ontotext

GATE

Module 13

Introduction to Semantic Technology, Ontologies and the Semantic Web

ontotext

GATE

Module 13 Outline

10.30-12.30	• Introduction to the Semantic Web • Ontologies • Semantic Web related standards
12.30-14.00	Lunch break
14.00-16.00	• Semantic Web related standards (part II) • Some Application of Semantic Technologies • Tools
16.00-16.30	Coffee

ontotext

GATE

About this tutorial

- The Web that we know → The Semantic Web
- Ontologies
- Semantic Web related standards
- Some Applications of Semantic Technologies
- Tools

Introduction to the Semantic Web

ontotext

GATE

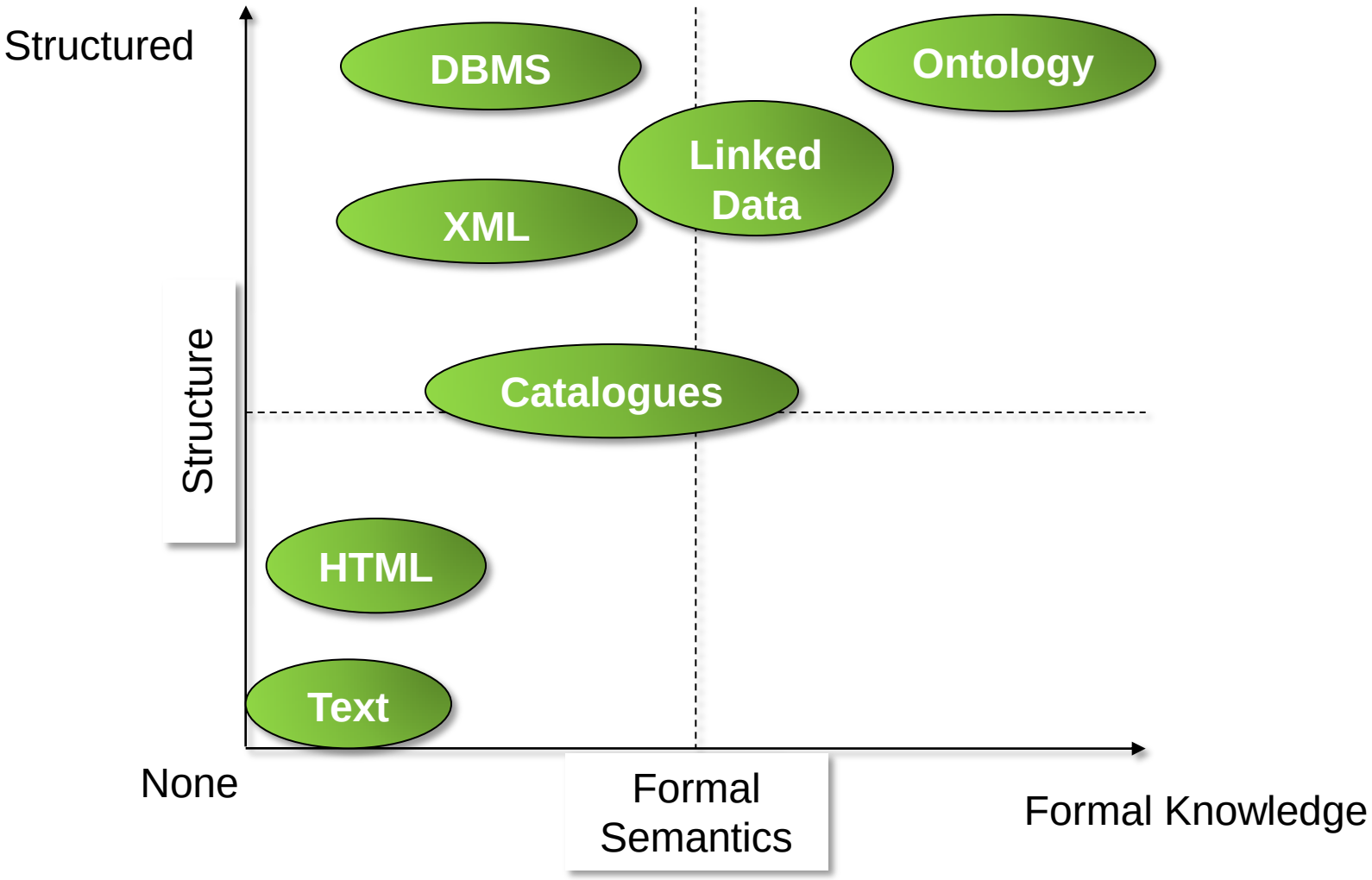
The Web as we know it

- *Target consumers*: humans
  - web 2.0 mashups provide *some* improvement
  - Rules about the *structure* and *visualisation* of information, but not about its *intended meaning*
  - Intelligent agents can't easily use the information
- *Granularity*: document
  - One giant distributed *filesystem* of documents
  - One *document* can link to other documents
- *Integration & reuse*: very limited
  - Cannot be easily automated
  - Web 2.0 mashups provide *some* improvement

Some problems with the current Web

- Finding information
- Data granularity
- Resource identification
- Data aggregation & reuse
- Data integration
- Inference of new information

Types of Data



The need for a smarter Web

- *"The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation."* (Tim Berners-Lee, 2001)

The need for a smarter Web (2)

- *"PricewaterhouseCoopers believes a Web of data will develop that fully augments the document Web of today. You'll be able to find and take **pieces of data** sets **from different places**, aggregate them without warehousing, and **analyze them in a more straightforward, powerful way** than you can now." (PWC, May 2009)*

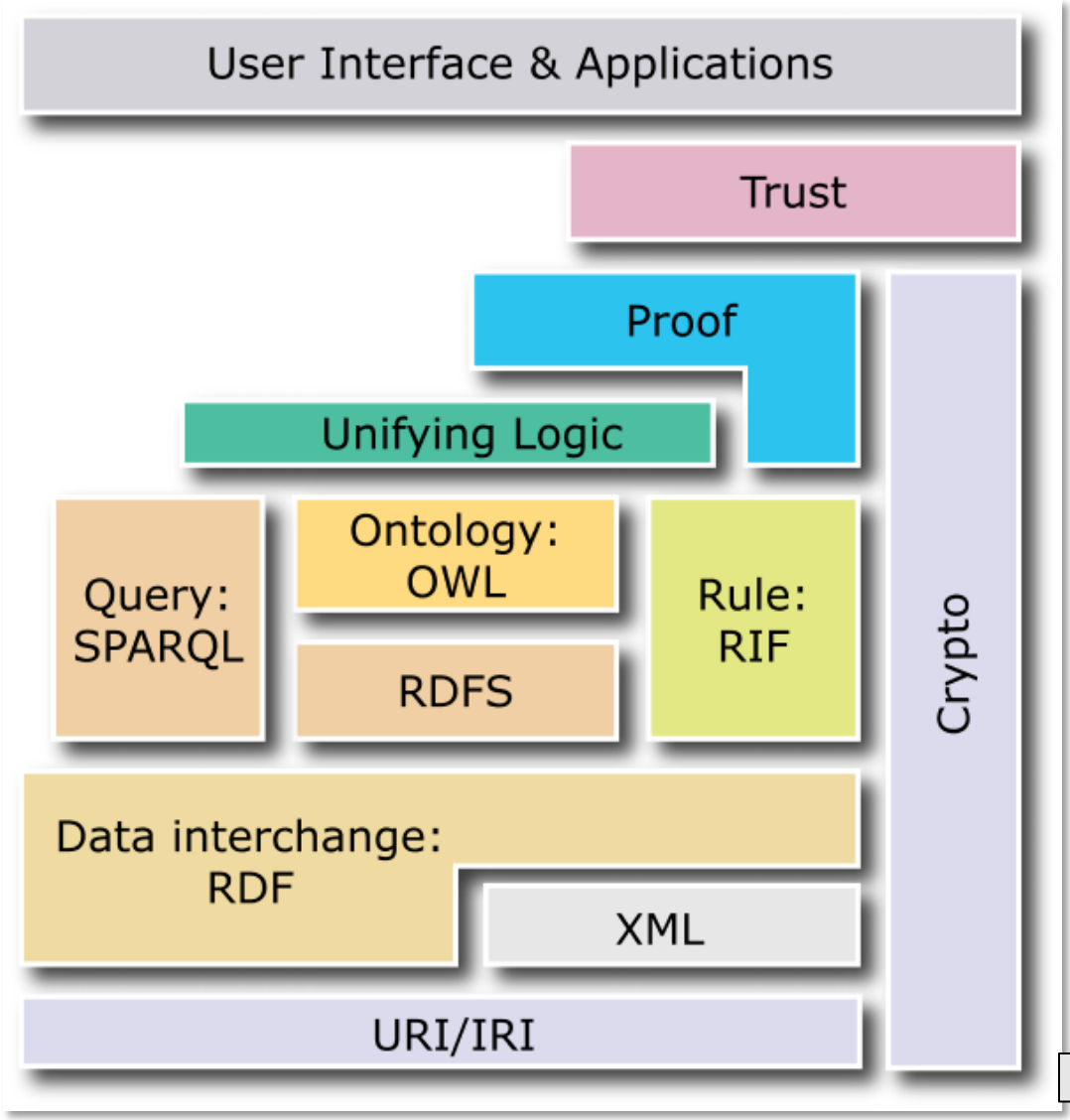
The Semantic Web

- *Target consumers*: intelligent agents
  - Explicit specification of the *intended meaning* information
  - Intelligent agents can make use the information
- *Granularity*: resource/fact
  - One giant distributed *database* of facts about resources
  - One *resource* can be linked (related) to other resources
- *Integration & reuse*: easier
  - Resources have unique identifiers
  - With explicit semantics transformation & integration can be automated

The Semantic Web vision (W3C)

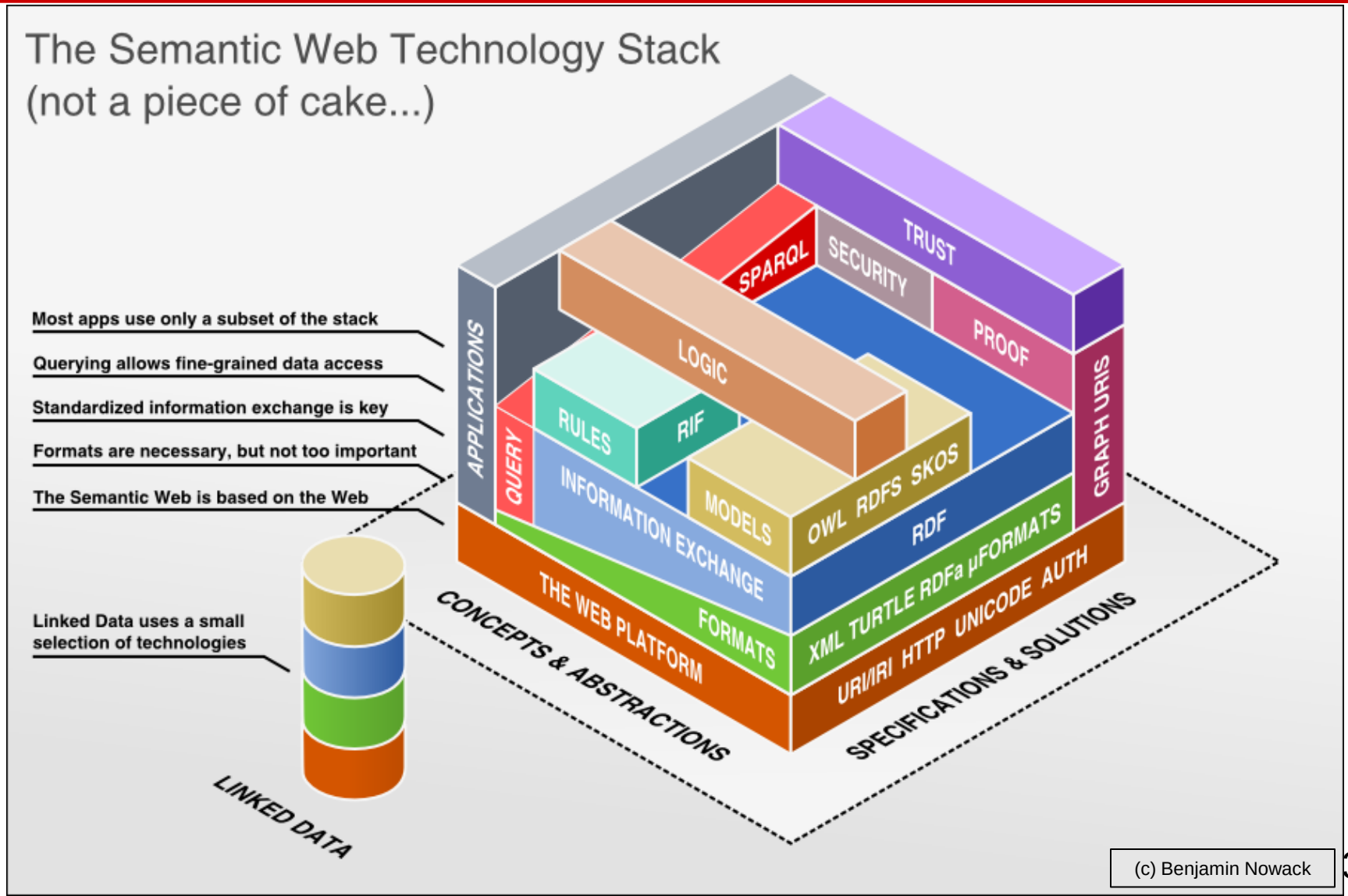
- Extend principles of the Web from documents to data
- Data should be accessed using the general Web architecture (e.g., URI-s, protocols, …)
- Data should be related to one another just as documents are already
- Creation of a common framework that allows
  - Data to be shared and reused across applications
  - Data to be processed automatically
  - New relationships between pieces of data to be inferred

The Semantic Web layer cake

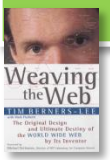


(c) W3C

The Semantic Web layer cake (2)



The Semantic Web timeline



RDF

DAML+OIL

OWL

OWL 2

SPARQL

SPARQL 1.1

RIF

RDFa

SAWSDL

Linked Open Data

POWDER

SKOS

HCLS

RDB2RDF

1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010

Ontologies

What is an ontology

- What is an ontology
  - A formal specification that provides sharable and reusable knowledge representation
  - Examples – taxonomies, thesauri, topic maps, E/R schemata*, formal ontologies
- Ontology specification includes
  - Description of the *concepts* in some domain and their properties
  - Description of the possible *relationships* between the concepts and the *constraints* on how the relationships can be used
  - Sometimes, the *individuals (*members of concepts)

Ontology dimensions (NIST, 2007)

	dimension	examples
Semantic	Degree of structure and formality	• Informal (specified in natural language) • taxonomy or topic hierarchy • very formal - unambiguous description of terms and axioms
	Expressiveness of the representation language	• different logic formalisms have different expressivity (and computational complexity)
	granularity	• simple taxonomies and hierarchies • detailed property descriptions, rules and restrictions
Pragmatic	Intended use	• data integration (of disparate datasources) • represent a natural language vocabulary (lexical ontology) • categorization and classification
	Role of automated reasoning	• is inference of new knowledge required? • simple reasoning (class/subclass transitivity inference) vs. complex reasoning (classification, theorem proving)
	Descriptive vs. prescriptive	• descriptive – less strict characterization, • prescriptive – strict characterization
	Design methodology	• bottom-up vs. top-down
	governance	• are there legal and regulatory implications • is provenance required?

