State of the Semantic Web

Beijing, China, 2006-10-16

Ivan Herman, W3C

What will I talk about?

- The history of the Semantic Web goes back to several years now
- It is worth looking at what has been achieved, where we are, and where we might be going…

Let us look at some results first!

The basics: RDF(S)

- We have a solid specification since 2004: well defined (formal) semantics, clear RDF/XML syntax
- *Lots* of tools are available. Are listed on W3C's wiki:
  - *RDF programming environment for 14+ languages, including C, C++, Python, Java, Javascript, Ruby, PHP,… (no Cobol or Ada yet sad smiley!)*
  - *13+ Triple Stores, ie, database systems to store (sometimes huge!) datasets*
  - *etc*
- Some of the tools are Open Source, some are not; some are very mature, some are not ☺: *it is the usual picture of software tools*, nothing special any more!
- *Anybody can start developing RDF-based applications today*

The basics: RDF(S) (cont.)

- There are lots of tutorials, overviews, and books around
  - *again, some of them good, some of them bad, just as with any other areas…*
- Active developers' communities
- Large datasets are accumulating. E.g.:
  - *IngentaConnect bibliographic metadata storage: over 200 million triplets*
  - *RDF version of Wikipedia: more than 47 million triplets*
  - *tracking the US Congress: data stored in RDF (around 25 million triplets)*
  - *RDFS/OWL Representation of Wordnet: also downloadable as 150MB of RDF/XML*
  - *"Département/canton/commune" structure of France published by the French Statistical Institute*

Ontologies: OWL

- This is also a stable specification since 2004
- Separate layers have beed defined, balancing expressibility vs. implementability (OWL-Lite, OWL-DL, OWL-Full)
  - *quite a controversial issue, actually…*
- Looking at the tool list on W3C's wiki again:
  - *a number programming environments (in Java, Prolog, …) include OWL reasoners*
  - *there are also stand-alone reasoners (downloadable or on the Web)*
  - *ontology editors come to the fore*
- OWL-DL and OWL-Lite relies on Description Logic, ie, can use a large body of accumulated knowledge

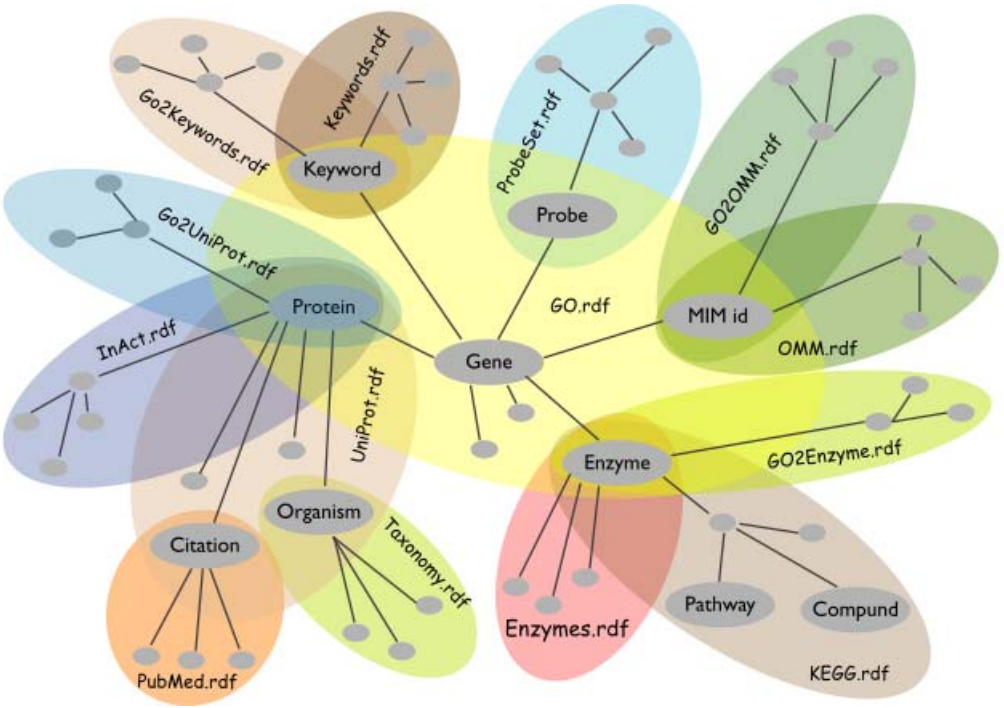
Ontologies

- Large ontologies are being developed (converted from other formats or defined in OWL)
  - *eClassOwl: eBusiness ontology for products and services, 75,000 classes and 5,500 properties*
  - *the Gene Ontology: to describe gene and gene product attributes in any organism*
  - *UniProt: protein sequence and annotation terminology and data*

