Building Semantic Web tools for Bioinformatics

Andrea Splendiani
(andrea.splendiani@bbsrc.ac.uk)
Outline

The building blocks of the Semantic Web

HowTo: export your data in semantic web formats, build a knowledge bases and query it.

Semantic Web application development

Building Semantic Web tools for Bioinformatics
6th annual meeting of the Bioinformatics ITalian Sociaety, Genova, 18-20 Apr. 2009
Outline

The building blocks of the Semantic Web

HowTo: export your data in semantic web formats, build a knowledge bases and query it.

Semantic Web application development

Extra time: Application examples
Outline

The building blocks of the Semantic Web

HowTo: export your data in semantic web formats, build a knowledge bases and query it.

Semantic Web application development

Building Semantic Web tools for Bioinformatics
6th annual meeting of the Bioinformatics ITalian Sociaety, Genova, 18-20 Apr. 2009
The building blocks of the Semantic Web

HowTo: export your data in semantic web formats, build a knowledge bases and query it.

Semantic Web application development
The building blocks of the Semantic Web

- The Semantic web is a set of technologies
- Different technologies address different needs, and not all of them need to be used
- Different technologies have different level of maturity
The building blocks of the Semantic Web

• The Semantic web is a set of technologies
• Different technologies address different needs, and not all of them need to be used
• Different technologies have different level of maturity

★ Mature technology
The building blocks of the Semantic Web

- The Semantic web is a set of technologies
- Different technologies address different needs, and not all of them need to be used
- Different technologies have different level of maturity

⭐ Mature technology
The building blocks of the Semantic Web

- The Semantic web is a set of technologies
- Different technologies address different needs, and not all of them need to be used
- Different technologies have different level of maturity

★ Mature technology
The building blocks of the Semantic Web

- The Semantic web is a set of technologies
- Different technologies address different needs, and not all of them need to be used
- Different technologies have different level of maturity

★ Mature technology
★ Refinements in progress, usable.
The building blocks of the Semantic Web

- The Semantic web is a set of technologies
- Different technologies address different needs, and not all of them need to be used
- Different technologies have different level of maturity

⭐ Mature technology
⭐⭐ Refinements in progress, usable.
The building blocks of the Semantic Web

• The Semantic web is a set of technologies
• Different technologies address different needs, and not all of them need to be used
• Different technologies have different level of maturity

🌟 Mature technology
🌟 Refinements in progress, usable.
The building blocks of the Semantic Web

- The Semantic web is a set of technologies
- Different technologies address different needs, and not all of them need to be used
- Different technologies have different level of maturity

★ Mature technology
★ Refinements in progress, usable.
The **building blocks** of the Semantic Web

- The Semantic web is a set of technologies
- Different technologies address different needs, and not all of them need to be used
- Different technologies have different level of maturity

⭐ Mature technology

⭐ Refinements in progress, usable.

⭐ Not so standard yet, but usable.
The building blocks of the Semantic Web

- The Semantic web is a set of technologies
- Different technologies address different needs, and not all of them need to be used
- Different technologies have different level of maturity

⭐ Mature technology
⭐⭐ Refinements in progress, usable.
⭐⭐⭐ Not so standard yet, but usable.
The building blocks of the Semantic Web
The **building blocks** of the Semantic Web

**URIs** are global names for information resources.

Examples:
The **building blocks** of the Semantic Web

URIs are global names for information resources.

Examples:

**The building blocks of the Semantic Web**

**URIs** are global names for information resources.

Examples:

- `http://purl.org/obo/owl/NCBITaxon#NCBITaxon_6`
The building blocks of the Semantic Web

URIs are global names for information resources.

Examples:

• http://purl.org/obo/owl/NCBITaxon#NCBITaxon_6
• http://bio2rdf.org/html/go:0032283
The building blocks of the Semantic Web

URIs are global names for information resources.

