivation – Competency Question

- *"Which compounds will produce a gas when HCl is added to the solid compound? HCl is a strong acid producing a yellow-green colored gas above the acid solution?"*

Background

- **The "Digital Aristotle" – a system that:**
  - Encompasses much of the world's scientific knowledge
  - Reasons over that knowledge
  - Answers novel scientific questions
  - Explains these answers
  - Is quite ambitous

- **The "Digital Aristotle" is a multi-stage effort:**
  - Start with a specific science (Chemistry)
  - Challenge with several teams
  - Answer AP-style questions
  - Refine existing system

Formalizing questions

- **Example**
- Which of the following compounds will produce a gas when HCl is added to the solid compound? HCl is a strong acid producing a yellow-green colored gas above the acid solution?
  - $Ba(OH)_2$ (s)
  - $CaCO_3$ (s)
  - $CuSO_4$ (s)
  - Na_3PO_4(s)
  - NaCl (s)

Formalized

```
(every QF1 has (context ((:pair "(a) Ba(OH)2(s)"
(a Reaction with (raw-material ((a HCl-Substance)
(a Ba_OH_2-Substance with (state ((a State-Value
with (value (*solid))))))))) (:pair "(b)
CaCO3(s)" (a Reaction with (raw-material ((a HCl-
Substance) (a CaCO3-Substance with (state ((a
State-Value with (value (*solid))))))))) (:pair
"(c) CuSO4(s)" (a Reaction with (raw-material ((a
HCl-Substance) (a CuSO4-Substance with (state ((a
State-Value with (value (*solid))))))))) (:pair
"(d) Na3PO4(s)" (a Reaction with (raw-material
((a HCl-Substance) (a Ionic-Compound-Substance
with (state ((a State-Value with (value
(*solid))))) (has-basic-structural-unit ((a
Ionic-Compound with (nested-atomic-chemical-
formula ((a Chemical-Formula with (term ((:seq
(:pair 3 Na) (:pair 1 P) (:pair 4
O))))))))))))) (:pair "(e) NaCl(s)" (a Reaction
with (raw-material ((a HCl-Substance) (a NaCl-
Substance with (state ((a State-Value with (value
*solid))))))))))) (output ((forall (the context
of Self) where (oneof2 (the result of (the2 of
It)) where ((the value of (the state of It2)) =
*gas)) (the1 of It) (comm [QF1-output-1]
Self)))))
```

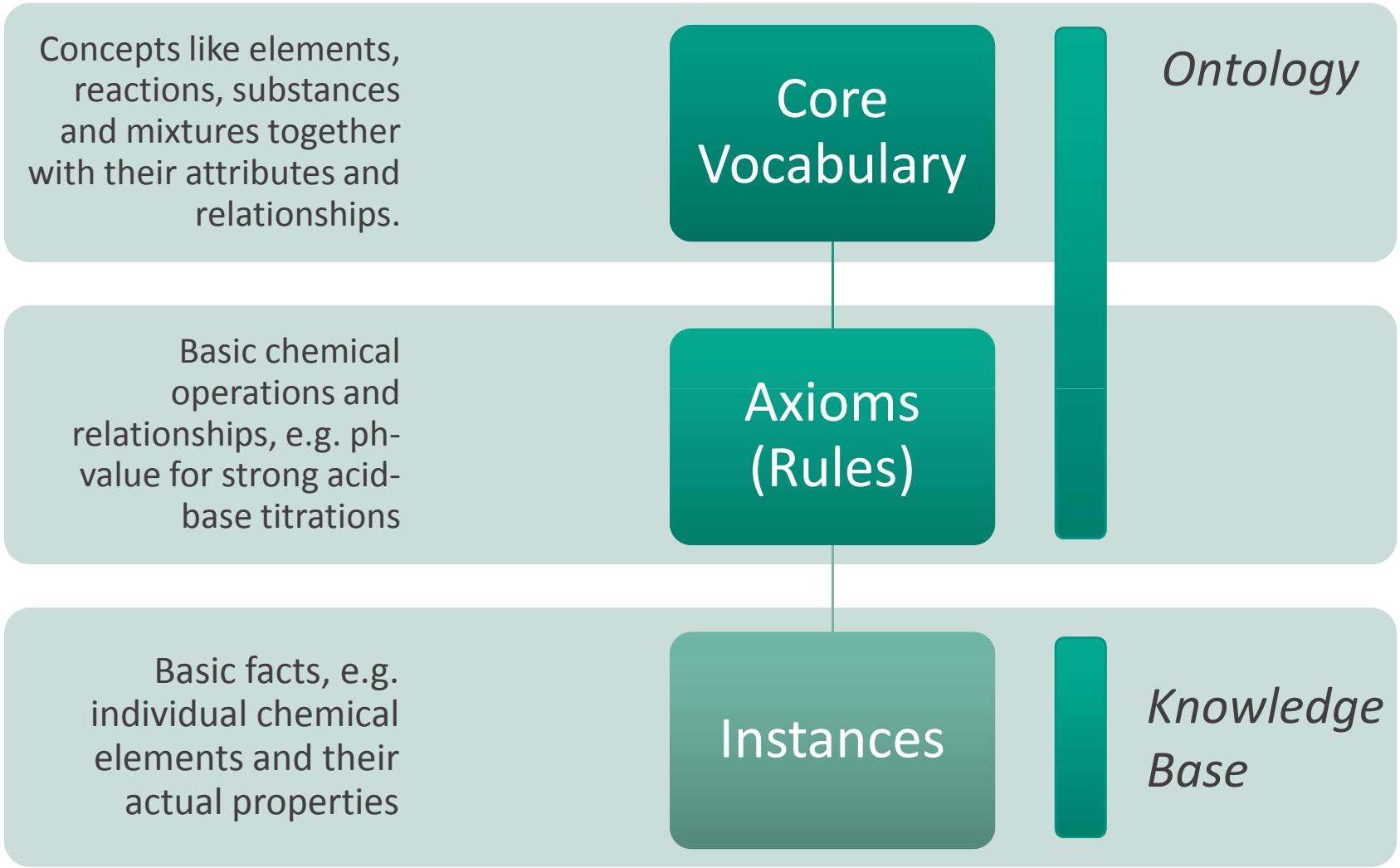
Background Knowledge

- Formalizing questions is “just” question understanding
- Needs (loads of) background knowledge (ontology)
- Needs a reasoning engine to answer the question using the ontology