Vocabularies

- There are also a number “core vocabularies” (not necessarily OWL based)
  - *SKOS Core: about knowledge systems*
  - *Dublin Core: about information resources, digital libraries, with extensions for rights, permissions, digital right management*
  - *FOAF: about people and their organizations*
  - *DOAP: on the descriptions of software projects*
  - *MusicBrainz: on the description of CDs, music tracks, …*
  - *SIOC: Semantically-Interlinked Online Communities*
  - *…*
- One should *never* forget: ontologies/vocabularies must be shared and reused!

A mix of ontologies (a life science example)…



Ontologies, Vocabularies

- Ontology and vocabulary *development* is still a complex task
- The W3C SW Best Practices and Deployment Working Group has developed some documents:
  - *"Best Practice Recipes for Publishing RDF Vocabularies"*
  - *"Defining N-ary relations"*
  - *"Representing Classes As Property Values"*
  - *"Representing "value partitions" and "value sets""*
  - *"XML Schema Datatypes in RDF and OWL"*

  the work is continuing in the (new) SW Deployment Working Group

Querying RDF: SPARQL

- Querying RDF graphs becomes essential
- SPARQL is almost here
  - *query language based on graph patterns*
  - *there is also a protocol layer to use SPARQL over, eg, HTTP*
  - *hopefully a Recommendation mid 2007*
- There are a number of implementations already
- There are also SPARQL “endpoints” on the Web:
  - *send a query and a reference to data over HTTP GET, receive the result in XML or JSON*
  - *applications may not need any direct RDF programming any more, just a SPARQL endpoint*

SPARQL as the *only* interface to RDF data?

- **http://www.sparql.org/sparql?query=…**
  with the query:

```
SELECT ?translator ?translationTitle ?originalTitle ?originalDate
FROM <http://…/TR_and_Translations.rdf>
WHERE {
   ?trans rdf:type trans:Translation;
                  trans:translationFrom ?orig;
                  trans:translator      [ contact:fullName ?translator ];
                  dc:language           "fr";
                  dc:title              ?translationTitle.
   ?orig  rdf:type rec:REC;
                  dc:date               ?originalDate;
                  dc:title              ?originalTitle.
}
ORDER BY ?translator ?originalDate
```

- yields…

A word of warning on SPARQL…

- It is *not* a Recommendation yet
- New issues may pop up at the last moment via reviews
  - *a query language needs very precise semantics and that is not that easy* ☹
- Some features *are* missing
  - *query on list/sequence/set membership*
  - *control and/or description on the entailment regimes of the triple store (RDFS? OWL-DL? OWL-Lite? …)*
  - *modify the triple store*
  - *…*

postponed to a next version…

Of course, not everything is so rosy…

- There are a number of issues, problems
  - *how to get RDF data*
  - *missing functionalities: rules, "light" ontologies, fuzzy reasoning, necessity to review RDF and OWL, …*
  - *misconceptions, messaging problems*
  - *need for more applications, deployment, acceptance*
  - *etc*

How to get RDF data?

