Publishing Data Using Semantic Web Technologies

An introduction for software engineers

Nikolaos Konstantinou, Ph.D.

National Documentation Centre / N.H.R.F.

Lecture Outline

- Introduction to the Semantic Web
- Semantic Web Languages
- Publishing RDF Using OpenLink’s Virtuoso
- Linked Open Data and Examples

The Problem (1)

- Keyword-based queries cannot be expressive
- E.g. search for:
  - Cities in the U.S. with more than 100,000 inhabitants
  - Italian painters of the 18th century
- Web resources
  - Do not (usually) convey their meaning

The Problem (2)

- Seeking specific information in the Web or a repository
- Integrating distributed data sources
- Need for data annotation
  - Necessary for data non-readable by human
    - E.g. binary information, multimedia
  - Annotation may be redundant, incomplete, or erroneous
  - When it is present it does not necessarily follow a standard pattern

The Semantic Web Paradigm (1)

- ‘Web of Data’ as in a ‘Web of Documents’
  - Web resources uniquely identified by their URI
- Assign an unambiguously defined meaning to information, its *semantics*
  - Ontology, a well defined vocabulary
  - Queries can be posed by any third parties
- Knowledge modeled in the form of a graph
  - subject, predicate, object
- Interconnected data sets on the Web
  - Provide context

The Semantic Web Paradigm (2)

- Enables semantic annotation, interoperability, integration of information
- Enables reasoning
  - Extract implicit information
  - Assure concept consistency
- Variety of mature, open source tools available
  - Protégé, Jena, Virtuoso, D2RQ, …
- Allows information to be exposed as Linked Open Data (to be discussed later on)
- Data ⇒ Information ⇒ Knowledge

What is an Ontology?

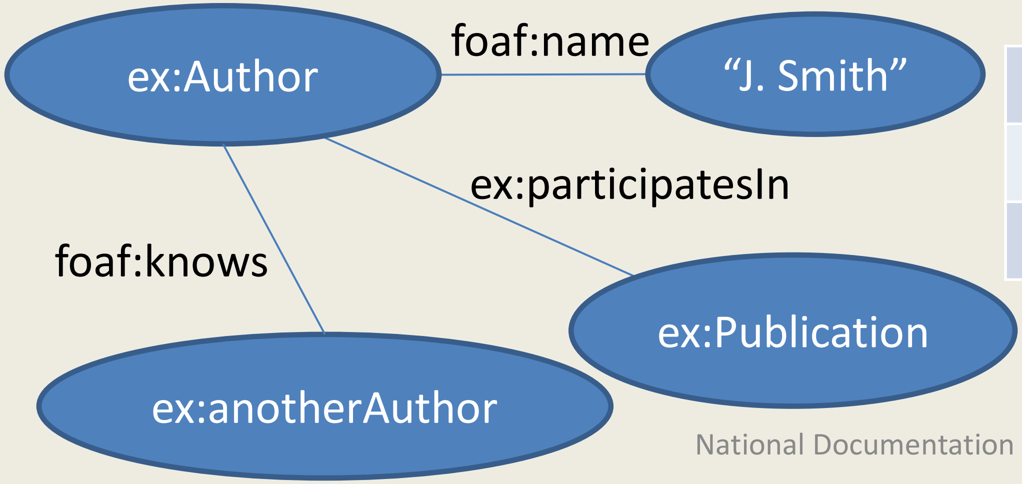
- In philosophy, Ontology is the study of beings
  - Onto (ὤν/ὄντος) + logy (λογία)
  - Along with their properties and relations
- In computer science, an ontology is the formal representation of knowledge
  - A formal, explicit specification of a shared conceptualisation
  - Concepts of a domain, objects and their relations
  - Allows complexity in schemas
- The RDF and OWL approaches

Lecture Outline

- Introduction to the Semantic Web
- Semantic Web Languages
- Publishing RDF Using OpenLink’s Virtuoso
- Linked Open Data and Examples

The Resource Description Framework

- The Resource Description Framework is about describing resources
  - Was initially proposed for describing Web resources
- RDF can be viewed as a graph where
  - Objects are graph nodes
  - Properties are graph edges



Graph triples

ex:Author	foaf:name	“J. Smith”
ex:Author	ex:participatesIn	ex:Publication
ex:Author	foaf:knows	ex:anotherAuthor

The RDF Schema (1)

- Describing Web Resources using RDF
  - rdfs:Resource
    - All things described by RDF are resources
  - rdfs:Class
    - The class of resources that are classes, i.e. the class of classes
  - rdf:type
    - States resource membership
    - E.g.: ex:Person rdf:type rdfs:Class
  - rdf:Property
    - The relations between subjects and objects

The RDF Schema (2)

- Describing Web Resources using RDF
  - rdfs:SubClassOf
    - foaf:Agent rdfs:subClassOf foaf:Person
  - rdfs:SubPropertyOf
    - Allow class *and* property hierarchies
    - E.g.: ex:hasFirstName rdfs:subpropertyOf ex:hasName
  - rdfs:domain
    - ex:employer rdfs:domain foaf:Person
  - rdfs:range
    - ex:employer rdfs:range foaf:Organization

The RDF Schema (3)

- Describing Web Resources using RDF
  - rdfs:Container
    - rdf:Bag
    - rdf:Seq
    - rdf:Alt
    - rdfs:ContainerMembershipProperty
    - rdfs:member
  - rdfs:label
  - rdfs:comment
  - rdf:List
    - rdf:first
    - rdf:rest
    - rdf:nil
  - rdf:statement
  - rdf:subject
  - rdf:predicate
  - rdf:object
  - rdf:value
  - rdfs:seeAlso
  - rdfs:isDefinedBy

The RDF Schema (4)

- Example 1

```
<rdfs:Class rdf:ID="animal" />
  <rdfs:Class rdf:ID="horse">
  <rdfs:subClassOf rdf:resource="#animal"/>
</rdfs:Class>
```

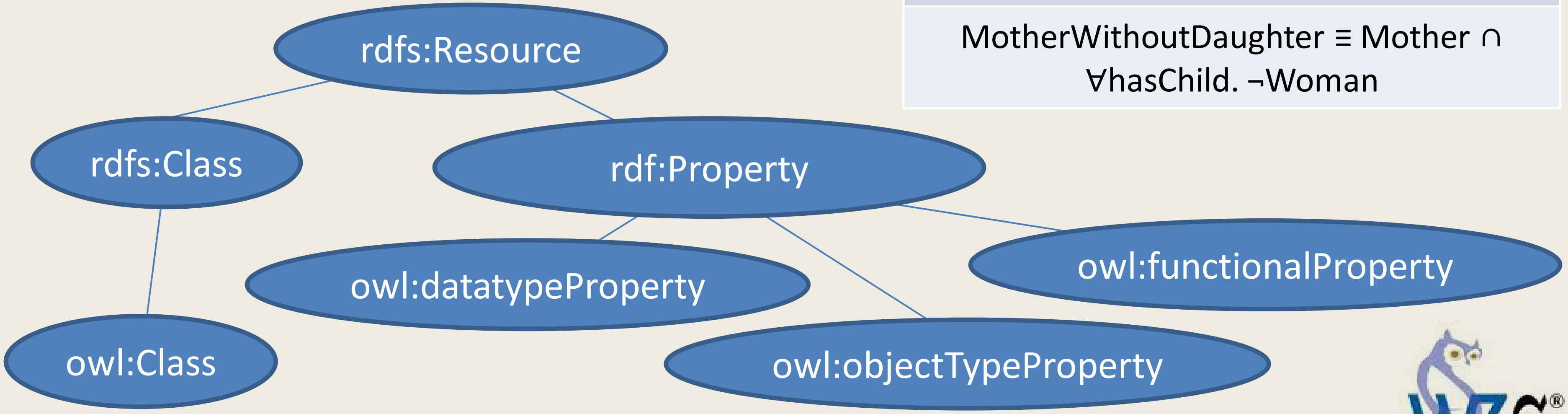
- Example 2

```
<rdf:Description rdf:about="http://www.ekt.gr">
 <dc:description>National Documentation Centre</dc:description>
 <dc:publisher>NHRF</dc:publisher>
 <dc:date>2001-02-16</dc:date>
 <dc:format>text/html</dc:format>
 <dc:language>el</dc:language>
</rdf:Description>
```

Web Ontology Language (1)

- Based on Description Logics
  - Decidable fragment of First Order Logic
- Allows more complex schema definitions
- OWL builds on top of RDF
- Current version is OWL 2

Woman ≡ Person ∩ Female
Father ≡ Man ∩ ∃hasChild.Person
Wife ≡ Woman ∩ ∃hasHusband.Man
MotherWithoutDaughter ≡ Mother ∩ ∀hasChild. ¬Woman



Web Ontology Language (2)

- Class description
  - owl:intersectionOf
  - owl:unionOf
  - owl:complementOf
  - owl:equivalentClass
  - owl:disjointWith
  - Cardinality
    - owl:maxCardinality
    - owl:minCardinality
    - owl:cardinality

- Property description
  - owl:datatypeProperty
  - owl:objectProperty
  - owl:equivalentProperty
  - owl:inverseOf
    - isTaughtBy ↔ teaches
  - owl:functionalProperty
  - owl:inverseFunctionalProperty
  - owl:transitiveProperty
  - owl:symmetricProperty

Web Ontology Language (3)

- owl:Thing
- owl:Nothing
- Version information
  - owl:versionInfo
  - owl:priorVersion
  - owl:backwardCompatibleWith
  - owl:incompatibleWith
  - owl:deprecatedClass
  - owl:deprecatedProperty
- Individuals
  - owl:sameAs
  - owl:differentFrom
  - owl:allDifferent
- Value constraints
  - owl:allValuesFrom
  - owl:someValuesFrom
  - owl:hasValue

Web Ontology Language (4)

- Example 1

```
:RedBordeaux rdf:type owl:Class ;
                 owl:equivalentClass [ rdf:type owl:Class ;
                 owl:intersectionOf ( :Bordeaux :RedWine ) ] .
```

- Example 2

```
:locatedIn rdf:type owl:ObjectProperty ,
                        owl:TransitiveProperty ;
            rdfs:domain owl:Thing ;
            rdfs:range    :Region .
```

- Example 3

```
:BordeauxRegion rdf:type owl:NamedIndividual ,
                              :Region ;
                   :locatedIn :FrenchRegion .
```

Web Ontology Language (5)

- Example 4

```
:hasColor rdf:type owl:FunctionalProperty ,
                   owl:ObjectProperty ;
          rdfs:domain :Wine ;
          rdfs:range     :WineColor ;
          rdfs:subPropertyOf :hasWineDescriptor .
```

- Example 5

```
:CabernetSauvignon rdf:type owl:Class
        owl:equivalentClass [ rdf:type owl:Class ;
                owl:intersectionOf ( :Wine
                    [ rdf:type owl:Restriction ;
                      owl:onProperty :madeFromGrape ;
                      owl:hasValue :CabernetSauvignonGrape
                    ] [ rdf:type owl:Restriction ;
                      owl:onProperty :madeFromGrape ;
                      owl:maxCardinality "1"^^xsd:nonNegativeInteger] ) ] ;
```

Web Ontology Language (6)