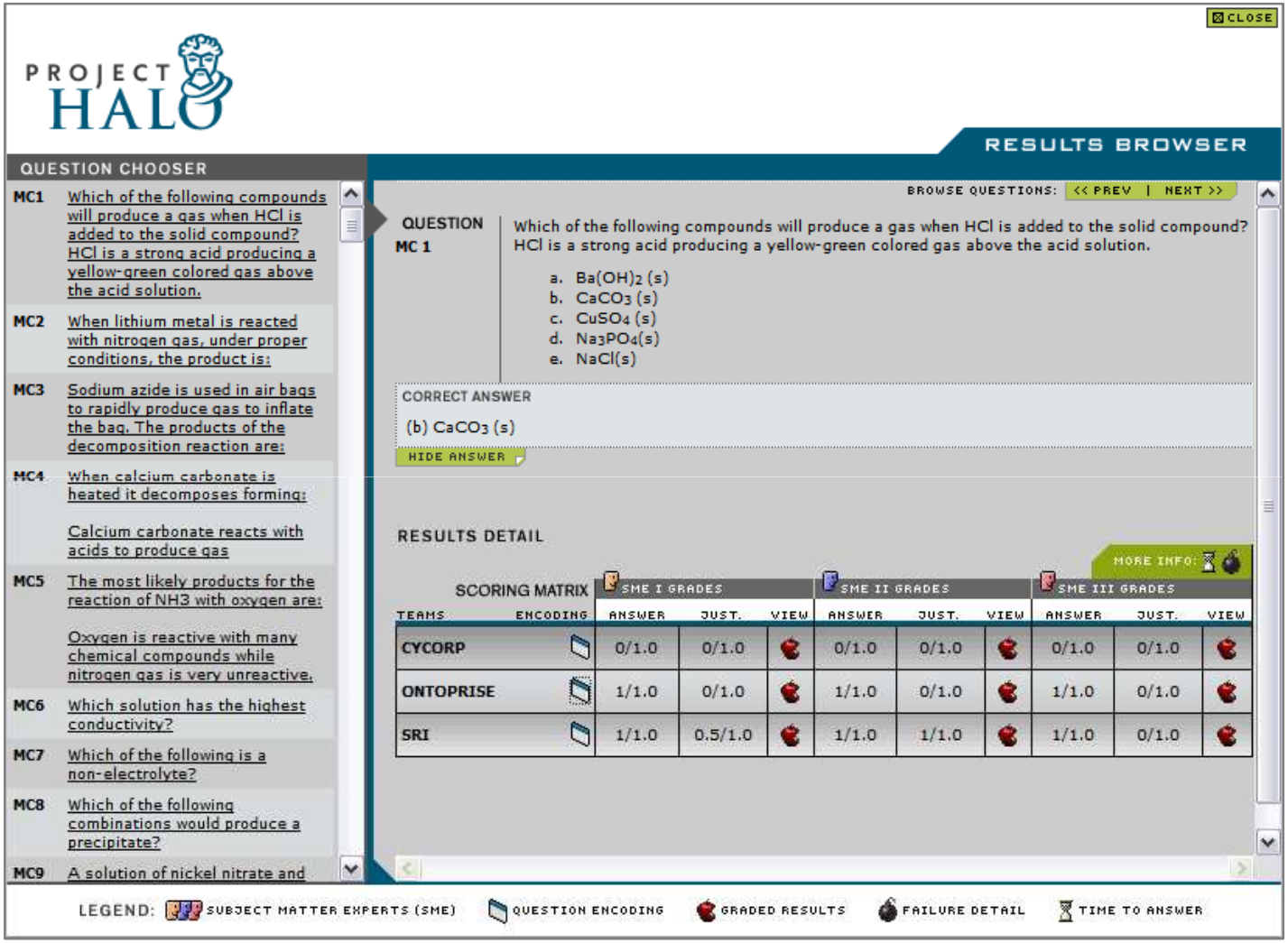
```
(every QF1 has (context
((:pair "(a) Ba(OH)2(s)"
(a Reaction with (raw-
material ((a HCl-
Substance) (a Ba_OH_2-
Substance with (state
((a State-Value with
(value (*solid))))))))))
(:pair "(b) CaCO3(s)" (a
Reaction with (raw-
material ((a HCl-
Substance) (a CaO3-
Substance

[…]
```

Overall Structure



Result browser



http://www.projecthalo.com/

Evaluation

- **General Results**
  - Correctness: high
  - Justification: considerably lower than correctness
  - Speed: was critical, but all systems faired well

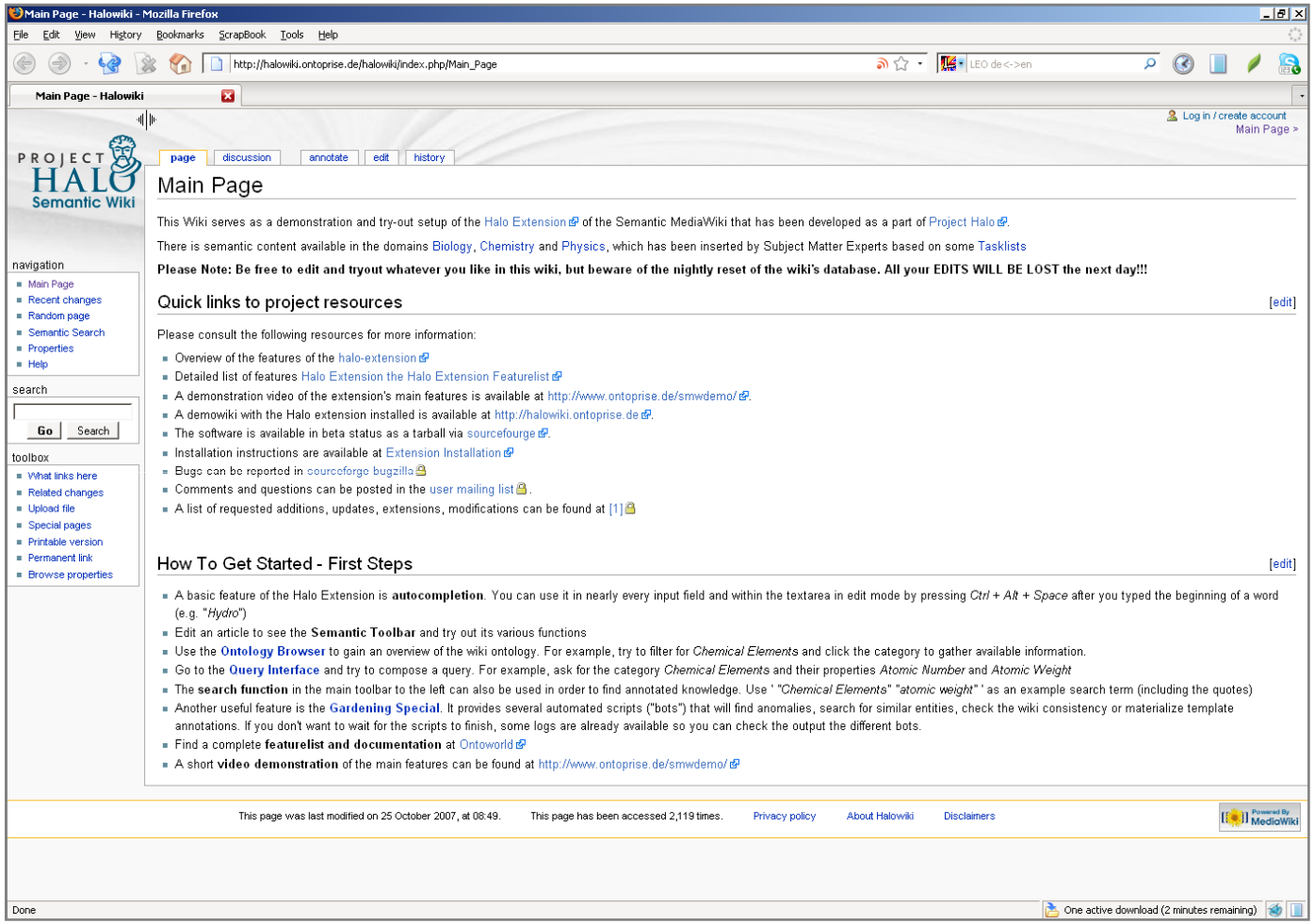
- **AP-Test Results**
  - Human mean average in this test is AP-2.82
  - Project Halo scored an AP-3 – they would have passed!