example

class Person

class Woman
subClassOf #Person

class Man
subClassOf #Person
complementOf #Woman

individual John
instanceOf #Man

individual Mary
instanceOf #Woman
hasSpouce #John

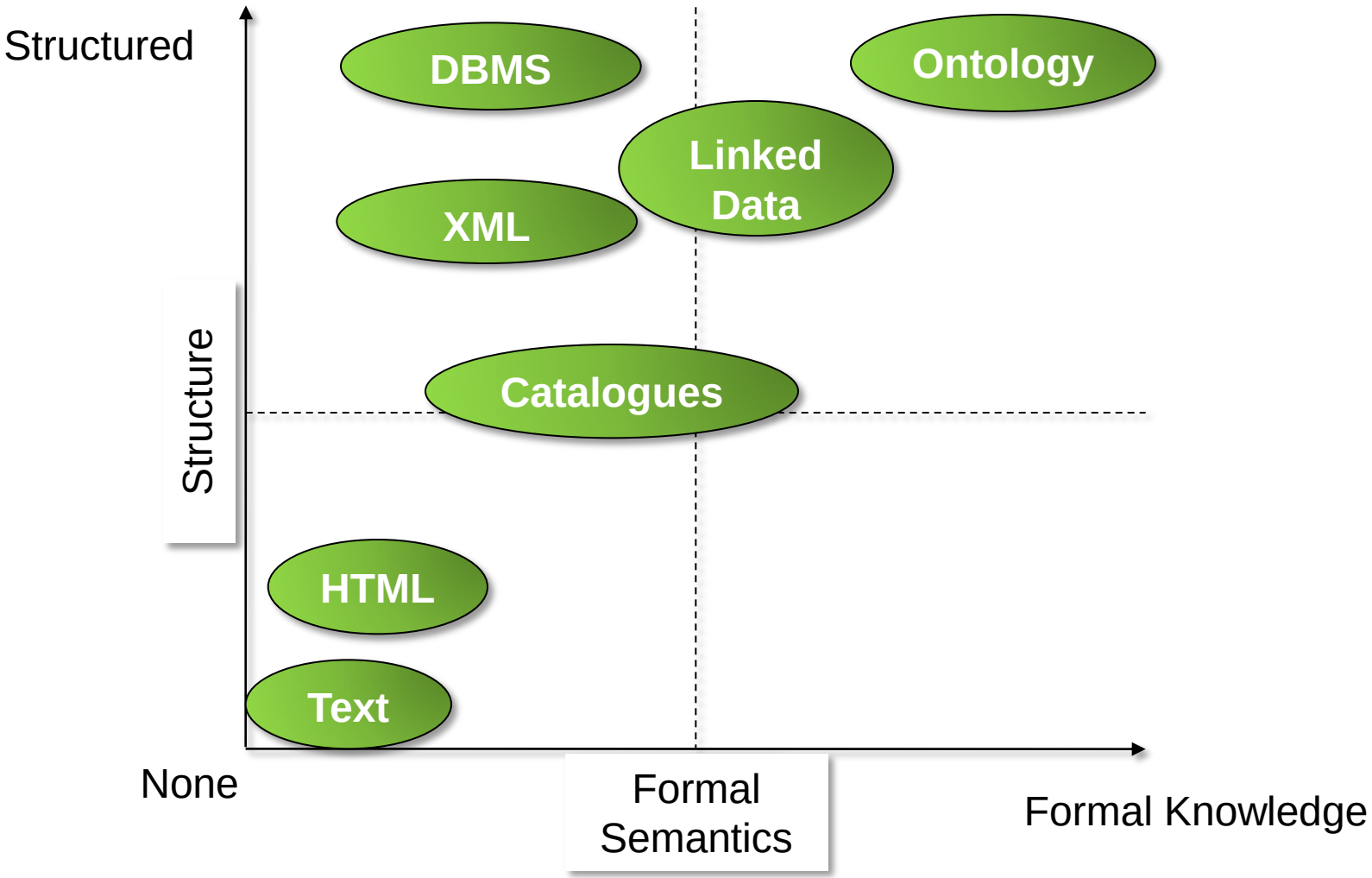
individual Jane
instance Of #Woman
hasParent #John
hasParent #Mary

property hasParent
domain #Person
range #Person
maxCardinality 2

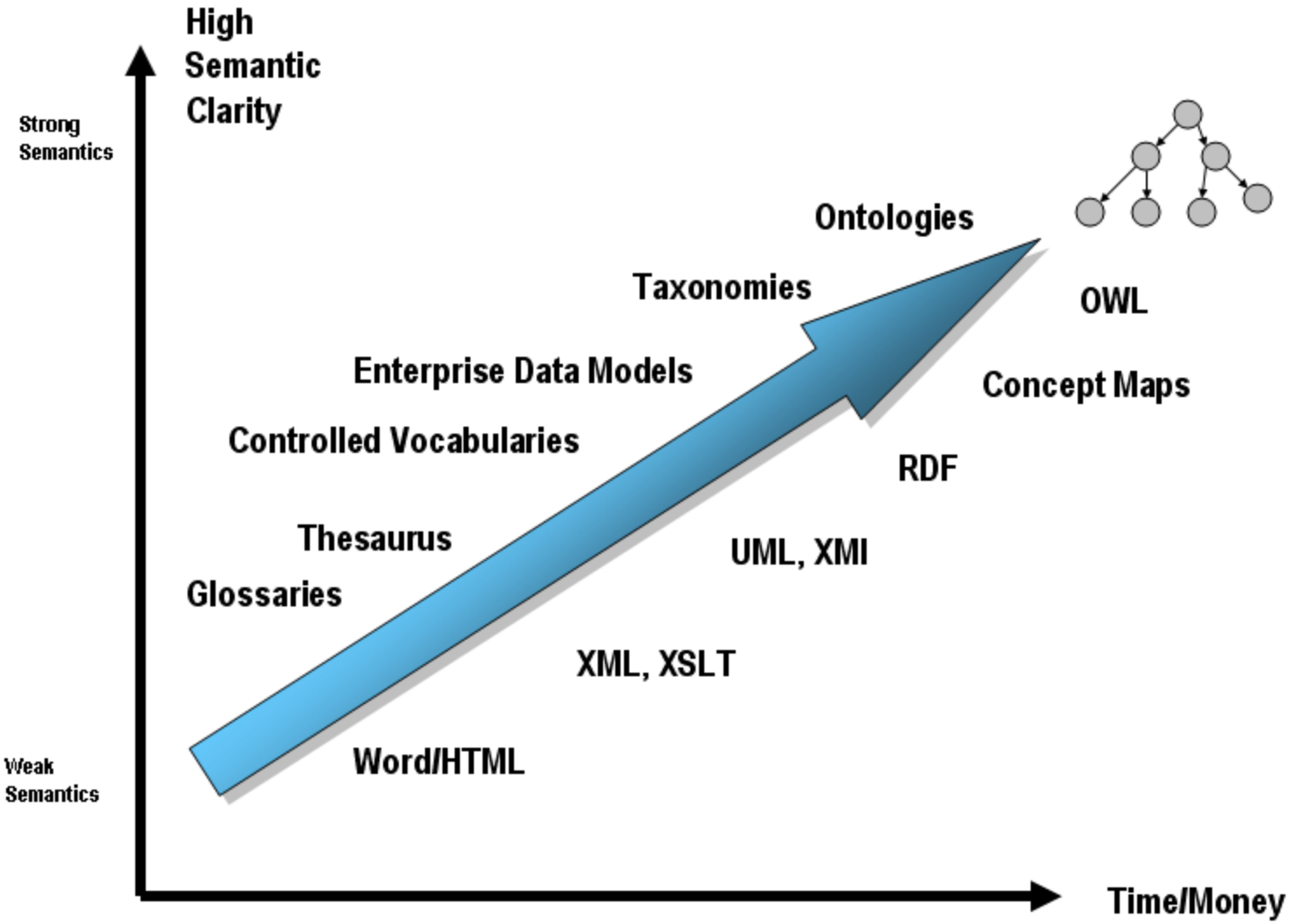
property hasChild
inverseOf #hasParent

property hasSpouce
domain #Person
range #Person
maxCardinality 1
symmetric

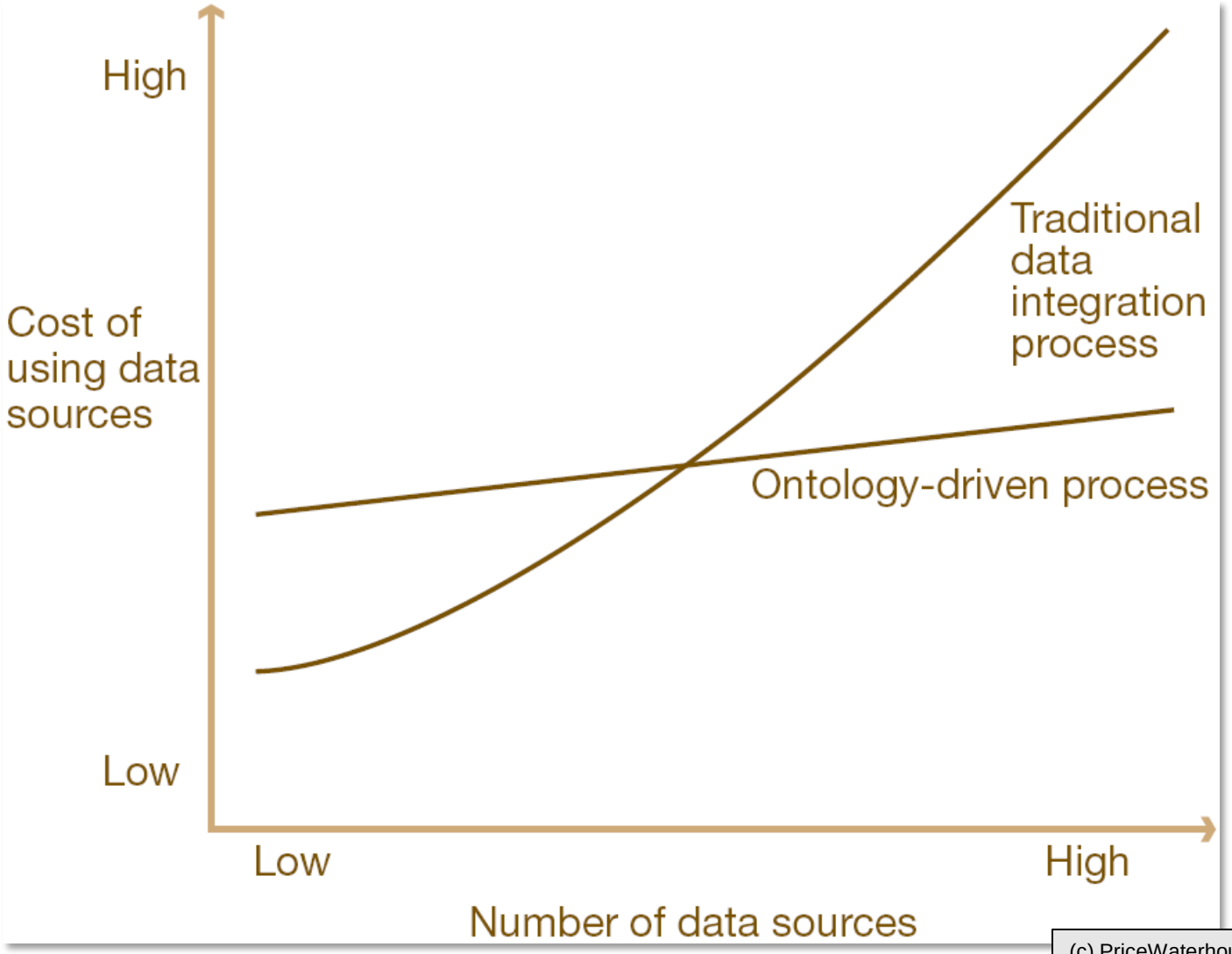
Types of Data



The cost of semantic clarity



Data integration cost



ontotext

GATE

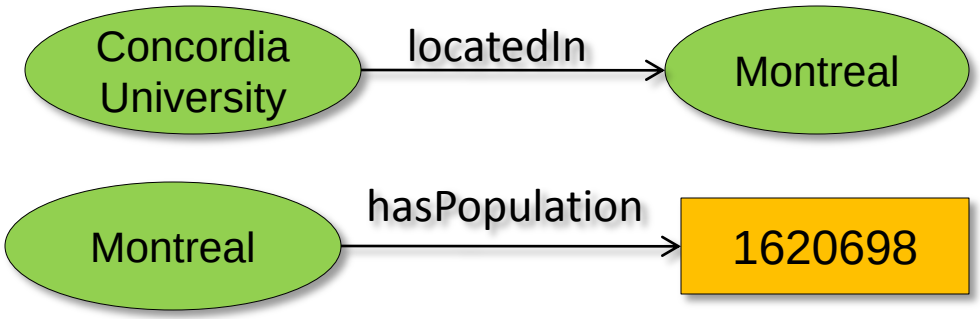
Semantic Web related standards

Resource Description Framework (RDF)

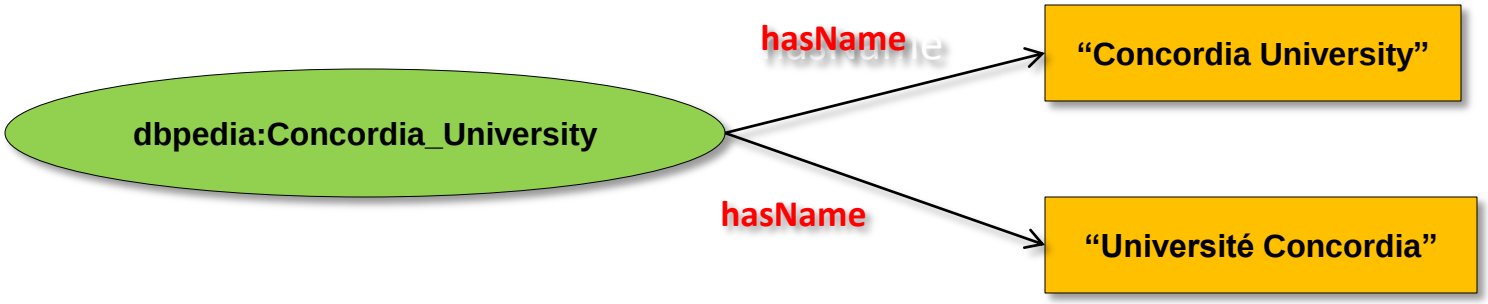
- A simple data model for
  - describing the *semantics* of information in a machine accessible way
  - representing meta-data (data about data)
- A set of representation syntaxes
  - XML (standard) but also N3, Turtle, …
- Building blocks
  - *Resources* (with unique identifiers)
  - *Literals*
  - Named *relations* between pairs of resources (or a resource and a literal)

RDF (2)

- Everything is a triple
  - **Subject** (resource), **Predicate** (relation), **Object** (resource or literal)
- The RDF graph is a collection of triples

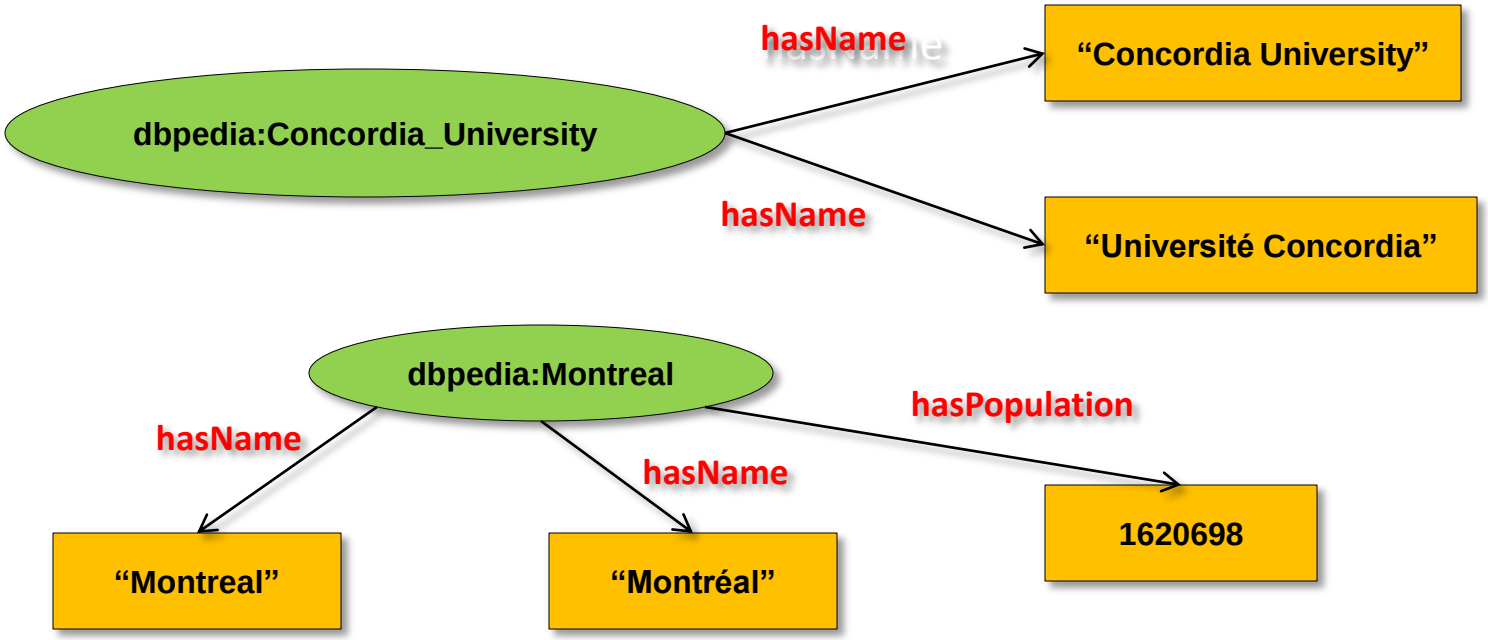


RDF (3)