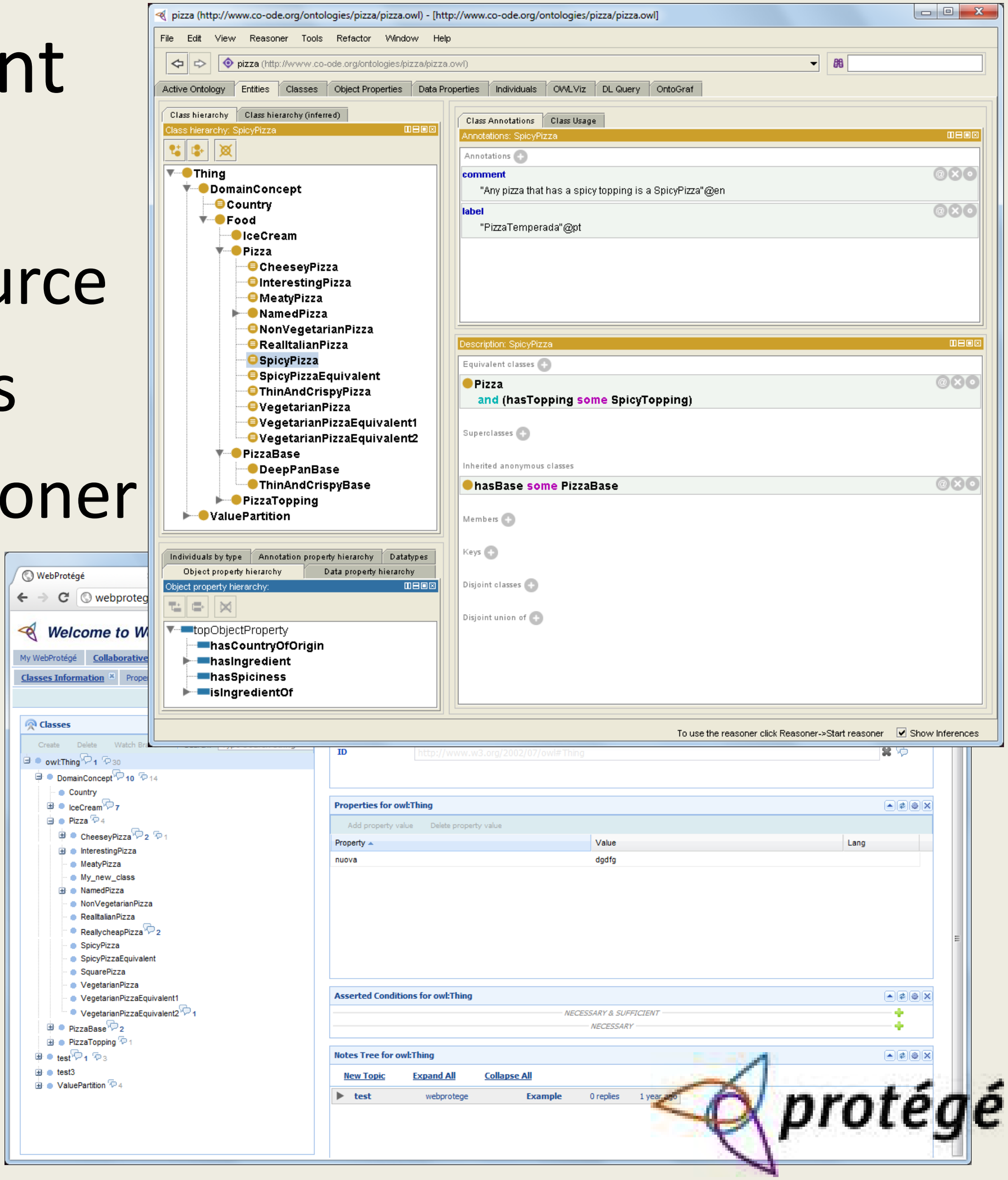
- OWL 1 flavors
  - OWL Full, full language expressivity
  - OWL DL, maximal subset allowing reasoner support
  - OWL Lite, minimal useful subset of language features
- OWL 2 profiles
  - OWL 2 EL, for large numbers of classes/properties
  - OWL 2 QL, large volume of instance data support, relational database-friendly
  - OWL 2 RL, RDFS with extra expressivity, scalable reasoning

Reasoning

- Check ontology consistency
- Class expression subsumption
- Concept satisfiability
- Infer implicit information
  - Produces extra (inferred) triples
- Numerous reasoners available
  - Free
    - Pellet, FaCT++, Jena, Hermit
  - Non-free
    - OWLIM, OntoBroker

Ontology Authoring (1)

- Protégé is a prominent GUI solution
  - Java-based, open-source
  - OWL/RDF capabilities
  - Includes FaCT++ reasoner
  - WebProtégé in beta
  - Extensible through plugins
    - E.g. Ontograf



comment "Any pizza that has a spicy topping is a SpicyPizza"@en

label "PizzaTemperada"@pt

Pizza and (hasTopping some SpicyTopping)

hasBase some PizzaBase

To use the reasoner click Reasoner->Start reasoner

Property	Value	Lang
nuova	dgdfg	

test webprotege Example 0 replies 1 year ago

Available online at http://protege.stanford.edu/

Ontology Authoring (2)

- Using HP’s Jena
  - Large, active community
  - Apache Maven group id com.hp.hpl.jena
  - API Example

```
String ns = "http://example.com/sample#";
 Model model = ModelFactory.createDefaultModel();
 Resource resource = model.createResource(ns +
 "Individual1");
 resource.addProperty(DC.title, title);
 model.write(file, “RDF/XML”);
```

Adding Reasoning Capabilities

- Using HP’s Jena
  - Create an RDFS model using the Jena API

```
String ns = "http://www.example.com/ex#";
Model rdfsEx = ModelFactory.createDefaultModel();
    Property p = rdfsEx.createProperty(ns, "p");
    Property q = rdfsEx.createProperty(ns, "q");
    rdfsEx.add(p, RDFS.subPropertyOf, q);
    rdfsEx.createResource(NS+"a").addProperty(p, "foo");
```

  - Adding the internal RDFS reasoner

```
Reasoner reasoner = ReasonerRegistry.getRDFSReasoner();
InfModel inf = ModelFactory.createInfModel(reasoner, rdfsEx);
```

Querying Ontologies

- SPARQL is to ontologies what SQL is to relational databases
  - W3C recommendation since 2008
- Designed using an SQL-like syntax
  - SELECT … FROM … WHERE
- The WHERE conditions are a triple pattern
- Returns graphs instead of a tables
- Example

  SELECT ?x ?y ?z WHERE { ?x ?y ?z }

  returns all the triples in the graph

Introduction to SPARQL (1)

- Selecting a single value

```
SELECT ?x
WHERE { ?x  <ex:hasName>  "John Smith" }
```

- Matching values from a graph

```
SELECT ?x ?fname
WHERE {?x  <ex:hasName>  ?fname}
```

- Also

```
SELECT ?name ?value
WHERE { ?x <ex:hasAttribute> ?attr .
                ?attr <ex:hasValue> ?value . }
```

Introduction to SPARQL (2)

- String matching using regular expressions

```
SELECT ?y
WHERE
{ ?x vcard:Given ?y .
  FILTER regex(?y, "r", "i") }
```

- Filtering values

```
SELECT ?resource
WHERE {
    ?resource info:age ?age .
    FILTER (?age >= 24) }
```

Introduction to SPARQL (3)

- The OPTIONAL construct to return information where available

```
 SELECT ?name ?age
 WHERE {
     ?person vcard:FN  ?name .
     OPTIONAL { ?person info:age ?age } }
```

- The UNION construct

```
 SELECT ?name
 WHERE {
    { [] foaf:name ?name } UNION { [] vCard:FN ?name }
```

Introduction to SPARQL (4)

- Result handling
  - ORDER BY, DISTINCT, OFFSET and LIMIT
    - Same as in SQL
  - CONSTRUCT
    - Ability to construct a new graph based on the results

```
 PREFIX foaf: <http://xmlns.com/foaf/0.1/>
   CONSTRUCT { <http://example.com/person#Alice> foaf:knows ?x }
   FROM <http://example.org/foaf/people>
   WHERE { ?x foaf:name ?name }
   ORDER BY desc(?name)
   LIMIT 10
```

Common Vocabularies (1)

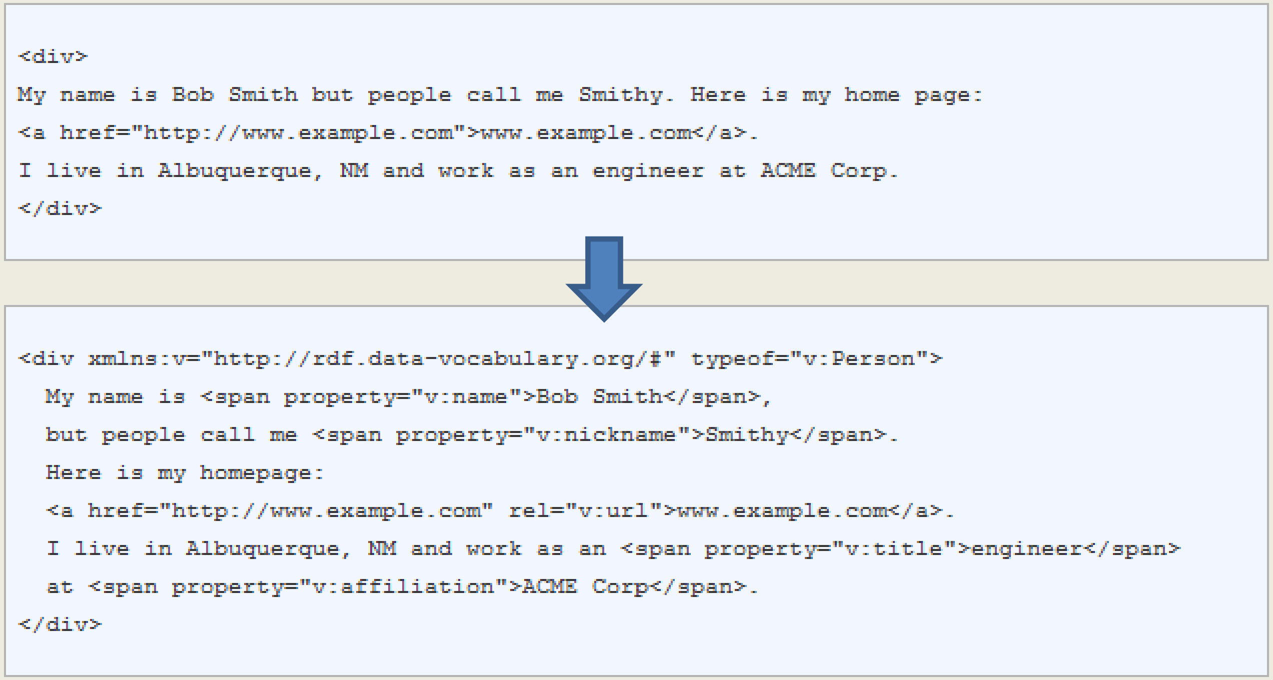
- DC
  - Describe library asset information
- SKOS
  - Simple Knowledge Organization Scheme
- FOAF
  - Friend of a friend
- SIOC
  - Semantically Interlinked Online Communities
- DBPedia
  - Extract structured information from Wikipedia

Common Vocabularies (2)

- Music ontology
  - Describe music concepts
- Good relations
  - Used in the e-commerce context
  - Supported by Google and Yahoo
- Basic Geo Vocabulary
  - Expresses spatial information using WGS84
- Creative commons
  - Express copyright information

Common Vocabularies (3)

- Microformats are open data standards for publishing structured information on the Web
- Simple, solve specific problems
- No change in display
- Examples
  - hCard
  - hCalendar
  - RDFa
- For SEO, see also schema.org

