BUILDING AN OBO FOUNDRY ONTOLOGY USING SEMANTIC WEB TECHNOLOGIES

Mélanie Courtot and Alan Ruttenberg
mcourtot@gmail.com
alanruttenberg@gmail.com

ICBO 2012
Goals of this section of tutorial

- Give a brief introduction to semantic web technologies for those who are unfamiliar with them
- Expose why these technologies are important for the OBO Foundry, and how we help the effort
- Detail specific practices used by Foundry developers
  - Registering a namespace
  - Authoring using Protege 4.2, saving as RDF/XML
  - Reusing other ontologies either by import or MIREOT
    - Initial imports: BFO, RO, ontology-metadata
  - ID/URI Policy (How to set up Protege to help)
  - Use of common metadata from ontology-metadata.owl
  - Use of existing relations, creating new (shortcut) relations
  - Use of axioms to constrain meaning of terms
  - Use of reasoner to check ontology
  - Having your ontology made accessible via Ontobee
The semantic web in a nutshell

**Adds to Web standards and practices (currently only for documents and services) encouraging**

- **Unambiguous names for things, classes, and relationships**
- **Well organized and documented in ontologies**
- With data expressed using uniform knowledge representation languages
  - Logic-based: RDF, OWL, RIF (a number of syntaxes available)
- To enable computationally assisted exploitation of information
  - Information for machines to work with, not only people
  - Allows consistency checking, precise query, inference
- That can be **easily integrated from different sources**
  - Both within and across public and organizational boundaries
How semantic web technologies help us achieve our goals (1)

The project of enabling effective communication and discovery in the biological and life sciences is complex.

Some pieces of this effort are handled by these technologies:

- We need tools for logical languages that have effective implementations. We get OWL, HerMIT, Pellet, FaCT++
- In contrast to RDBs, tools work on combined schema (ontology) and data.
  - Both are expressed logically
  - Both are queryable together at once
- We get a free data model: RDF without any more effort beyond building the ontology.
  - Saves us the trouble of designing a "data model" unless there is some good reason to do so.
  - Building "data models" is a common source of non-integrable data - everyone wants their own
Resource Description Framework

- **Resources (= nodes)**
  - Identified by Unique Resource Identifier (URI)

- **Properties (= edges)**
  - Identified by Unique Resource Identifier (URI)
  - Binary relations between 2 resources

Courtesy Natalia Villanueva- Rosales
One dataset

Source 1

<table>
<thead>
<tr>
<th>Owner</th>
<th>Pet</th>
</tr>
</thead>
<tbody>
<tr>
<td>Joe</td>
<td>Benji</td>
</tr>
<tr>
<td>Tim</td>
<td>Felix</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Joe hasPet Benji

Tim hasPet Felix

I can query about owners and the name of their pets.

Courtesy Natalia Villanueva- Rosales
Two datasets

**Source 1**

<table>
<thead>
<tr>
<th>Owner</th>
<th>Pet</th>
</tr>
</thead>
<tbody>
<tr>
<td>Joe</td>
<td>Benji</td>
</tr>
<tr>
<td>Tim</td>
<td>Felix</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

**Source 2**

<table>
<thead>
<tr>
<th>Pet</th>
<th>Species</th>
</tr>
</thead>
<tbody>
<tr>
<td>Benji</td>
<td>Dog</td>
</tr>
<tr>
<td>Felix</td>
<td>Cat</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

---

Courtesy Natalia Villanueva-Rosales
Data merge

Source 1

<table>
<thead>
<tr>
<th>Owner</th>
<th>Pet</th>
</tr>
</thead>
<tbody>
<tr>
<td>Joe</td>
<td>Benji</td>
</tr>
<tr>
<td>Tim</td>
<td>Felix</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Source 2

<table>
<thead>
<tr>
<th>Pet</th>
<th>Species</th>
</tr>
</thead>
<tbody>
<tr>
<td>Benji</td>
<td>Dog</td>
</tr>
<tr>
<td>Felix</td>
<td>Cat</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Same Identifier

Joe hasPet BENJI
Tim hasPet FELIX

Merge data! (no join needed)

Courtesy Natalia Villanueva- Rosales
I can query about owners of pets of a certain species, which is not in Source 1 or source 2, but in the union of both.
Now I can ask about possible customers of a new brand of pet meat: owners of carnivore pets.
RDF statement

• Triple subject, object, property
• SPARQL query by graph matching
PREFIX example: http://semanticscience.org/resource/ws_example#
SELECT ?name
WHERE {
  ?name rdf:type example:Person.
  ?name example:hasPet ?pet.
  ?pet rdf:type example:Carnivore.
}
Inference rule

SPARQL

PREFIX example: http://semanticscience.org/resource/ws_example#
SELECT ?name
WHERE {
  ?name rdf:type example:Person.
  ?name example:hasPet ?pet.
  ?pet rdf:type example:Carnivore.
}
How semantic web technologies help us achieve our goals (2)

• We get databases that support those data models
  o Triple stores: Virtuoso, Stardog, OWLIM, more...

