Semantic Web
Methods, Tools and Applications

November 05, 2007

Peter Haase and Denny Vrandečić,
AIFB, Universität Karlsruhe (TH)
Asian Autumn School for the Semantic Web
AASSW 2007
Busan, Korea
Dr. Peter Haase
- Researcher at AIFB, University of Karlsruhe
- Project leader: NeOn – Lifecycle support for networked ontologies
- Interests and expertise: Semantic technologies and Semantic Web Infrastructure

Denny Vrandečić
- Researcher at AIFB, University of Karlsruhe
- Developer of Semantic MediaWiki
- Interests and expertise: Social Semantic Web, Ontology Evaluation
Agenda

- Motivation
  - Semantic Technologies and Ontologies
  - Ontology Lifecycle

- Methods and Tools
  - Ontology Development
  - Ontology Learning
  - Ontology Mapping
  - Ontology Evaluation and Evolution
  - Ontology Management: Repositories and Reasoners

- Applications
  - Content Management / Information Search and Retrieval
  - Expert and Assistance Systems
  - Social and collaborative applications
  - Web Applications
Problem: What does the available information really mean?

Find all companies specialising in semantic technologies in Europe.

Could he find a company related to RDF?

Search is based on keywords, and not on the actual meaning/semantics.
**Approach:** Annotate information elements (including links) with information about their meaning.

- **Company** specializesIn **Technology**
- **European Company** specializesIn **Semantic Technology**
- **SEMpany** specializesIn **RDF**
- **Ontology about the domain of companies**

**Semantic annotations** link to **companies** and **Europe**.
Semantic Technologies Provide Dynamic Integration Solutions to EII

- Data Integration **most costly** and challenging task for enterprises
- E.g. up to 80% of migration costs due to data integration
- Semantic technologies enable dynamic integration solution which federate queries and merge data from heterogeneous data sources

**Benefits** of ontologies
- **Shared interpretation** of distributed data
- **Conceptually adequate and expressive** data model to integrate heterogeneous data
- **Ease of maintenance** because of declarative specification of integration rules
Ontology Development Environments
Protege, NeOn Toolkit

Ontology Repositories
Sesame, Jena, …

Ontology Mapping Tools
Prompt, OntoMap…

Ontology Reasoners
Pellet, KAON2, Racer,…

Ontology Learning Tools
Text2Onto, OntoLT, …
The Protégé-OWL editor enables users to:

- Load and save OWL and RDFS files.
- Edit and visualize classifications, properties, and OWL individuals.
- Define logical axioms and OWL expressions.
- Execute reasoners such as description logic classifiers.
- Edit OWL individuals for Semantic Web markup.
Ontology Development with the NeOn Toolkit
# Ontology Development Tools

<table>
<thead>
<tr>
<th>Developer</th>
<th>Protégé OWL</th>
<th>Semantic Works</th>
<th>TopBraid Composer</th>
<th>IODT</th>
<th>SWOOP</th>
<th>OntoStudio (NeOn Toolkit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stanford</td>
<td>Stanford</td>
<td>Altova</td>
<td>TopQuadrant</td>
<td>IBM</td>
<td>Univ. of Maryland</td>
<td>Ontoprise (NeOn Found.)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Primary Ontology Language</th>
<th>OWL</th>
<th>OWL</th>
<th>OWL</th>
<th>OWL</th>
<th>OWL</th>
<th>F-Logic (+OWL)</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>View</th>
<th>Form Text</th>
<th>Form Text Graph</th>
<th>Form Text (UML-like) Graph</th>
<th>(UML-like) Graph</th>
<th>Browser-like</th>
<th>Forms</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Platform</th>
<th>Java</th>
<th>.NET</th>
<th>Eclipse</th>
<th>Eclipse</th>
<th>Browser + Java</th>
<th>Eclipse</th>
</tr>
</thead>
</table>

| Supported Reasoner     | Via DIG      | None            | Pellet, (built-in) Via DIG  | RACER, Pellet   | Pellet        | OntoBroker (+ KAON2) |