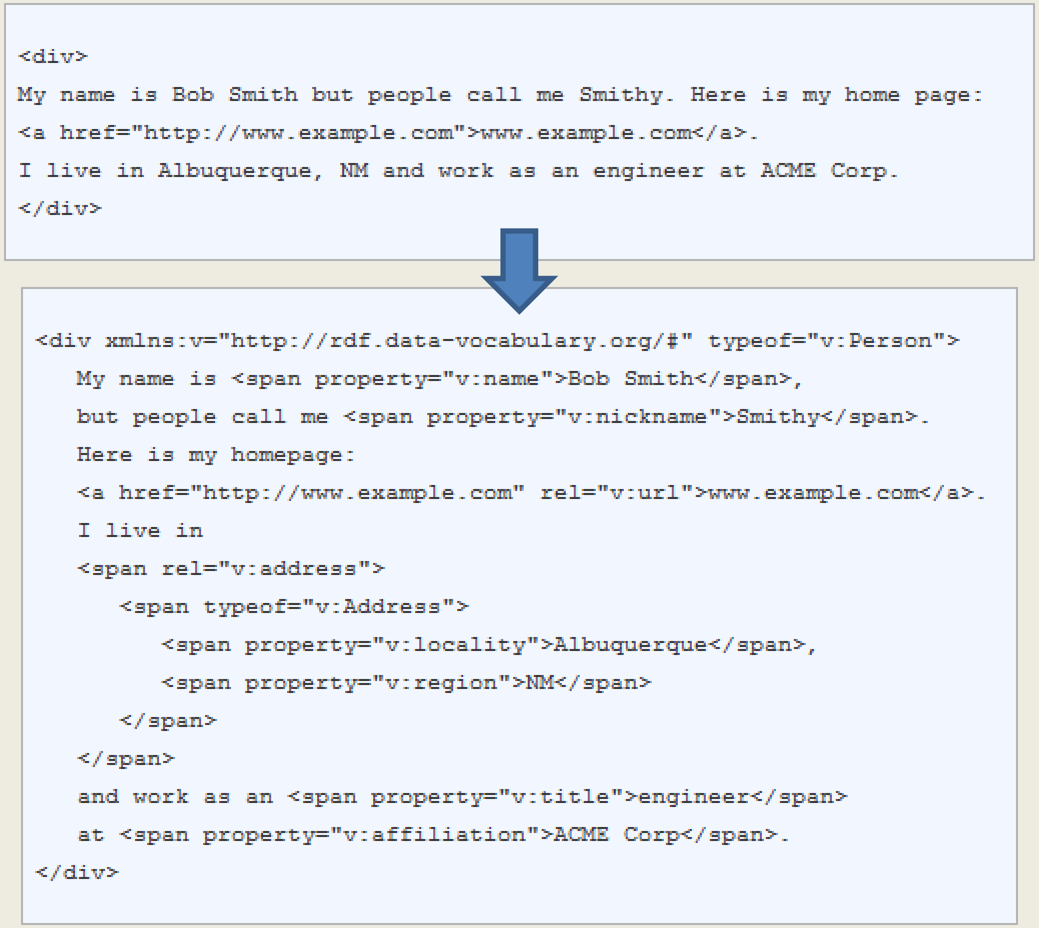


```
<div>
My name is Bob Smith but people call me Smithy. Here is my home page:
<a href="http://www.example.com">www.example.com</a>.
I live in Albuquerque, NM and work as an engineer at ACME Corp.
</div>
```

```
<div xmlns:v="http://rdf.data-vocabulary.org/#" typeof="v:Person">
  My name is <span property="v:name">Bob Smith</span>,
  but people call me <span property="v:nickname">Smithy</span>.
  Here is my homepage:
  <a href="http://www.example.com" rel="v:url">www.example.com</a>.
  I live in Albuquerque, NM and work as an <span property="v:title">engineer</span>
  at <span property="v:affiliation">ACME Corp</span>.
</div>
```

Embedded RDF

- RDFa
  - Embed RDF in XHTML documents
  - Uses <span>, <div>
  - Allows nested descriptions
- GRDDL
  - Obtain RDF from HTML pages
  - Uses XSLT for XML



```
<div>
My name is Bob Smith but people call me Smithy. Here is my home page:
<a href="http://www.example.com">www.example.com</a>.
I live in Albuquerque, NM and work as an engineer at ACME Corp.
</div>
```

```
<div xmlns:v="http://rdf.data-vocabulary.org/#" typeof="v:Person">
   My name is <span property="v:name">Bob Smith</span>,
   but people call me <span property="v:nickname">Smithy</span>.
   Here is my homepage:
   <a href="http://www.example.com" rel="v:url">www.example.com</a>.
   I live in
   <span rel="v:address">
      <span typeof="v:Address">
         <span property="v:locality">Albuquerque</span>,
         <span property="v:region">NM</span>
      </span>
   </span>
   and work as an <span property="v:title">engineer</span>
   at <span property="v:affiliation">ACME Corp</span>.
</div>
```

Both RDFa and GRDDL are W3C recommendations

Ontology Serialisation Formats

- RDF+XML

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns:dc="http://purl.org/dc/elements/1.1/">
   <rdf:Description rdf:about="http://en.wikipedia.org/wiki/Tony_Benn">
      <dc:title>Tony Benn</dc:title>
      <dc:publisher>Wikipedia</dc:publisher>
   </rdf:Description>
</rdf:RDF>
```

- N3 and Turtle (Turtle ⊆ N3)

```
@prefix dc: <http://purl.org/dc/elements/1.1/>.
<http://en.wikipedia.org/wiki/Tony_Benn>
  dc:title "Tony Benn";
  dc:publisher "Wikipedia".
```

- ... and, of course, in the database!

Lecture Outline

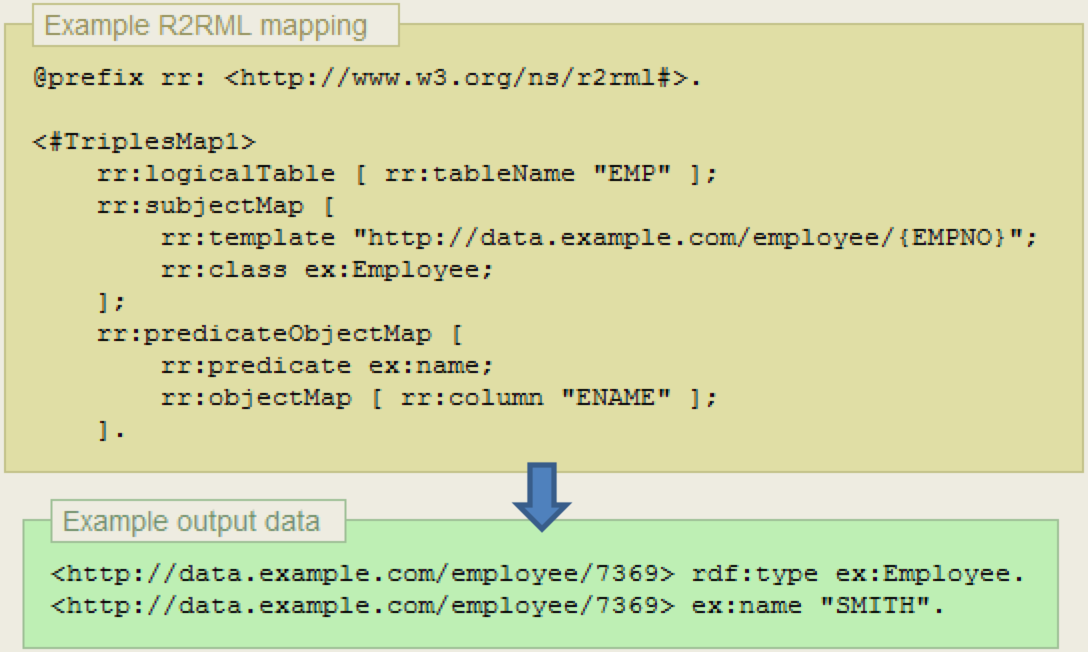
- Introduction to the Semantic Web
- Semantic Web Languages
- Publishing RDF Using OpenLink’s Virtuoso
- Linked Open Data and Examples

Triplestores

- A triplestore contains records in the form
  - (subject, predicate, object)
- Uses a relational database backend
- Saving an Ontology in a triplestore
  - Jena
    - http://jena.sourceforge.net
  - Sesame
    - http://www.openrdf.org
  - Virtuoso
    - http://virtuoso.openlinksw.com
  - Oracle
    - http://www.oracle.com/technetwork/database

RDB to RDF Mapping Language

- Several tools proposed in the early years
  - Triplify, D2OMapper, DB2OWL, VisaVis, R2O, MapOnto, …
- R2RML: a W3C working draft
  - Implementations
    - D2RQ
    - Virtuoso



Example R2RML mapping

```
@prefix rr: <http://www.w3.org/ns/r2rml#>.

<#TriplesMap1>
    rr:logicalTable [ rr:tableName "EMP" ];
    rr:subjectMap [
        rr:template "http://data.example.com/employee/{EMPNO}";
        rr:class ex:Employee;
    ];
    rr:predicateObjectMap [
        rr:predicate ex:name;
        rr:objectMap [ rr:column "ENAME" ];
    ].
```

Example output data