- Of course, one could create RDF data manually…
- … but that is unrealistic on a large scale
- Goal is to generate RDF data automatically when possible and “fill in” by hand only when necessary

Data may be around already…

- Part of the (meta)data information is present in tools … but thrown away at output
  - *e.g., a business chart can be generated by a tool: it "knows" the structure, the classification, etc. of the chart, but, usually, this information is lost*
- storing it in web data would be easy!
- "SW-aware" tools are around (even if you do not know it…), though more would be good:
  - *Photoshop CS stores metadata in RDF in, say, jpg files (using XMP)*
  - *RSS 1.0 feeds are generated by (almost) all blogging systems (a huge amount of RDF data!)*
  - *…*

Data may be extracted (a.k.a. “scraped”)

- Different tools, services, etc, come around every day:
  - *get RDF data associated with images, for example:*
    - service to get RDF from flickr images (see example)
    - service to get RDF from XMP (see example)
  - *XSLT scripts to retrieve microformat data from XHTML files*
  - *scripts to convert spreadsheets to RDF*
  - *etc*
- Most of these tools are still individual “hacks”, but show a general tendency
- Hopefully more tools will emerge

GRDDL Working Group

- GRDDL WG's goal is a more systematic way of defining "scrapers" for XHTML files (eg, for microformats)

```
<html xmlns="http://www.w3.org/1999/">
  <head profile="http://www.w3.org/2003/g/data-view">
    <title>Some Document</title>
    <link rel="transformation" href="http:…/dc-extract.xsl"/>
    <meta name="DC.Subject" content="Some subject"/>
    ...
  </head>
  ...
  <span class="date">2006-01-02</span>
  ...
```

- yields, by running the file through `dc-extract.xsl`:

```
<rdf:Description rdf:about="…">
  <dc:subject>Some subject</dc:subject>
  <dc:date>2006-01-02</dc:date>
</rdf:Description>
```

Another Future Solution: RDFa

- RDFa (formerly known as RDF/A) extends XHTML by:
  - *extending the* `link` *and* `meta` *to include child elements*
  - *add metadata to any elements (a bit like the* `class` *in microformats, but via dedicated properties)*
- It is very similar to microformats, but with more rigor:
  - *it is a general framework (instead of an "agreement" on the meaning of, say, a* `class` *attribute value)*
  - *terminologies can be mixed more easily*
- The W3C Working Group on SW Deployment has this on its charter
- May be considered as an alternative serialization of (part of) RDF; may be bound to GRDDL in practice

RDFa example

- For example

```
<div about="http://uri.to.newsitem">
  <span property="dc:date">March 23, 2004</span>
  <span property="dc:title">Rollers hit casino for £1.3m</span>
  By <span property="dc:creator">Steve Bird</span>. See
  <a href="http://www.a.b.c/d.avi" rel="dcmtype:MovingImage">
  also video footage</a>…
</div>
```

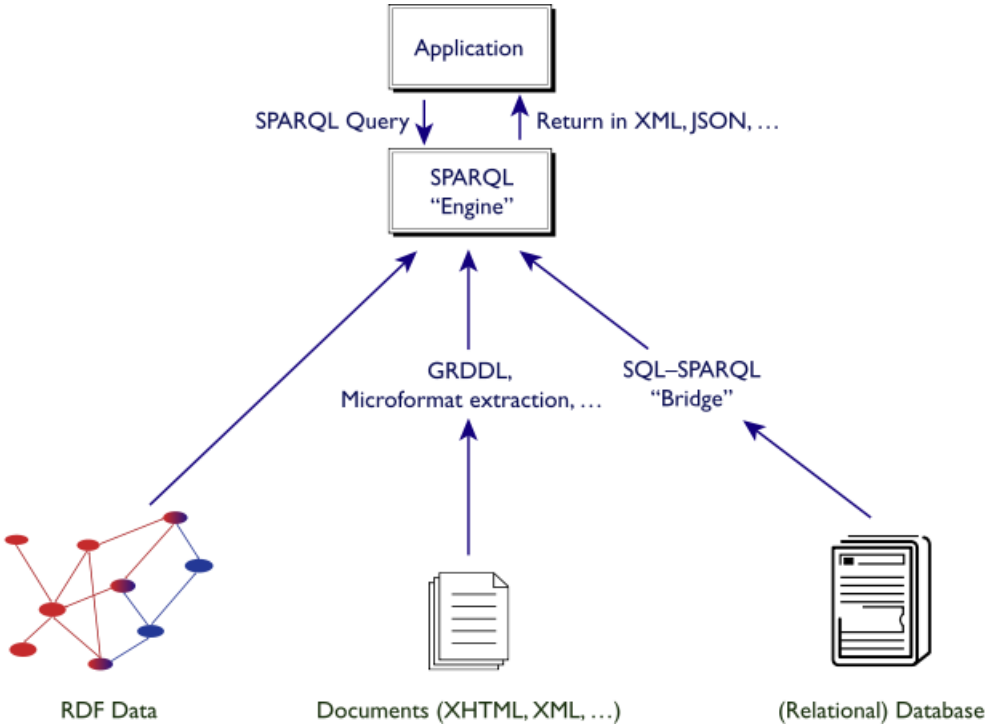
- yields, by running the file through a processor:

```
<http://uri.to.newsitem>
  dc:date              "March 23, 2004";
  dc:title             "Rollers hit casino for £1.3m;
  dc:creator           "Steve Bird";
  dcmtype:MovingImage <http://www.a.b.c/d.avi>.
```