Refinement Phase

- **Task**
- Gather knowledge faster, cheaper, better…

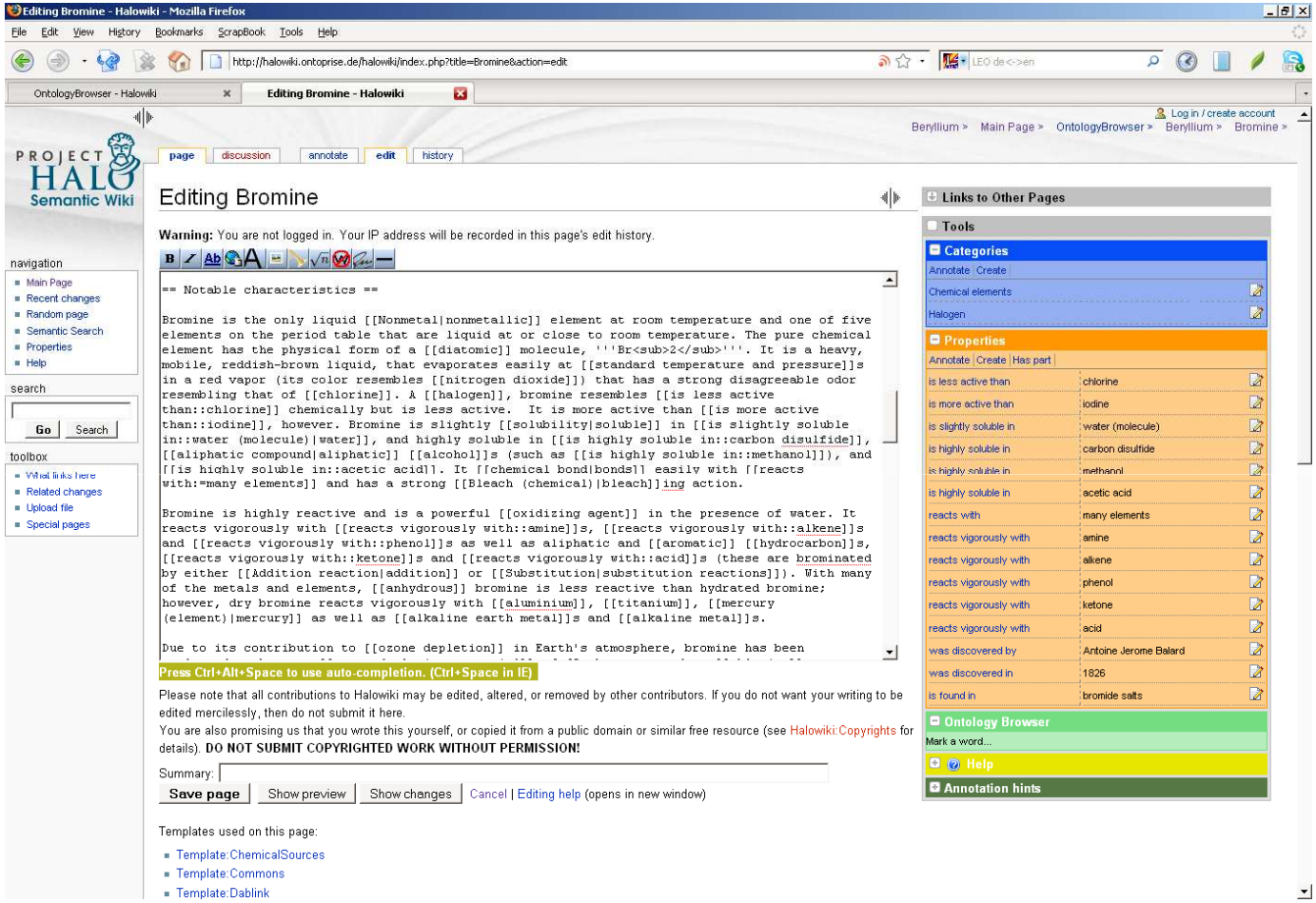
- **Approach**
  - Semantic Wiki for Knowledge Capturing
  - As a base, Semantic MediaWiki was taken (open source, developed by AIFB)
  - Several enhancements to usability of the system (SMW+, developed by ontoprise)