```
<http://data.example.com/employee/7369> rdf:type ex:Employee.
<http://data.example.com/employee/7369> ex:name "SMITH".
```

Virtuoso Overview (1)

- Open source and commercial version
- Can be used as
  - A web application server
  - A relational database repository
    - Offers a JDBC Driver
    - Collaborates with Jena
    - Offers Conductor, a GUI for server administration
  - A web service server
  - A triplestore
    - Export RDF data from same DB or others

Virtuoso Overview (2)

- RDF Views
  - Export relational data as triples
- SPARQL 1.1 support, plus
  - Full Text Queries
  - Geo Spatial Queries
  - Business Analytics and Intelligence
  - SQL Stored Procedure and Built-In Function exploitation from SPARQL
  - Create, Update, and Delete (SPARUL)
- Cluster Configuration
  - Parallel and Horizontal scaling

Virtuoso Overview (3)

- Extendable through VAD* Packages
  - Interactive SPARQL Query Builder
    - A GUI to create SPARQL queries
  - Sponger Middleware
    - Offers RDF Mappers to import data into Virtuoso
  - PubSubHub Protocol (for RSS)
    - Can be used to allow *push* behavior and subscriptions by clients
  - OAT (OpenLink AJAX Toolkit) Framework
    - Rich web application development

* Virtuoso Application Distribution

Virtuoso Reasoning Engine

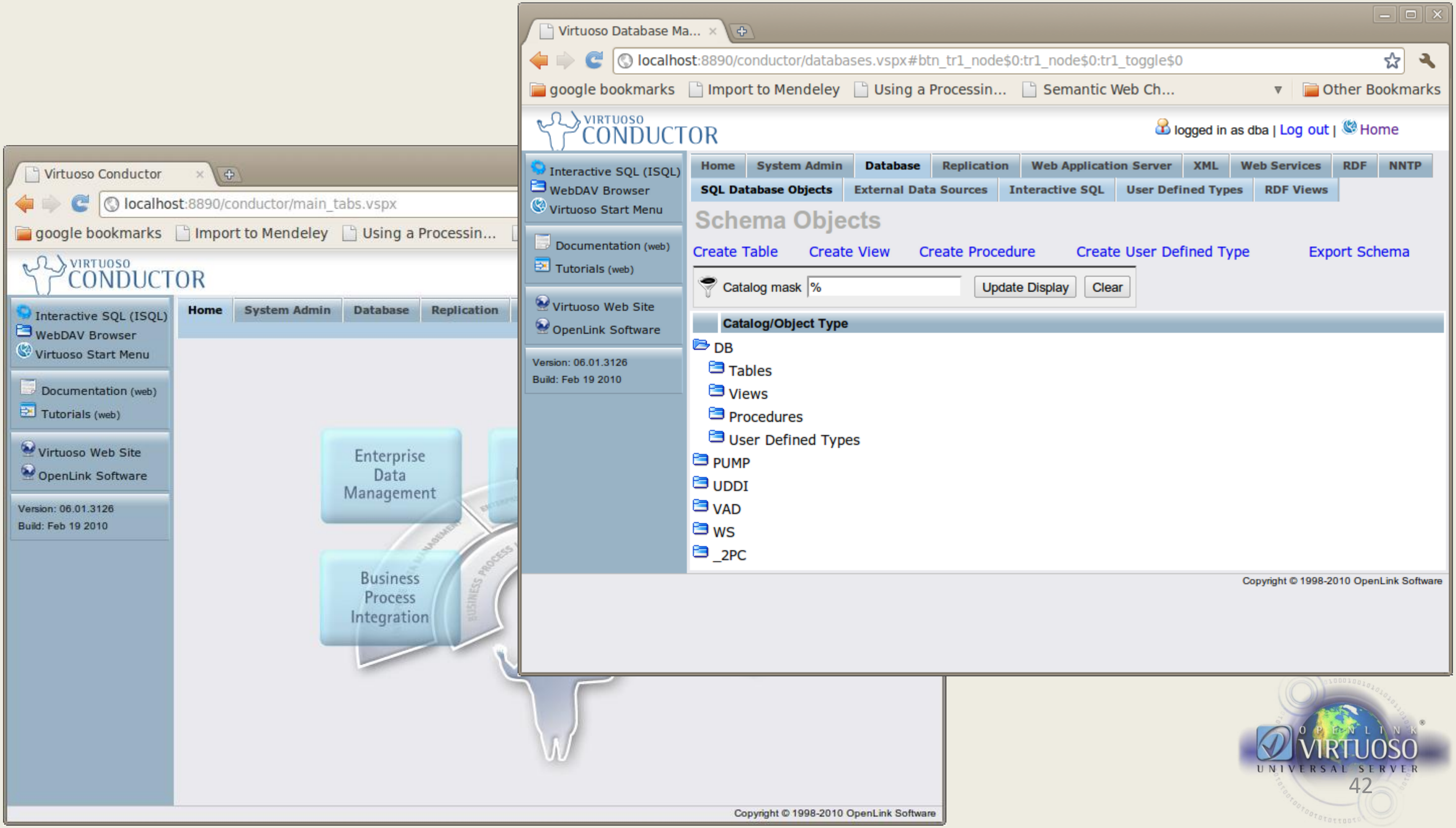
- Backward-chaining OWL reasoner coverage
  - rdfs:subClassOf
  - rdfs:subPropertyOf
  - owl:sameAs
  - owl:equivalentClass
  - owl:equivalentProperty
  - owl:InverseFunctionalProperty
  - owl:inverseOf
  - owl:SymmetricalProperty
  - owl:TransitiveProperty

Virtuoso Sponger

- An RDF-iser to bring data into the Semantic Web
- Sponger extracts RDF data from non-RDF sources
- A *Cartridge* per data source
- XSLT templates do the work
- Customisable and Programmable
  - Virtuoso PL, C++, Java

Virtuoso as a DB Server (1)

- Conductor: a GUI for server administration



Virtuoso Conductor

localhost:8890/conductor/main_tabs.vspx

google bookmarks | Import to Mendeley | Using a Processin...

VIRTUOSO CONDUCTOR

Interactive SQL (ISQL)
WebDAV Browser
Virtuoso Start Menu

Documentation (web)
Tutorials (web)

Virtuoso Web Site
OpenLink Software

Version: 06.01.3126
Build: Feb 19 2010

Home | System Admin | Database | Replication

Copyright © 1998-2010 OpenLink Software

Virtuoso Database Ma...

localhost:8890/conductor/databases.vspx#btn_tr1_node$0:tr1_node$0:tr1_toggle$0

google bookmarks | Import to Mendeley | Using a Processin... | Semantic Web Ch... | Other Bookmarks

VIRTUOSO CONDUCTOR

logged in as dba | Log out | Home

Interactive SQL (ISQL)
WebDAV Browser
Virtuoso Start Menu

Documentation (web)
Tutorials (web)

Virtuoso Web Site
OpenLink Software

Version: 06.01.3126
Build: Feb 19 2010

Home | System Admin | Database | Replication | Web Application Server | XML | Web Services | RDF | NNTP

SQL Database Objects | External Data Sources | Interactive SQL | User Defined Types | RDF Views

Schema Objects

Create Table | Create View | Create Procedure | Create User Defined Type | Export Schema

Catalog mask %   Update Display   Clear

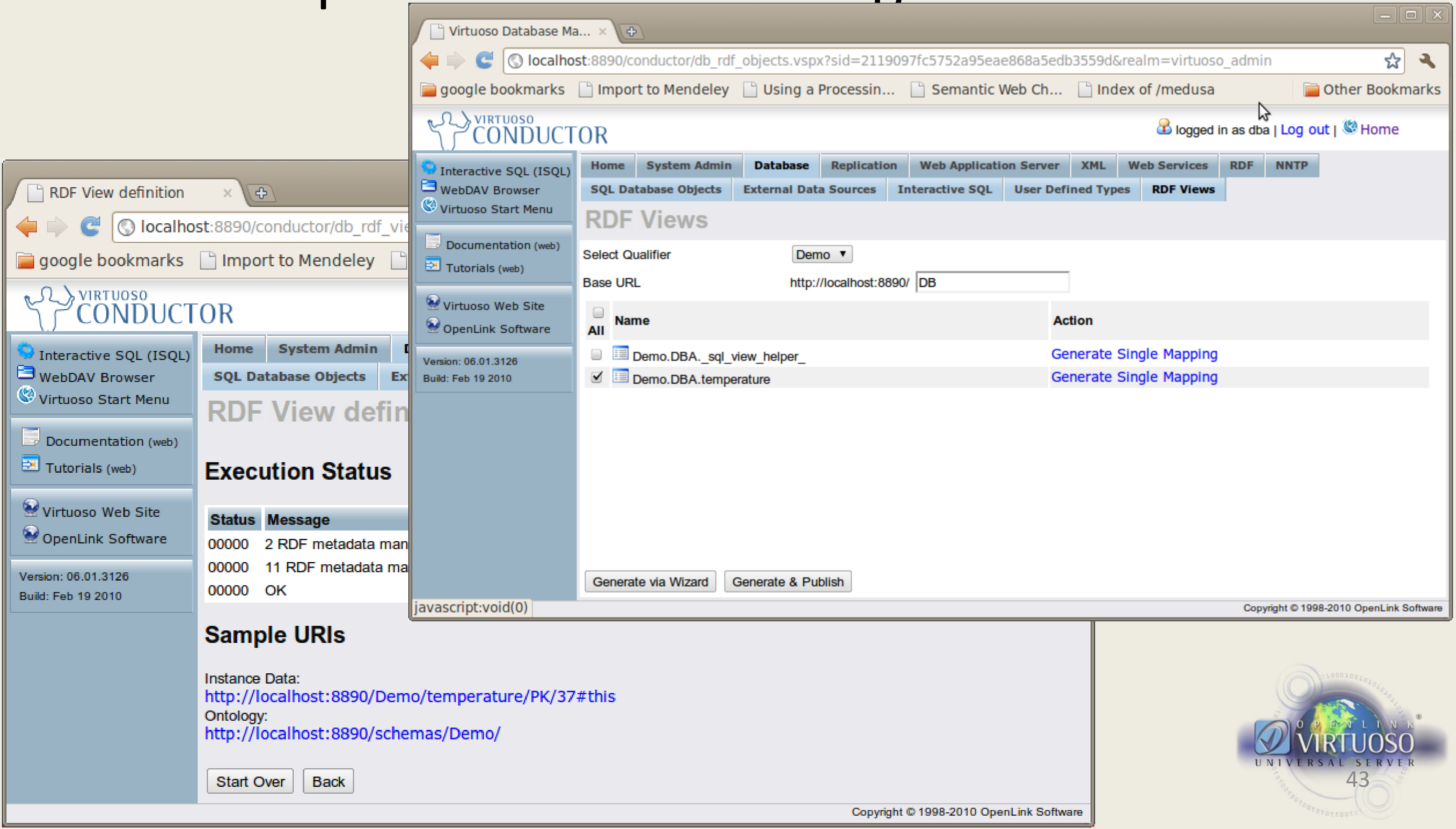
Catalog/Object Type

- DB
  - Tables
  - Views
  - Procedures
  - User Defined Types
- PUMP
- UDDI
- VAD
- WS
- _2PC

Copyright © 1998-2010 OpenLink Software

Virtuoso as a DB Server (2)

- Can export data as RDF using RDF Views



Virtuoso Database Ma...

localhost:8890/conductor/db_rdf_objects.vspx?sid=2119097fc5752a95eae868a5edb3559d&realm=virtuoso_admin

Version: 06.01.3126
Build: Feb 19 2010

RDF Views

Select Qualifier: Demo ▾

Base URL: http://localhost:8890/ DB

□ All	Name	Action
□	Demo.DBA._sql_view_helper_	Generate Single Mapping
☑	Demo.DBA.temperature	Generate Single Mapping

Copyright © 1998-2010 OpenLink Software

RDF View definition

localhost:8890/conductor/db_rdf_vie

Version: 06.01.3126
Build: Feb 19 2010

RDF View defin

Execution Status

Status	Message
00000	2 RDF metadata man
00000	11 RDF metadata ma
00000	OK

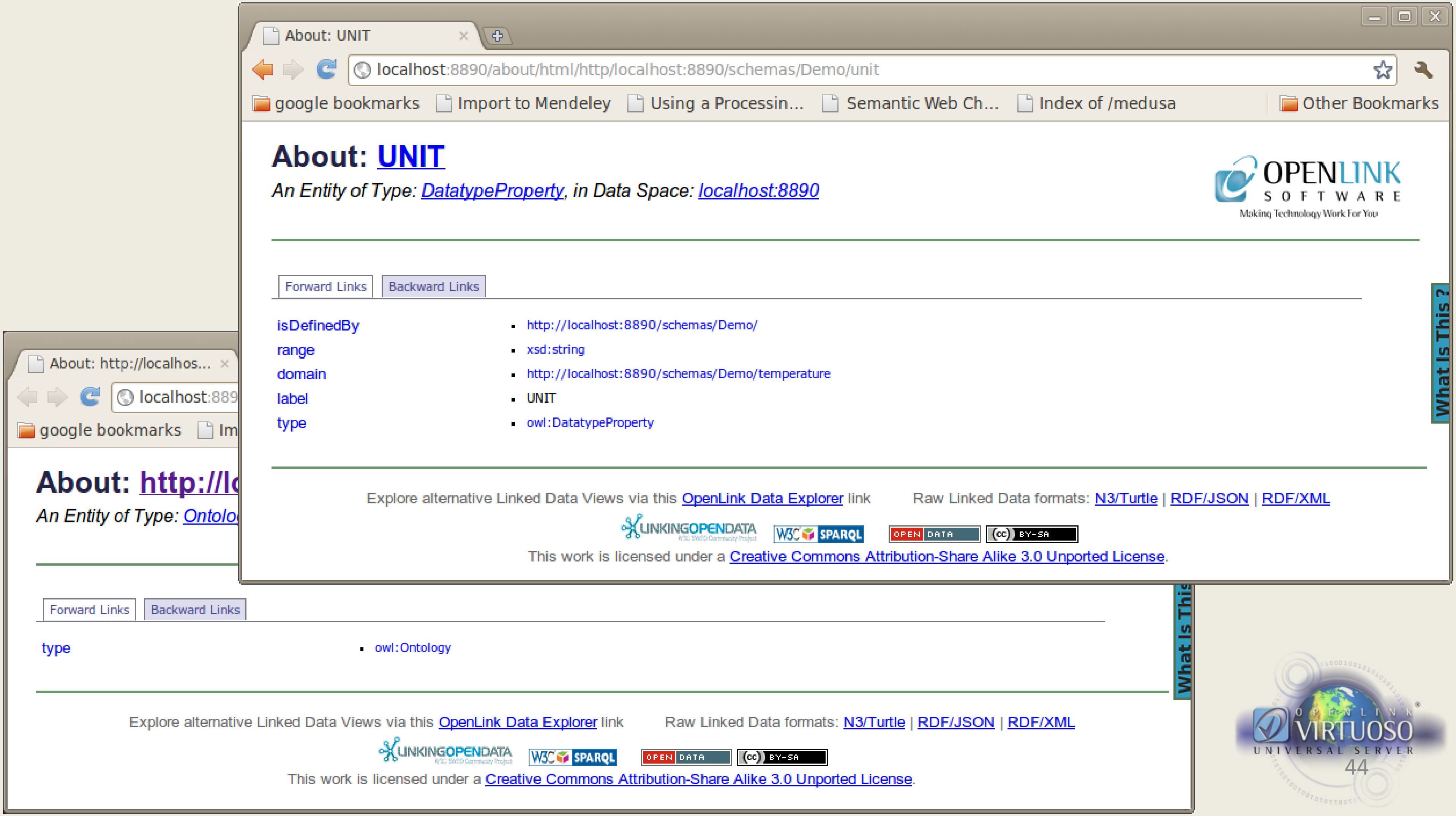
Sample URIs

Instance Data:
http://localhost:8890/Demo/temperature/PK/37#this
Ontology:
http://localhost:8890/schemas/Demo/

Copyright © 1998-2010 OpenLink Software

Virtuoso as an RDF Server (1)

- A URI for every resource, browseable repository



About: UNIT

localhost:8890/about/html/http/localhost:8890/schemas/Demo/unit

google bookmarks | Import to Mendeley | Using a Processin... | Semantic Web Ch... | Index of /medusa | Other Bookmarks

About: UNIT

An Entity of Type: DatatypeProperty, in Data Space: localhost:8890

Forward Links | Backward Links

isDefinedBy	http://localhost:8890/schemas/Demo/
range	xsd:string
domain	http://localhost:8890/schemas/Demo/temperature
label	UNIT
type	owl:DatatypeProperty

Explore alternative Linked Data Views via this OpenLink Data Explorer link   Raw Linked Data formats: N3/Turtle | RDF/JSON | RDF/XML

This work is licensed under a Creative Commons Attribution-Share Alike 3.0 Unported License.

About: http://localhos...

localhost:889

google bookmarks | Im

About: http://lo

An Entity of Type: Ontolo

Forward Links | Backward Links

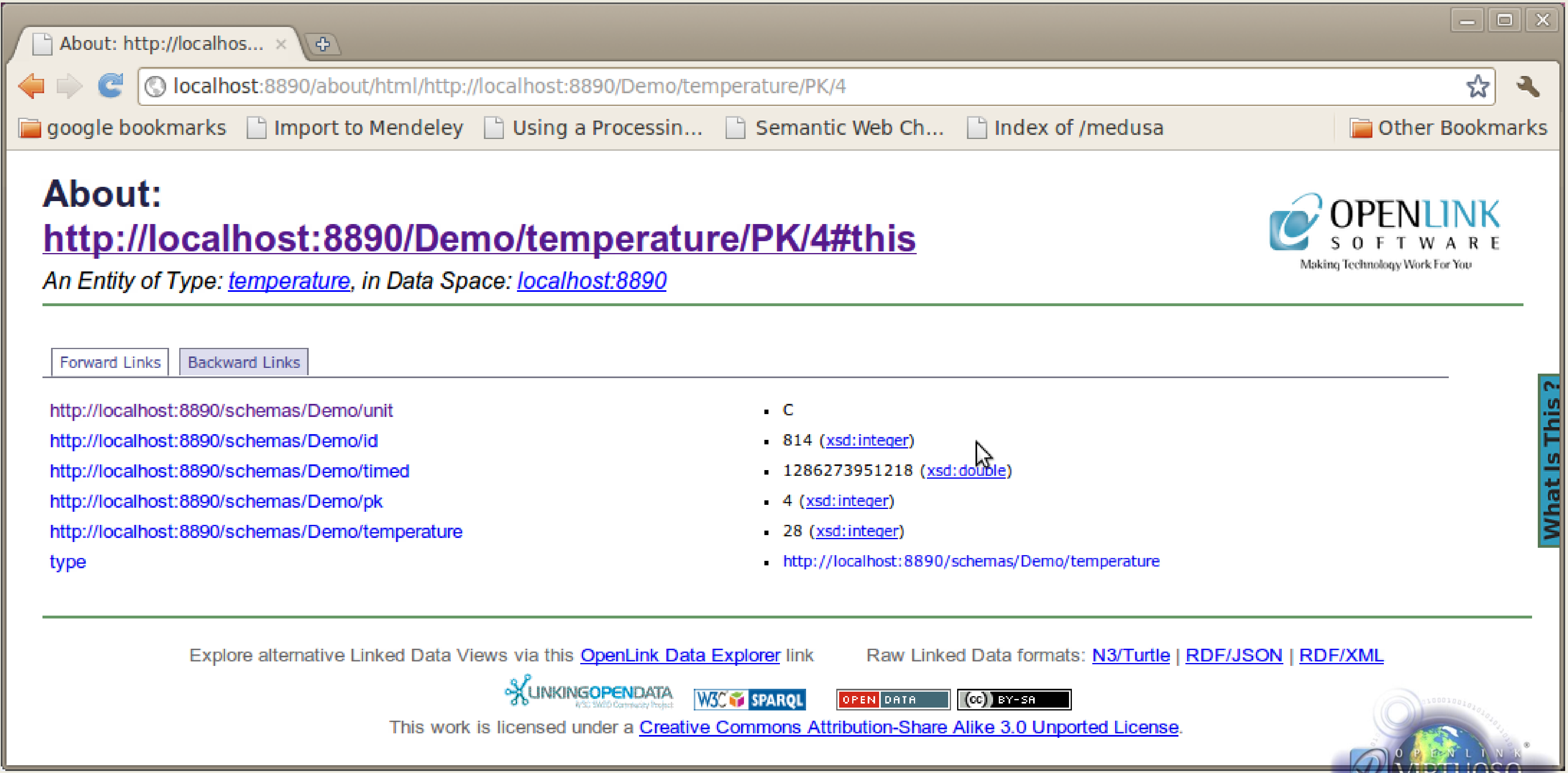
type	owl:Ontology

Explore alternative Linked Data Views via this OpenLink Data Explorer link   Raw Linked Data formats: N3/Turtle | RDF/JSON | RDF/XML

This work is licensed under a Creative Commons Attribution-Share Alike 3.0 Unported License.

Virtuoso as an RDF Server (2)

- Example: Measurement URI



About:
http://localhost:8890/Demo/temperature/PK/4#this

An Entity of Type: temperature, in Data Space: localhost:8890

Forward Links | Backward Links

http://localhost:8890/schemas/Demo/unit	C
http://localhost:8890/schemas/Demo/id	814 (xsd:integer)
http://localhost:8890/schemas/Demo/timed	1286273951218 (xsd:double)
http://localhost:8890/schemas/Demo/pk	4 (xsd:integer)
http://localhost:8890/schemas/Demo/temperature	28 (xsd:integer)
type	http://localhost:8890/schemas/Demo/temperature

Explore alternative Linked Data Views via this OpenLink Data Explorer link   Raw Linked Data formats: N3/Turtle | RDF/JSON | RDF/XML

This work is licensed under a Creative Commons Attribution-Share Alike 3.0 Unported License.

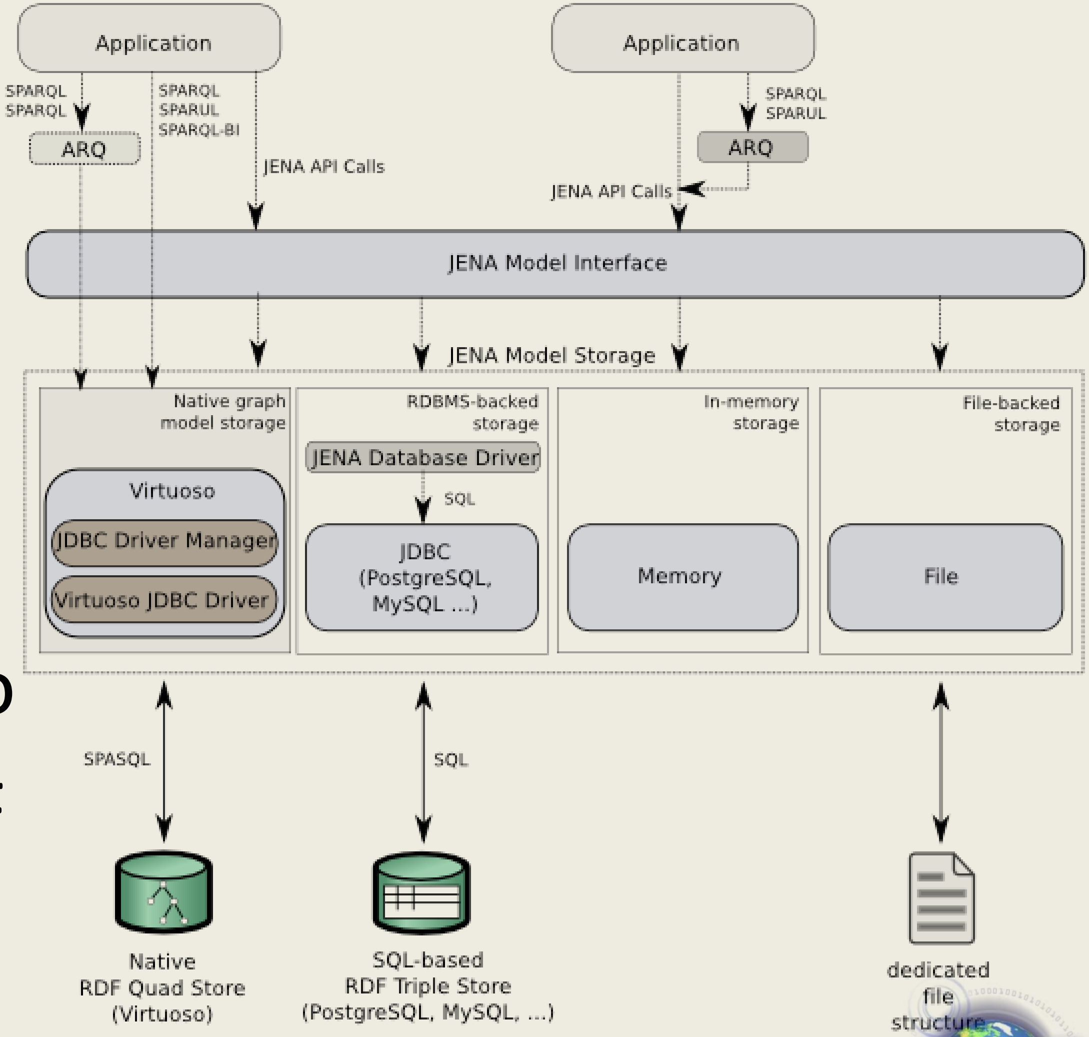
Virtuoso as an RDF Server (3)

- RDF data also accessible via
  - ODBC, JDBC, OLE DB, XMLA, ADO.NET
- Difficulties in extracting RDF Data
  - Tables *must* have a primary key
  - Mappings are defined using regular expressions and tend to be complicated