Linking to SQL

- A huge amount of data in Relational Databases
- Although tools exist, it is not feasible to *convert* that data into RDF
- Instead: SQL ⇋ RDF "bridges" are being developed:
  - *a query to RDF data is transformed into SQL on-the-fly*
  - *the modalities are governed by small, local ontologies or rules*
- An active area of development, on the radar screen of W3C!

SPARQL as a unifying point?



Missing features, functionalities…

- Everybody has a favorite item, ie, the list tends to infinite…
- W3C is a *standardization* body, and has to look at where a consensus can be found

Rules

- OWL-DL and OWL-Lite are based on Description Logic
- There are things that DL cannot express
  - *(though there are things that are difficult to express with rules and easy in DL…)*
- A well known examples is Horn rules (eg, the "uncle" relationship):
  - *$(P_1 \wedge P_2 \wedge \ldots) \rightarrow C$*
  - *e.g.: for any «X», «Y» and «Z»: "if «Y» is a parent of «X», and «Z» is a brother of «Y» then «Z» is the uncle of «X»"*
- Several attempts already to combine Semantic Web with Rules (Metalog, RuleML, SWRL, WRL, cwm, …)

Some typical use cases

- Negotiate eBusiness contracts across platforms: supply vendor-neutral representation of your business rules so that others may find you
- Describe privacy requirements and policies, and let clients “merge” those (e.g., when paying with a credit card)
- Medical decision support, combining rules on diagnoses, drug prescription conditions, etc,
- Extend OWL with rule-based statements (e.g., the uncle example)

But: it is not easy!

- From a theoretical viewpoint, Description Logic and Logic Programming are different:
  - *DL is based on FOL Model Theory, while LP not exactly*
  - *Open vs. Closed Worlds, monotonicity vs. non-monotonicity: OWL operates on an Open World, Rules usually don't*

  …hence it is not easy to combine these
- Rule systems often operate with procedural rules ("execute this and this Java procedure if…")

Rules Interchange Format Working Group

- The W3C Working Group started at the beginning of November 2005
- Work is planned in two “phases”:
  1. *construct an extensible format for rule interchange with simple rule systems*
  2. *define more complex extensions*
- Great interest from financial services, business rules, life science community, …
- Work is going on!

“Light” ontologies

- For a number of applications RDFS is not enough, but even OWL Lite is too much
  - *OWL-Lite needs a DL reasoner to operate properly*
- There may be a need for a “light” version of OWL, just a few extra possibilities v.a.v. RDFS
- There are a number of proposals, papers, prototypes around: RDFS++, OWL Feather, pD*, …
  - *pD*, for example, has property characterization (symmetric, transitive, inverse), class and property equivalence, and property restrictions with some or all values*
- This might consolidate in the coming years

Revisions of RDF and OWL?

- Such specifications have their own life
- Missing features come up, errors show up
- There will probably be a next version at some point

Revision of the RDF model?

- Some restrictions in RDF may be unnecessary (bNodes as predicates, literals as subject, …)
- Issue of “named graph”: possibility to give a URI to a set of triplets and make statements on those
- Syntax issues in RDF/XML (eg, QNames in properties)
- Alternative XML serializations?
- Add a time tag to statements? A probability value? A measure of “fuzzyness”?
- Internationalization issues with literals (how do I set “bidi” writing?)