<table>
<thead>
<tr>
<th>Repository</th>
<th>Files, RDBMS</th>
<th>Files</th>
<th>Files, RDBMS</th>
<th>RDF on RDBMS</th>
<th>Files</th>
<th>Files, RDBMS</th>
</tr>
</thead>
</table>
The Problem

- *Heterogeneous ontologies require mappings for interoperability*
  - Numerous independent Ontologies
  - No single Domain Model
  - Modeling same or overlapping Knowledge

Main challenges

- Identifying mappings (correspondences between Entities)
- Representing these Relations
- Utilizing Mapping for querying, reasoning, ontology integration, translation and exchange
Great number of Techniques
- Syntactic, Semantic, External
- Element-Level, Structure-Level
- Schema or Instance Level mapping

Mapping Tools
- Several mapping systems already available
  (GLUE, PROMPT, FOAM, ONION, MAFRA)
- Manual, visual creation of mappings between ontologies
- Integration of (relational databases): automated ontology lifting and query answering
  (OntoMap, ODEMapster)

Best results
- Find best approximate Matches -> Similarity
- Semi-automatic
- Requires human Domain Expert
Challenge: Ontology development is a bottleneck!

**Solution: Extraction of (domain) ontologies from natural language text**
- Natural Language Processing
- Machine Learning

**Ontology Learning tasks**
- Concepts, instances
- Taxonomic relations: subclass-of, instance-of
- Relations: specializesIn ...
- Relation instantiations: specializesIn ...

**Ontology Population**
- Ontology Learning framework developed at AIFB since 2004 (successor of TextToOnto by Alexander Mädche)
- Explicit change management and incremental learning
- Implemented algorithms:
  - Concept extraction
    - TFIDF, entropy ...
  - Instance extraction
    - TFIDF ...
  - Similarity extraction
    - Context vectors
  - Concept classification
    - Heuristic, WordNet, patterns
  - Instance classification
    - Patterns, context similarity
  - Relation extraction
    - Subcategorization frames
<table>
<thead>
<tr>
<th>Domain</th>
<th>Range</th>
<th>Confidence</th>
</tr>
</thead>
<tbody>
<tr>
<td>fusion process</td>
<td>process</td>
<td>1.0</td>
</tr>
<tr>
<td>paper extract</td>
<td>extract</td>
<td>1.0</td>
</tr>
<tr>
<td>method</td>
<td>knowledge</td>
<td>1.0</td>
</tr>
<tr>
<td>template</td>
<td>model</td>
<td>1.0</td>
</tr>
<tr>
<td>column</td>
<td>information</td>
<td>1.0</td>
</tr>
<tr>
<td>contents</td>
<td>information</td>
<td>1.0</td>
</tr>
<tr>
<td>internet</td>
<td>system</td>
<td>1.0</td>
</tr>
<tr>
<td>datum</td>
<td>knowledge</td>
<td>1.0</td>
</tr>
<tr>
<td>template</td>
<td>knowledge</td>
<td>1.0</td>
</tr>
<tr>
<td>template</td>
<td>content</td>
<td>1.0</td>
</tr>
</tbody>
</table>

[ subclass-of( internet, network ), 1.0 ]
<table>
<thead>
<tr>
<th>Organization</th>
<th>System</th>
<th>Ontology Learning Subtasks</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Terms</td>
</tr>
<tr>
<td>AIFB, Univ. Karlsruhe</td>
<td>TextToOnto/Text2Onto</td>
<td>X</td>
</tr>
<tr>
<td>Amir Kabir Univ., Teheran</td>
<td>HASTI</td>
<td>X</td>
</tr>
<tr>
<td>CNTS, Univ. Antwerpen</td>
<td>OntoBasis</td>
<td></td>
</tr>
<tr>
<td>DFKI</td>
<td>OntoLT / RelExt</td>
<td>X</td>
</tr>
<tr>
<td>Economic Univ. Prague</td>
<td>TextToOnto Extensions</td>
<td></td>
</tr>
<tr>
<td>ISI, USC</td>
<td>CBC / DIRT</td>
<td></td>
</tr>
<tr>
<td>Keio Univ.</td>
<td>DOODLE</td>
<td></td>
</tr>
<tr>
<td>Univ. Paris-Sud</td>
<td>ASIUM/ Mo’K</td>
<td></td>
</tr>
<tr>
<td>Univ. Rome</td>
<td>OntoLearn</td>
<td>X</td>
</tr>
<tr>
<td>Univ. Salford</td>
<td>ATRACT</td>
<td>X</td>
</tr>
</tbody>
</table>
Ontology Learning - Conclusions