```
DB.DBA.URLREWRITE_CREATE_REGEX_RULE ( 'demo_rule6‘, 1, '/Demo/objects/([^#]*)‘, vector('path'), 1,
'/sparql?query=DESCRIBE+%%3Chttp%%3A//^{URIQADefaultHost}^/Demo/objects/%U%%3E+FROM+%
%3Chttp%%3A//^{URIQADefaultHost}^/Demo%%23%%3E&format=%U‘, vector('path', '*accept*'), null,
'(text/rdf.n3)|(application/rdf.xml)‘, 2, null );
```

Virtuoso Jena Provider

- Offered by OpenLink
- Native Graph Model Storage Provider
- Enables access to the Virtuoso RDF Quad store through Jena



Querying Remote Repositories

- XML over HTTP (RESTful approach)
  - http://demo.openlinksw.com/sparql?default-graph-uri=urn:lsid:ubio.org:namebank:11815&should-sponge=soft&query=SELECT+*+WHERE+{?s+?p+?o}&format=text/html
- No create/update/delete capabilities

SPARQL Query Interface (1)

- SPARQL queries can be named and stored
  - A query named *sparql-demo* listens to: http://localhost:8890/DAV/sparql-demo
- Can return results over HTTP (XML by default)
- MIME type of the RDF data
  - 'rdf+xml' (default) | 'n3' | 'turtle' | 'ttl‘

SPARQL Query Interface (2)

- SPARQL results example in RDF/XML