These are just ideas floating around…

Revision of OWL? (OWL 1.1)

- There is a group working on this (outside W3C for now)
- Small additions to the current OWL:
  - *"qualified cardinality restrictions" (i.e., "class instance must have two black cats")*
  - *disjoint properties*
  - *reflexive, irreflexive properties*
  - *own datatype construct instead of complex XML Schema datatypes*
  - *some syntactic sugar (eg, disjoint union)*
  - *…*
- At this moment not yet decided how, if, and when this would become a W3C document

Other items…

- Fuzzy logic
  - *look at alternatives of Description Logic based on fuzzy logic*
  - *alternatively, extend RDF(S) with fuzzy notions*
- Probabilistic statements
  - *have an OWL class membership with a specific probability*
  - *combine reasoners with Bayesian networks*
- Security, trust, provenance
  - *combining cryptographic techniques with the RDF model, sign a portion of the graph, etc*
- Ontology merging, alignment, term equivalences, versioning, development, …
- etc

(Need a new PhD topic?)

A major problem: messaging

- Some of the messaging on Semantic Web has gone terribly wrong ☹. See these statements:
  - *"the Semantic Web is a reincarnation of Artificial Intelligence on the Web"*
  - *"it relies on giant, centrally controlled ontologies for "meaning" (as opposed to a democratic, bottom–up control of terms)"*
  - *"one has to add metadata to all Web pages, convert all relational databases, and XML data to use the Semantic Web"*
  - *"it is just an ugly application of XML"*
  - *"one has to learn formal logic, knowledge representation techniques, description logic, etc, to use it"*
  - *"it is, essentially, an academic project, of no interest for industry"*
  - *…*
- Some simple messages should come to the fore!

RDF ≠ RDF/XML!

- *RDF is a model*, and RDF/XML is only *one* possible serialization thereof
  - *lots of people prefer, for example, Turtle*
  - *a good percentage of the tools have Turtle parsers, too!*
- The model is, after all, simple: interchange format for Web resources. That is it ☺!

RDF ≠ RDF/XML! (cont.)

- RDF/XML is indeed a very complex serialization format
- Certainly not the nicest possible XML application
  - *good to know that it was created when XML was not yet final…*
- Again: it is only syntactic sugar!
- One has to emphasize: RDF is *not* an XML application!

RDF is not *that* complex…

- Of course, the formal semantics of RDF *is* complex
- But the average user should not care, it is all "under the hood"
  - *how many users of SQL have ever read its formal semantics?*
  - *it is not much simpler than RDF…*
- *People should "think" in terms of graphs*, the rest is syntactic sugar!

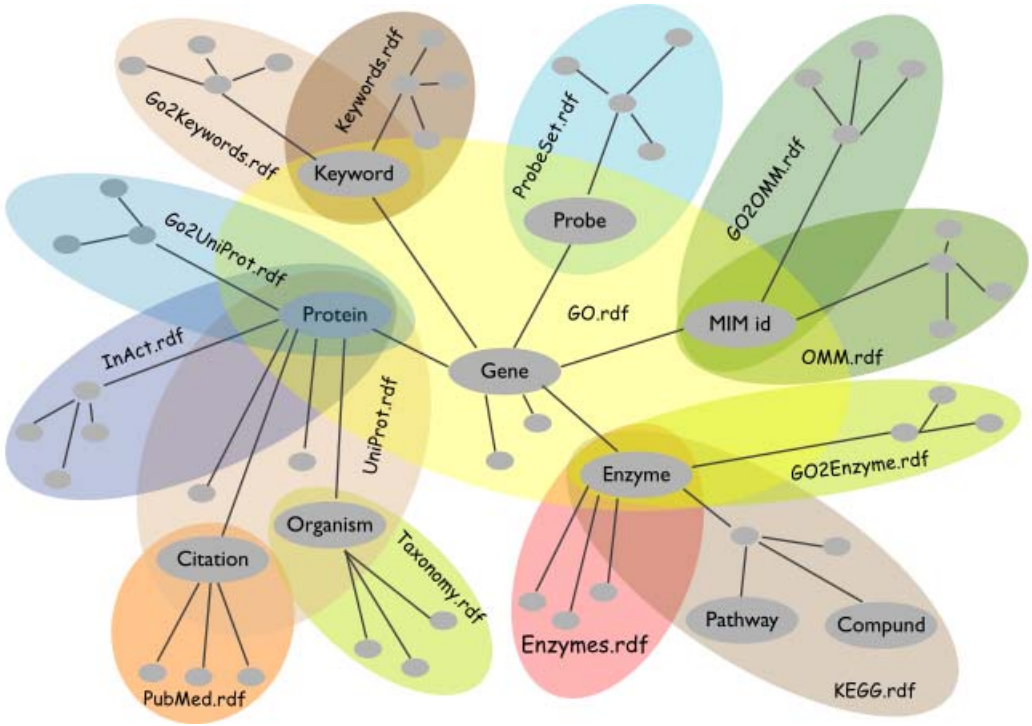
Semantic Web ≠ Ontologies on the Web!