Knowledge Elicitation via Semantic Wikis



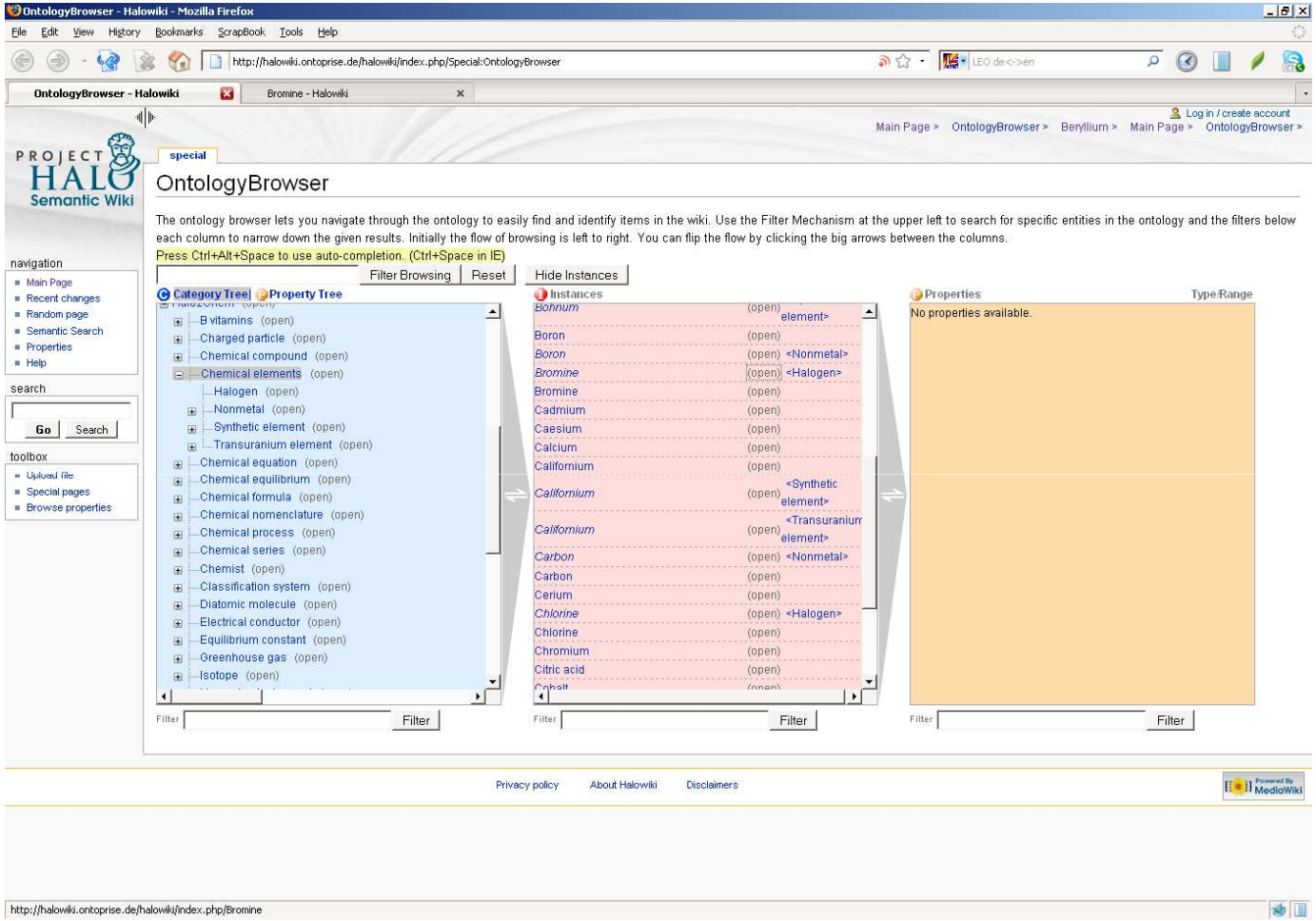
http://halowiki.ontoprise.de/

Knowledge Elicitation via Semantic Wikis



http://halowiki.ontoprise.de/

Knowledge Elicitation via Semantic Wikis



http://halowiki.ontoprise.de/

Karlsruhe Institute of Technology

AIFB

Data Modeling and Integration

SCENARIO: OVERFISHING ALERT SYSTEM

Background

Goal:
Fish **S**tock **D**epletion **A**ssessment **S**ystem

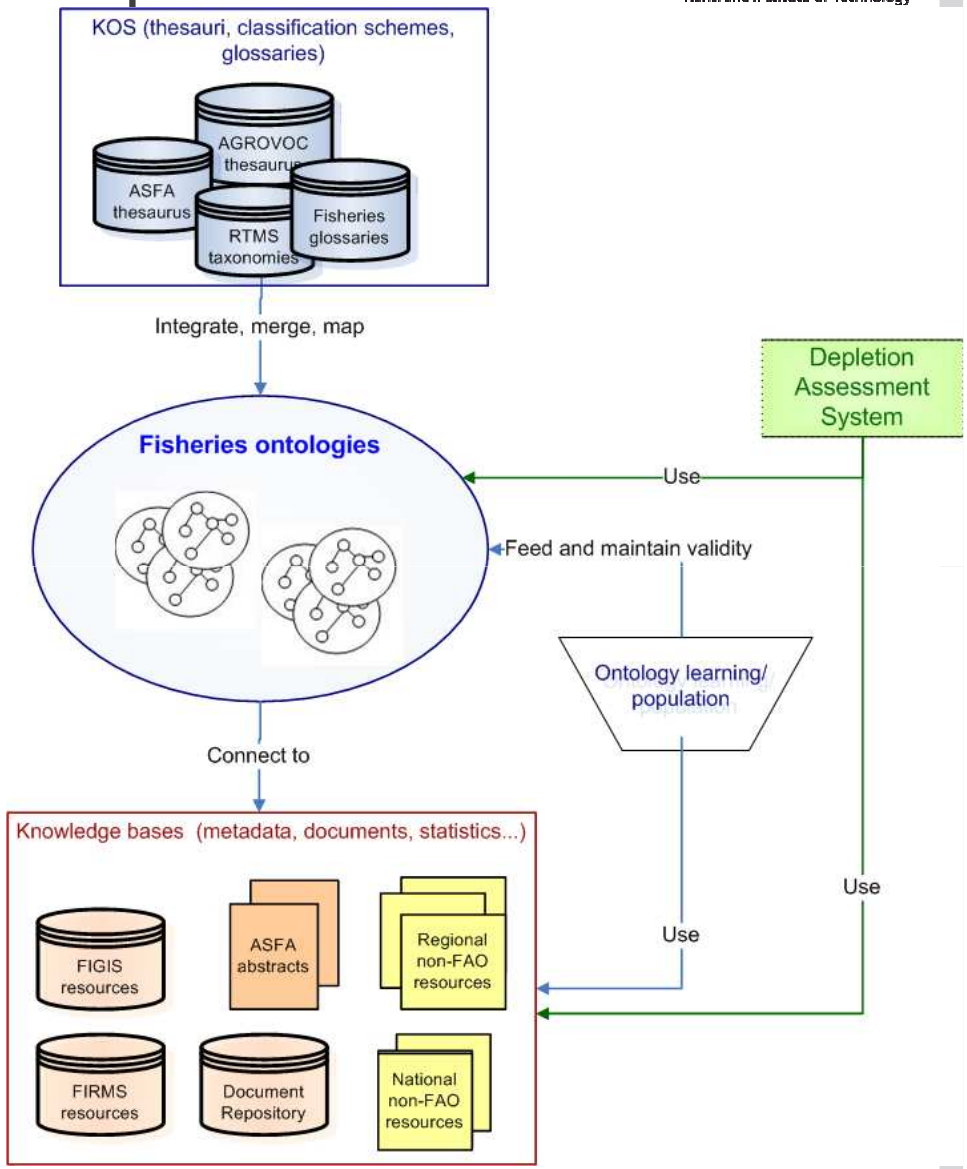
Decision support system to help Fisheries experts analyzing the status and trends of world's fish stocks

Case study:
FAO - Food and **A**griculture **O**rganization of the United Nations

Project:
NeOn– "Lifecycle Support for Networked Ontologies"

FSDAS – Ontology Runtime Requirements

- Ubiquitous and easy access to
  - status of fish stock
  - factors affecting fish depletion
- **Integration and querying** of heterogeneous (non-) ontological resources through the exploitation of the Fisheries ontologies
- return relevant results to the client
- Integrate with advanced annotation and visualization tools