• Tools are standardized and there is a growing workforce we can tap.
  o This scales better than making our own tools, training people

• Our goal of universal easy access is shared with semweb efforts. The web is good at supporting this.
  o We need tools for logical languages that have effective implementations. We get OWL, HerMIT, Pellet, FaCT++
  o By using these tools early we get to help guide their future
How we help the semweb effort

These standards and tools are still young. Ontology building is hard. Too many semantic web efforts suffer from poor ontology design and the consequences of that—failure to integrated, confused messages.

We help that by

• Providing and teach how to build high quality ontologies
• Provide a positive example of how a coordinated effort can develop good practices using the tools
• Provide feedback to standards efforts and tool developers that help improve successive versions
• Documenting our approaches so that others can benefit from them

The semantic web needs us!
Step by step: Building a foundry ontology for semantic web deployment
1. Register a namespace

If you plan to build an orthogonal ontology, you do this step once you are clear what your domain is and that id does not overlap another:

- Read http://obofoundry.org/id-policy.shtml
- Send mail to obo-discuss@obofoundry.org asking for your namespace (a short string of letters used as part of your identifier)
- Provide a description of your (planned) work, email for a point of contact, and location of your ontology
  - We recommend google code for setting up new projects
- Use the "New entities" preference in Protege to have protege create ids like:
  
  http://purl.obolibrary.org/obo/ XXX_0000000
2. Use Protégé to author your ontology

- Download the most recent version of protege - currently 4.2 beta
- After setting preferences create a new ontology
- Answer http://purl.obolibrary.org/obo/xxx.owl for the ontology IRI (xxx is lower case of your namespace)
- Save. When it asks format accept the default RDF/XML

If something doesn't seem to work in Protégé don't assume that you are doing something wrong. Ask for help on the discussion lists
3. ID/URI policy

- OBO Foundry aims at building orthogonal interoperating resources

- Reuse is successfully achieved by using the identifier from the source ontology in your own ontology.
- See http://obofoundry.org/id-policy.owl
The case for http: URIs

• A global namespace promotes generic tools
  • Query, inference, cross-reference, data integration

• URIs coordinate with web standards
  • Created with and for the Web
  • IETF and W3C recommended for naming
  • HTTP, HTML, RDF, OWL, SPARQL

• http: URIs are universally understood
  • Most people will know what to do with an http: URI

• http: URIs can identify anything
  • Not only a web page, but any kind of entity

• http: URIs are as reliable as anything else
  • Durability doesn’t depend on protocol
  • http: URIs are not tied to the HTTP protocol
OBO Foundry URIs

• Unique resource identifier: denotes each entity
• OBO Foundry ID policy: 
  [http://obofoundry.org/id-policy.shtml](http://obofoundry.org/id-policy.shtml)
• FOUNDRY_OBO_URI ::= "http://purl.obolibrary.org/obo/" IDSPACE "_" LOCALID
• Current ontology document:
  • Current OWL: http://purl.obolibrary.org/obo/IDSPACE.owl
  • Current OBO: http://purl.obolibrary.org/obo/IDSPACE.obo
• Versions of ontology:
  • OWL: http://purl.obolibrary.org/obo/<idspace>/YYYY-MM-DD/<idspace>.owl
  • OBO: http://purl.obolibrary.org/obo/<idspace>/YYYY-MM-DD/<idspace>.obo
Additional stable links

• Home page
  • http://purl.obolibrary.org/obo/IDSPACE

• Issue tracker
  • http://purl.obolibrary.org/obo/IDSPACE/tracker

• Ontology browsing
  • http://purl.obolibrary.org/obo/IDSPACE/browse

• Wiki
  • http://purl.obolibrary.org/obo/IDSPACE/wiki
URI dereferencing

Client

302
Location: http://sw.neurocommons.org/obiterm/OBI_0000225

303
Location: http://ashby.csail.mit.edu/cgi-bin/obiterm?ref=OBI_0000225

200
Content-type: application/xml

PURL Server

Neurocommons Server
Ontobee

Ontology for biomedical investigations

Keywords:  

Class: cell culture

- Term IRI: http://purl.obolibrary.org/obo/OBI_0100090
- definition: a cell culture is a material entity consisting of a population of cells that is maintained in vitro

Annotations

- definition editor: Bjorn Peters
- has curation status: ready for release
- example of usage: Jurkat cell line in RPMI w/ 10% FCS. PBMCs were purified from blood sample and put into tissue culture media. Purification of recombinant human growth hormone from CHO cell culture supernatant by Gradiflow preparative electrophoresis technology. Protein Expr Purif. 2003 Nov;32(1):126-34. PMID: 14680949
- definition source: GROUP: OBI Biomaterial Branch

Class Hierarchy

Thing
  + entity
    + continuannt
      + independent continual
    + material entity
      + processed material
    + device
      - polyacrylamide gel
      - xenograft
      - precipitate
    + amplified DNA
    + extract
    + cloning vector
HTML for human, RDF for machines
4. Reuse other ontologies

There are two ways of using other ontologies.