Examples:

- http://purl.org/obo/owl/NCBITaxon#NCBITaxon_6
  (http://www.ebi.ac.uk/ego/GTerm?id=GO:0032283)
The **building blocks** of the Semantic Web

**URIs** are global names for information resources.

**Examples:**

- http://purl.org/obo/owl/NCBITaxon#NCBITaxon_6
  (http://www.ebi.ac.uk/ego/GTerm?id=GO:0032283)
- info:pmid/18460179
The **building blocks** of the Semantic Web

**URIs** are global names for information resources.

Examples:

- http://purl.org/obo/owl/NCBITaxon#NCBITaxon_6
  (http://www.ebi.ac.uk/ego/GTerm?id=GO:0032283)
- info:pmid/18460179
- http://biocyc.org/biopax/biopax#smallMolecule84998
The **building blocks** of the Semantic Web

**URIs** are global names for information resources.

Examples:

- http://purl.org/obo/owl/NCBITaxon#NCBITaxon_6
  (http://www.ebi.ac.uk/ego/GTerm?id=GO:0032283)
- info:pmid/18460179
- http://biocyc.org/biopax/biopax#smallMolecule84998
- http://www.reactome.org/biopax#H2O__ChEBI_15377_
The **building blocks** of the Semantic Web

There are a few proposals for standard ways to define **URIs**, a consensus will be hopefully reached soon.

In a “local” integration processes, any style of URIs will work. In general, a conversion to a different “style” is relatively straightforward.

**URIs** are “global” names. There are also “local names” (blank nodes). They refer to objects whose identity is not identified across resources.

- [http://neurocommons.org/page/Shared_names](http://neurocommons.org/page/Shared_names)
- [http://esw.w3.org/topic/HCLSIG_BioRDF_Subgroup/Tasks/URI_Best_Practices/Recommendations](http://esw.w3.org/topic/HCLSIG_BioRDF_Subgroup/Tasks/URI_Best_Practices/Recommendations)
The building blocks of the Semantic Web
RDF is the language to express types and relations among information resource (what can be identified through a URI).

- It is made of a set of simple statements: subject predicate object
- It is conceptually a graph
- Several syntaxes support its representation: XML, N3, N-Triples...
The **building blocks** of the Semantic Web

example of **RDF**

More info:

Building Semantic Web tools for Bioinformatics
6th annual meeting of the Bioinformatics ITalian Sociaety, Genova, 18-20 Apr. 2009
The building blocks of the Semantic Web

example of RDF

Readability can be improved by the use of namespaces

More info:

Building Semantic Web tools for Bioinformatics
6th annual meeting of the Bioinformatics ITalian Sociaety, Genova, 18-20 Apr. 2009
The building blocks of the Semantic Web

Example of RDF in XML

```xml
<rdf:RDF
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"/>
<rdf:Description rdf:about="http://purl.org/obo/owl/GO#GO_0006915">
  <rdfs:label xml:lang="en">apoptosis</rdfs:label>
  <rdfs:subClassOf rdf:resource="http://purl.org/obo/owl/GO#GO_0012501"/>
</rdf:Description>
<rdf:Description rdf:about="http://purl.org/obo/owl/GO#GO_0012501">
  <rdfs:label xml:lang="en">programmed cell death</rdfs:label>
</rdf:Description>
</rdf:RDF>

• Not the only XML serialization possible!
• Some more details (language, types...)

Building Semantic Web tools for Bioinformatics
6th annual meeting of the Bioinformatics ITalian Sociaety, Genova, 18-20 Apr. 2009
The building blocks of the Semantic Web

example of RDF in XML

```xml
<rdf:RDF
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
>
<rdf:Description rdf:about="http://purl.org/obo/owl/GO#GO_0006915">
  <rdfs:label xml:lang="en">apoptosis</rdfs:label>
  <rdfs:subClassOf>
    <rdf:Description rdf:about="http://purl.org/obo/owl/GO#GO_0012501">
      <rdfs:label xml:lang="en">programmed cell death</rdfs:label>
    </rdf:Description>
  </rdfs:subClassOf>
</rdf:Description>
</rdf:RDF>
```

- Not the only XML serialization possible!
- Some more details (language, types...)

- http://www.w3.org/TR/rdf-syntax-grammar/
example of RDF in N3-Triple

<http://purl.org/obo/owl/GO#GO_0006915> <http://www.w3.org/2000/01/rdf-schema#label> "apoptosis".


<http://purl.org/obo/owl/GO#GO_0012501> <http://www.w3.org/2000/01/rdf-schema#label> "programmed cell death".
The building blocks of the Semantic Web

Blank nodes in RDF
(XML and N3-Triple)

<rdf:Description rdf:nodeID="abc">
  <ex:city="Bedford" />
</rdf:Description>
The **building blocks** of the Semantic Web

The complex picture...

Building **Semantic Web** tools for Bioinformatics
6th annual meeting of the Bioinformatics ITalian Sociaety, Genova, 18-20 Apr. 2009
The building blocks of the Semantic Web

Don’t focus on the syntax, focus on the RDF Conceptual graph, and use a library to serialize it! (more later...)

http://www.w3.org/RDF/
The **building blocks** of the Semantic Web

![Diagram of building blocks of the Semantic Web]

- **Query**: SPARQL
- **Ontology**: OWL
- **Rules**: RIF
- **Data interchange**: RDF
- **XML**
- **URI**
- **Unicode**

**Building Semantic Web tools for Bioinformatics**

6th annual meeting of the Bioinformatics ITalian Society, Genova, 18-20 Apr. 2009
The building blocks of the Semantic Web

SPARQL is a language to query RDF graphs.

Example:

```sparql
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT ?x
WHERE {
  ?x rdfs:subClassOf <http://purl.org/obo/owl/GO#GO_0012501> .
}
```

More info:
Building Semantic Web tools for Bioinformatics
6th annual meeting of the Bioinformatics ITalian Society, Genova, 18-20 Apr. 2009
SPARQL is a language to query RDF graphs.

What SPARQL does for you:
• allows to query resources in RDF
• allows to make queries over http

What SPARQL does not for you:
• it cannot specify which inference to use
• it cannot alter graphs (sparql/update)

More info:
• http://www.w3.org/TR/rdf-sparql-query/ 

More examples later...
The **building blocks** of the Semantic Web
The **building blocks** of the Semantic Web

**OWL** specifies properties of relations and types used in RDF (it specifies an **ontology**). **RDF-S** has a similar function, but is more limited in scope.

The **building blocks** of the Semantic Web

**OWL** specifies properties of relations and types used in RDF (it specifies an **ontology**).

**RDF-S** has a similar function, but is more limited in scope.
The **building blocks** of the Semantic Web

**OWL** specifies properties of relations and types used in RDF (it specifies an **ontology**). **RDF-S** has a similar function, but is more limited in scope.

**RDF-S:**
- subClassOf
- subPropertyOf
- ...

**OWL:**
- Class Union/Intersection
- Universal/Existential restriction
- Property Domain/Range
- Transitive/Reflexive/Functional properties
- ...

More info:
Building Semantic Web tools for Bioinformatics
6th annual meeting of the Bioinformatics ITalian Sociaety, Genova, 18-20 Apr. 2009
The building blocks of the Semantic Web

OWL

Example:

Cell motility part of localization of cell

<owl:Class rdf:about="GO#GO_0006928">
  <rdfs:label xml:lang="en">cell motility</rdfs:label>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="obo#part_of"/></owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
The building blocks of the Semantic Web

OWL

Example:
cell motility part of localization of cell

```
<owl:Class rdf:about="GO#GO_0006928">
  <rdfs:label xml:lang="en">cell motility</rdfs:label>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="obo#part_of"/>
      <owl:someValuesFrom rdf:resource="GO#GO_0051674"/>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
```

More info:
Building Semantic Web tools for Bioinformatics
6th annual meeting of the Bioinformatics ITalian Sociaet, Genova, 18-20 Apr. 2009
Why OWL?

OWL is designed for automatic classification: Given a description in OWL of two biological processes, a “reasoner” could establish whether two processes are equivalent, if one is a more specific kind of the other, or if they are disjoint.

More info:

http://www.co-ode.org/resources/tutorials/ProtegeOWLTutorial.pdf
The building blocks of the Semantic Web

WHICH OWL?

RDF-S

OWL

OWL-Lite

OWL-DL

OWL-Full

RDF-S

Based on
Description Logic
(decidable subset of first order logic)
Computable (mostly) in polynomial
time...

More info:
Building Semantic Web tools for Bioinformatics
6th annual meeting of the Bioinformatics ITalian Society, Genova, 18-20 Apr. 2009
The building blocks of the Semantic Web

- **ontology:** OWL
- **Rules:** RIF
- **RDF-S**

**HOW to use OWL?**

**Editor**
- Protégé
- SWOOP

**Reasoner**
- Pellet
- Fact++
- ...

More info:
6th annual meeting of the Bioinformatics ITalian Sociaety, Genova, 18-20 Apr. 2009
Confused about OWL?

- You don’t need to use OWL unless you want to model what you know in formal terms!
- Most of the time you will see OWL, you will encounter only simple OWL expressions.

**OWL is not an Object Oriented language!!!**

- Classes and properties have different semantics in OO and OWL.
- OWL is based on an Open World Assumption, OO on a Closed World Assumption
The building blocks of the Semantic Web
Rules can specify additional semantics that cannot be represented in OWL.

Example:

\[ \text{hasFather}(x,y) \text{ and } \text{hasBrother}(y,z) \rightarrow \text{hasUncle}(x,z) \]

- The definition of rules varies in their syntax and in their expressivity.
- One common language to represent rules is SWRL.
- In general, unrestricted rules may result in undecidable OWL.
hasFather(x, y) and hasBrother(y, z) $\rightarrow$ hasUncle(x, z)

in SWRL:

```
<ruleml:imp>
  <ruleml:_rlab ruleml:href="#example1"/>
  <ruleml:_body>
    <swrlx:individualPropertyAtom
      swrlx:property="hasParent">
      <ruleml:var>x1</ruleml:var>
      <ruleml:var>x2</ruleml:var>
    </swrlx:individualPropertyAtom>
    <swrlx:individualPropertyAtom
      swrlx:property="hasBrother">
      <ruleml:var>x2</ruleml:var>
      <ruleml:var>x3</ruleml:var>
    </swrlx:individualPropertyAtom>
  </ruleml:_body>
  <ruleml:_head>
    <swrlx:individualPropertyAtom
      swrlx:property="hasUncle">
      <ruleml:var>x1</ruleml:var>
      <ruleml:var>x3</ruleml:var>
    </swrlx:individualPropertyAtom>
  </ruleml:_head>
</ruleml:imp>
```
The **building blocks** of the Semantic Web

- You can use inference rules through reasoners (Pellet), tools (Protégé) or libraries (Jena,...)

- [http://www.w3.org/Submission/SVRL/](http://www.w3.org/Submission/SVRL/)
- [http://www.ruleml.org/](http://www.ruleml.org/)
Outline

HowTo:
- export your data in semantic web formats,
- build a knowledge bases and query it.

The building blocks of the Semantic Web

Semantic Web application development
HowTo:

export your data in semantic web formats,
build a knowledge bases and query it.

The building blocks of the Semantic Web

Semantic Web application development

Building Semantic Web tools for Bioinformatics
6th annual meeting of the Bioinformatics ITalian Society, Genova, 18-20 Apr. 2009
HowTo: **export** your data in semantic web formats, **build** a knowledge bases and **query** it.
HowTo: **export** your data in semantic web formats, **build** a knowledge bases and **query** it.

How do I convert a text file in RDF?
How To: **export** your data in semantic web formats, **build** a knowledge bases and **query** it.

<table>
<thead>
<tr>
<th>CAS reg. Number</th>
<th>Chemical name</th>
<th>CHEBI ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>17804-35-2</td>
<td>Benomyl</td>
<td>3015</td>
</tr>
<tr>
<td>36734-19-7</td>
<td>Iprodione</td>
<td>28909</td>
</tr>
<tr>
<td>65277-42-1</td>
<td>Cis-Ketoconazole</td>
<td>47519</td>
</tr>
</tbody>
</table>

Let's see how this looks in semantic web format:

```
<http://www.purl.org/obo/owl/CHEBI#CHEBI_3015> <http://ex.org/hasChemName> "Benomyl".
<http://www.purl.org/obo/owl/CHEBI#CHEBI_3015> <http://ex.org/hasCASN> "17804-35-2".

<http://www.purl.org/obo/owl/CHEBI#CHEBI_28909> <http://ex.org/hasChemName> "Iprodione".
<http://www.purl.org/obo/owl/CHEBI#CHEBI_28909> <http://ex.org/hasCASN> "36734-19-7".

<http://www.purl.org/obo/owl/CHEBI#CHEBI_47519> <http://ex.org/hasChemName> "Cis-Ketoconazole".
<http://www.purl.org/obo/owl/CHEBI#CHEBI_47519> <http://ex.org/hasCASN> "65227-42-1".
```
HowTo: **export** your data in semantic web formats, **build** a knowledge bases and **query** it.

How do I convert a relational database in RDF?
HowTo: export your data in semantic web formats, build a knowledge bases and query it.

How do you **export** the content of a relational database **in RDF**?

• Build your own script (use a RDF library, **more later...**)
• Use a relational to RDF mapping tool. These tools (which varies in “strategy”) can be used to:
  • dump the content of a relational database in RDF
  • map queries addressing the RDF representation to the underlying SQL representation

We present briefly one of these tools: **D2RQ**

HowTo: **export** your data in semantic web formats, **build** a knowledge bases and **query** it.

### D2RQ

- It is a relational to RDF mapping tool that can be used to:
  - dump the content of a relational db in RDF
  - provide an RDF-query front-end to a relational database
  - provide a view on sql data for RDF libraries
- D2RQ analyzes the db structure to generate a candidate mapping file