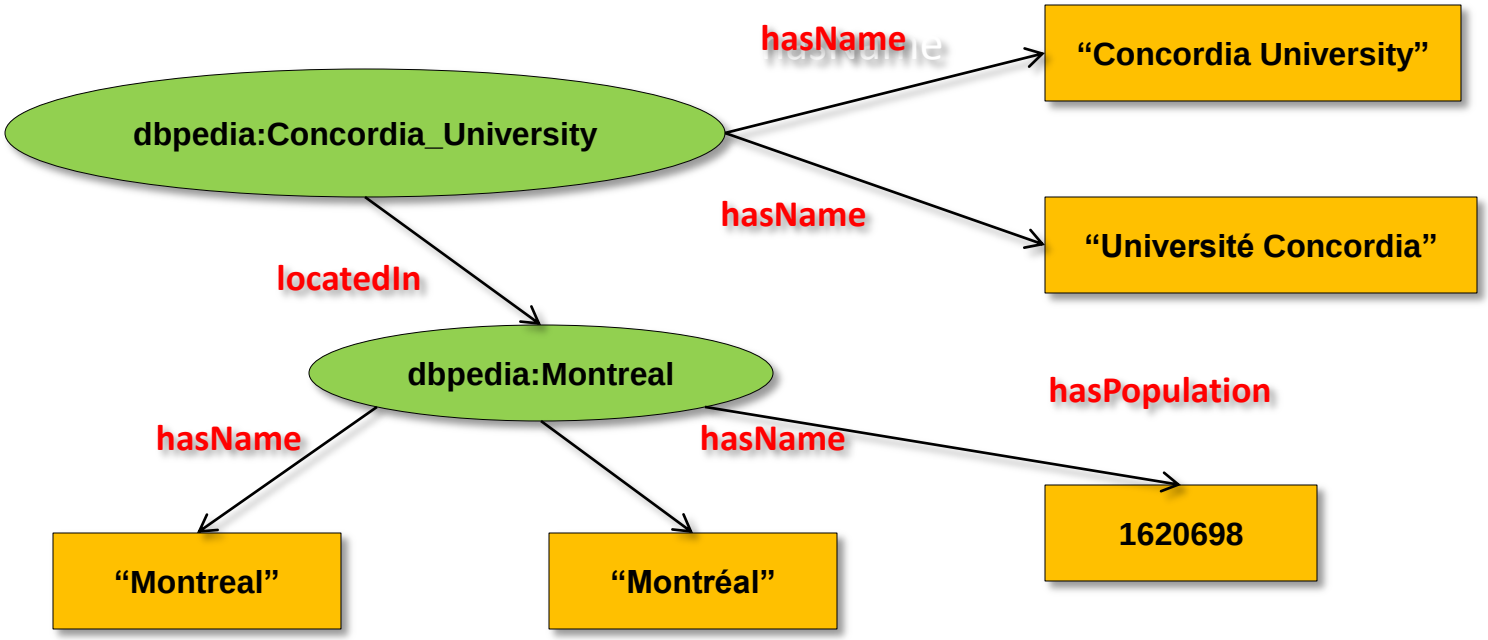
Subject	Predicate	Object
http://dbpedia.org/resource/Concordia_University	hasName	"Concordia University"
http://dbpedia.org/resource/Concordia_University	hasName	"Université Concordia"

RDF (4)



Subject	Predicate	Object
http://dbpedia.org/resource/Montreal	*hasName*	*"Montreal"*
http://dbpedia.org/resource/Montreal	*hasPopulation*	*1620698*
http://dbpedia.org/resource/Montreal	*hasName*	*"Montréal"*
http://dbpedia.org/resource/Concordia_University	**hasName**	**"Concordia University"**
http://dbpedia.org/resource/Concordia_University	**hasName**	**"Université Concordia"**

RDF (5)



Subject	Predicate	Object
http://dbpedia.org/resource/Montreal	hasName	"Montreal"
http://dbpedia.org/resource/Montreal	hasPopulation	1620698
http://dbpedia.org/resource/Montreal	hasName	"Montréal"
http://dbpedia.org/resource/Concordia_University	*locatedIn*	*http://dbpedia.org/resource/Montreal*
http://dbpedia.org/resource/Concordia_University	hasName	"Concordia University"
http://dbpedia.org/resource/Concordia_University	hasName	"Université Concordia"

ontotext

GATE

RDF (6)

- RDF advantages
  - Simple but expressive data model
  - Global identifiers of all resources
    - Remove ambiguity
  - Easier & incremental data integration
    - Can handle incomplete information
    - Open world assumption
  - Schema agility
  - Graph structure
    - Suitable for a large class of tasks
    - Data merging is easier

ontotext

GATE

SPARQL Protocol and RDF Query Language (SPARQL)

- SQL-like query language for RDF data
- Simple protocol for querying remote databases over HTTP
- Query types
  - *select* – projections of variables and expressions
  - *construct* – create triples (or graphs)
  - *ask* – whether a query returns results (result is true/false)
  - *describe* – describe resources in the graph

ontotext

GATE

SPARQL (2)

- Anatomy of a SPARQL query
  - List of namespace prefix shortcuts
  - Query result definition (variables)
  - List of datasets
  - Graph patterns (restrictions)
    - Conjunctions, disjunctions, negation
  - Modifiers
    - Sort order, grouping

SPARQL (3)

```
PREFIX rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX dbpedia: <http://dbpedia.org/resource/>
PREFIX dbp-ont: <http://dbpedia.org/ontology/>

SELECT DISTINCT ?university ?students
WHERE {
   ?university rdf:type dbpedia:Academic_institution .
   ?university dbp-ont:numberOfStudents ?students .
   ?university dbp-ont:city dbpedia:Montreal .
   FILTER (?students > 5000)
}
ORDER BY DESC (?students)
```

FactForge

SPARQL Query

Results for PREFIX rdf:<http://www.w3... (6) View as Exhibit Downlo

university	students
dbpedia:Con_U	43944
dbpedia:HEC_Montreal	12000
dbpedia:Collège_Ahuntsic	10100
dbpedia:John_Molson_School_of_Business	8026
dbpedia:CEGEP_Vanier	6100
dbpedia:Cégep_du_Vieux_Montréal	6100

RDF Schema (RDFS)

- RDFS provides means for
  - Defining *Classes* and *Properties*
  - Defining hierarchies (of classes and properties)
- RDFS differs from XML Schema (XSD)
  - Open World Assumption
  - RDFS is about describing resources, not about validation
- Entailment rules (axioms)
  - Infer new triples

ontotext

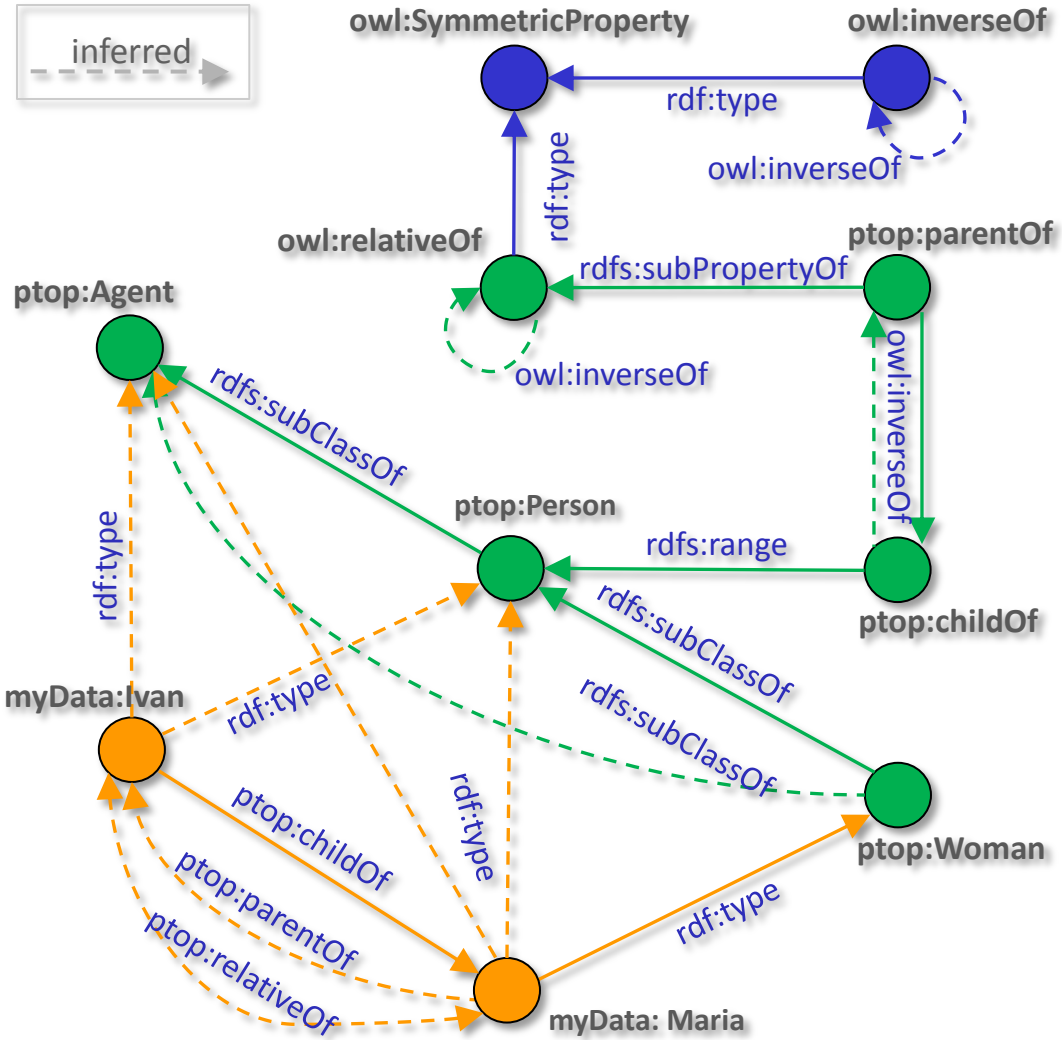
GATE

RDFS (2)

- Entailment rules

1:	$s\ p\ o$ (if o is a literal)		$\Rightarrow$ _:n rdf:type rdfs:Literal
2:	p rdfs:domain x	& $s\ p\ o$	$\Rightarrow$ s rdf:type x
3:	p rdfs:range x	& $s\ p\ o$	$\Rightarrow$ o rdf:type x
4a:	$s\ p\ o$		$\Rightarrow$ s rdf:type rdfs:Resource
4b:	$s\ p\ o$		$\Rightarrow$ o rdf:type rdfs:Resource
5:	p rdfs:subPropertyOf q	& q rdfs:subPropertyOf r	$\Rightarrow$ p rdfs:subPropertyOf r
6:	p rdf:type rdf:Property		$\Rightarrow$ p rdfs:subPropertyOf p
7:	$s\ p\ o$	& p rdfs:subPropertyOf q	$\Rightarrow$ $s\ q\ o$
8:	s rdf:type rdfs:Class		$\Rightarrow$ s rdfs:subClassOf rdfs:Resource
9:	s rdf:type x	& x rdfs:subClassOf y	$\Rightarrow$ s rdf:type y
10:	s rdf:type rdfs:Class		$\Rightarrow$ s rdfs:subClassOf s
11:	x rdfs:subClassOf y	& y rdfs:subClassof z	$\Rightarrow$ x rdfs:subClassOf z
12:	p rdf:type rdfs:ContainerMembershipProperty		$\Rightarrow$ p rdfs:subPropertyOf rdfs:member
13:	o rdf:type rdfs:Datatype		$\Rightarrow$ o rdfs:subClassOf rdfs:Literal

RDFS (3)


inferred
owl:SymmetricProperty
owl:inverseOf
rdf:type
owl:inverseOf
rdf:type
owl:relativeOf
ptop:parentOf
rdfs:subPropertyOf
ptop:Agent
owl:inverseOf
owl:inverseOf
rdfs:subClassOf
ptop:Person
rdf:type
rdfs:range
ptop:childOf
rdfs:subClassOf
myData:Ivan
rdf:type
rdfs:subClassOf
ptop:childOf
rdf:type
ptop:Woman
ptop:parentOf
ptop:relativeOf
rdf:type
myData: Maria

Web Ontology Language (OWL)

- More expressive than RDFS
  - Identity equivalence/difference
    - *sameAs, differentFrom, equivalentClass/Property*
  - More expressive class definitions
    - Class intersection, union, complement, disjointness
    - Cardinality restrictions
  - More expressive property definitions
    - Object/Datatype properties
    - Transitive, functional, symmetric, inverse properties
    - Value restrictions

OWL (2)

- What can be done with OWL?
  - *Consistency checks* - Are there contradictions in the logical model
  - *Satisfiability checks* - Are there classes that cannot have any instances?
  - *Classification* - What is the type of a particular instance?
- OWL sublanguages
  - ***OWL Lite*** – low expressiveness / low computational complexity
  - ***OWL DL*** – high expressiveness / decidable & complete
  - ***OWL Full*** – max expressiveness / no guarantees

ontotext

GATE

Simple Knowledge Organization System (SKOS)

- Represent simple knowledge bases
  - Taxonomies, thesauri, classifications, vocabularies, etc
  - RDF based
- SKOS essentials
  - *Concepts* – describe entities
  - *Labels* – lexical means to refer to a concept
  - *Relationships* – hierarchy (*skos:broader*, *skos:narrower*) or relatedness (*skos:related*)
  - *Notes* – human readable documentation
  - *Schemes* – compilation of concepts

Rule Interchange Format (RIF)

- Goals
  - Define a framework for rule languages for the Semantic Web
    - If <*condition*> then <*conclusion*>
  - Define a standard format/syntax for interchanging rules
- Several dialects defined so far
  - different expressivity & complexity
  - ***RIF BLD*** (Basic Logic Dialect)
    - Rule condition/conclusion are *monotonic* (like in OWL/RDF)
  - ***RIF PRD*** (Production Rule dialect)
    - Condition/conclusion are *non-monotonic* (retraction)

RDFa

- A set of XHTML attributes
  - Embed RDF annotations in web pages
  - The DC and FOAF vocabularies can be easily used for most simple annotations
    - Creator, title, contact info, …

Original HTML

```
<div >
  <ul>
    <li>
      <a href="http://example.com/bob/">Adam</a>
    </li>
    <li>
      <a href="http://example.com/eve/">Eve</a>
    </li>
  </ul>
</div>
```

Annotated XHTML

```
<div xmlns:foaf="http://xmlns.com/foaf/0.1/">
  <ul>
    <li typeof="foaf:Person">
      <a property="foaf:name“
         rel="foaf:homepage"
         href="http://example.com/bob/">Adam</a>
    </li>
    <li typeof="foaf:Person">
      <a property="foaf:name"
         rel="foaf:homepage"
         href="http://example.com/eve/">Eve</a>
    </li>
  </ul>
</div>
```

RDFa (2)