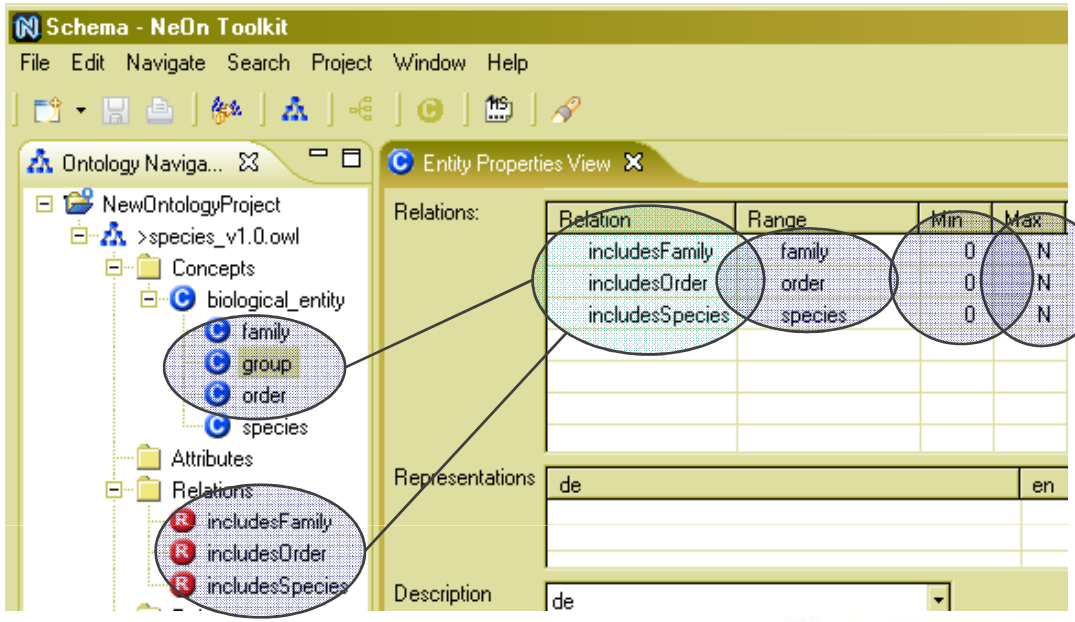


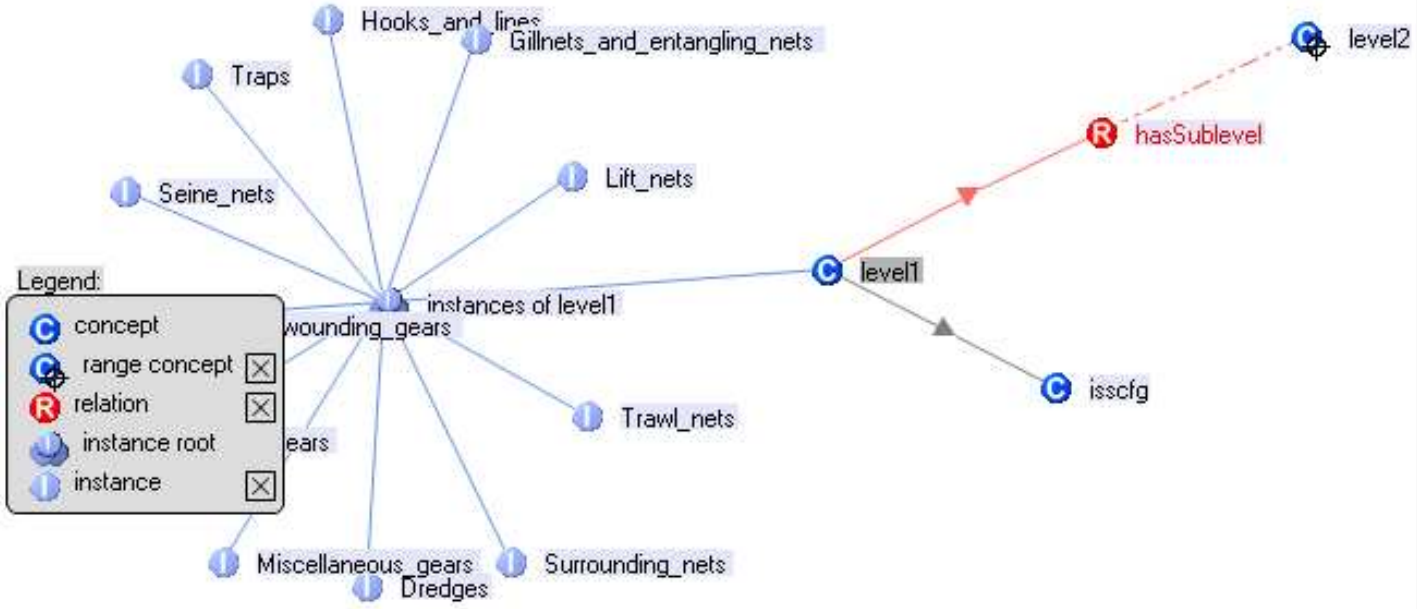
Fisheries Lifecycle Management – Ontology Engineering Requirements

- Provide support to ontology engineers and subject experts for:
  - modeling, populating, deploying, versioning ontologies
  - keeping them updated through an editorial workflow
  - managing mappings and relations between them

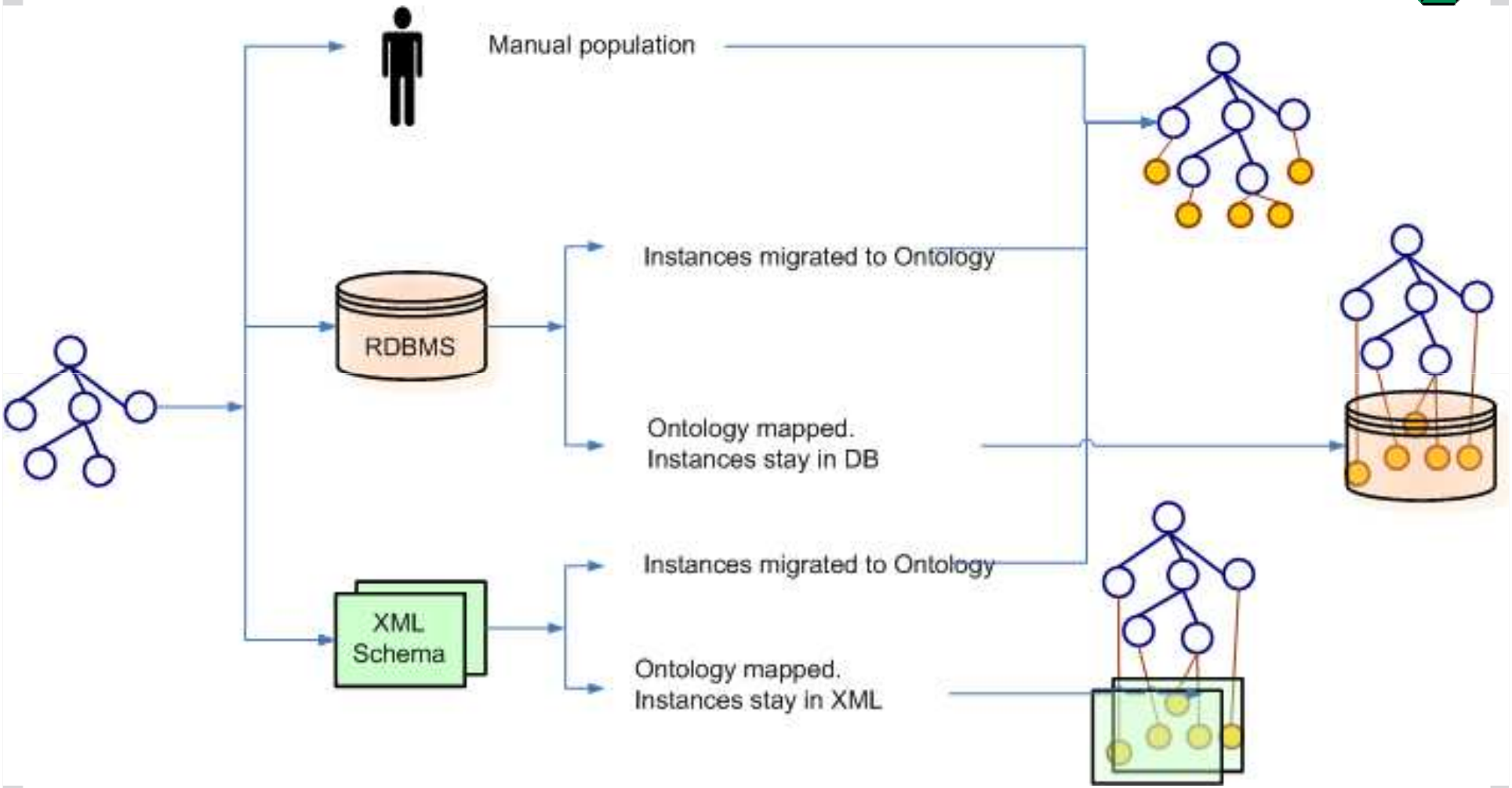
- Fisheries ontologies are:
  - multilingual ontologies
  - distributed / networked