```bash
./generate-mapping -u XXXX -p XXXX -d org.postgresql.Driver -o phibasemapping.n3 -b http://phi-base.org jdbc:postgresql://127.0.0.1/phibase
```

```bash
./dump-rdf -m phibasemapping.n3 -o phibaserdfv1.xml
```

- [http://www4.wiwiss.fu-berlin.de/bizer/d2rq/](http://www4.wiwiss.fu-berlin.de/bizer/d2rq/)
HowTo: **export** your data in semantic web formats, **build** a knowledge bases and **query** it.

Details from **D2RQ** mapping file

<table>
<thead>
<tr>
<th><strong>Author</strong></th>
<th><strong>Paper</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>ID (key)</td>
<td>ID (key)</td>
</tr>
<tr>
<td>Name</td>
<td>PubmedID</td>
</tr>
<tr>
<td></td>
<td>...</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Author2p</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>AuthorID (fkey)</td>
</tr>
<tr>
<td>PaperID (fkey)</td>
</tr>
<tr>
<td>...</td>
</tr>
</tbody>
</table>
HowTo: **export** your data in semantic web formats, **build** a knowledge bases and **query** it.

Details from **D2RQ** mapping file: default mapping

<table>
<thead>
<tr>
<th>Author</th>
<th>Paper</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID (key)</td>
<td>ID (key)</td>
</tr>
<tr>
<td>Name</td>
<td>PubmedID</td>
</tr>
<tr>
<td></td>
<td>...</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Author2p</th>
<th>PaperID (fkey)</th>
</tr>
</thead>
<tbody>
<tr>
<td>AuthorID (fkey)</td>
<td></td>
</tr>
<tr>
<td>PaperID (fkey)</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
</tbody>
</table>

```plaintext
# Table Author
map:author a d2rq:ClassMap;
   d2rq:dataStorage map:database;
   d2rq:uriPattern "interaction/@@Author.ID@@";
   d2rq:class ex:author;
   d2rq:classDefinitionLabel "author";
   .

map:interaction__label a d2rq:PropertyBridge;
   d2rq:belongsToClassMap map:author;
   d2rq:property rdfs:label;
   d2rq:pattern "interaction #@@Author.ID@@";
   .

map:interaction_interaction_id a d2rq:PropertyBridge;
   d2rq:belongsToClassMap map:author;
   d2rq:property ex:author_name;
   d2rq:propertyDefinitionLabel "Author_name";
   d2rq:column "Author.Name";
   d2rq:datatype xsd:string;
   .
```

Building **Semantic Web** tools for **Bioinformatics**

6th annual meeting of the Bioinformatics ITalian Sociaty, Genova, 18-20 Apr. 2009
HowTo: export your data in semantic web formats, build a knowledge bases and query it.

Details from **D2RQ** mapping file: default mapping

# Table Author
map:author a d2rq:ClassMap;
  d2rq:dataStorage map:database;
  d2rq:uriPattern "author#@Author.ID@";
  d2rq:class foaf:Person;
# d2rq:classDefinitionLabel "author";

  .

map:author__label a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:author;
  d2rq:property rdfs:label;
  d2rq:pattern "#@Author.Name@";

  .

map:author_name a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:author;
  d2rq:property foaf:name;
  d2rq:column "author.Name";
# d2rq:propertyDefinitionLabel "name";

.
HowTo: export your data in semantic web formats, build a knowledge bases and query it.

Details from **D2RQ** mapping file: default mapping

<table>
<thead>
<tr>
<th>Author</th>
<th>Paper</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID (key)</td>
<td>ID (key)</td>
</tr>
<tr>
<td>Name</td>
<td>PubmedID</td>
</tr>
</tbody>
</table>

```
# Table Paper
map:paper a d2rq:ClassMap;
  d2rq:dataStorage map:database;
  d2rq:uriPattern "info:pubmed/@@Paper.ID@@";
  d2rq:class ex:paper;
  d2rq:classDefinitionLabel "paper";
  d2rq:condition "PubmedID !=''";
map:paper__label a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:paper;
  d2rq:property rdfs:label;
  d2rq:pattern "paper: @@Paper.ID@@";
```

Building **Semantic Web** tools for **Bioinformatics**
6th annual meeting of the Bioinformatics ITalian Society, Genova, 18-20 Apr. 2009
HowTo: \textit{export} your data in semantic web formats, \textbf{build} a knowledge bases and \textit{query} it.

Details from \textbf{D2RQ} mapping file: default mapping

