EFO and applications

How we use ontologies in our applications

James Malone
malone@ebi.ac.uk
Gene Expression: Archive to Atlas

ArrayExpress

Manually curated submissions

direct submissions

GEO

Gene Expression Omnibus

Curation

Curation

Gene Expression Atlas

Subset re-curated with ontologies then analysed to produce summaries

BioPortal

efo

EMBL-EBI
Motivation: Use Cases in 2007

- Query support (e.g., query for 'cancer' and get also ‘leukemia’)
- Data visualisation – e.g., presenting an ontology tree to the user of what is in the database
- Data integration by ontology terms – e.g., 'kidney' in independent studies roughly means the same
- Intelligent template generation for different experiment types in submission or data presentation
- Summary level data
- Nonsense detection – e.g. telling us that something marked as cancer can not be marked as healthy
Lots of stuff, lots of types of stuff

- Use cases remain same but problem size keeps growing

<table>
<thead>
<tr>
<th></th>
<th>Annotations</th>
<th>Archive</th>
<th>Atlas</th>
</tr>
</thead>
<tbody>
<tr>
<td>Species</td>
<td>~1821</td>
<td></td>
<td>73</td>
</tr>
<tr>
<td>Samples</td>
<td>1,126,457</td>
<td>98,631</td>
<td></td>
</tr>
<tr>
<td>Annotations on samples</td>
<td>7,672,825</td>
<td>463,797</td>
<td></td>
</tr>
<tr>
<td>Unique sample annotations</td>
<td>223,650</td>
<td></td>
<td>22,339</td>
</tr>
<tr>
<td>Assays (Hybridizations)</td>
<td>965,638</td>
<td>100,654</td>
<td></td>
</tr>
<tr>
<td>Annotations on assays</td>
<td>3,248,298</td>
<td>265,971</td>
<td></td>
</tr>
<tr>
<td>Unique assay annotations</td>
<td>189,381</td>
<td></td>
<td>19,136</td>
</tr>
</tbody>
</table>
Also lots of ontology stuff

- There are hundreds of ontologies
- Which do we pick, how do we present these?
Primary needs are:

- An ontology that serves our use cases
- Reuses existing resources where appropriate
- Has some level of automated update
- Does not involve users having to familiarise with many ontologies and interfaces
- Find sweet spot between building it efficiently and most accurate implementation
EFO’s position in the world

- EFO is an application ontology, built for use in applications
- This brings with it implications that don’t necessarily apply to all ontologies
Implications of an Application Ontology

• Applications rely on EFO to work properly so they can work properly

• To do this we treat EFO like a software eng project and use continuous integration methods:
  • Changes are integrated to revision system frequently
  • Automated build process on commit (Bamboo)
  • Self-testing – on commit, code is tested, every time
  • Predictable release cycles

• In addition we use agile methods for weekly prioritisation of tickets
Continuous Integration – Managing Edits

URiGEN the URI generation service

Latest URI's...

<table>
<thead>
<tr>
<th>URI</th>
<th>Label</th>
<th>Ontology</th>
<th>User</th>
<th>Creation time</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="http://www.ebi.ac.uk/efo/EFO_0004909">http://www.ebi.ac.uk/efo/EFO_0004909</a></td>
<td>test2</td>
<td>EFO</td>
<td>helen</td>
<td>2012-10-29 17:44:59</td>
</tr>
<tr>
<td><a href="http://www.ebi.ac.uk/efo/EFO_0004908">http://www.ebi.ac.uk/efo/EFO_0004908</a></td>
<td>test</td>
<td>EFO</td>
<td>helen</td>
<td>2012-10-29 17:42:40</td>
</tr>
<tr>
<td><a href="http://www.ebi.ac.uk/efo/EFO_0004907">http://www.ebi.ac.uk/efo/EFO_0004907</a></td>
<td>laboratory</td>
<td>EFO</td>
<td>helen</td>
<td>2012-10-26 13:39:35</td>
</tr>
<tr>
<td><a href="http://www.ebi.ac.uk/efo/EFO_0004906">http://www.ebi.ac.uk/efo/EFO_0004906</a></td>
<td>metastasis to lymph node</td>
<td>EFO</td>
<td>helen</td>
<td>2012-10-22 16:51:56</td>
</tr>
<tr>
<td><a href="http://www.ebi.ac.uk/efo/EFO_0004905">http://www.ebi.ac.uk/efo/EFO_0004905</a></td>
<td>induced pluripotent stem cell</td>
<td>EFO</td>
<td>james</td>
<td>2012-10-22 13:51:56</td>
</tr>
<tr>
<td><a href="http://www.ebi.ac.uk/efo/EFO_0004904">http://www.ebi.ac.uk/efo/EFO_0004904</a></td>
<td>hereditary genetic disorder</td>
<td>EFO</td>
<td>james</td>
<td>2012-10-22 10:07:02</td>
</tr>
<tr>
<td><a href="http://www.ebi.ac.uk/efo/EFO_0004903">http://www.ebi.ac.uk/efo/EFO_0004903</a></td>
<td>array design element</td>
<td>EFO</td>
<td>james</td>
<td>2012-10-15 16:31:18</td>
</tr>
<tr>
<td><a href="http://www.ebi.ac.uk/efo/EFO_0004902">http://www.ebi.ac.uk/efo/EFO_0004902</a></td>
<td>European HapMap cell line</td>
<td>EFO</td>
<td>james</td>
<td>2012-09-03 16:56:44</td>
</tr>
<tr>
<td><a href="http://www.ebi.ac.uk/efo/EFO_0004901">http://www.ebi.ac.uk/efo/EFO_0004901</a></td>
<td>Northern and Western European</td>
<td>EFO</td>
<td>james</td>
<td>2012-09-03 16:49:27</td>
</tr>
<tr>
<td><a href="http://www.ebi.ac.uk/efo/EFO_0004900">http://www.ebi.ac.uk/efo/EFO_0004900</a></td>
<td>Yoruba</td>
<td>EFO</td>
<td>james</td>
<td>2012-09-03 16:41:43</td>
</tr>
</tbody>
</table>