- Ontologies are important, but use them *only when necessary*
- You can be a perfectly decent citizen of the Semantic Web if you do *not* use Ontologies, not even RDFS!
- *The Semantic Web is about integrating data on the Web*; ontologies (and/or rules) are tools to achieve that when necessary
- Remember the “light ontologies” issue?

SW Ontologies ≠ some central, big ontology!

- The “ethos” of the Semantic Web is on *sharing*, ie, sharing possibly small ontologies
- A huge, central ontology would be unmanageable
- OWL includes statements for versioning, for equivalence and disjointness of terms
  - *a revision of those may be necessary, but the goal is clear*
- The practice:
  - *SW applications using ontologies always mix large number of ontologies and vocabularies (FOAF, DC, and others)*
  - *the real advantage comes from this mix: that is also how new relationships may be discovered*

The mix of ontologies…



Semantic Web ≠ an academic research only!

- SW has indeed a strong foundation in research results
- But remember:
  - *(1) the Web was born at CERN…*
  - *(2) …was first picked up by high energy physicists…*
  - *(3) …then by academia at large…*
  - *(4) …then by small businesses and start-ups…*
  - *(5) "big business" came only later!*
- network effect kicked in early…
- Semantic Web is now at #4, and moving to #5!

May start with small communities

- The needs of a deployment application area:
  - *have serious problem or opportunity*
  - *have the intellectual interest to pick up new things*
  - *have motivation to fix the problem*
  - *its data connects to other application areas*
  - *have an influence as a showcase for others*
- The high energy physics community played this role for the Web in the 90's

Some RDF deployment areas (cont)

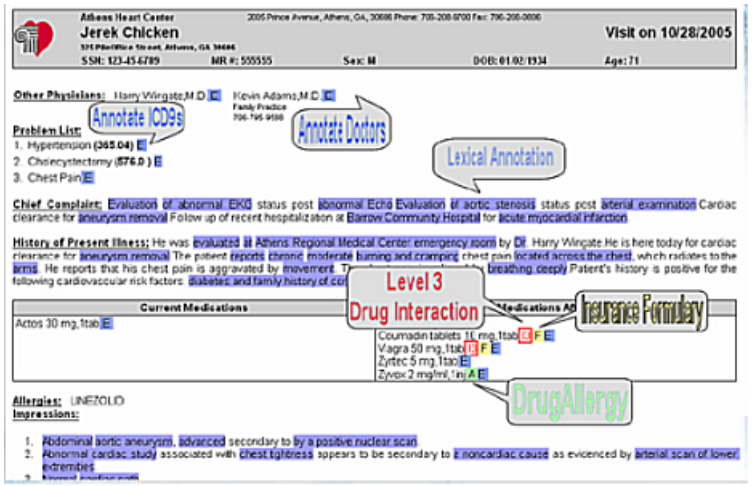
- Some deployment areas are already very active: Health Care and Life Sciences, Digital Libraries, Defense
  - *also at W3C, in the form of an Interest Group for HCLS*
- Others are coming to the fore: eGovernment, energy sector (oil industry), financial services, …