```
# Map author2paper
map:author2paper a d2rq:PropertyBridge;
d2rq:belongsToClassMap map:paper;
d2rq:property ex:hasAuthor;
d2rq:refersToClassMap map:author;
d2rq:join "Author2p.AuthorID = Author.ID";
```

- **Author**
  - ID (key)
  - Name

- **Paper**
  - ID (key)
  - PubmedID
  - ...

- **Author2p**
  - AuthorID (fkey)
  - PaperID (fkey)
  - ...

- [http://www4.wiwiss.fu-berlin.de/bizer/d2rq/spec](http://www4.wiwiss.fu-berlin.de/bizer/d2rq/spec)
HowTo: **export** your data in semantic web formats, **build** a knowledge bases and **query** it.

- “Default” mapping can be misleading.
- Understand what is a global identifier (URI) and what is not. URIs are stable and shared...
- Weigh pros and cons of tools: respect to tools’ maturity, to dump data in RDF is less critical than to query it.
HowTo: export your data in semantic web formats, build a knowledge bases and query it.

How do I build a knowledge base?
HowTo: **export** your data in semantic web formats, **build** a knowledge bases and **query** it.

How to build a Semantic Web based knowledge base

What:

- **Triple stores** Yes!
- **Ontology design** No...
- **Reasoning/Inference** A little, later...

*http://www4.wiwiss.fu-berlin.de/bizer/d2rq/*
HowTo: **export** your data in semantic web formats, **build** a knowledge bases and **query** it.

How to build a Semantic Web based knowledge base: **TripleStores**

- A triplestore is *like a db engine* for RDF
- It manages persistence (either through a relational database or not)
- It can provide some form of inference
- It can support for SPARQL queries

[http://www4.wiwiss.fu-berlin.de/bizer/d2rq/](http://www4.wiwiss.fu-berlin.de/bizer/d2rq/)
HowTo: **export** your data in semantic web formats, **build** a knowledge bases and **query** it.

---

**How to build a Semantic Web based knowledge base:** *TripleStores*

<table>
<thead>
<tr>
<th>TripleStore</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jena TDB</td>
<td>1.7B</td>
</tr>
<tr>
<td>Sesame</td>
<td></td>
</tr>
<tr>
<td>Virtuoso(os)</td>
<td>1+B</td>
</tr>
<tr>
<td>Garlik JXT</td>
<td>9.8B</td>
</tr>
<tr>
<td>Yars2</td>
<td>7B</td>
</tr>
<tr>
<td>BigOWLIM</td>
<td>3.36B</td>
</tr>
<tr>
<td>Mulgara</td>
<td>500M</td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
</tbody>
</table>

- [http://esw.w3.org/topic/LargeTripleStores](http://esw.w3.org/topic/LargeTripleStores)
HowTo: export your data in semantic web formats, build a knowledge bases and query it.

How to build a Semantic Web based knowledge base: **TripleStores**

- The use of triplestores is straightforward
- Load into it all RDF files and Ontologies that make your knowledge base
- By virtues of URIs the resulting knowledge base is already “connected”

• http://www4.wiwiss.fu-berlin.de/bizer/d2rq/
HowTo: export your data in semantic web formats, build a knowledge bases and query it.

How to build a Semantic Web based knowledge base: **TripleStores**

```sql
# NAME     : Get psoriasis proteins
# PARAMETER: psoriasis: the disease name
# FUNCTION : returns all the proteins that have 'psoriasis' in
# their Swiss-Prot disease description
# and their
# interacting proteins (if known)

BASE <http://www.semantic-systems-biology.org/>
PREFIX rdfs:<http://www.w3.org/2000/01/rdf-schema#>
PREFIX ssb:<http://www.semantic-systems-biology.org/SSB#>
SELECT distinct ?protein_name ?disease_description
?interacts_with ?encoded_by
WHERE {
  GRAPH <uniprot_sprot> {
    OPTIONAL {
    }
  }
  FILTER regex(?disease_description, 'psoriasis').
}
```

**Example SPARQL Query:**