Ontology Conceptualization

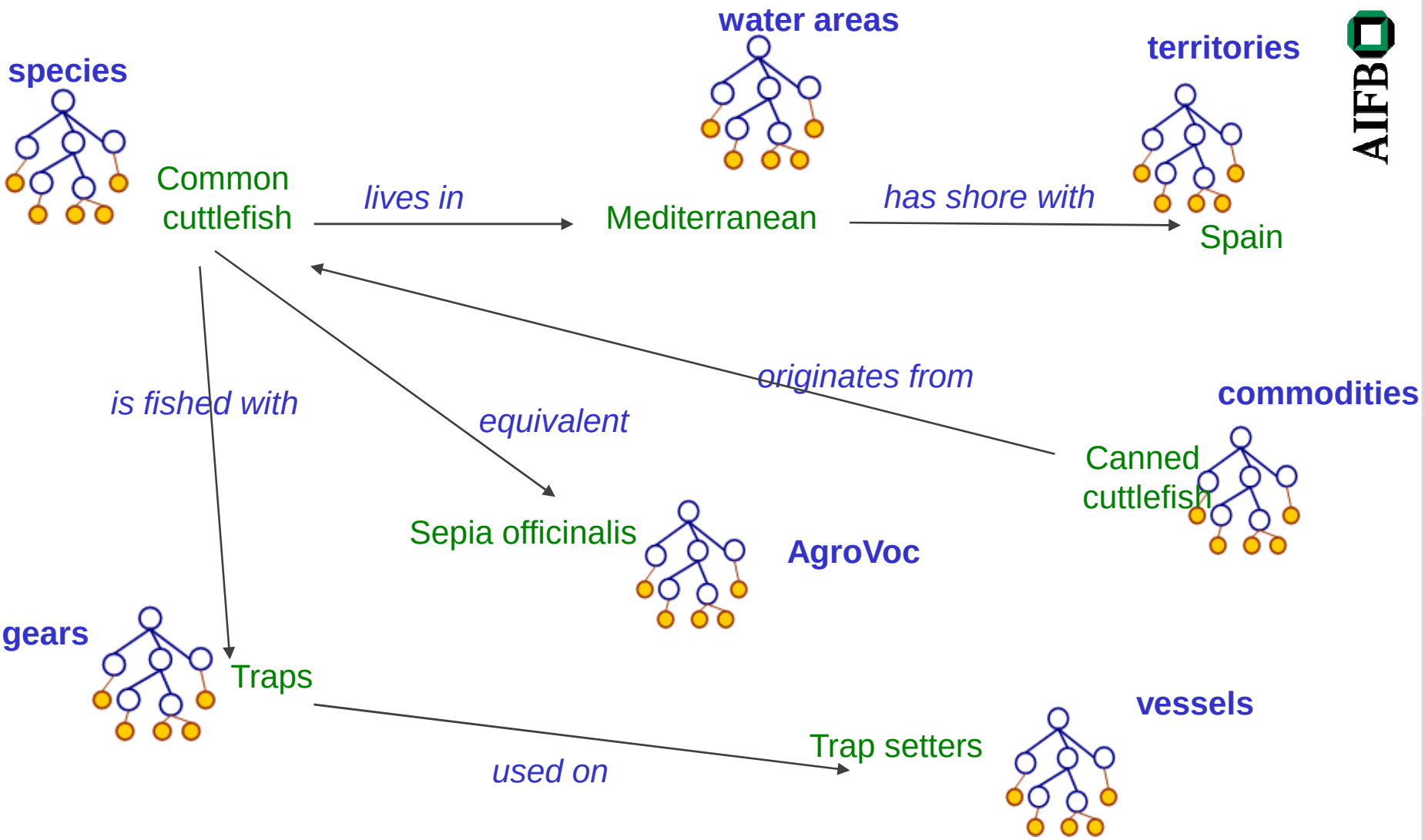




Ontology Population



Ontology Mapping to Create a Network



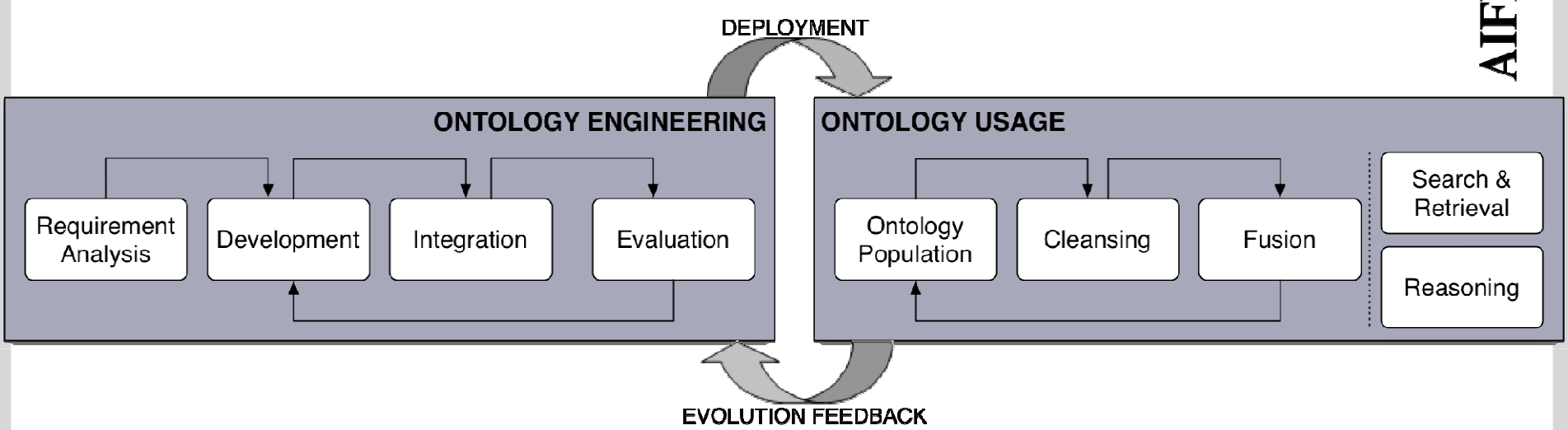
Aspects of Engineering Ontology–based Applications

- Engineering of ontology-based applications is a complex task, which involves
  - Ontology engineering
  - Software engineering
  - **Ontology management throughout the entire lifecycle**