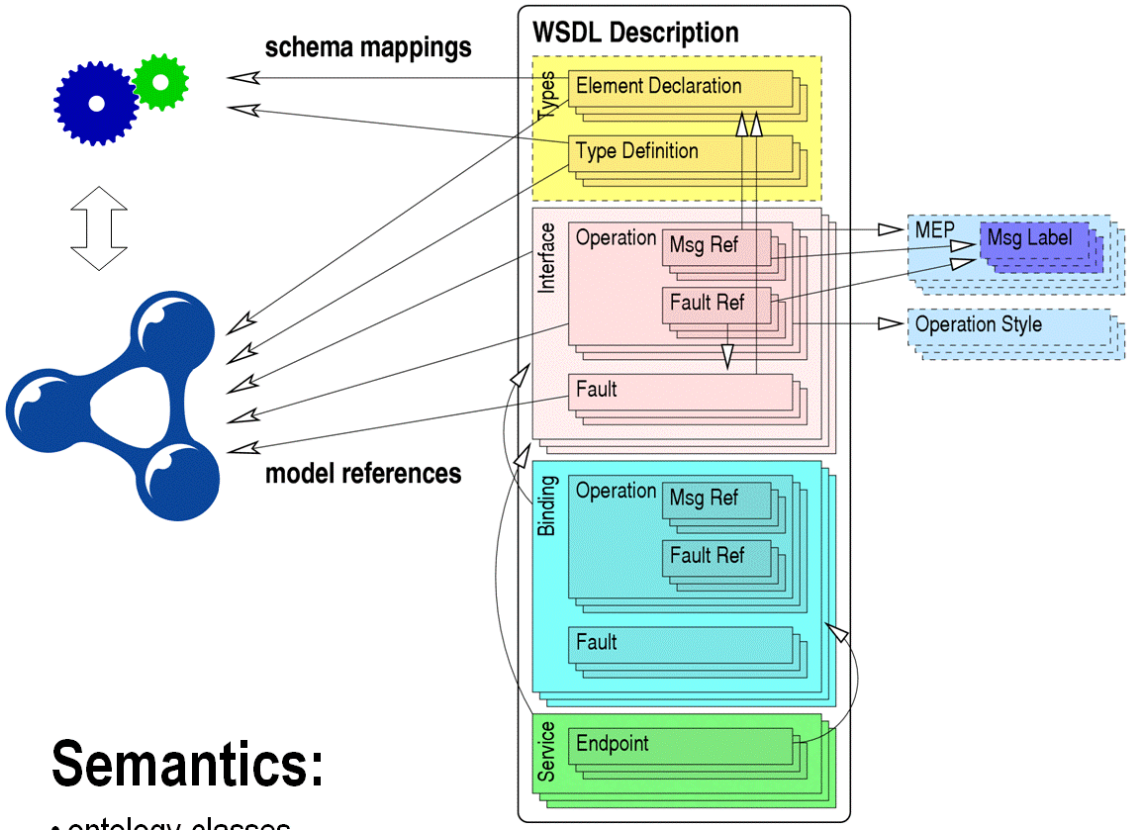
- Less than 5% of web pages have RDFa annotations (Google, 2010)
- But many organizations already publish or consume RDFa:
  - Google, Yahoo
  - Facebook, MySpace, LinkedIn
  - Best Buy, Tesco, O'Reilly,
  - SlideShare, Digg
  - WhiteHouse.gov, Library of Congress, UK government
  - Newsweek, BBC

Semantic Annotations for WSDL (SAWSDL)

- Semantic annotations for Web Services
  - Embed semantic annotations within WSDL 2.0 service descriptions
  - Annotations are agnostic of the ontology language
  - Based on the extensibility mechanism of WSDL 2.0
    - Existing tools that do not understand the semantics will just ignore the annotations
- Elements
  - *Model reference* – associate a WSDL element with a concept in some ontology
  - *Lifting/lowering schema* – mappings between the XML and the ontology data

Web Services and Business Process Management (2)

- SAWSDL at a glance



Semantics:

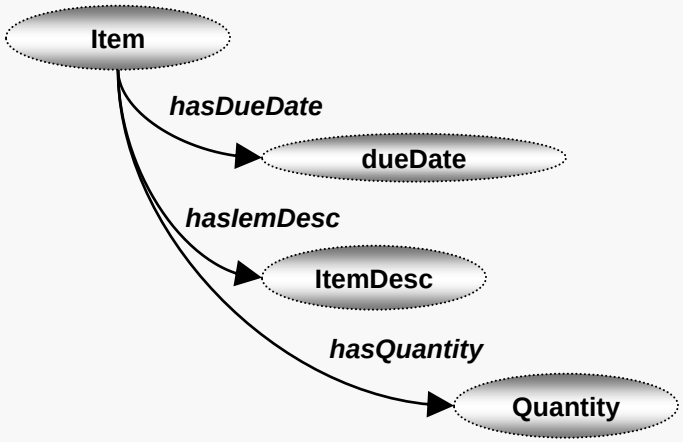
- ontology classes
  - discovery, composition
  - filtering, ranking
- lifting/lowering mappings
  - mediation, invocation
- functionality categories
  - publishing, discovery, composition
- anything, really

Web Services and Business Process Management (3)

- SAWSDL example

```
<complexType name="POItem" >
 <all>
  <element name="dueDate" nillable="true" type="dateTime"
    sawsdl:modelReference=”
      http://www.w3.org/2002/ws/sawsdl/spec/ontology/purchaseorder#DueDate”/>
  <element name="qty" type="float"
    sawsdl:modelReference=”
      http://www.w3.org/2002/ws/sawsdl/spec/ontology/purchaseorder#Quantity”/>
  <element name="EANCode" nillable="true" type="string"
    sawsdl:modelReference=”
      http://www.w3.org/2002/ws/sawsdl/spec/ontology/purchaseorder#ItemCode”/>
  <element name="itemDesc" nillable="true" type="string"
    sawsdl:modelReference=”
   http://www.w3.org/2002/ws/sawsdl/spec/ontology/purchaseorder#ItemDesc” />
  </all>
</complexType>
```

WSDL complex type element



OWL ontology

ontotext

GATE

Some Applications of Semantic Technologies

ontotext

GATE

Linked Data

"To make the Semantic Web a reality, it is necessary to have a large volume of data available on the Web in a standard, reachable and manageable format. In addition the relationships among data also need to be made available. This collection of interrelated data on the Web can also be referred to as ***Linked Data****. Linked Data lies at the heart of the Semantic Web: large scale integration of, and reasoning on, data on the Web."* (W3C)

- *Linked Data* is a set of principles that allows publishing, querying and browsing of RDF data, distributed across different servers
  - similar to the way HTML is currently published & consumed

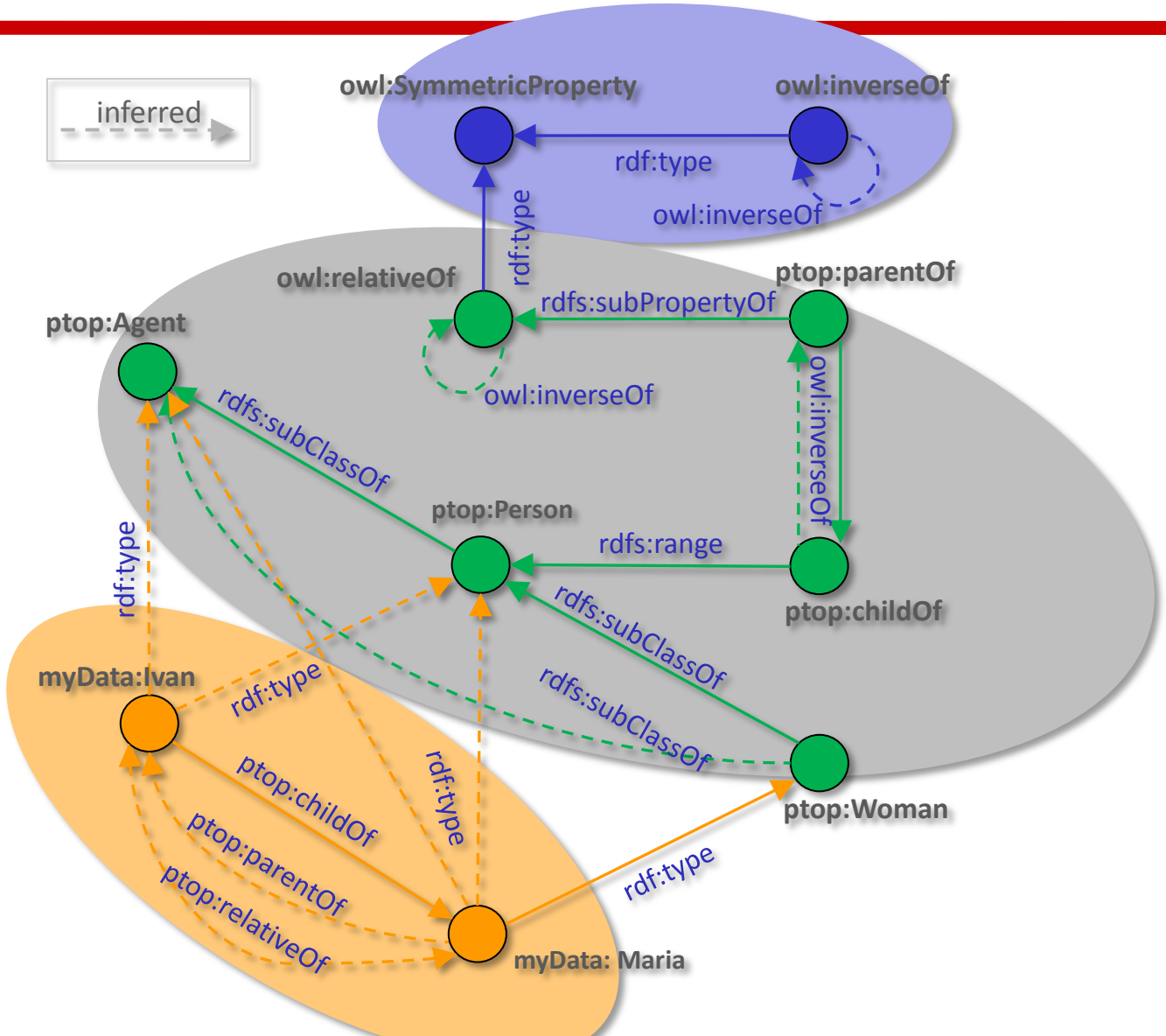
ontotext

GATE

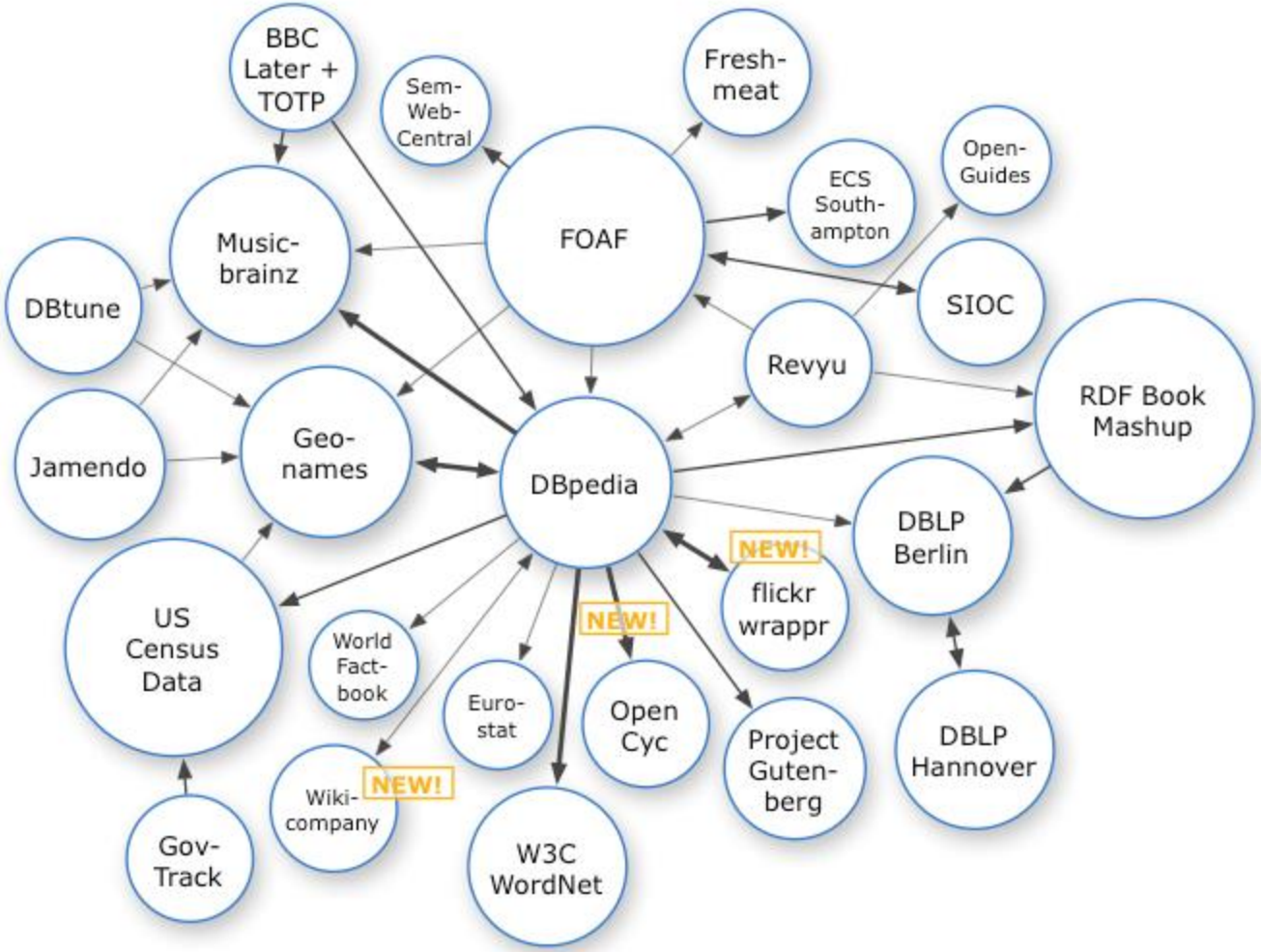
Linked Data design principles

1. Unambiguous identifiers for objects (resources)
   - Use URIs as names for things
2. Use the structure of the web
   - Use HTTP URIs so that people can look up the names
3. Make is easy to discover information about an object (resource)
   - When someone lookups a URI, provide useful information
4. Link the object (resource) to related objects
   - Include links to other URIs

Linked Data (2)