- Ontologies can not be learned automatically, especially not from text
  - Techniques for lexical acquisition too bad?
  - No, it is an inherent problem…
  - Ontologies can not be “learned” because they represent a way of viewing things and thus a process of reflection and agreement

- Ontology learning needs procedures for (formal) self-evaluation

- Ontology learning and reasoning strictly interact with each other

- Ontologies and the world evolve, so ontology learning should address the evolution of data and knowledge
Ontology Evaluation

- **Evaluation**
  - User Guidance Tools for Design Support
  - Consistency checks during development process
  - Support for error recovery and debugging

- **What is a ‘good’ ontology?**
  - Usage, application performance, data coverage, corpus fit, reasoning adequacy?
  - **Well-known: OntoClean** (formal evaluation of taxonomic relationships), developed by Guarino and Welty
  - System of conceptual tags (Rigidity, Unity, Identity, Independence) + constraints on possible subsumption relations (e.g. ~R can’t subsume +R)
  - OntoClean is based on philosophical principles and is even hard to understand for experts
  - AEON approach tries to (automatically) approximate the tagging with OntoClean
Challenge: Domain ontologies are ever changing!

- **Solution**: A systematic method to handle the resolution of changes is needed

![Diagram showing relationships between concepts](image)

- **Traceability (of changes)**
  - which change to the ontology was performed by whom and when, possibly why?
  - if the change has been generated as a byproduct of updating the corpus, keep a reference to the segment in a corpus which triggered the change
### Ontology Evolution (Process)

#### Core Components

- **Capturing**
  - Explicit changes – intended by Ontology Engineer
  - Implicit changes: *usage-driven, data-driven, structure-driven*

- **Representation**
  - Fine grained (simple) change operation
  - Coarse grained (complex/composite) change operations

- **Semantics of change**
  - Consistency: *structural consistency, logical consistency, user-defined consistency*
  - Verification: *a posterior, a priori*
  - Realization: *declarative, procedural*

- **Change propagation**
  - Pull based, Push based
  - also: Ontology Dependent Consistency, Replication Ontology Consistency

- **Change Implementation**
  - Change Notification, Change Application, Change Logging

- **Change validation**
  - Justification / redoing based on user request
Ontology Repositories
- Focus on storing, manipulating and retrieving large sets of data
- Typically, with some query language (e.g. SPARQL)
- File system vs. DBMS backend

Ontology Reasoners
- Reasoning tasks (standard / non-standard)
- Reasoning methods

Often, repository and reasoning are combined, sometimes via the same API
Ontology Repositories

- **Jena Framework**
  - OWL and RDF API
  - Reading and writing RDF in RDF/XML, N3 and N-Triples
  - In-memory and persistent storage
  - SPARQL query engine

- **Sesame**
  - RDF API
  - Native store for scalable storage and querying with reliable persistence
  - Support for several back-end stores
  - Support for several RDF query languages including SPARQL and SeRQL

- **Others: YARS, Kowari, OWLIM**
Standard Reasoning Tasks

- ...check whether an ontology is consistent
  - satisfiability checking: does the ontology have a model?