➢ *Require systematic approach to*

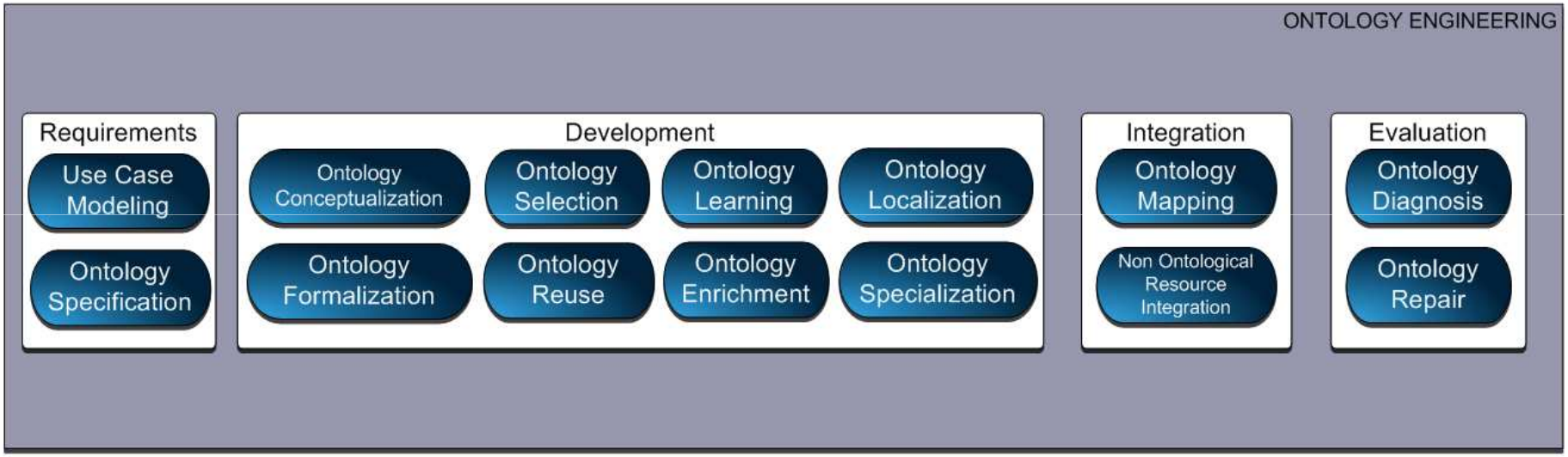
- the ontology and application development process,
- the ontology life cycle,
- the methods and methodologies for building ontologies and applications,
- and tool support

Ontology Lifecycle in Applications



- Close interaction between ontology engineering and runtime usage

Ontology Engineering Activities



How to build an ontology? Initial Steps

- Requirements specification
- Initial lexicon
- Knowledge extraction

Requirements Specification (I)

- **Domain and Goal**
  - what is the objective
- **Design Guidelines**
  - description of domain in use
- **Supported Applications**
  - brief characteristics of planned application
  - specification of system environment

Requirements Specification (II)

- **Knowledge Sources**
  - types of knowledge sources may be very different
    - domain experts
    - (reusable) ontologies
    - documents / systems
      - dictionaries
      - thesauri
      - product descriptions
      - organisational charts
      - employee role descriptions
      - ...

Requirements Specification (III)

- **Usage Scenarios** (Users and Use cases)
  - describe users/user groups
  - identify stakeholders
  - describe usage scenarios
    - how do they want to use the system?
    - what kind of support do they expect ?
    - use e.g. UML use-case diagrams
- **Competency Questions**
  - define collection of queries that should be supported by the system
  - analyze queries to find relevant lexical entries (concepts and relations)
  - explore scenarios
  - collect competency questionnaire

Competency Questionnaire

Competency Questionnaire No. 1			
Name: skill-man-ontology **Date**: 2001/03/22 **Ontology Engineer**: T. Model **Domain Expert:** X. Pert			
No.	Competency Question	Lexical Entries	Type
Q1	Which of our consultants has experience with JAVA programming language?	consultant	concept
		consultant *is a* employee	*isA* relation
		JAVA	concept
		programming language	concept
		JAVA *is a* program- ming language	*isA* relation
		programming language *is a* skill	*isA* relation
		employee *has experience with* skill	relation
Q2	What is the salary of a senior programmer? ...	salary	concept
		...	
Q3			

Initial lexicon - example

Competency Questionnaire No. 1			
Name: skill-man-ontology **Date**: 2001/03/22 **Ontology Engineer**:T. Model **Domain Expert:** X.Pert			
No.	**Competency Question**	**Lexical Entries**	**Type**
Q1	Which of our consultants has experience with JAVA programming language?	consultant	concept
		consultant *is a* employee	*isA* relation
		JAVA	concept
		programming language	concept
		JAVA *is a* programming language	*isA* relation
		programming language *is a* skill	*isA* relation
		employee *has experience with* skill	relation
Q2	What is the salary of a senior programmer?	salary	concept
		...	
Q3	...		

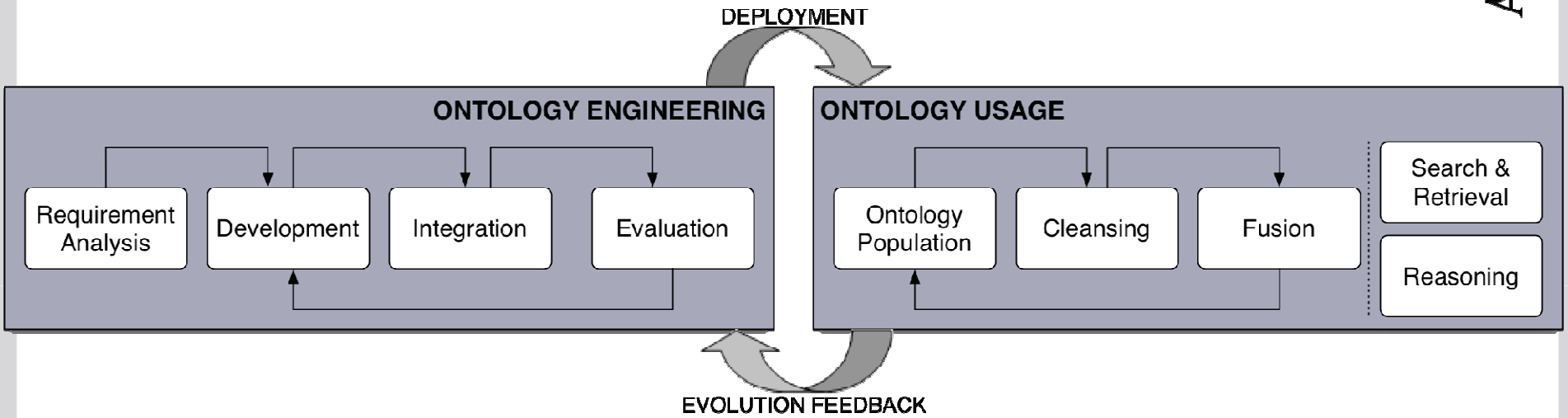
lexical entries

potential concepts	potential relations
Consultant Employee JAVA Programming language Experience Skill Programmer Project Customer Industry ...	HasExperienceWith WorksIn Contains ...