```
<ROOT>
<rdf:RDF  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#” xmlns:rs="http://www.w3.org/2005/sparql-results#"
xmlns:xsd="http://www.w3.org/2001/XMLSchema#">
<rs:results rdf:nodeID="rset">
    <rs:result rdf:nodeID="sol193">
        <rs:binding rdf:nodeID="sol193-0" rs:name="x“>
            <rs:value rdf:resource="http://localhost:8890/Demo/temperature/PK/4#this"/>
        </rs:binding>
        <rs:binding rdf:nodeID="sol193-1" rs:name="y“>
            <rs:value rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#type"/></rs:binding>
        <rs:binding rdf:nodeID="sol193-2" rs:name="z“>
            <rs:value rdf:resource="http://localhost:8890/schemas/Demo/temperature"/>
        </rs:binding>
    </rs:result>
…
</rs:results>
</rdf:RDF>
</ROOT>
```

Lecture Outline

- Introduction to the Semantic Web
- Semantic Web Languages
- Publishing RDF Using OpenLink’s Virtuoso
- Linked Open Data and Examples

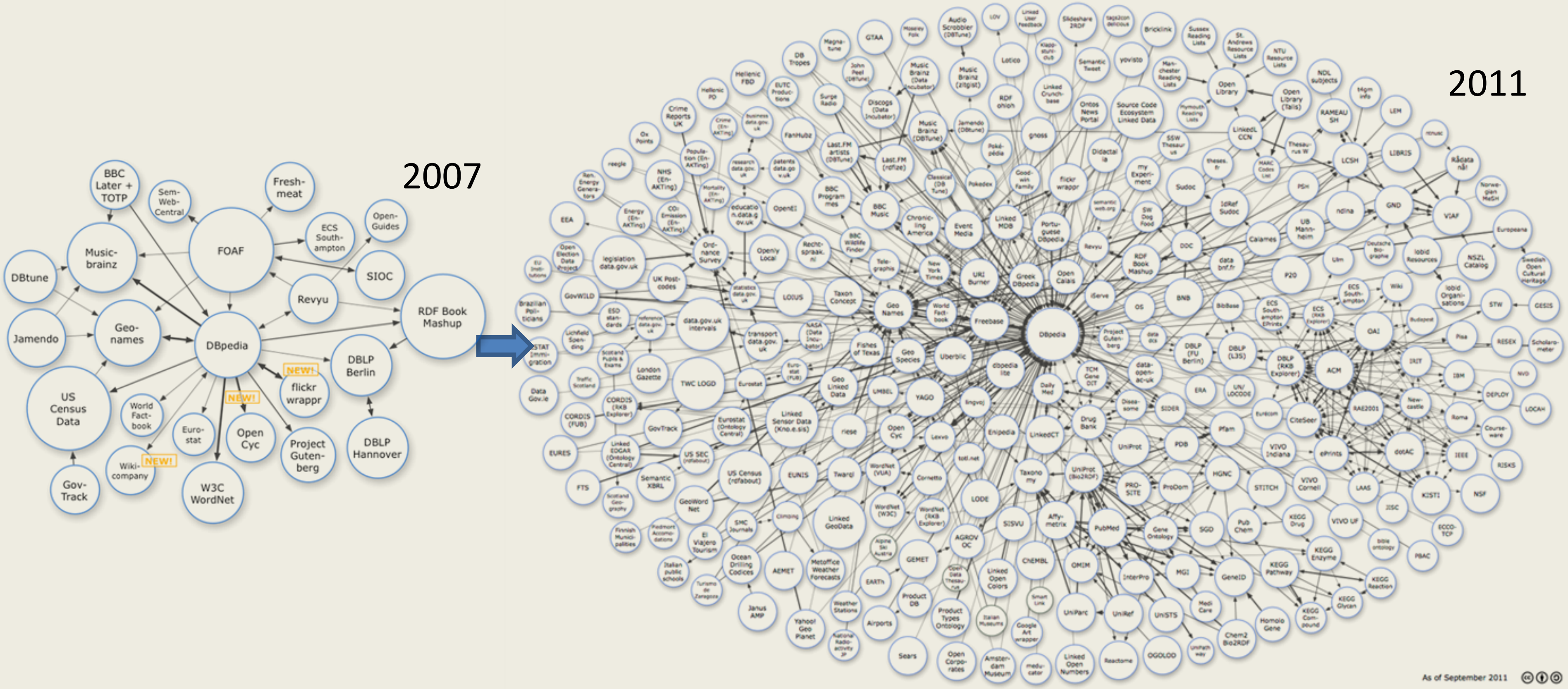
The Linked Open Data Cloud (1)

- Data available on the Web
  - Under an open license
- Available as structured data
  - Excel sheet instead of a scanned image
- Use non-proprietary format
  - CSV, RDF instead of DOC, XLS
- Use linked data format
  - URIs to identify things
- Linked to other people’s data
  - Provision of context

Also see: http://lab.linkeddata.deri.ie/2010/star-scheme-by-example/

The Linked Open Data Cloud (2)

- Interconnected datasets using URI’s and RDF



Source: http://linkeddata.org

Also see the ckan datahub: http://thedatahub.org/group/lodcloud

The Linked Open Data Cloud (3)

- Consumer capabilities
  - Access it, print it, store it locally, enter the data in another system
  - Process, aggregate, visualise, manipulate, export in another format, reuse
  - Avoid vendor lock-ins
- Publisher capabilities
  - Make data discoverable
  - Increase the value of the data
    - Allow added-value services
  - Fine-granular control over the data

Also see: http://lab.linkeddata.deri.ie/2010/star-scheme-by-example/

Open Government Data

- Prominent examples include
  - – data.gov (US)
  - – data.gov.uk, (UK)
  - – data.london.gov.uk (UK)
  - – digitaliser.dk (DK)
  - – data.govt.nz (NZ)
  - – linkedopendata.it (IT)
  - – geodata.gov.gr (GR)

DATA.GOV
EMPOWERING PEOPLE

data.gov.uk BETA
Opening up government

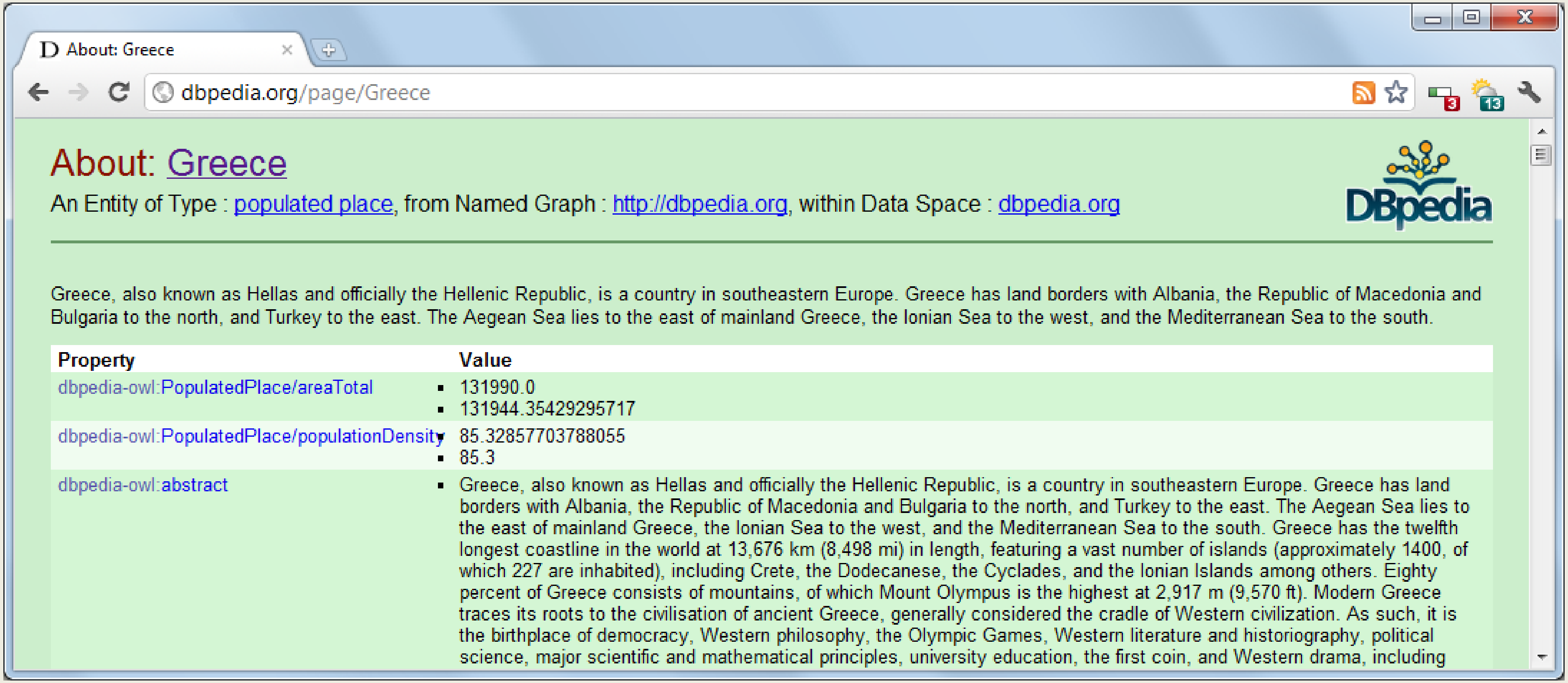
Digitalisér dk