- ...check if a class C is consistent
  - i.e. whether C can have non-empty extension

- ...check if a is an instance of C: C(a)
  - basis for query answering

- ...check for class inclusion of two classes: C v D
  - i.e. whether instances of C are always instances of D
  - subsumption checking
  - equivalent to query containment

- Selecting the right reasoner depends on many aspects
  - Reasoning task
  - Complexity of ontology (underlying logical language)
  - Size of ontology (ABox vs. TBox)
## Overview of Reasoners

<table>
<thead>
<tr>
<th></th>
<th>Cerebra (Web Methods)</th>
<th>FACT++ (U of Manchester)</th>
<th>KAON2 (ontoprise)</th>
<th>Pellet (U of Maryland)</th>
<th>Racer (Racer Systems)</th>
<th>Ontobroker (ontoprise)</th>
<th>OWLIM (ontotext)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Interfaces</strong></td>
<td>OWL API</td>
<td>DIG</td>
<td>KAON2 API</td>
<td>DIG, OWL API, Jena API</td>
<td>DIG, OWL API</td>
<td>KAON2 API</td>
<td>Sesame API</td>
</tr>
<tr>
<td><strong>Reasoning</strong></td>
<td>Tableaux</td>
<td>Tableaux</td>
<td>Resolution</td>
<td>Tableaux</td>
<td>Tableaux</td>
<td>Datalog</td>
<td>Forward Chaining</td>
</tr>
<tr>
<td><strong>Approach</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Supported</strong></td>
<td>OWL DL</td>
<td>SHIQI</td>
<td>SHIQ + DL safe rules</td>
<td>SROIQ + DL safe rules</td>
<td>SHIQ</td>
<td>F-Logic</td>
<td>OWL DLP</td>
</tr>
<tr>
<td><strong>Logic</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Based on</strong></td>
<td>C++</td>
<td>C++</td>
<td>Java</td>
<td>Java</td>
<td>Lisp</td>
<td>Java</td>
<td>Java</td>
</tr>
</tbody>
</table>

*OWL DLP: OWL-DL Reasoner*
**What is the NeOn Toolkit?**

The NeOn Toolkit is a comprehensive framework designed to support the development and management of networked ontologies. It encompasses various phases of the ontology lifecycle, including modeling, integration, verification, testing, and deployment. Key features of the toolkit include:

- **Dual language approach**
- **Modular**
- **Extensible**
- **Platform**
- **Based on Eclipse**
- **Succeeds OntoStudio**

Supporting the lifecycle of networked ontologies, the NeOn Toolkit integrates dual language approaches and utilizes tools like F-Logic rules for frame-like modeling and OWL-DL for ontology modeling.
NeOn Architecture to Support the Entire Ontology Lifecycle

- GUI components
  - Standard GUI components [class tree, …]
  - Text-based Interface
  - Graph-based Interface

- Engineering components
  - Property Editors
  - Mapping Editors
  - Visualization Algorithms
  - Collab. Support
  - Translation
  - Data Annotation
  - Loosely coupled services

- Infrastructure services
  - OWL Frames/Rules Reasoner
  - OWL Frames/Rules Model API
  - Storage Basic Query
  - Query Metadata
  - Repository Service Registry Service
  - Orthogonal Functionality
  - Security
  - Service Infrastructure
  - Orthogonal Functionality
  - Versioning
Applications

- Content Management / Information Retrieval
  - BT Digital Library

- Expert and Assistance Systems
  - HALO

- Social and collaborative applications
  - Ontoworld

- Web Applications
  - PiggyBank
  - eMerges: Semantic Web Services GIS based Emergency Management
Bob works as technology analyst for British Telecom. His daily work includes research on new technological trends, market developments as well as the analysis of competitors.

Bob’s company maintains a digital library that gives access to a repository of internal surveys and analysis documents. The company also has a license with an academic research database which is accessed via a separate interface.

Depending on his work context, Bob uses the topic hierarchies, the full-text search functionalities or metadata search facilities provided by the two libraries to get access to the relevant data.

However, Bob is often annoyed by the differing topic hierarchies and metadata schemes used by the two libraries as well as by a cumbersome syntax for metadata queries.
Why Ontology-Based Digital Libraries?