- Using the OWL import mechanism
  - We recommend importing
    - A version of BFO
      - BFO 1, stable: http://ifomis.org/bfo/1.1
      - BFO 2, under development: http://purl.obolibrary.org/obo/bfo.owl
    - A version of the relation ontology
      - For BFO 1, http://www.obofoundry.org/ro/ro.owl
      - For BFO 2, http://purl.obolibrary.org/obo/ro.owl
    - The ontology metadata annotation properties
      - http://purl.obolibrary.org/obo/ontology-metadata.owl
  - If you will try BFO 2, attend Wednesday's tutorial and engineer to be able to make changes as it evolves.

- Import selected terms only, using MIREOT
MIREOT

- Minimum Information to Reference an External ontology term
- ~ copy/paste of terms into your own ontology
- Terms in OBO Foundry ontologies stand on their own
- If their meaning changes, they are deprecated

=> *denotation* of individual terms remain stable
=> they can be seen as *individual units* of meaning
MIREOT in practice

**xxx.owl**
Main ontology file, imports external and externalDerived.owl files.

**IMPORTS**

**external.owl**
Contains minimal information about mapped classes

**externalDerived.owl**
Contains additional information about mapped classes
OntoFox: a Web Server for MIREOTing

• Based on the MIREOT principle
• Web-based data input and output
• Output OWL file can be directly imported in your ontology
• Easy to use
• No programming needed for users

http://ontofox.hegroup.org
MIREOT - extras

- Common metadata set for the OBO Foundry resources
- Developed within the Information Artifact Ontology (IAO)
  - IAO working session tonight
- MIREOT can be used to import anything you want
  - For example, we currently import superclasses (parents) from NCBITaxon

- Other ways to use MIREOT
  - Lisp scripts (OBI)
  - Protégé plugin (in progress, EBI)
5. Use of common metadata

By ontology metadata, we mean assertions about the ontology representational units (terms) rather than assertions about what they mean.

E.g. A label, who defined the term, who edited the term, what is an English definition of the term, etc.

Common metadata supports common tools such as MIREOT as well as browsing tools such as Ontobee. Protégé now supports numeric IDs for label display.
6. Using / sharing relations

- Relations ontology
  
  [link to Relations ontology](http://code.google.com/p/obo-relations/)
  
- BFO2 contains “core” relations
  
- RO will include biology specific relations
  
- RO includes *macro* or *shortcut* relations

**ObjectProperty: capable of**

- **Term IRI:** [http://purl.obolibrary.org/obo/RO_0002215](http://purl.obolibrary.org/obo/RO_0002215)
- **definition:** A relation between a material entity (such as a cell) and a process. This is a shortcut relation, translation rule for which is: capable_of P <-> bearer_of (some realized_by only P). Example: osteoclast capable of bone resorption.

**Annotations**

- definition editor: Chris Mungall
- alternative term: has function realized in
- has curation status: pending final vetting
- example of usage: osteoclast SubClassOf capable_of some 'bone resorption'
- expand expression to: BFO_0000053 some (BFO_0000054 only ?Y)
- definition source: PMID:21208450, PMID:20123131
7. Axioms constrain meaning of your terms

BFO and RO provide a taxonomy under which to put your terms. However, unless you add further axioms they won't be able to help you make a quality ontology.

Here are some examples that should consider adding:

- If every instance of an entity type E has certain types of parts P1, P2 make E subclass of (has part some P1) and (has part some P2)
- If you define a role type, say what kind of entity it inheres in and define and relate it to its realization process type. e.g. 'traffic guard role': 'inheres in' some 'homo sapiens' and 'is realized by only 'escorting person across road''

During reasoning these axioms interact with the ones from BFO and RO the reasoner will be able to detect some mistakes.
Reasoning - consistency checking

http://www.co-ode.org/ontologies/pizza/2007/02/12/
8. Have a deliberate release process

The goal of a deliberate release process is to prepare a version that is aimed at users (which might be other developers or biologists) rather than the developers of this ontology.