Knowledge Extraction

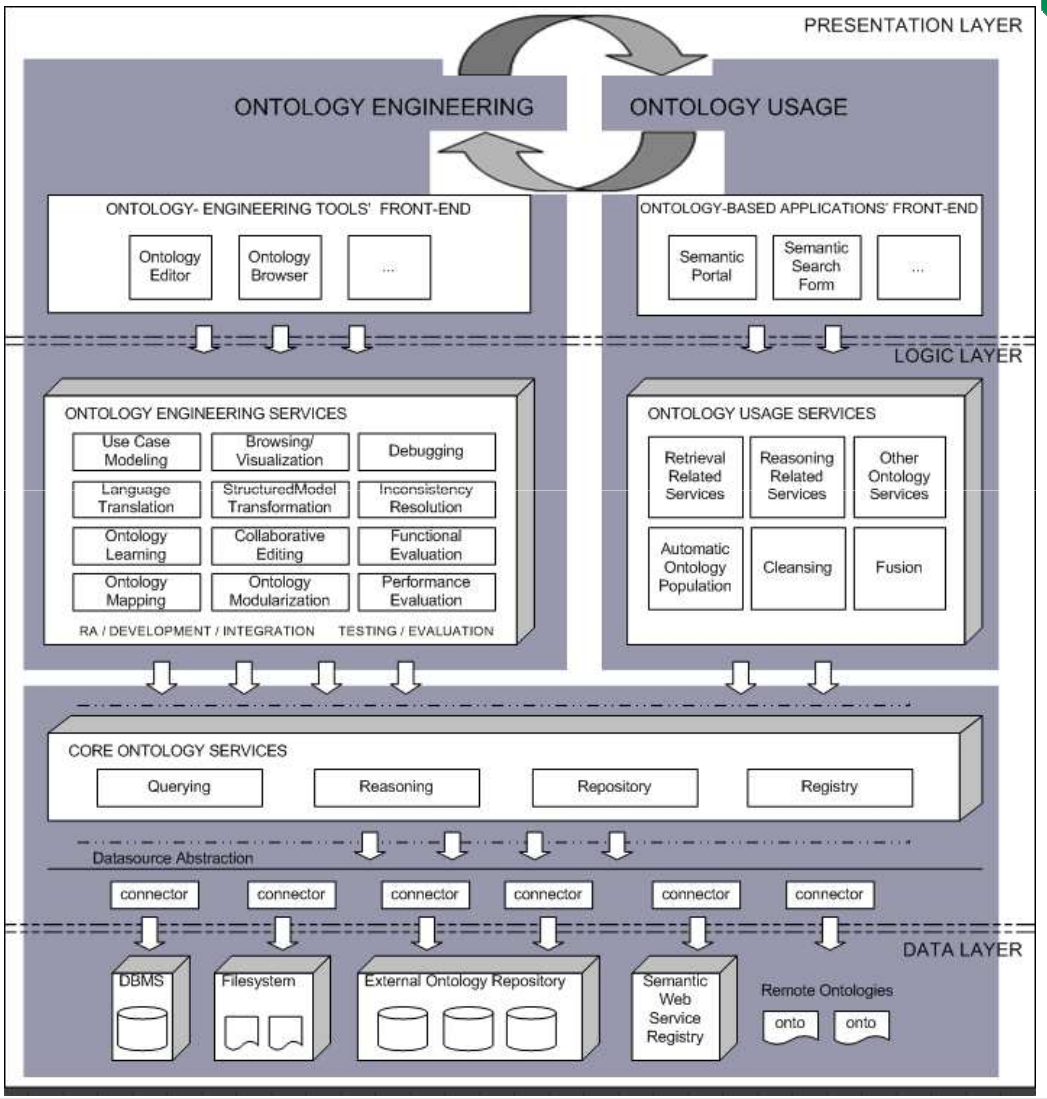
- **top-down** (modeling concepts and relationships on a very generic level)
- **bottom-up** (relevant concepts are extracted semi-automatically from available sources)
- **middle-out** (identify the most important concepts which will then be used to obtain the remainder of the hierarchy by generalization and specialization)

Ontology Lifecycle in Applications



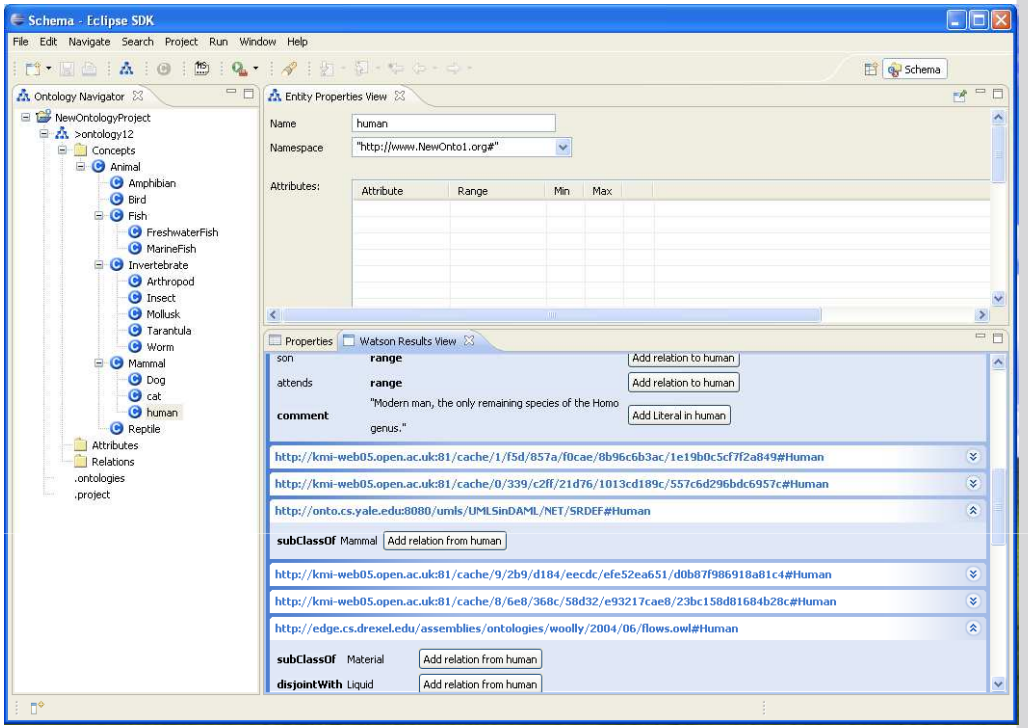
Generic Architecture with Lifecycle Support

- Inspired by SE best practices (SOA, JEE)
- Dynamic interaction of engineering and usage activities
- *Layered Organization*
- Presentation Layer
  - Thin client vs. Rich client
- Logic Layer
  - Business services / objects
  - Ontology services
- Data Layer
  - Ontological sources
  - Non-ontological sources



NeOn Toolkit

- Reference implementation of a generic architecture
- Infrastructure and reusable software components
- Based on Eclipse platform
- Based on a set of standardized APIs
- Plugins support lifecycle activities



http://www.neon-toolkit.org/

Summary

- Wide spectrum of types of ontologies
- Wide spectrum of ontology-based applications
  - Organisation of resources
  - Data integration
  - Modeling of generic knowledge
- Methods, methodologies and tools for engineering ontologies and ontology-based applications