The “corporate” landscape is moving

- See, for example, the Semantic Technology Conference series
  - *not a scientific conference, but commercial people making real money!*
  - *speakers in 2006: from IBM, Cisco, BellSouth, GE, Walt Disney, Nokia, Oracle, …*
  - *not all referring to Semantic Web (eg, RDF, OWL, …) but semantics in general*
  - *but they might come around!*
- Major companies offer (or will offer) Semantic Web tools or systems using Semantic Web: Adobe, Oracle, IBM, HP, Software AG, WebMethods, Northrop Gruman, Altova, …
- “Corporate Semantic Web” listed as major technology by Gartner in 2006

Applications are not always very complex…

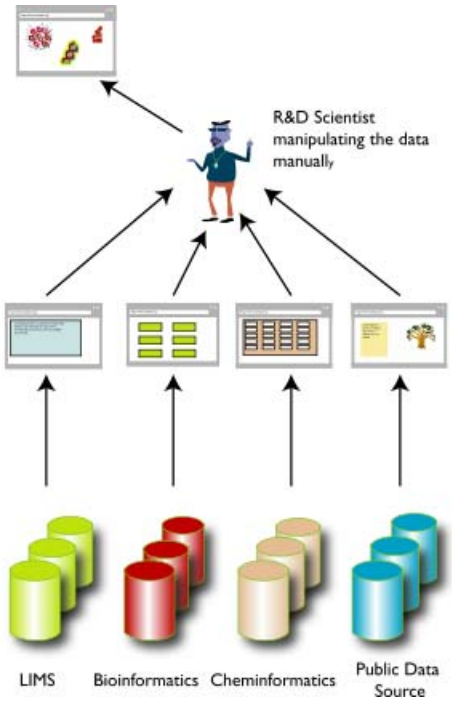
- Eg: simple semantic annotations of patients' data greatly enhances communications among doctors
- What is needed: some simple ontologies, an RDFa/microformat type editing environment
- Simple but powerful!

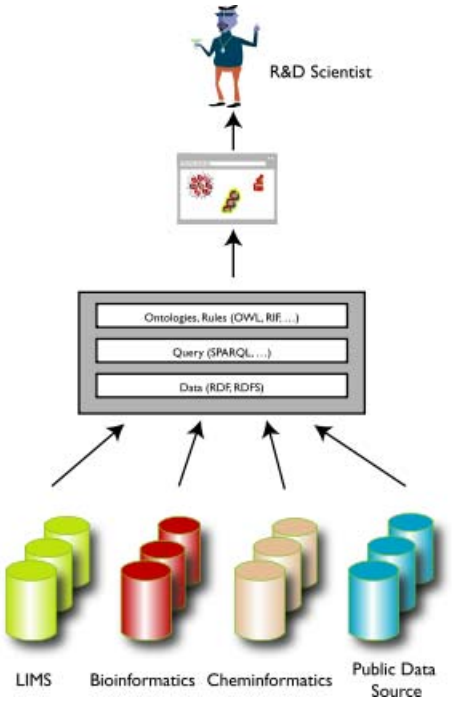


Data integration

- Data integration comes to the fore as one of *the* SW Application areas
- Very important for large application areas (life sciences, energy sector, eGovernment, financial institutions), as well as everyday applications (eg, reconciliation of calendar data)
- Life sciences example:
  - *data in different labs…*
  - *data aimed at scientists, managers, clinical trial participants…*
  - *large scale public ontologies (genes, proteins, antibodies, …)*
  - *different formats (databases, spreadsheets, XML data, XHTML pages)*
  - *etc*

Life Sciences (cont.)