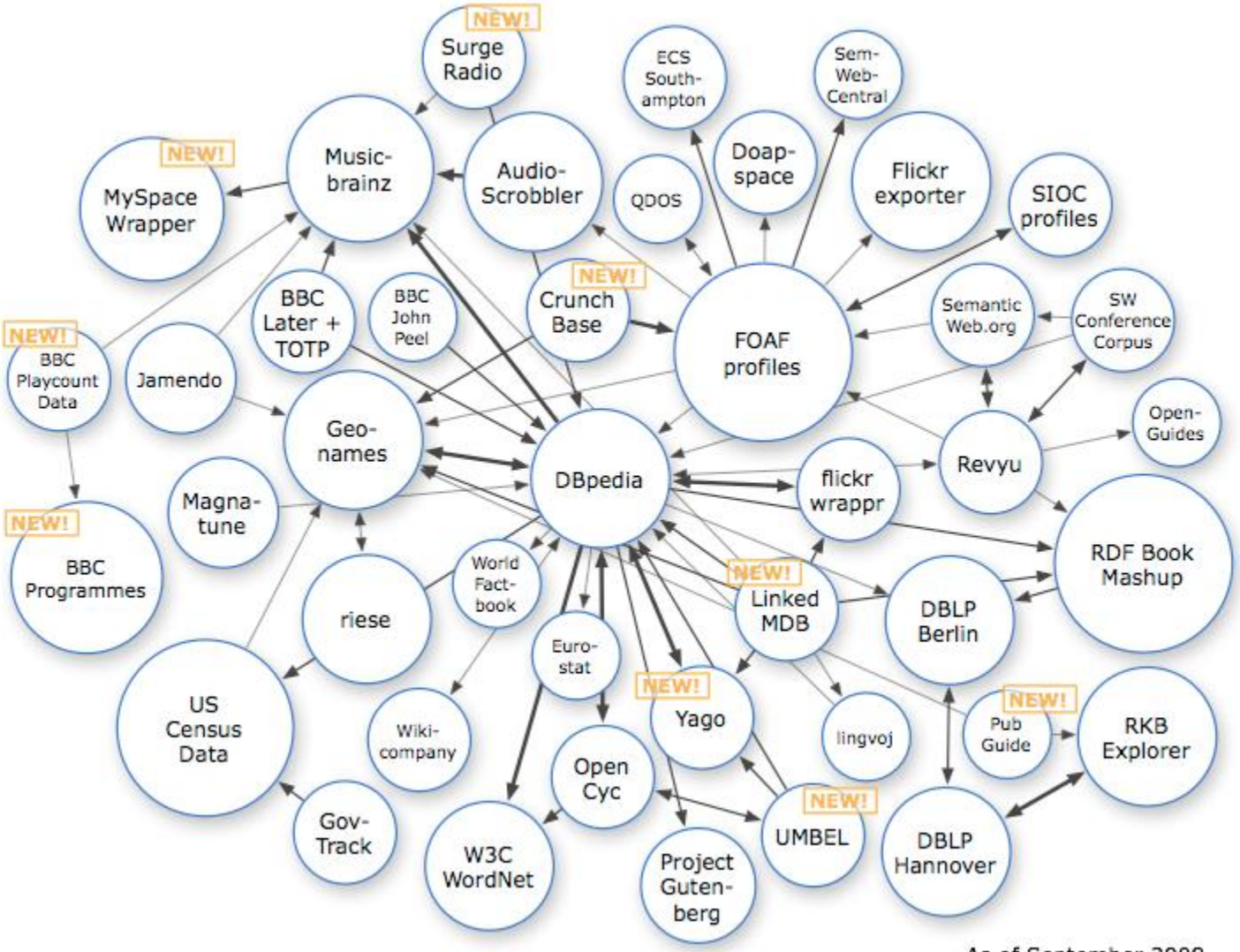
Linked Data evolution – Oct 2007



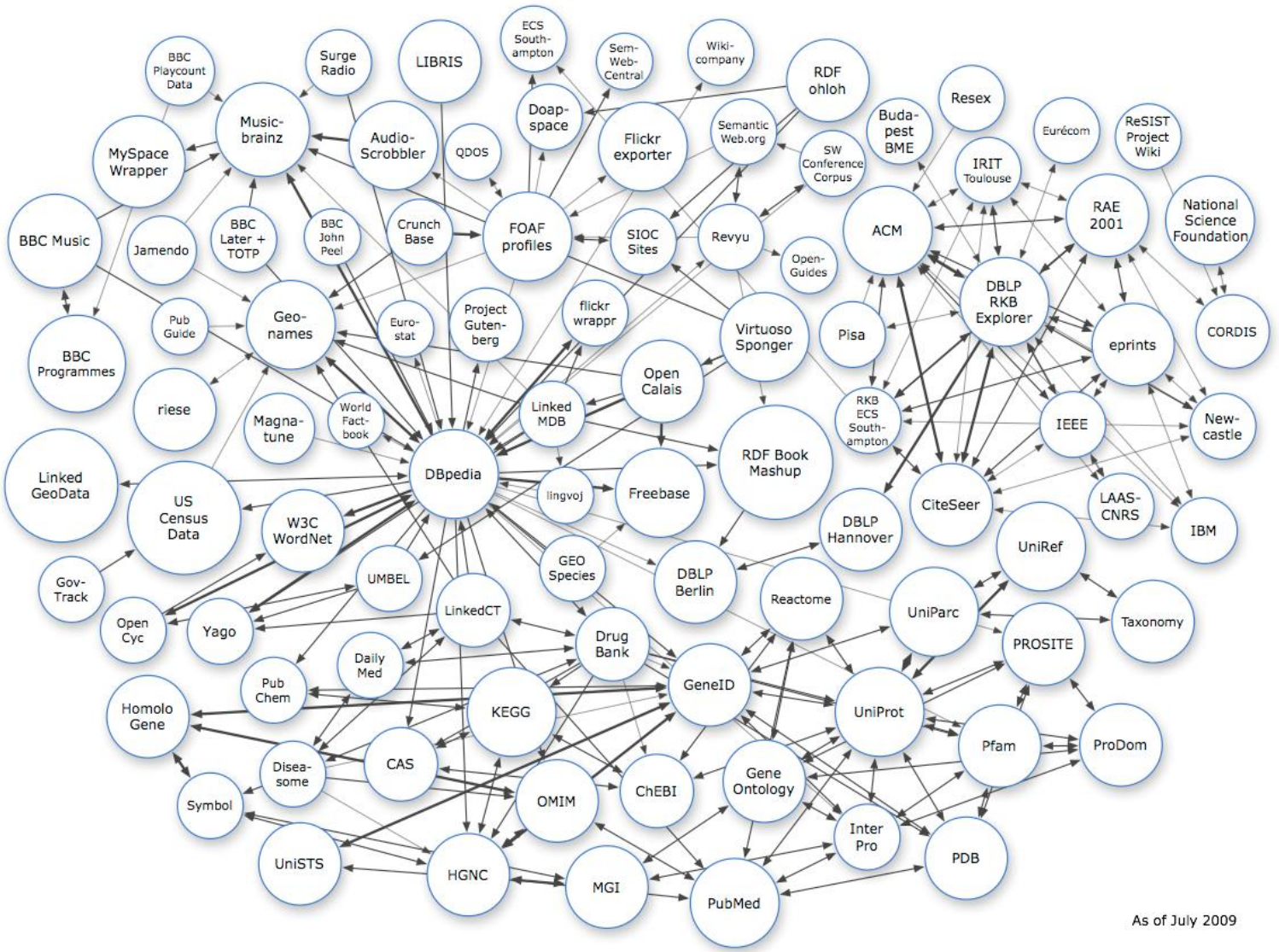
ontotext

GATE

Linked Data evolution – Sep 2008

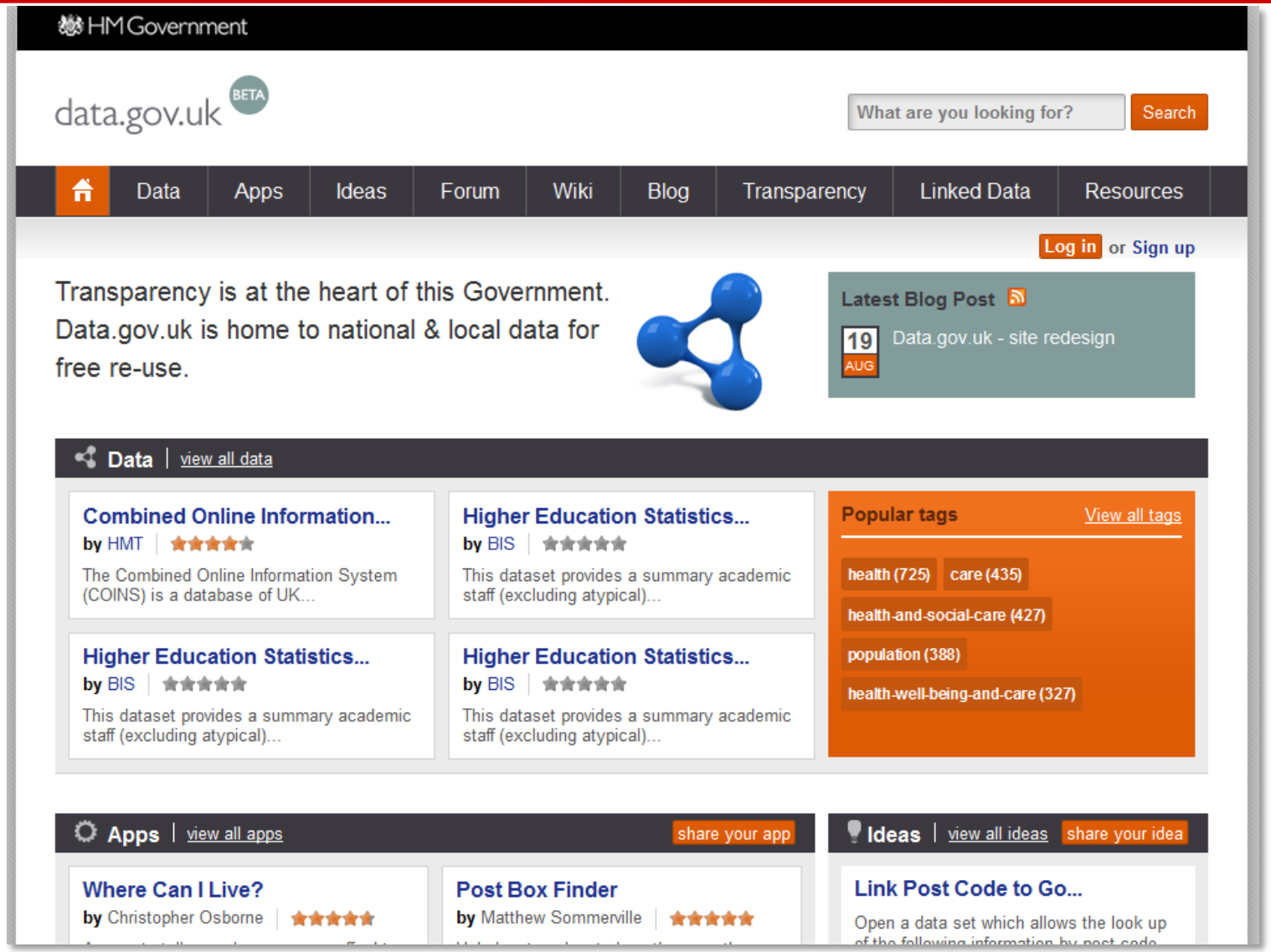


Linked Data evolution – Jul 2009



Open Government Data

Open Government Data (2)



Linked Data success stories

- BBC Music
  - Integrates information from MusicBrainz and Wikipedia for artist/band infopages
  - Information also available in RDF (in addition to web pages)
  - 3rd party applications built on top of the BBC data
  - BBC also contributes data back to the MusicBrainz
- NY times
  - Maps its thesaurus of 1 million entity descriptions (people, organisations, places, etc) to DBpedia and Freebase

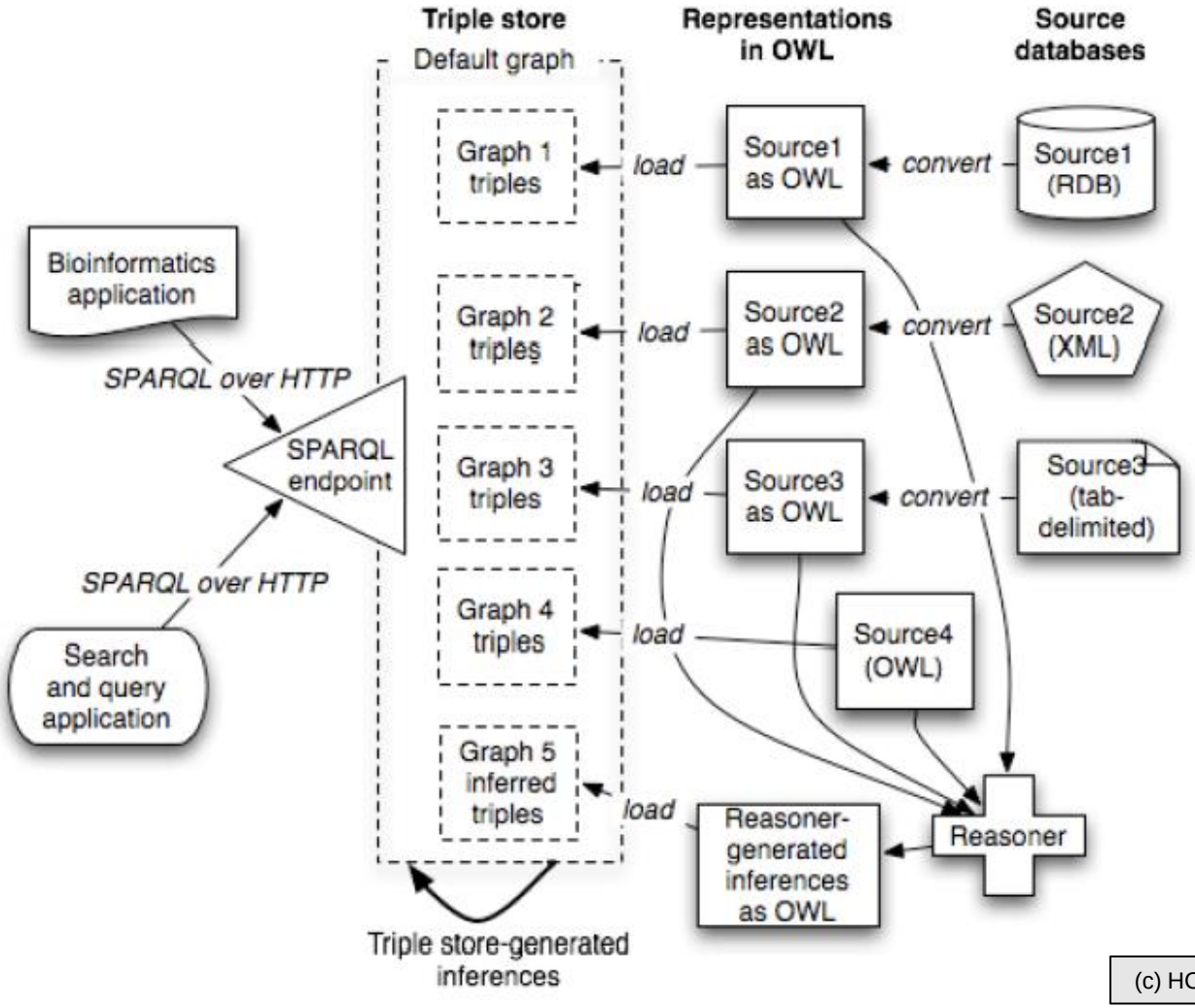
ontotext

GATE

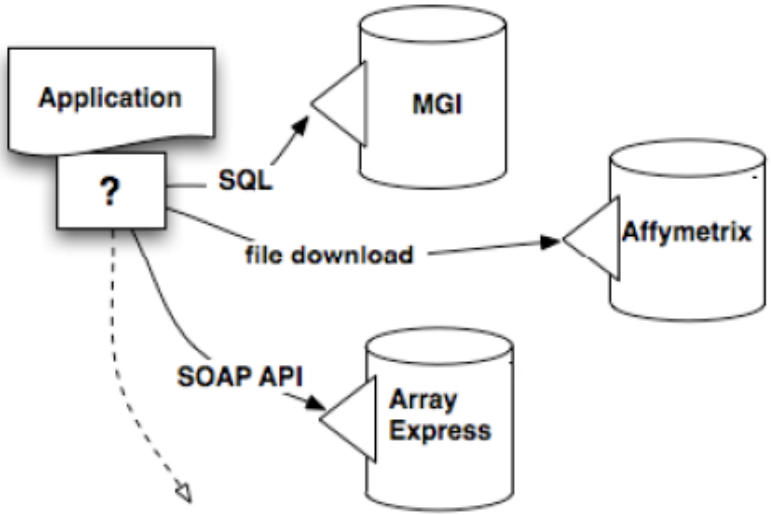
Life Sciences

- Semantic Web Health Care and Life Sciences (HCLS) @ W3C
  - http://www.w3.org/blog/hcls/
- Potential benefits of ST
  - Interoperability – re-using common models
  - Data integration – too many disparate datasources
  - Efficient query answering

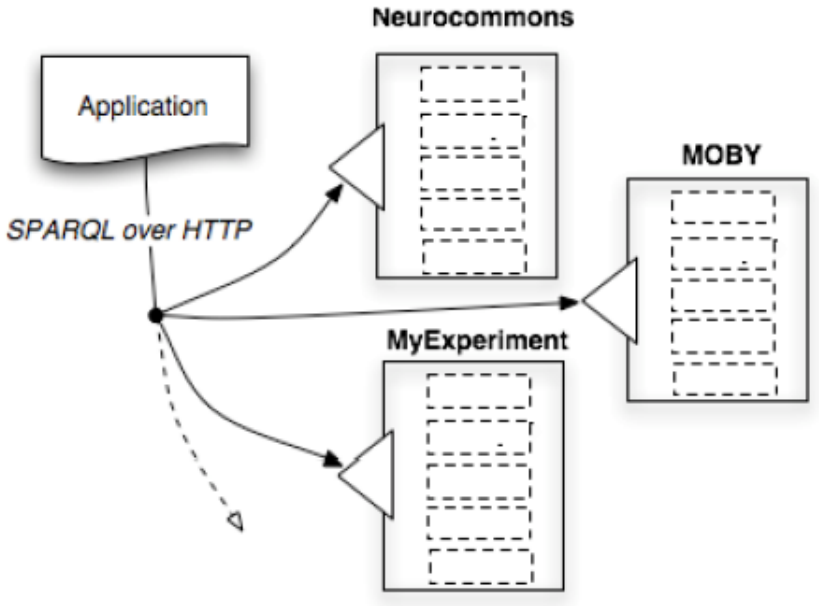
Life Sciences (2)



Life Sciences (3)