```
<data
 geo="ανοικτά">
</data>
```

Promotional video: http://opengovernmentdata.org/film/

The DBPedia Project

- Structured information based on Wikipedia
- SPARQL endpoint: dbpedia.org/sparql
- Example: Greece on DBPedia



About: Greece

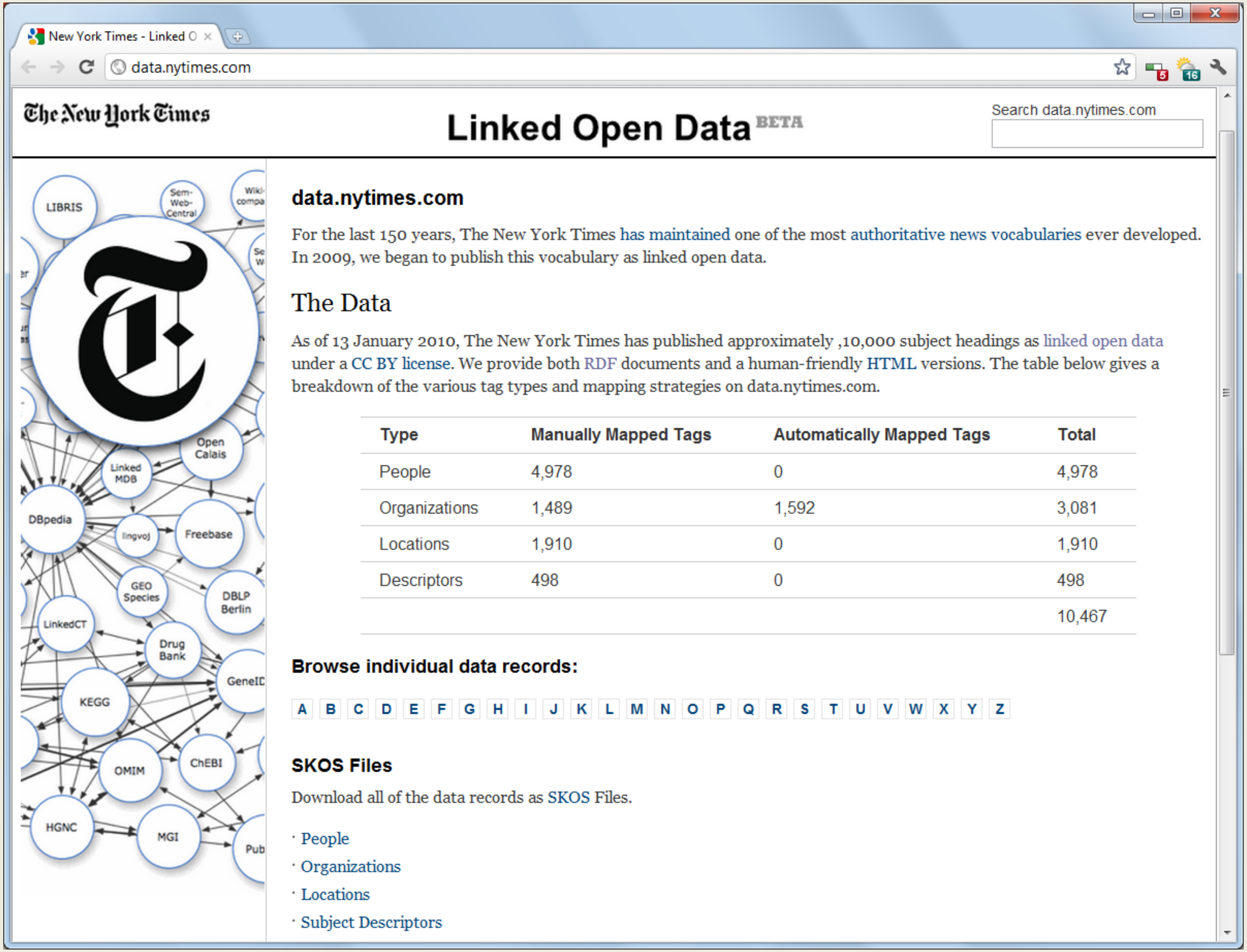
An Entity of Type : populated place, from Named Graph : http://dbpedia.org, within Data Space : dbpedia.org

Greece, also known as Hellas and officially the Hellenic Republic, is a country in southeastern Europe. Greece has land borders with Albania, the Republic of Macedonia and Bulgaria to the north, and Turkey to the east. The Aegean Sea lies to the east of mainland Greece, the Ionian Sea to the west, and the Mediterranean Sea to the south.

Property	Value
dbpedia-owl:PopulatedPlace/areaTotal	▪ 131990.0 ▪ 131944.35429295717
dbpedia-owl:PopulatedPlace/populationDensity	▪ 85.32857703788055 ▪ 85.3
dbpedia-owl:abstract	▪ Greece, also known as Hellas and officially the Hellenic Republic, is a country in southeastern Europe. Greece has land borders with Albania, the Republic of Macedonia and Bulgaria to the north, and Turkey to the east. The Aegean Sea lies to the east of mainland Greece, the Ionian Sea to the west, and the Mediterranean Sea to the south. Greece has the twelfth longest coastline in the world at 13,676 km (8,498 mi) in length, featuring a vast number of islands (approximately 1400, of which 227 are inhabited), including Crete, the Dodecanese, the Cyclades, and the Ionian Islands among others. Eighty percent of Greece consists of mountains, of which Mount Olympus is the highest at 2,917 m (9,570 ft). Modern Greece traces its roots to the civilisation of ancient Greece, generally considered the cradle of Western civilization. As such, it is the birthplace of democracy, Western philosophy, the Olympic Games, Western literature and historiography, political science, major scientific and mathematical principles, university education, the first coin, and Western drama, including

New York Times Public Data

- News data



The New York Times

Linked Open Data BETA

Search data.nytimes.com

data.nytimes.com

For the last 150 years, The New York Times has maintained one of the most authoritative news vocabularies ever developed. In 2009, we began to publish this vocabulary as linked open data.

The Data

As of 13 January 2010, The New York Times has published approximately ,10,000 subject headings as linked open data under a CC BY license. We provide both RDF documents and a human-friendly HTML versions. The table below gives a breakdown of the various tag types and mapping strategies on data.nytimes.com.

Type	Manually Mapped Tags	Automatically Mapped Tags	Total
People	4,978	0	4,978
Organizations	1,489	1,592	3,081
Locations	1,910	0	1,910
Descriptors	498	0	498
			10,467

Browse individual data records:

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

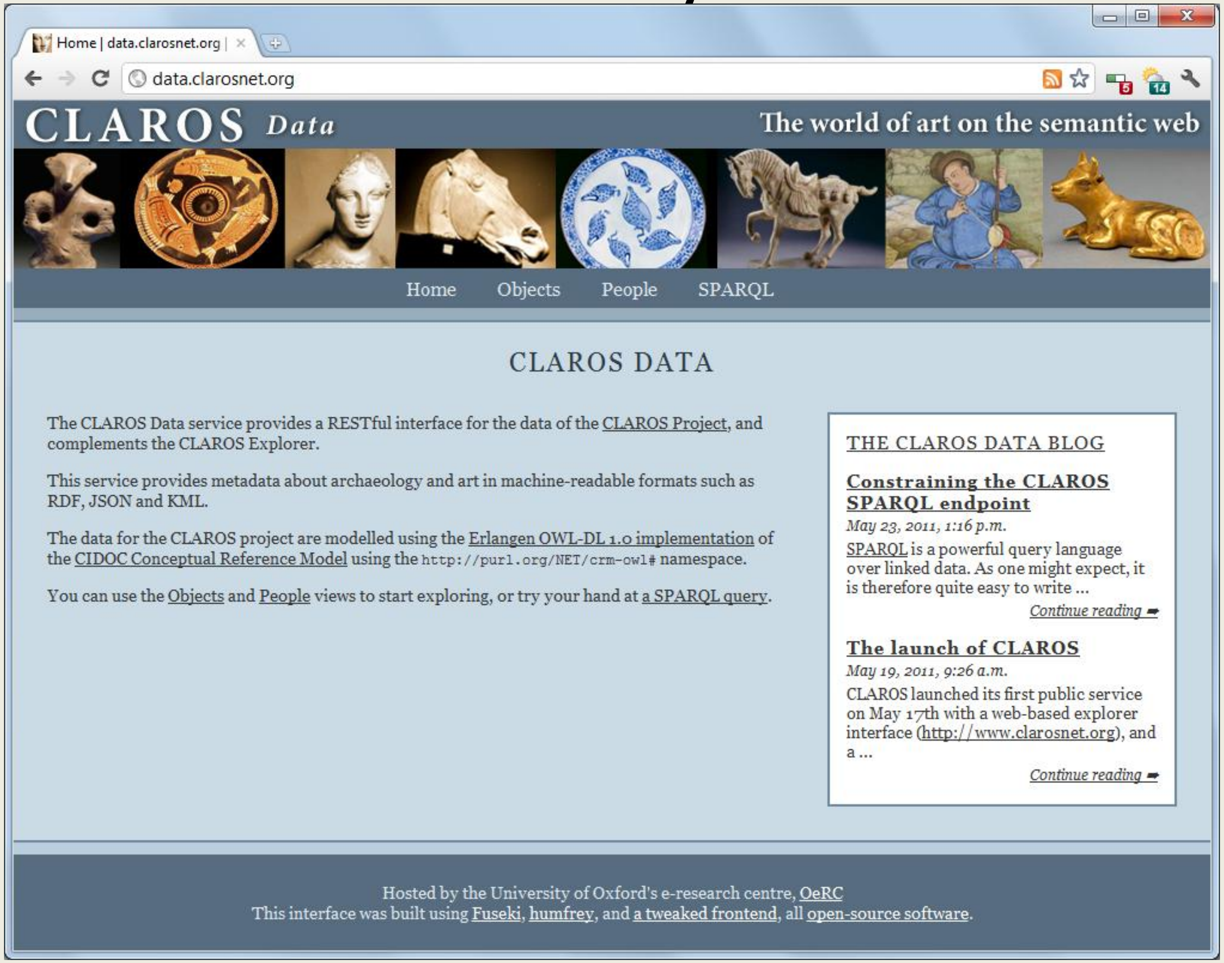
SKOS Files

Download all of the data records as SKOS Files.

- People
- Organizations
- Locations
- Subject Descriptors

The Claros Project

- The world of art on the Semantic Web
- Uses the CIDOC-CRM vocabulary to describe
  - Objects
  - Places
  - Periods
  - People
- OWL DL
- RESTful


Home | data.clarosnet.org
data.clarosnet.org
CLAROS Data
The world of art on the semantic web
Home
Objects
People
SPARQL

CLAROS DATA

The CLAROS Data service provides a RESTful interface for the data of the CLAROS Project, and complements the CLAROS Explorer.

This service provides metadata about archaeology and art in machine-readable formats such as RDF, JSON and KML.

The data for the CLAROS project are modelled using the Erlangen OWL-DL 1.0 implementation of the CIDOC Conceptual Reference Model using the `http://purl.org/NET/crm-owl#` namespace.

You can use the Objects and People views to start exploring, or try your hand at a SPARQL query.

THE CLAROS DATA BLOG

Constraining the CLAROS SPARQL endpoint

May 23, 2011, 1:16 p.m.

SPARQL is a powerful query language over linked data. As one might expect, it is therefore quite easy to write ...

Continue reading ➡

The launch of CLAROS

May 19, 2011, 9:26 a.m.

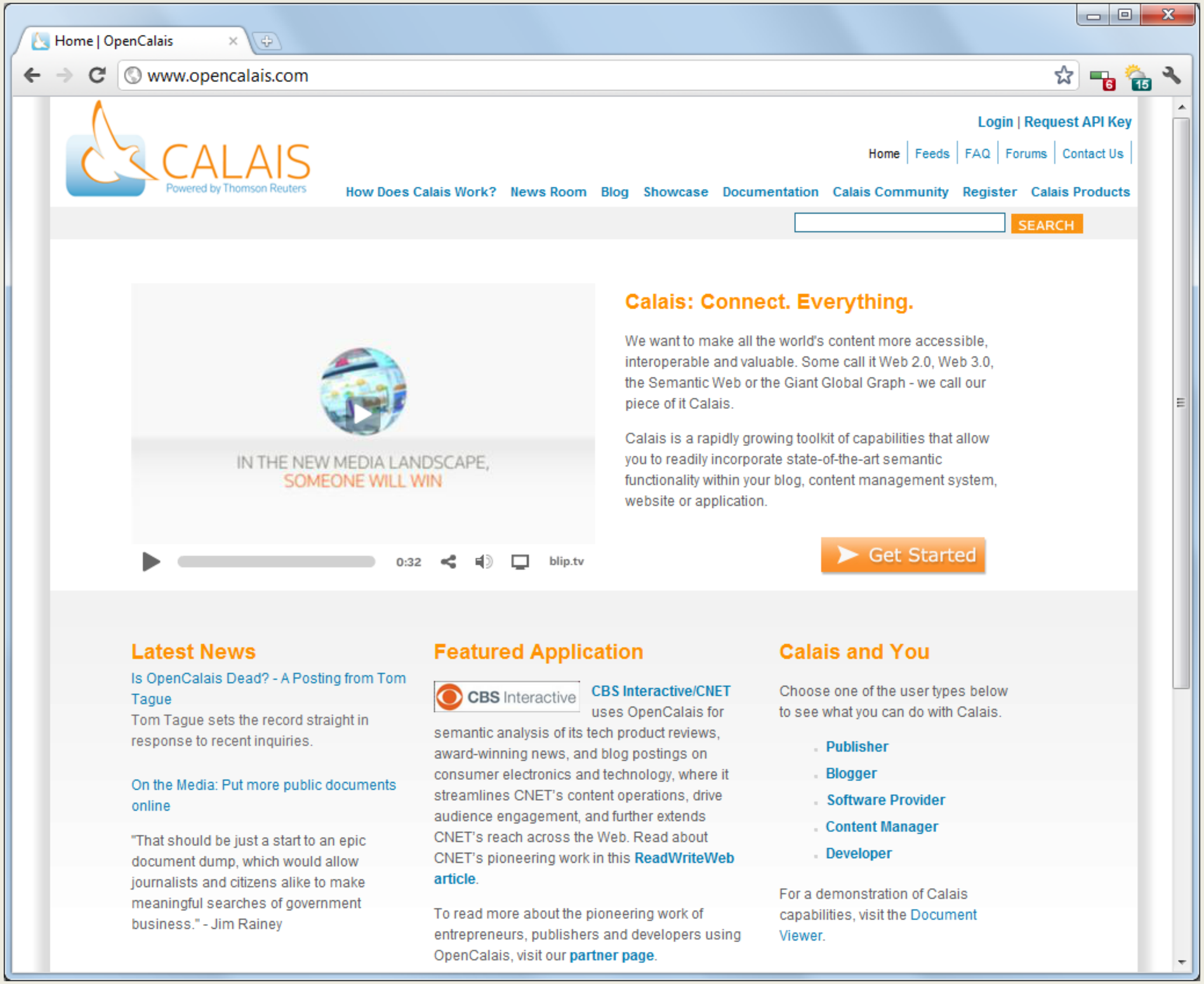
CLAROS launched its first public service on May 17th with a web-based explorer interface (http://www.clarosnet.org), and a ...

Continue reading ➡

Hosted by the University of Oxford's e-research centre, OeRC
This interface was built using Fuseki, humfrey, and a tweaked frontend, all open-source software.

The OpenCalais Project

- Creates semantic metadata for submitted content
- By Thomson Reuters
- Free to all
- Extracts RDF using NLP
- Uses WS


Home | OpenCalais
www.opencalais.com
CALAIS
Powered by Thomson Reuters
Login | Request API Key
Home | Feeds | FAQ | Forums | Contact Us
How Does Calais Work? News Room Blog Showcase Documentation Calais Community Register Calais Products
SEARCH
IN THE NEW MEDIA LANDSCAPE,
SOMEONE WILL WIN
0:32
blip.tv
Get Started
CBS Interactive

Calais: Connect. Everything.

We want to make all the world's content more accessible, interoperable and valuable. Some call it Web 2.0, Web 3.0, the Semantic Web or the Giant Global Graph - we call our piece of it Calais.