```sql
http://www.semantic-systems-biology.org/biogateway/endpoint?
default-graph-uri=&query=%23+NAME+++++%3A+Get+psoriasis+proteins
%0D%0A%23+PARAMETER%3A+psoriasis%3A+the+disease+name
%0D%0A%23+FUNCTION+%3A+returns+all+the+proteins+that+have+%27psoriasis
%27+in%0D%0A%23+their+Swiss-Prot+disease+description
%0D%0A%27&format=text%2Fhtml&debug=on
```

*Building Semantic Web tools for Bioinformatics*

6th annual meeting of the Bioinformatics ITalian Societaey, Genova, 18-20 Apr. 2009
HowTo: export your data in semantic web formats, build a knowledge bases and query it.

How do I build a knowledge base?
HowTo: **export** your data in semantic web formats, 
**build** a knowledge bases and **query** it.

An introduction to **SPARQL**

Simple **Select Query**

```
<ex.org#pers1> <ex.org#name> “Marc”
<ex.org#pers1> <ex.org#age> “40”^^xsd:integer
<ex.org#pers2> <ex.org#name> “Mary”
<ex.org#pers2> <ex.org#age> “38”^^xsd:integer
<ex.org#pers3> <ex.org#sonOf> <ex.org#pers2>
<ex.org#pers3> <ex.org#age> “10”^^xsd:integer
<ex.org#pers3> <ex.org#sonOf> <ex.org#pers1>
<ex.org#pers3> <ex.org#sonOf> <ex.org#pers1>

SELECT ?n
WHERE ?x <ex.org#name> ?n .
“Marc”
“Mary”
“Tom”

SELECT ?x ?y
WHERE <ex.org#pers1> ?x ?y .
<ex.org#name> “Marc”
<ex.org#age> 40

SELECT ?y
WHERE { ?x <ex.org#sonOf> ?y .
    ?y <ex.org#name> “Marc” .
}
HowTo: **export** your data in semantic web formats, build a knowledge bases and **query** it.

An introduction to **SPARQL**

Simple **Select Query + Filters** and **Optional**

```
PREFIX ex: <ex.org#>
SELECT ?name ?parent
WHERE { ?x ex:name ?name .
    OPTIONAL { ?x ex:sonOf ?parent }
}

"Marc"
"Mary"
"Tom"
"Tom" "Marc"
"Tom" "Mary"
```
HowTo: export your data in semantic web formats, build a knowledge bases and query it.

An introduction to **SPARQL**

**Simple Construct Query**

```sparql
PREFIX ex: <ex.org#>
CONSTRUCT { ?p1 ex:older ?p2}
    FILTER (?a1 > ?a2) }
```

"Marc" ex:older "Mary"
"Marc" ex:older "Tom"
"Mary" ex:older "Tom"
HowTo: **export** your data in semantic web formats, **build** a knowledge bases and **query** it.

An introduction to **SPARQL** **Named Graphs**

```sparql
<ex.org#family1> {  
  <ex.org#pers1> <ex.org#name> "Marc"  
  ...  
  <ex.org#pers3> <ex.org#sonOf> <ex.org#pers2>  
}  

<ex.org#family2> {  
  <ex.org#pers2> <ex.org#name> "John"  
  ...  
}  
```

```sparql
PREFIX ex: <ex.org#>  
SELECT ?name  
FROM NAMED <ex.org#family1>  
FROM NAMED <ex.org#family2>  
WHERE { ?x ex:name ?name . }  
"Marc"  
...  
"John"  
...  
```

Building **Semantic Web** tools for **Bioinformatics**  
6th annual meeting of the Bioinformatics ITalian Society, Genova, 18-20 Apr. 2009
How To: export your data in semantic web formats, build a knowledge bases and query it.

Playing with **SPARQL**

- [http://www.w3.org/TR/rdf-sparql-query/](http://www.w3.org/TR/rdf-sparql-query/)
- [http://www4.wiwiss.fu-berlin.de/bizer/ng4j/](http://www4.wiwiss.fu-berlin.de/bizer/ng4j/)

Building Semantic Web tools for Bioinformatics
6th annual meeting of the Bioinformatics ITalian Sociaety, Genova, 18-20 Apr. 2009
Outline

The building blocks of the Semantic Web

HowTo:
export your data in semantic web formats,
built a knowledge bases and query it.