- Immediate support for unified structured queries against metadata and documents
- Easy integration of heterogeneous knowledge sources
- Easy integration with knowledge elicitation methods from unstructured content
- Mapping to natural language queries
- Generic, flexible and modular architecture
Knowledge Portal

Conceptual Architecture

Query Translation Component
Query Interface
Reasoning Engine
Query Answering Component
Ontology Model and Knowledge Base
Internal Storage
Data Connectors
Document Metadata
Full-text documents

Built-In Library
Full Text Indexing
Text Classification
Ontology Learning

Ontology-based Question Answering for Digital Libraries - ECDL 2007, Budapest
Ontology Model and Knowledge Base

- **Ontology**
  - global conceptual model
  - aligned with established schemas (e.g. Dublin Core)

- **Knowledge base of the digital library**
  - actual bibliographical metadata, topic hierarchies, and full-text document content
  - data aligned with global ontology via mapping axioms

```
swrc:Book rdfs:subClassOf protont:Document
expl:document5127 rdf:type swrc:InProceedings
expl:document5127 protont:title "Digital Libraries"
```

- **Query answering against knowledge base (SPARQL)**

```
SELECT ?x WHERE {
  ?x <http://proton.semanticweb.org/2005/04/protonu#hasSubject> ?y .
  match(?z,"Intellectual Capital")
}
```
Presentation layer for underlying content

Interaction via standard interfaces
- keyword-search, topic browsers etc.

Interaction via natural language queries
- converts natural language queries into SPARQL

Translation step comprises
- deep parsing of the questions
- roughly, linguistic frames become query constraints
- lexicon describes possible lexical realizations of ontology elements

"Who wrote books on 'digital libraries'?"
"Which journal articles were written by 'Tim Berners-Lee' (and for which journal)?"
Overview of the ORAKEL System

Natural Language Interface

- FrameMapper
- Query Interpreter
- Query Converter
- Answer Generation
- Domain-specific Lexicon
- Domain-independent Lexicon
- Domain Ontology
- Knowledge Base
Scenario Revisited

"Which journal articles were written by 'Tim Berners-Lee' for which journal?"

PREFIX protonu:<http://proton.semanticweb.org/2005/04/protonu#>
PREFIX protont:<http://proton.semanticweb.org/2005/04/protont#>

SELECT ?x ?z WHERE {
  ?x rdf:type protonu:Article .
  match(?ys, "Tim Berners Lee") .
  ?z rdf:type protonu:Journal .
  ?x protonu:publishedWithin ?z
}

"The Semantic Web"          "The Scientific American"
"WWW: Past, Present, and Future"  "IEEE Computer"
[...]                      [...]
The BT Digital Library

Screenshot from BT Digital Library

2007-09-17 36

Ontology-based Question Answering for Digital Libraries - ECDL 2007, Budapest

Slide 36
“Building a 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 ambitious

Multi-stage effort:
- Start with a specific science (Chemistry)
- Challenge with several teams
- Answer AP-style questions

Complete information at http://www.projecthalo.com/
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$_3$PO$_4$(s)
- NaCl (s)

Formalized questions:

(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) CaCO$_3$(s)" (a Reaction with (raw-material ((a HCl-Substance) (a CaCO$_3$-Substance with (state ((a State-Value with (value (*solid)))))))) (:pair "(c) CuSO$_4$(s)" (a Reaction with (raw-material ((a HCl-Substance) (a CuSO$_4$-Substance with (state ((a State-Value with (value (*solid)))))))) (:pair "(d) Na$_3$PO$_4$(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))))))

Slide 38
Formalizing questions is “just” question understanding

Needs a huge amount of background knowledge = ontology

And a reasoner to answer the question using the ontology
Evaluation

- Correctness
  - Was pretty high

- Justification
  - Considerably lower than correctness

- Speed
  - Was critical, but all systems faired well

- Results:
  - Human mean average in this test is AP-2.82
  - Project Halo scored an AP-3 – they would have passed!
http://www.projecthalo.com/halotempl.asp?cid=2135#
A semantic wiki for the semantic web community
The Semantic Web

- Persons
- Events
- Tools
- Publications

Events

You can find information about many events and papers within this wiki. Using semantic annotation, it is possible to query for particular events.