Typical activities

- Creation of dated release directory in repository
- Copy project files to release directory
- Update of MIREOT related files
- Merge in one release file for ease of use
- Run reasoner to add inferred axioms
- OWL and OBO format
- Quality checks
- Create dated and "latest" PURLs as stable URIs for your ontology
OORT – release tool

• An application to help manage ontologies and manage the ontology release process.
• Takes as input an *editors* version of an ontology and creates a number of *release* versions in both obo and owl format.
• These versions currently include:
  • The *main* version, with additional non-redundant links added via an OWL reasoner (i.e. *classified* in advance).
  • A *simple* version, which corresponds to the main version with any imported or "mireoted" classes removed
  • A version that has not had been reasoned over - the *non-classified* version
**BFO2**

- All OBO ontologies built under Basic Formal Ontology
- New version as draft release for ICBO
Pyramidal neuron relevant genes and processes

Inference required
### Results

| Gene       | Function                                                                 |
|------------|---------------------------------------------------------------------------|---|---|---|---|
| DRD1, 1812 | adenylate cyclase activation                                               |   |   |   |   |
| ADRB2, 154 | adenylate cyclase activation                                               |   |   |   |   |
| ADRB2, 154 | arrestin mediated desensitization of G-protein coupled receptor protein signaling pathway |
| DRD1, 1812 | dopamine receptor, adenylate cyclase activating pathway                   |   |   |   |   |
| DRD2, 1813 | dopamine receptor, adenylate cyclase inhibiting pathway                   |   |   |   |   |
| GRM7, 2917 | G-protein coupled receptor protein signaling pathway                      |   |   |   |   |
| GNG3, 2785 | G-protein coupled receptor protein signaling pathway                      |   |   |   |   |
| GNG12, 55970 | G-protein coupled receptor protein signaling pathway                       |   |   |   |   |
| DRD2, 1813 | G-protein coupled receptor protein signaling pathway                      |   |   |   |   |
| ADRB2, 154 | G-protein coupled receptor protein signaling pathway                      |   |   |   |   |
| CALM3, 808 | G-protein coupled receptor protein signaling pathway                      |   |   |   |   |
| HTR2A, 3356 | G-protein coupled receptor protein signaling pathway                      |   |   |   |   |
| DRD1, 1812 | G-protein signaling, coupled to cyclic nucleotide second messenger        |   |   |   |   |
| SSR5, 6755 | G-protein signaling, coupled to cyclic nucleotide second messenger        |   |   |   |   |
| MTNR1A, 4543 | G-protein signaling, coupled to cyclic nucleotide second messenger       |   |   |   |   |
| CNR2, 1269 | G-protein signaling, coupled to cyclic nucleotide second messenger        |   |   |   |   |
| HTR6, 3362 | G-protein signaling, coupled to cyclic nucleotide second messenger        |   |   |   |   |
| GRIK2, 2898 | glutamate signaling pathway                                               |   |   |   |   |
| GRIN1, 2902 | glutamate signaling pathway                                               |   |   |   |   |
| GRIN2A, 2903 | glutamate signaling pathway                                               |   |   |   |   |
| GRIN2B, 2904 | glutamate signaling pathway                                               |   |   |   |   |
| ADAM10, 102 | integrin-mediated signaling pathway                                       |   |   |   |   |
| GRM7, 2917 | negative regulation of adenylate cyclase activity                        |   |   |   |   |
| LRP1, 4035 | negative regulation of Wnt receptor signaling pathway                    |   |   |   |   |
| ADAM10, 102 | Notch receptor processing                                                |   |   |   |   |
| ASCL1, 429 | Notch signaling pathway                                                  |   |   |   |   |
| HTR2A, 3356 | serotonin receptor signaling pathway                                      |   |   |   |   |
| ADRB2, 154 | transmembrane receptor protein tyrosine kinase activation (dimerization) |   |   |   |   |
| PTPRG, 5793 | transmembrane receptor protein tyrosine kinase signaling pathway        |   |   |   |   |
| EPHA4, 2043 | transmembrane receptor protein tyrosine kinase signaling pathway        |   |   |   |   |
| NRTN, 4902 | transmembrane receptor protein tyrosine kinase signaling pathway        |   |   |   |   |
| CTNND1, 1500 | Wnt receptor signaling pathway                                           |   |   |   |   |

Many of the genes are indeed related to Alzheimer’s Disease through gamma secretase (presenilin) activity.
Links

- ID policy http://obofoundry.org/id-policy.shtml
- Obo-discuss mailing-list: https://lists.sourceforge.net/lists/listinfo/obo-discuss
- Ontology metadata: http://purl.obolibrary.org/obo/ontology-metadata.owl
- Ontobee: http://www.ontobee.org/
- OORT (release tool): http://code.google.com/p/owltools/wiki/Oort
- Neurocommons SPARQL endpoint
  - http://sparql.neurocommons.org/
  - http://sparql.obo.neurocommons.org/: OBO specific instance