Semantic Web application development
The building blocks of the Semantic Web

HowTo:
- **export** your data in semantic web formats,
- **build** a knowledge bases and **query** it.
Programmatic access to resources and tools: Jena

The Jena Semantic Web library (Java) provides:

- An API to parse/access RDF and OWL files
- A persistence layer
- An hybrid reasoner
- A query server

Jena is only one of the tools available (at least to be cited: Sesame, OWLApi)

http://esw.w3.org/topic/SemanticWebTools
Parsing and Persistence

```java
Model myRDF = ModelFactory.createDefaultModel()
Model.read("file://my/file.rdf");

IDBConnection dbConnection = new DBConnection("DB_URL", "DB_USER", "DB_PASSWD", "M_DB");
ModelRDB myDBModel = ModelRDB.createModel(dbConnection, "name");
MyDBModel.read("file://my/file.rdf");

StoreDesc storeDesc =
    new StoreDesc(LayoutType.LayoutTripleNodesHash, DatabaseType.Derby);
JDBC.loadDriverDerby();
String jdbcURL = "jdbc:derby:DB/SDB2";
SDBConnection conn = new SDBConnection(jdbcURL, null, null);
Store store = SDBFactory.connectStore(conn, storeDesc);
// Store store = SDBFactory.connectStore("sdb.ttl");
Model model = SDBFactory.connectDefaultModel(store);

outFileStream = new FileOutputStream(new File(file));
write(outFileStream, "RDF/XML-ABBREV");
write(outFileStream, "N3");
```
Accessing Elements of RDF

```java
boolean containsStat =
    myRDFModel.contains(
        ModelFactory.createResource("http://example/ex1"),
        ModelFactory.createProperty("http://example/p1"),
        null);

StmIterator stats = myRDF.listStatements();

ResIterator resWithProp =
    myRDFModel.listResourcesWithProperty(
        ModelFactory.createProperty("http://example/p1"));
```

value of p1 for ex1

list all statements

all subjects that have some p1
Accessing Elements of OWL

```java
myOntModel=ModelFactory.createOntologyModel();
myOntModel.read("example.owl");

ExtendedIterator myOntModel.listIndividuals();
ExtendedIterator myOntModel.listSymmetricalProperties();
ExtendedIterator myOntModel.listUnionClasses();

StmtIterator stats =myRDF.listStatements();
```

- [http://jena.sourceforge.net/javadoc/](http://jena.sourceforge.net/javadoc/)
myModel=ModelFactory.createOntologyModel(OntModelSpec.OWL_MEM_TRANS_INF,myRDFModel);

myModel=ModelFactory.createOntologyModel(PelletReasonerFactory.THE_SPEC,myRDFModel);
myModel.prepare();


• http://jena.sourceforge.net/inference/
The end

HowTo:
export your data in semantic web formats,
build a knowledge bases and query it.

The building blocks of the Semantic Web

Semantic Web application development
Reading further...

- http://www.w3.org/2005/04/fresnel-info/
- http://esw.w3.org/topic/HCLS/Banff2007Demo

The building blocks of the Semantic Web

HowTo: export your data in semantic web formats,

Semantic Web application development

Building Semantic Web tools for Bioinformatics
6th annual meeting of the Bioinformatics ITalian Society, Genova, 18-20 Apr. 2009
Example 1: implementing non standard semantics (UnificationXrefs)

```
(?x owl:sameAs ?y) <-
(?x bp:unificationXrefs ?x1)
(?y bp:unificationXrefs ?y1)
(?x1 bp:ID ?i)(?y1 bp:ID ?i)
(?x1 bp:version ?v)(?y1 bp:version ?v)
```
Extra time

Example 2: enhancing readability (PeP)

Example 3: enhancing readability (PeP)

[Define-interaction:
(?interactor1 new:interacts ?interactor2) <-
(?x rdf:type bp:interaction)
(?x new:elementInteracting ?interactor1)
(?x new:elementInteracting ?interactor2)]
Extra time

Example 4: inference of causal relations

\[\text{influence}: (\forall x \exists y \text{ ak:influences } y) \iff (\exists p1 \text{ rdf:type bp:biochemicalReaction}) (\exists p2 \text{ rdf:type bp:biochemicalReaction}) (\exists p1 \text{ bp:RIGHT } k1) (\exists k1 \text{ bp:PHYSICAL-ENTITY } k) (\exists p2 \text{ bp:LEFT } k2) (\exists k2 \text{ bp:PHYSICAL-ENTITY } k) ((\exists p1 \text{ bp:LEFT } k3) (\exists k3 \text{ bp:PHYSICAL-ENTITY } x) (\exists p2 \text{ bp:RIGHT } k4) (\exists k4 \text{ bp:PHYSICAL-ENTITY } y) \]