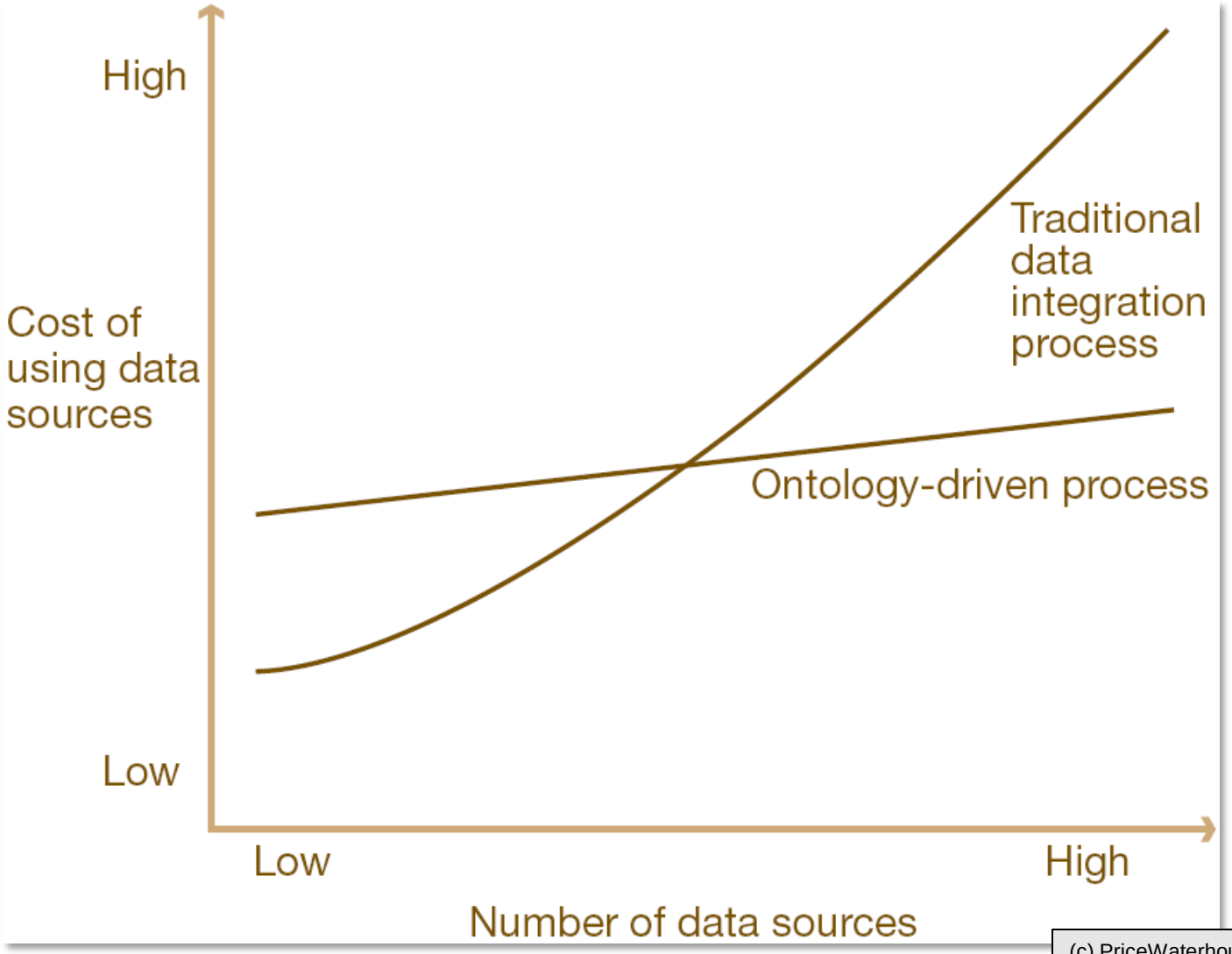


distributed querying at present



distributed querying with RDF and SPARQL

Data integration cost



ontotext

GATE

Semantic Life Science Models

OWL	OBO
Generic or top-down	Specific or bottom-up
Describe any domain (in theory)	Focus on supporting existing users and applications
Background in AI	Background in genome annotations
Ontology (strict semantics)	Vocabularies (relaxed semantics)

ontotext

GATE

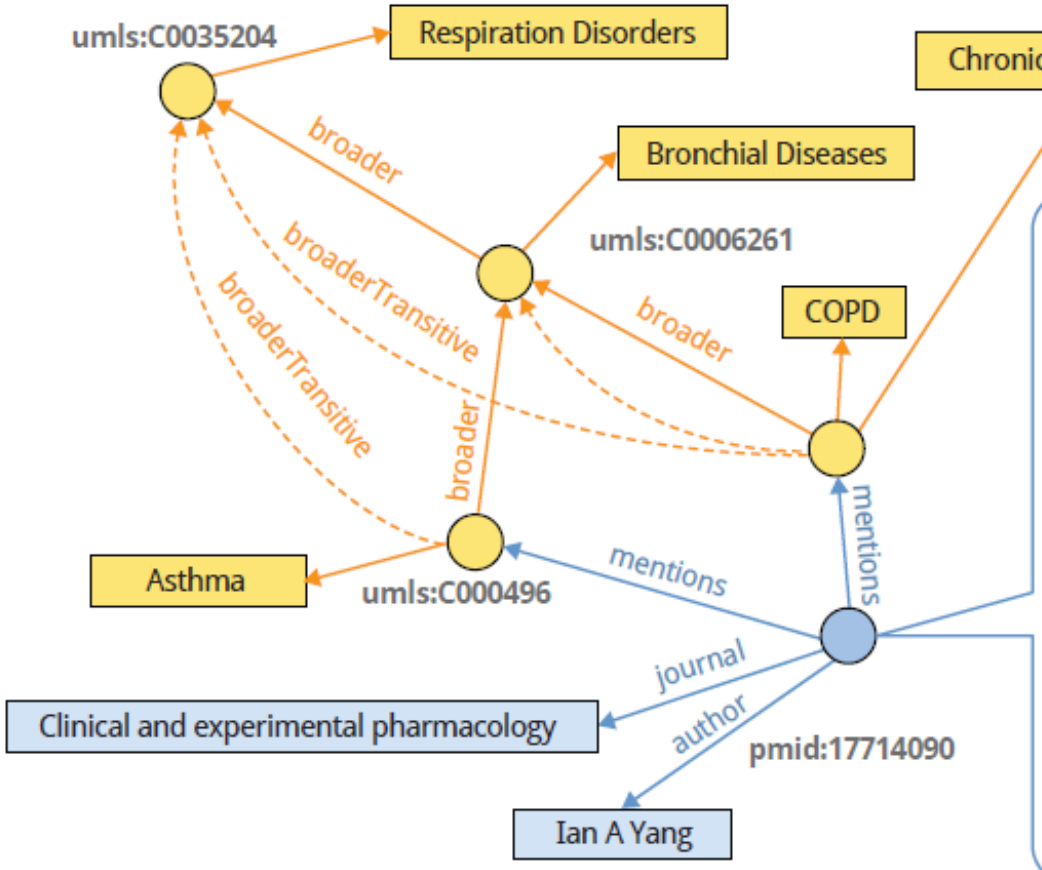
Semantic Annotation

- Semantic annotation (of text)
  - The process of linking text fragments to structured information
    - Organisations, Places, Products, Human Genes, Diseases, Drugs, etc.
  - Combines Text Mining (Information Extraction) with Semantic Technologies
- Benefits of semantic annotations
  - *Improves the text analysis process*
    - by employing Ontologies and knowledge from external Knowledge Bases / structured data sources

Semantic Annotation (2)

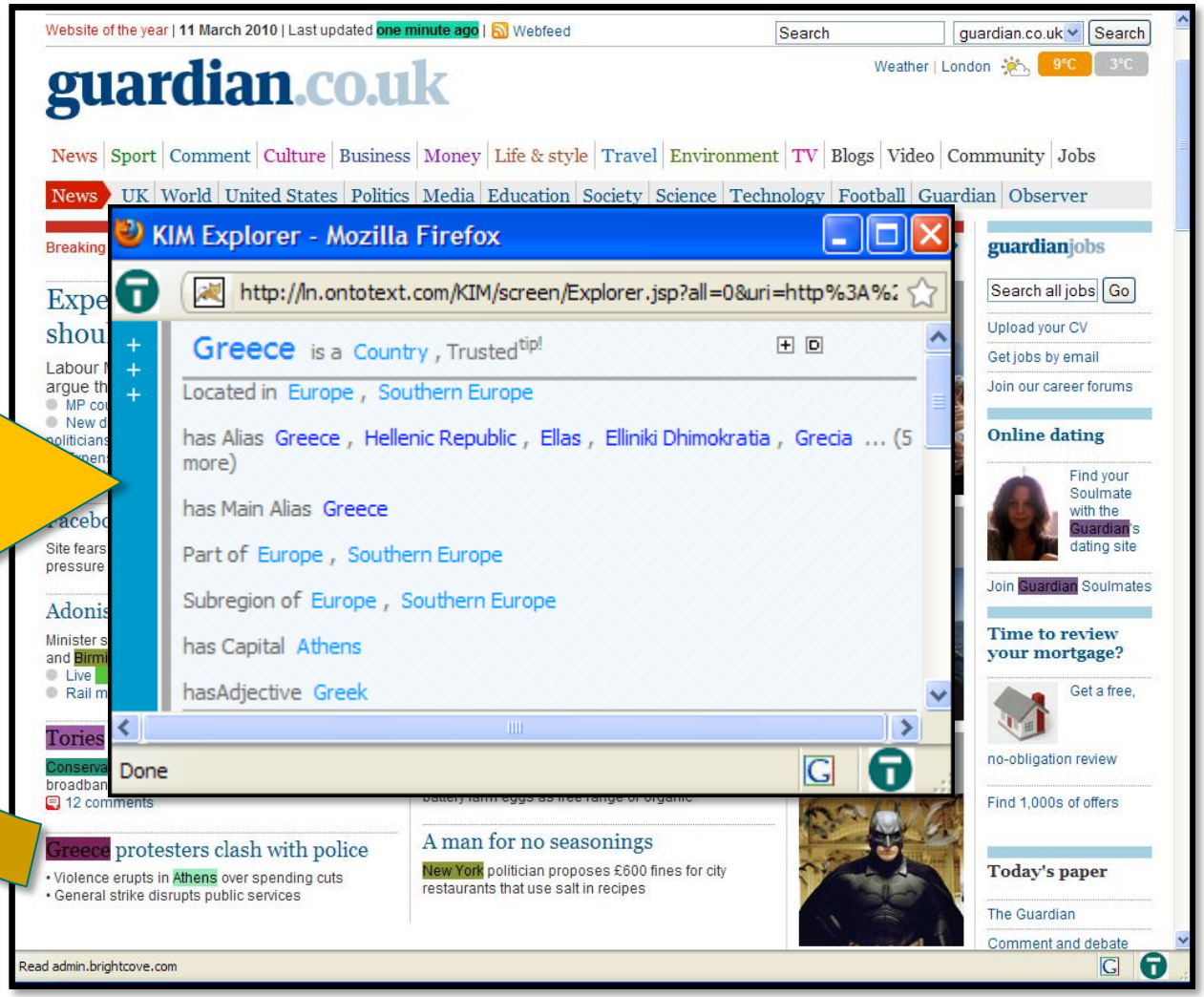
- Benefits of semantic annotations (cont.)
  - Provides *unambiguous (global) references for entities* discovered in text
    - Different from tagging
  - Provide the means for *semantic search*
    - Together or independently of the original text
  - Improved *data integration*
    - Documents from different data sources can share the same semantic concepts

Semantic Annotation (2)



Asthma and chronic obstructive pulmonary disease (COPD) are chronic airway diseases characterized by airflow obstruction. The beta(2)-adrenoceptor mediates bronchodilatation in response to exogenous and endogenous beta-adrenoceptor agonists. Single nucleotide polymorphisms in the beta(2)-adrenoceptor gene (ADRB2) cause amino acid changes (e.g. Arg16Gly, Gln27Glu) that potentially alter receptor function.

Semantic Annotation (3)



Semantic Search

- Semantic Search
  - In addition to the terms/keywords, explore the entity descriptions found in text
  - Make use of the semantic relations that exist between these entities
- Example
  - Query – "*Documents about a telecom companies in Europe related to John Smith from Q1 or Q2/2010*"
  - Document containing "*At its meeting on the 10th of May, the board of Vodafone appointed John G. Smith as CTO*" will **not match**

Semantic Search (2)

- Classical IR will fail to recognise that
  - Vodafone is a mobile operator, and mobile operator is a type of telecom
  - Vodafone is in the UK , which is part of Europe
  - => Vodafone is a “telecom company in Europe”
  - 5th of May is in Q2
  - John G. Smith may be the same as John Smith

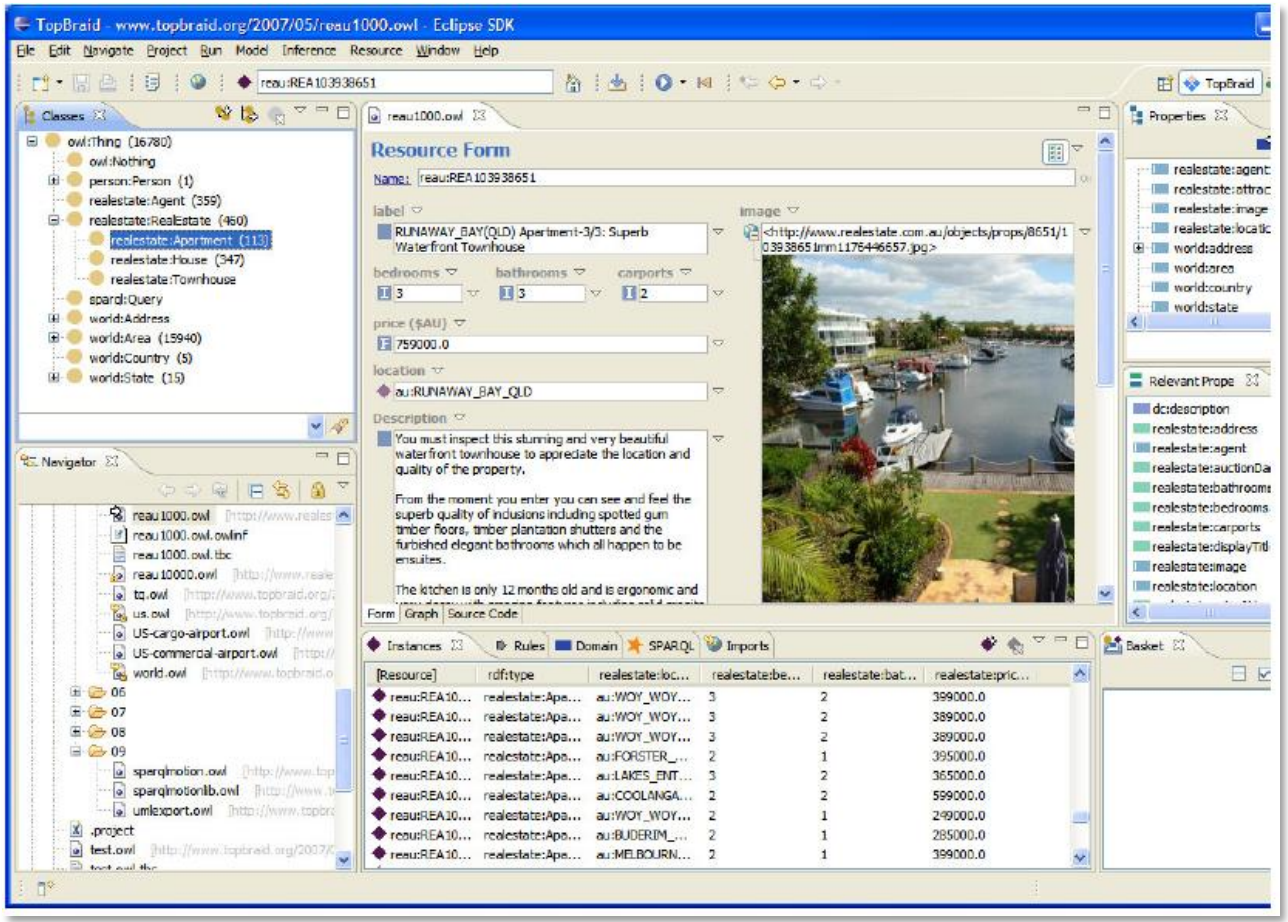
Web Services and Business Process Management

- Potential benefits of using ST
  - Automated mediation & interoperability
  - Capability based discovery
  - Automated composition of complex workflows
- Relevant activities & standards
  - SAWSDL – www.w3.org/2002/ws/sawsdl
    - Embed semantic annotations within WSDL descriptions
  - WSMO / Conceptual Models for Services (CMS) - http://cms-wg.sti2.org
    - Ontology and methodology for semantic annotation of services and business processes