**Upcoming conferences, etc.:** SAAKM2007 (Whitney, 2007), Ontology Engineering (Baden-Baden, 2 November), FEWS (Busan, 12 November 2007), ISWC2007 (Busan, 11 November), ISWC2007+ASWC2007 (Busan, 11 November)

**Upcoming submission deadlines:** PIM2007 (1 November 2007), WWW2008 (1 November 2007), ESWC2008 (14 December)

Organising an event? Advertise it here by creating an article! Just enter the event's abbreviation in the search box below to get an edit box with further documentation to fill.

---

**People**

This wiki contains annotated articles for a large number of community members. If you should read the documentation and create a new article. To be listed below, use Category:Person, e.g. by using the person template on their pages.

The following list is cached for better performance. If it appears to be out of date, clicking the link at the bottom of this page. You can also directly browse the current list.

<table>
<thead>
<tr>
<th>People</th>
<th>Affiliation</th>
</tr>
</thead>
<tbody>
<tr>
<td>A Min Tjoa</td>
<td>TU Wien</td>
</tr>
<tr>
<td>Aaron Stanton</td>
<td>QualitySmith</td>
</tr>
<tr>
<td>Aaron Swartz</td>
<td></td>
</tr>
<tr>
<td>Abhita Chugh</td>
<td>Stanford University</td>
</tr>
<tr>
<td>Abir Qasem</td>
<td>Lehigh University</td>
</tr>
<tr>
<td>Abraham Bernstein</td>
<td>I FI</td>
</tr>
<tr>
<td>Achille Fokoue</td>
<td>IBM Watson Research</td>
</tr>
<tr>
<td>Adam Cheyer</td>
<td>SRI International</td>
</tr>
<tr>
<td>Adina Sirbu</td>
<td>DERI Innsbruck</td>
</tr>
<tr>
<td>Aditya Kalyanpur</td>
<td>IBM Watson Research</td>
</tr>
<tr>
<td>Adrian Mocan</td>
<td>DERI Innsbruck</td>
</tr>
<tr>
<td>Afraz Jaffri</td>
<td>University of Southampton</td>
</tr>
<tr>
<td>Agnes Koschmider</td>
<td>University of Karlsruhe</td>
</tr>
</tbody>
</table>
Why should the community care?

- High Google rank
- Person articles and event articles ranked high, easy to find
- Collects information
  - Overview over several conferences, dates, etc.
  - But also PC /OC memberships

- Growing steadily
- Easy to add
- Easy to extend
- Firefox extension
- Collects data on the web while browsing
- Has a cute logo
- Collected data can be ...
  - Explored as you like
  - Aggregated from different sources
  - Compared
  - Browsed and viewed
  - Queried

- Mashup whatever you want whenever you want!

- Try it out - http://simile.mit.edu/wiki/Piggy_Bank
- Loads data from current page
- Either data is available in a standard format
- Or there is a screenscraper for that page
- Data you have just collected viewable in many different ways

- Like HousingMap.com, a Mashup that combines CraigsList and Google Maps

- But you did it yourself in your browser!
- Supporting Emergency Planning for Essex County Council

- Many heterogeneous data sources

- Need a simple integration interface
  - Users are potentially under high stress
Define a region (spatial object)
Assign a type to it.
Uniform representation: Spatial Object have context-dependent affordances (actions), and features (properties)
Here retrieves the shelters in an area
All GMaps features are available and retrieved spatial objects can be fields (e.g. chemical cloud over Stansted airport)
Any xml based WS can be integrated, here snow level (grey), and Instant Messaging presence (green/yellow)
- Any ideas? Try them out in your projects…

- Next: Hands-on Session!


- Enrico Motta, Marta Sabou: **Next Generation Semantic Web Applications**. ASWC 2006: 24-29
Noah S. Friedland, Paul G. Allen, and many more: Project Halo – Towards a Digital Aristotle, in AI Magazine 2004