Calais is a rapidly growing toolkit of capabilities that allow you to readily incorporate state-of-the-art semantic functionality within your blog, content management system, website or application.

Latest News

Is OpenCalais Dead? - A Posting from Tom Tague

Tom Tague sets the record straight in response to recent inquiries.

On the Media: Put more public documents online

"That should be just a start to an epic document dump, which would allow journalists and citizens alike to make meaningful searches of government business." - Jim Rainey

Featured Application

CBS Interactive/CNET uses OpenCalais for semantic analysis of its tech product reviews, award-winning news, and blog postings on consumer electronics and technology, where it streamlines CNET's content operations, drive audience engagement, and further extends CNET's reach across the Web. Read about CNET's pioneering work in this ReadWriteWeb article.

To read more about the pioneering work of entrepreneurs, publishers and developers using OpenCalais, visit our partner page.

Calais and You

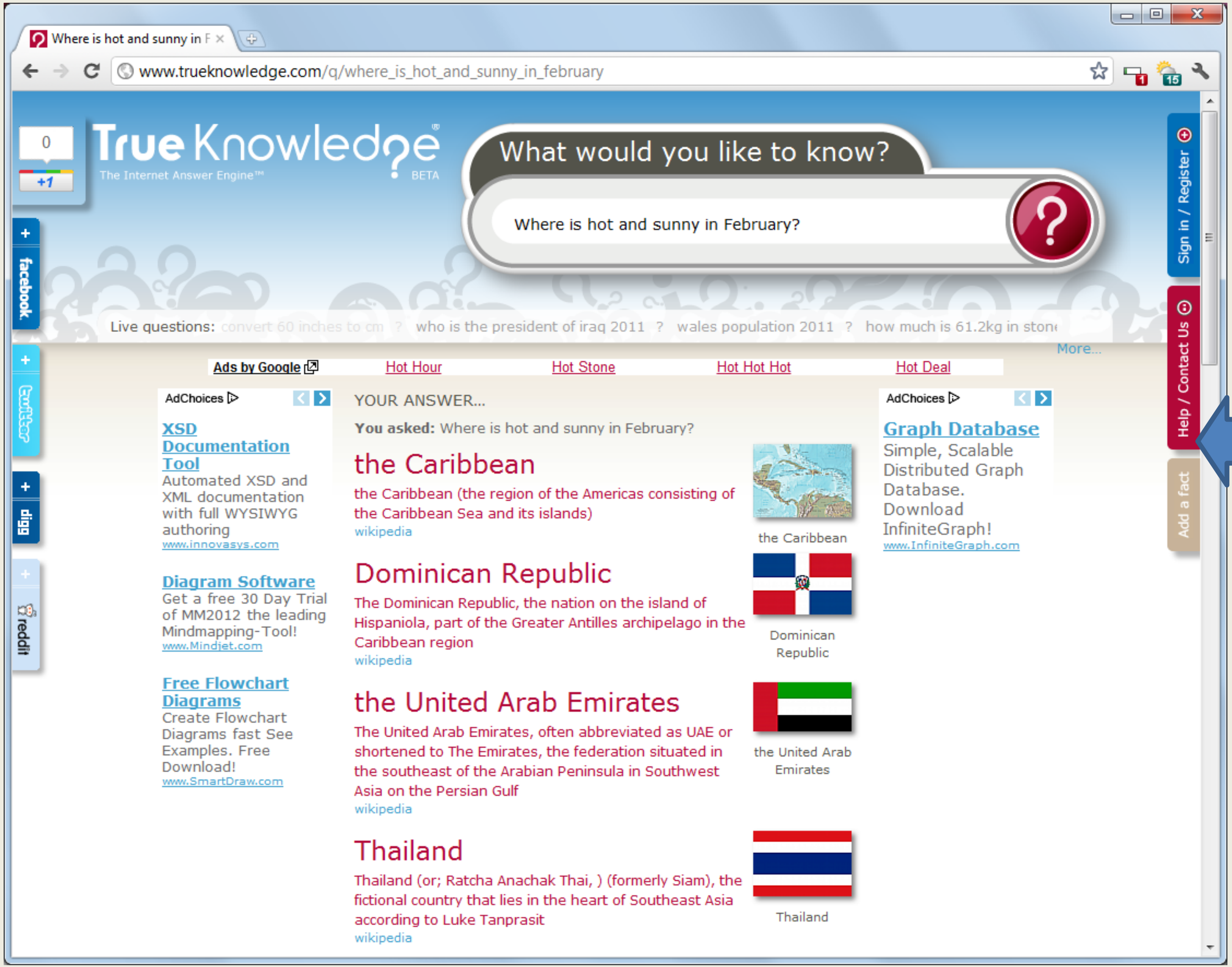
Choose one of the user types below to see what you can do with Calais.

- Publisher
- Blogger
- Software Provider
- Content Manager
- Developer

For a demonstration of Calais capabilities, visit the Document Viewer.

The TrueKnowledge Engine

- Based on facts, not keywords


Where is hot and sunny in F
www.trueknowledge.com/q/where_is_hot_and_sunny_in_february
0
+1
True Knowledge
The Internet Answer Engine™
BETA
facebook
twitter
digg
reddit
Sign in / Register
Help / Contact Us
Add a fact

What would you like to know?

Where is hot and sunny in February?

Live questions: convert 60 inches to cm ? who is the president of iraq 2011 ? wales population 2011 ? how much is 61.2kg in ston

More...

Ads by Google Hot Hour Hot Stone Hot Hot Hot Hot Deal

AdChoices

XSD Documentation Tool
Automated XSD and XML documentation with full WYSIWYG authoring
www.innovasys.com

Diagram Software
Get a free 30 Day Trial of MM2012 the leading Mindmapping-Tool!
www.Mindjet.com

Free Flowchart Diagrams
Create Flowchart Diagrams fast See Examples. Free Download!
www.SmartDraw.com

YOUR ANSWER...

You asked: Where is hot and sunny in February?

the Caribbean

the Caribbean (the region of the Americas consisting of the Caribbean Sea and its islands)
wikipedia

the Caribbean

Dominican Republic

The Dominican Republic, the nation on the island of Hispaniola, part of the Greater Antilles archipelago in the Caribbean region
wikipedia

Dominican Republic

the United Arab Emirates

The United Arab Emirates, often abbreviated as UAE or shortened to The Emirates, the federation situated in the southeast of the Arabian Peninsula in Southwest Asia on the Persian Gulf
wikipedia

the United Arab Emirates

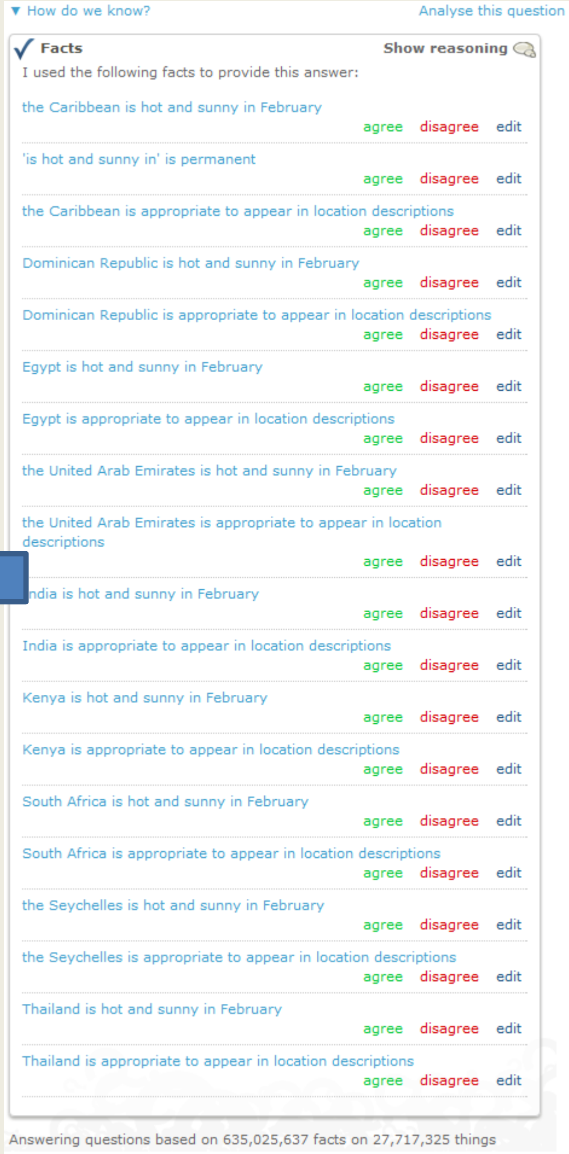
Thailand

Thailand (or; Ratcha Anachak Thai,) (formerly Siam), the fictional country that lies in the heart of Southeast Asia according to Luke Tanprasit
wikipedia

Thailand

AdChoices

Graph Database
Simple, Scalable Distributed Graph Database. Download InfiniteGraph!
www.InfiniteGraph.com



✓ **Facts**

I used the following facts to provide this answer:

the Caribbean is hot and sunny in February
agree disagree edit

'is hot and sunny in' is permanent
agree disagree edit

the Caribbean is appropriate to appear in location descriptions
agree disagree edit

Dominican Republic is hot and sunny in February
agree disagree edit

Dominican Republic is appropriate to appear in location descriptions
agree disagree edit

Egypt is hot and sunny in February
agree disagree edit

Egypt is appropriate to appear in location descriptions
agree disagree edit

the United Arab Emirates is hot and sunny in February
agree disagree edit

the United Arab Emirates is appropriate to appear in location descriptions
agree disagree edit

ndia is hot and sunny in February
agree disagree edit

India is appropriate to appear in location descriptions
agree disagree edit

Kenya is hot and sunny in February
agree disagree edit

Kenya is appropriate to appear in location descriptions
agree disagree edit

South Africa is hot and sunny in February
agree disagree edit

South Africa is appropriate to appear in location descriptions
agree disagree edit

the Seychelles is hot and sunny in February
agree disagree edit

the Seychelles is appropriate to appear in location descriptions
agree disagree edit

Thailand is hot and sunny in February
agree disagree edit

Thailand is appropriate to appear in location descriptions
agree disagree edit

Answering questions based on 635,025,637 facts on 27,717,325 things

Thank you for your attention! Questions?

Appendix

- Installing Virtuoso as a system service
  - Windows 7
    - Download and extract from http://virtuoso.openlinksw.com/dataspace/dav/wiki/Main/VOSDownload
    - Open a command line prompt as administrator
    - Register required DLL
      - regsvr32 virtodbc.dll
    - Install service
      - virtuoso-t +service screate +instance "DB" +configfile virtuoso.ini
  - Ubuntu 10.04 LTS
    - sudo apt-get install virtuoso-server