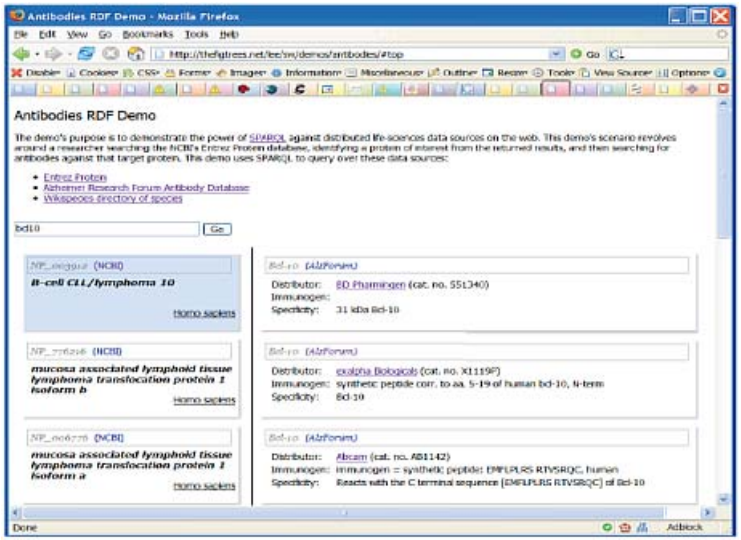
General approach

1. Map the various data onto RDF
   - *assign URI-s to your data*
   - *"mapping" may mean on-the-fly SPARQL to SQL conversion, "scraping", etc*
2. Merge the resulting RDF graphs (with a possible help of ontologies, rules, etc, to combine the terms)
3. Start making queries on the whole!

- Remember the role of SPARQL?

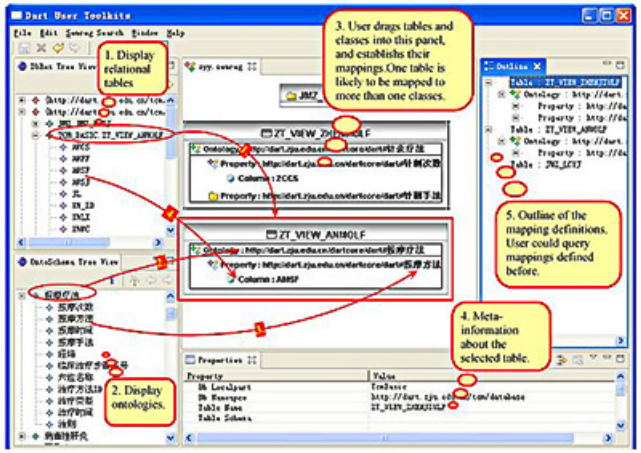
Example: antibodies demo

- Scenario: find the known antibodies for a protein in a specific species
- Combine (“scrape”…) three different data sources
- Use SPARQL as an integration tool (see also demo online)



There has been lots of R&D

- Boeing, MITRE Corp., Elsevier, EU Projects like Sculpteur and Artiste, national projects like MuseoSuomi, DartGrid from Zheijang University, …
- Developments are under way at various places in the area

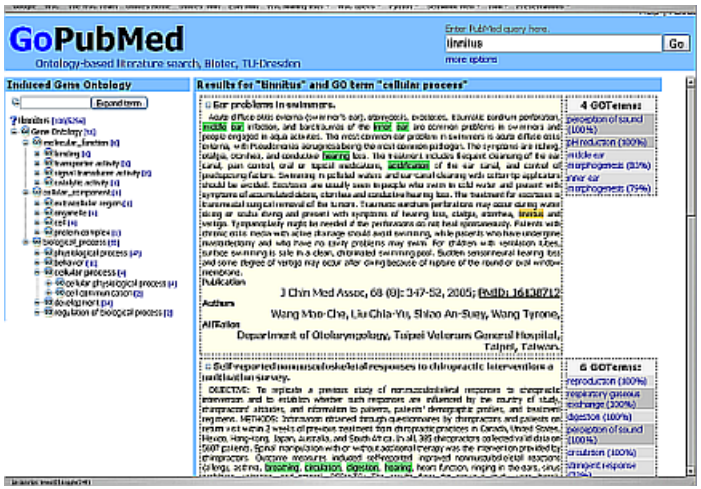


Portals

- Vodafone's Live Mobile Portal
  - *search application (e.g. ringtone, game, picture) using RDF*
    - page views per download decreased 50%
    - ringtone up 20% in 2 months
- A number of other portal examples: Sun's White Paper Collections and System Handbook collections; Nokia's S60 support portal; Harper's Online magazine linking items via an internal ontology; Oracle's virtual press room; Opera's community site,…

Improved Search via Ontology: GoPubMed

- Improved search on top of pubmed.org
  - *search results are ranked using the specialized ontologies*
  - *extra search terms are generated and terms are highlighted*
- Importance of *domain specific ontologies* for search improvement



Thank you for your attention!

(These slides are publicly available on http://www.w3.org/2006/Talks/1016-Beijing-IH/)