Tools

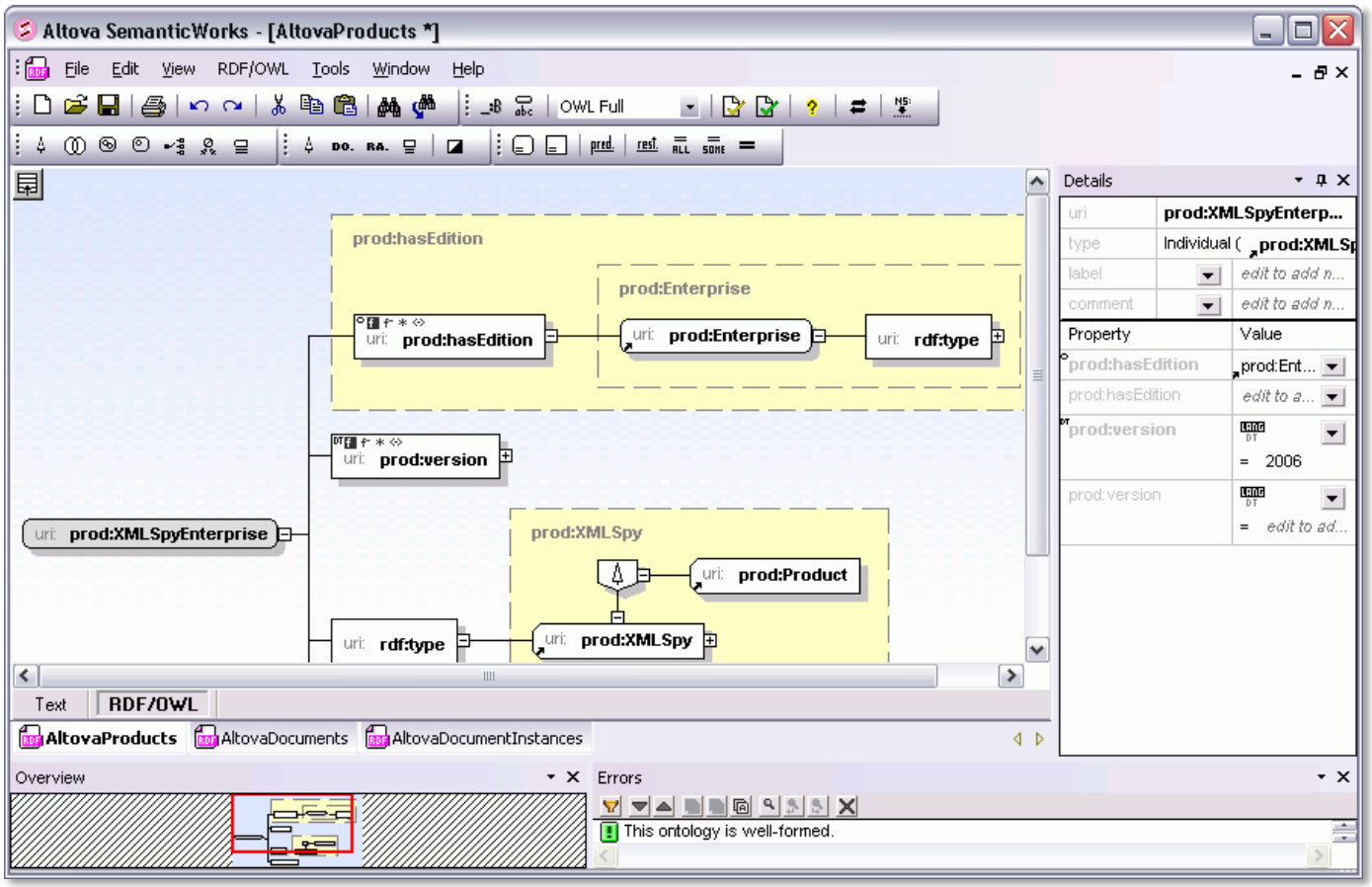
Ontology editors

- TopBraid Composer
  - http://www.topquadrant.com



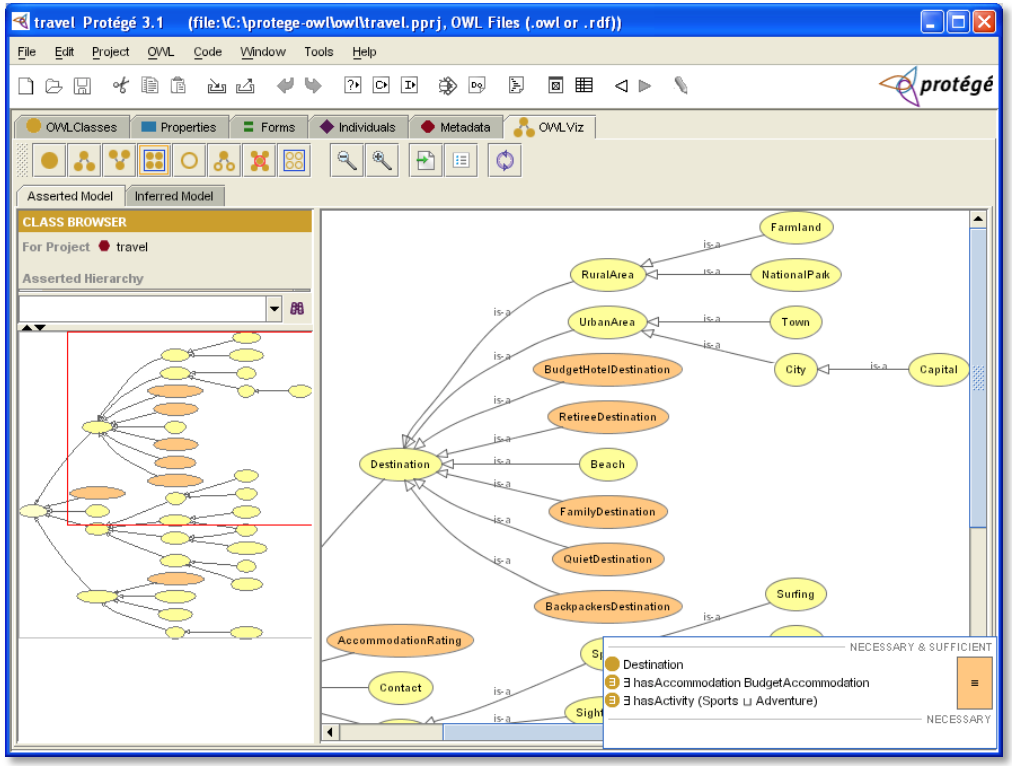
Ontology editors (2)

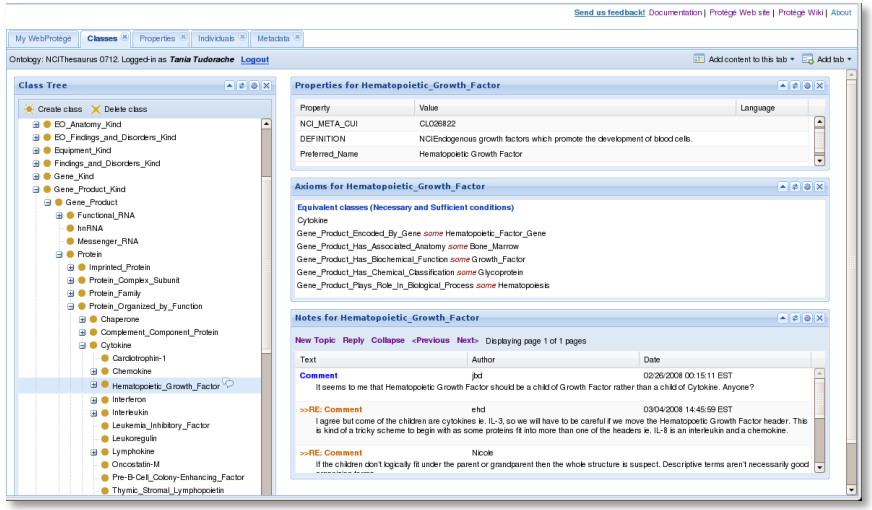
- Altova SemanticWorks
  - http://www.altova.com/semanticworks.html



Ontology editors (3)

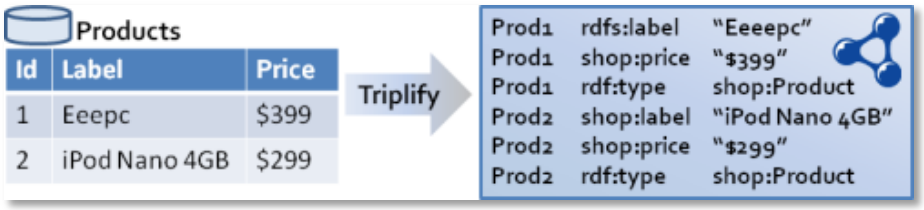
- Protégé
  - http://protege.stanford.edu
  - http://webprotege.stanford.edu



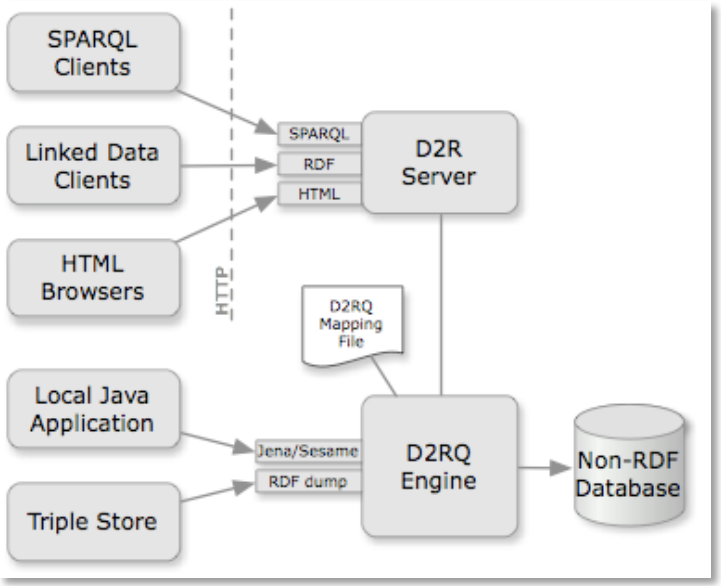


RDF-izers

- Triplify
  - http://triplify.org
  - Transform relational data into RDF / Linked Data



- D2RQ platform
  - http://www4.wiwiss.fu-berlin.de/bizer/d2rq/index.htm
  - D2RQ mapping language
  - D2RQ plugin for Sesame/Jena
  - D2R server
    - Linked Data & SPARQL endpoint



ontotext

GATE

RDF APIs

- Jena
  - http://jena.sourceforge.net
  - RDF/OWL API (Java)
  - In-memory or persistent storage
  - SPARQL query engine
- OpenRDF (Sesame)
  - http://www.openrdf.org
  - RDF API (Java), high performance parser
  - Persistent storage
  - SPARQL query engine

ontotext

GATE

Triplestores

- OWLIM
  - http://www.ontotext.com/owlim/
  - Unmatched materialisation & query performance, replication cluster, RDFRank, “sameAs” optimisation
  - 10-20 billion triples
- Virtuoso
  - http://virtuoso.openlinksw.com
  - RDBMS integration, geo-spatial extensions, federated / virtual databases
  - 10-20 billion triples

ontotext

GATE

Triplestores (2)

- Oracle
  - RDBMS integration, parallel inference/query, built-in compression
- AllegroGraph RDFStore
  - http://www.franz.com/agraph/allegrograph
  - Geo-spatial extensions, built-in SNA functionality, federated / virtual databases, built-in compression
- 4Store
  - http://4store.org
  - distributed cluster

ontotext

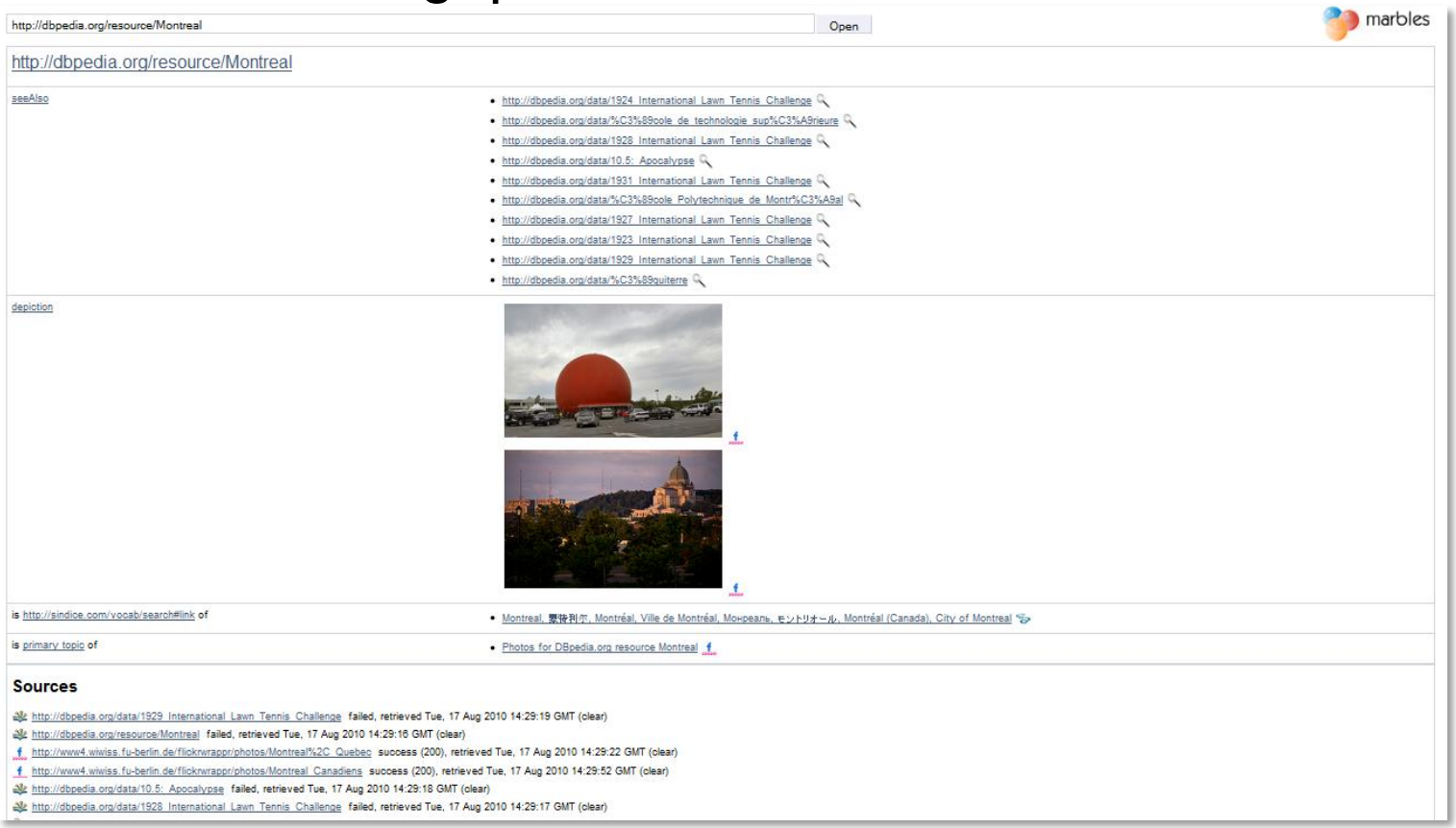
GATE

Reasoners

- Pellet
  - http://clarkparsia.com/pellet
  - OWL DL, OWL2 Profiles, datatypes, SWRL extensions, ontology analysis & repair, incremental reasoning, integration with Oracle DB
- Fact++
  - http://code.google.com/p/factplusplus
  - OWL DL, partial OWL2 and datatype support
- Racer
  - http://www.racer-systems.com

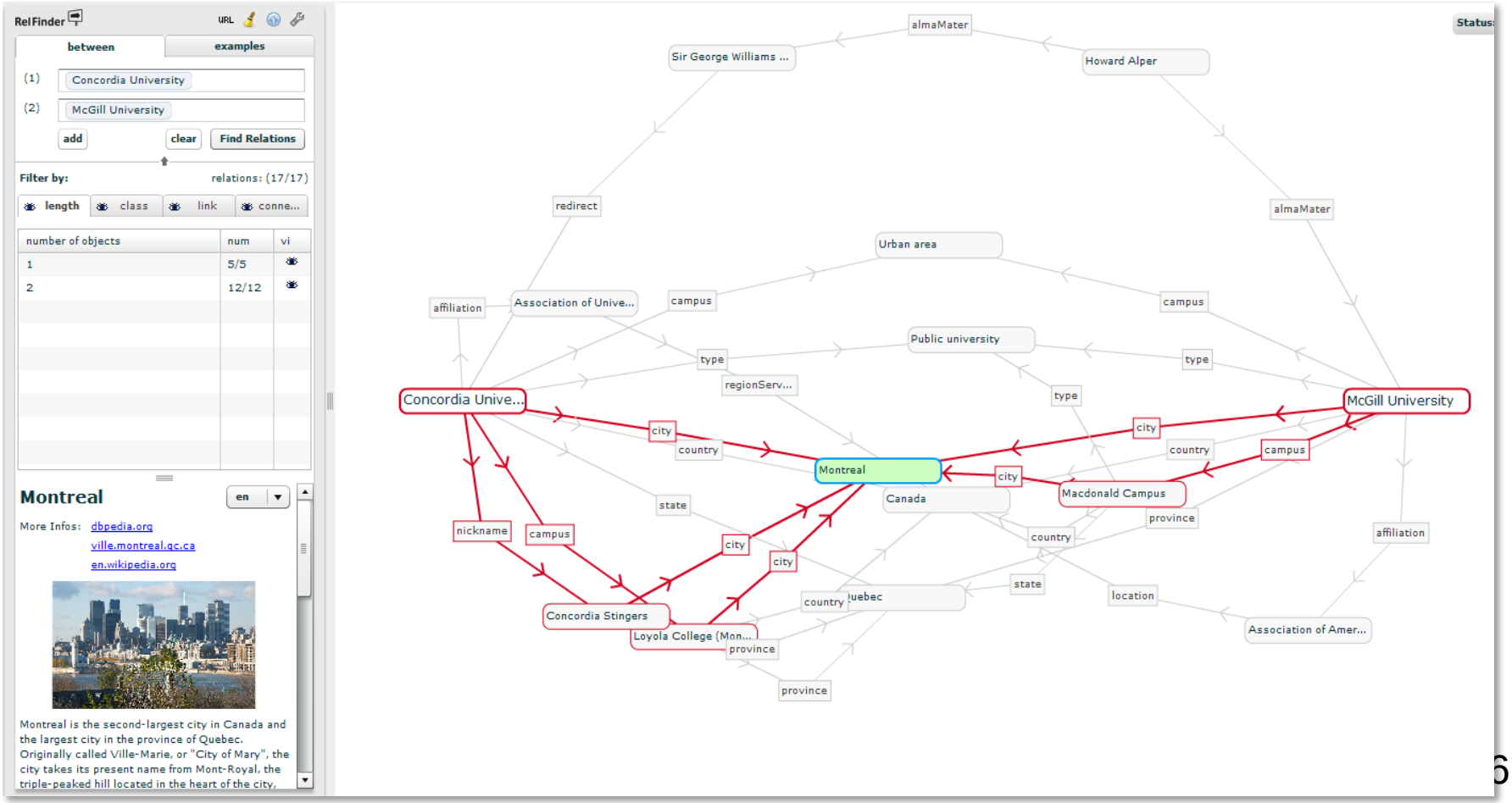
Linked Data browsers – Marbles

- http://marbles.sourceforge.net
- XHTML views of RDF data (SPARQL endpoint), caching, predicate traversal



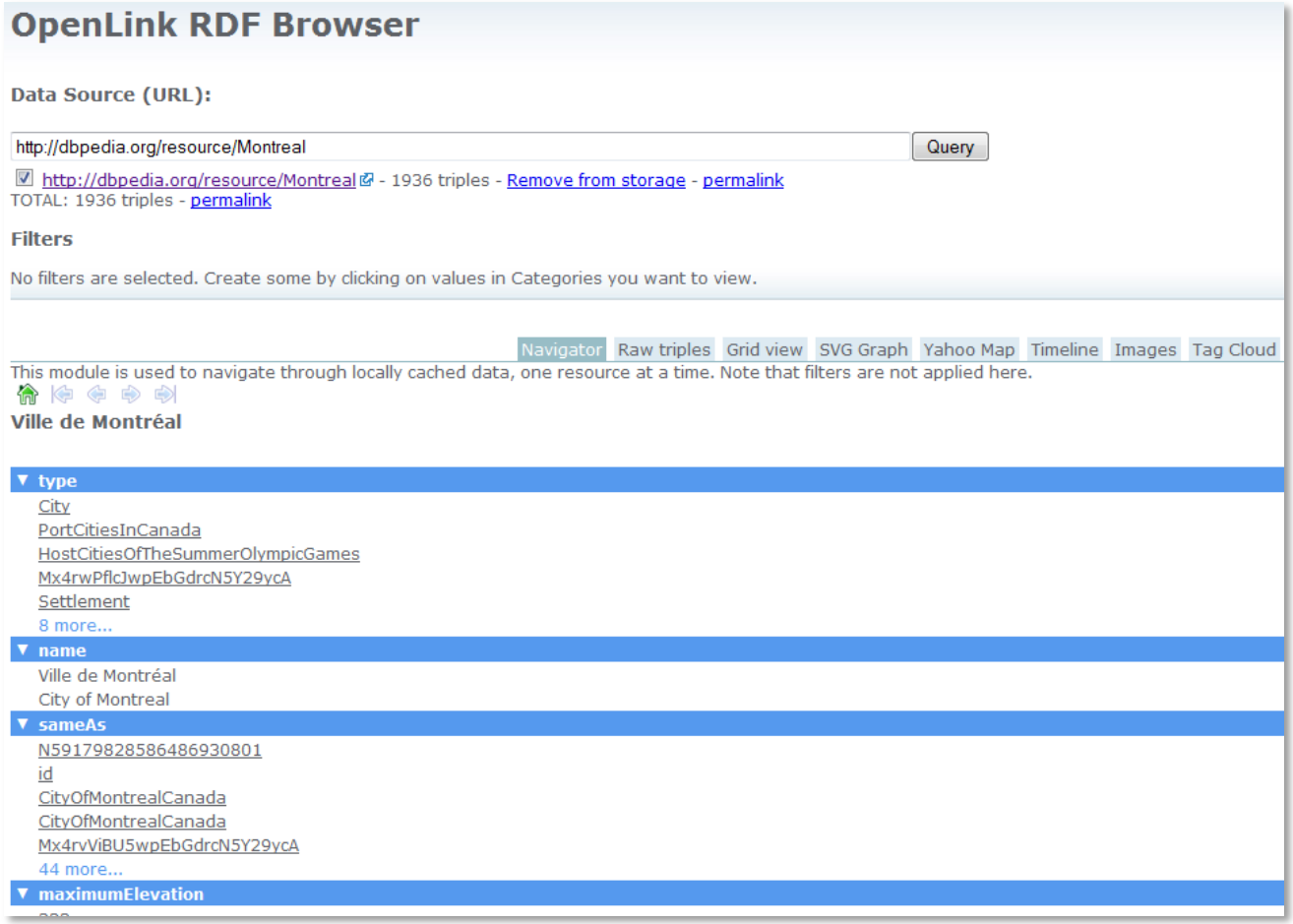
Linked Data browsers – RelFinder

- http://relfinder.dbpedia.org
- Explore & navigate relationships in a RDF graph



Linked Data browsers – OpenLink RDF Browser

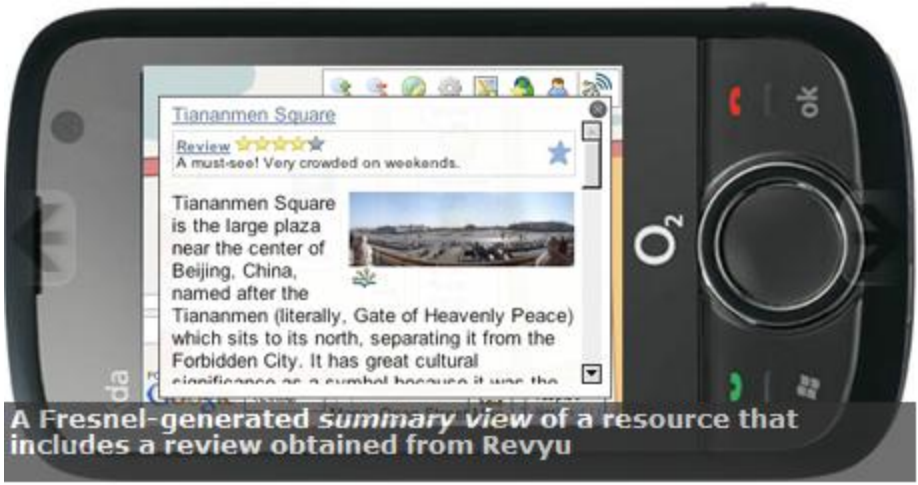
- http://demo.openlinksw.com/DAV/JS/rdfbrowser/index.html
- Explore & navigate relationships in a RDF graph



DBPedia Mobile

- http://wiki.dbpedia.org/DBpediaMobile
- Based on user's GPS position, renders a map with nearby places of interest (from DBpedia)

DBpedia Mobile's *map view* of resources in the user's proximity

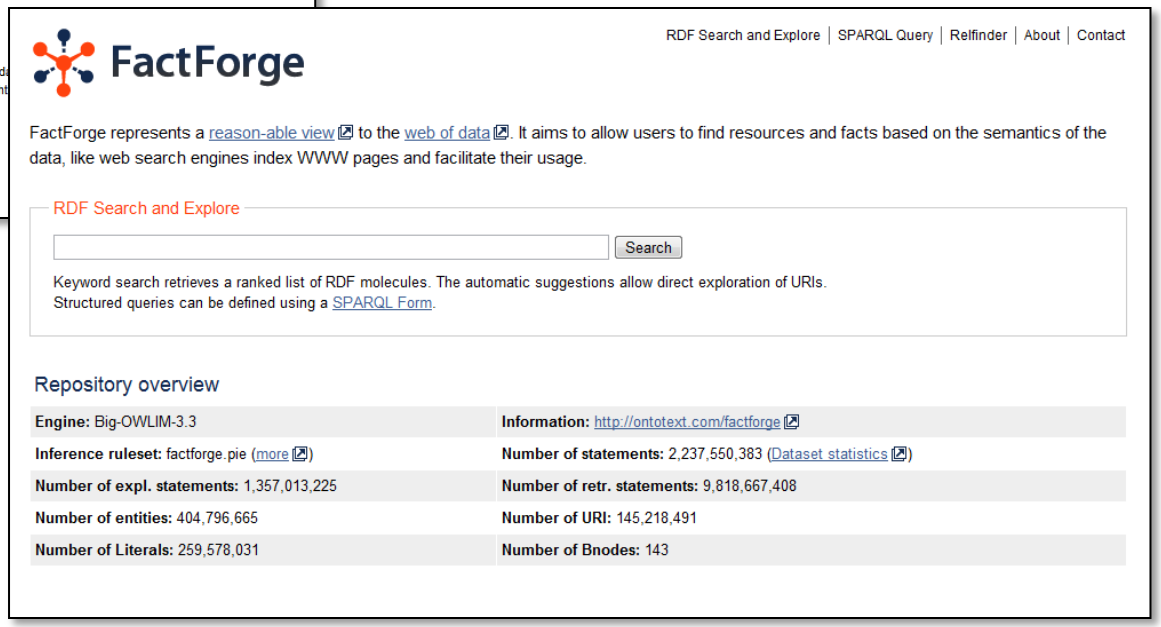


A Fresnel-generated *summary view* of a resource that includes a review obtained from Revyu

FactForge and LinkedLifeData

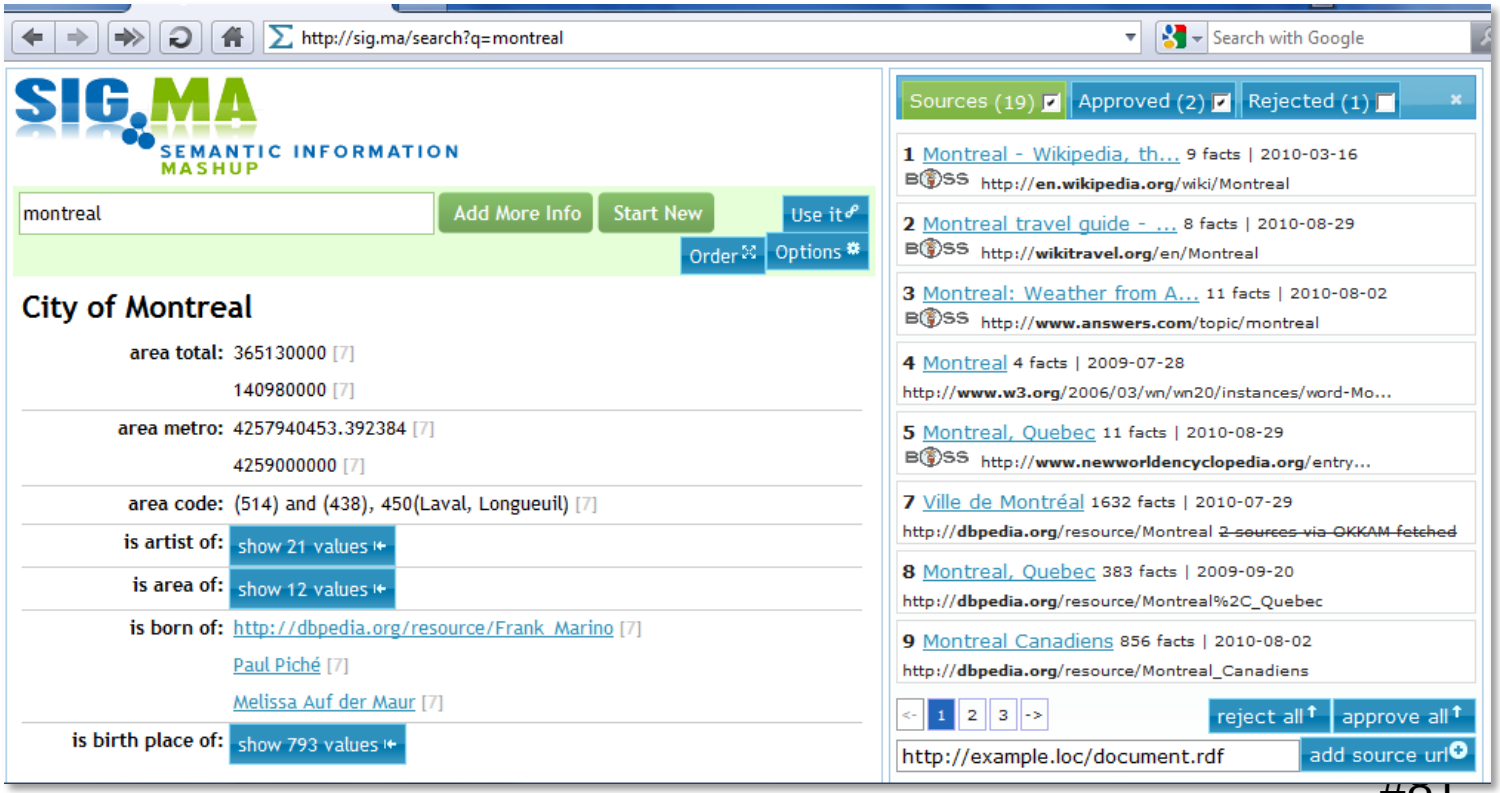
- *FactForge*
  - Integrates some of the most central LOD datasets
  - General-purpose information (not specific to a domain)
  - 1.2B explicit plus 1B inferred statements (10B retrievable)
  - The largest upper-level knowledge base
  - **http://www.FactForge.net/**
- *Linked Life Data*
  - 25 of the most popular life-science datasets
  - 2.7B explicit and 1.4B inferred triples
  - **http://www.LinkedLifeData.com**

FactForge and LinkedLifeData (2)



Sig.ma – the semantic mashup platform

- Aggregate RDF/RDFa data sources for „live views“ of the Web of Data
  - http://sig.ma



Sindice – the Semantic Web index

- Crawls RDF/RDFa pages on the web and provides a consolidated index
  - http://sindice.com
  - 120+ million indexed pages

Summary of this module

- The Semantic Web is an extension of the current web, where information has well-defined meaning, so that it is usable by intelligent agents
- Ontologies provide means to share and reuse knowledge, by formal modelling of the concepts and relationships between them in a domain
- Since 1999 W3C has provided a rich family of Semantic Web related standards for modelling knowledge and rules, querying knowledge bases, embedding semantic annotations in web pages, etc

Summary of this module (2)

- Semantic Technologies are applicable for a wide set of domains where interoperability, data integration and knowledge reuse is crucial
- There is a rich tool support for Semantic Technologies

Useful links

- RDF
  - Resource Description Framework (RDF): Concepts and Abstract Syntax – http://www.w3.org/TR/rdf-concepts/
- Linked Data
  - http://linkeddata.org/