Logged in as: james [log ou]
Show my API key
Standard designs

• Use ontology design patterns for a lot of axiomatisation
• We try and agree these with external ontologies
• Minimum set of metadata for each class

  Cell line x is
  i. derived from a cell type
  ii. which is part of some organism part and
  iii. which is part of some organism
Continuous Integration - Testing

- We need a consistent API – read here class URIs, annotation properties, etc.
- Every commit triggers sets of unit tests looking for:
  - URI deletion
  - URI is in allowed form (i.e. namespaces permitted)
  - Label duplication
  - Synonym to label duplication
  - White spaces in labels
  - Inconsistencies (through reasoning)
Testing in Bamboo build

Plan Summary

Current Activity
No builds are currently running.

Recent History

<table>
<thead>
<tr>
<th>PR</th>
<th>Updated by</th>
<th>Date</th>
<th>Build Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>#34</td>
<td>James Malone</td>
<td>6 days ago</td>
<td>Testless build</td>
</tr>
<tr>
<td>#33</td>
<td>James Malone</td>
<td>6 days ago</td>
<td>Testless build</td>
</tr>
<tr>
<td>#32</td>
<td>James Malone</td>
<td>1 week ago</td>
<td>Testless build</td>
</tr>
<tr>
<td>#31</td>
<td>James Malone</td>
<td>1 week ago</td>
<td>Testless build</td>
</tr>
<tr>
<td>#30</td>
<td>James Malone</td>
<td>1 week ago</td>
<td>Testless build</td>
</tr>
<tr>
<td>#29</td>
<td>Dani Welter</td>
<td>3 weeks ago</td>
<td>Testless build</td>
</tr>
<tr>
<td>#28</td>
<td>Dani Welter</td>
<td>3 weeks ago</td>
<td>Testless build</td>
</tr>
<tr>
<td>#27</td>
<td>James Malone</td>
<td>1 month ago</td>
<td>Testless build</td>
</tr>
<tr>
<td>#26</td>
<td>James Malone</td>
<td>1 month ago</td>
<td>Testless build</td>
</tr>
</tbody>
</table>

Plan Statistics

- 25 builds
- 72% successful
- 30s average duration
Testing in Bamboo build

Logs

The following logs have been generated by the Jobs in this Plan.

<table>
<thead>
<tr>
<th>Job</th>
<th>Logs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Check EFO Default Stage</td>
<td></td>
</tr>
<tr>
<td>20-Sep-2012 17:21:56</td>
<td># Report of validation of EFO generated by efovalidator</td>
</tr>
<tr>
<td>20-Sep-2012 17:21:56</td>
<td># log file : /ebi/microarray/home/fgpt/sw/efovalidator/validation.log</td>
</tr>
<tr>
<td>20-Sep-2012 17:21:56</td>
<td># EFO input : efo_release_candidate.owl</td>
</tr>
<tr>
<td>20-Sep-2012 17:21:56</td>
<td># generated : 20 September 2012 @ 05:21:56 PM</td>
</tr>
<tr>
<td>20-Sep-2012 17:21:56</td>
<td># For more information, see:</td>
</tr>
<tr>
<td>20-Sep-2012 17:21:56</td>
<td># <a href="http://www.ebi.ac.uk/fgpt/sw/efoimporter/index.html">http://www.ebi.ac.uk/fgpt/sw/efoimporter/index.html</a></td>
</tr>
<tr>
<td>20-Sep-2012 17:21:56</td>
<td>#</td>
</tr>
<tr>
<td>20-Sep-2012 17:22:01</td>
<td># Summary</td>
</tr>
<tr>
<td>20-Sep-2012 17:22:01</td>
<td># Critical errors</td>
</tr>
<tr>
<td>20-Sep-2012 17:22:01</td>
<td># 1) Check input is in valid OWL format</td>
</tr>
<tr>
<td>20-Sep-2012 17:22:01</td>
<td># 2) Check version number</td>
</tr>
<tr>
<td>20-Sep-2012 17:22:01</td>
<td># 3) Check date</td>
</tr>
<tr>
<td>20-Sep-2012 17:22:01</td>
<td># 4) Check class IDs are URIs formed from a valid namespace name</td>
</tr>
<tr>
<td>20-Sep-2012 17:22:01</td>
<td># 5) Check EFO IDs used in class IRIs are valid</td>
</tr>
<tr>
<td>20-Sep-2012 17:22:01</td>
<td># 6) Check every class has a single, unique label</td>
</tr>
</tbody>
</table>

Powered by a free Atlassian Bamboo open source license for EBI. Try Bamboo - The Zen of Continuous Integration.
Continuous Integration - Diffs

Number of classes that have changed: 120
Number of classes that have been added: 258
Number of classes that have been deleted: 111

Perform another diff Export results as XML Export results as text

Classes modified:

Class: http://purl.obolibrary.org/obo/GO_0007608
Label: sensory perception of smell
- SubClassOf(sensory perception of smell 'sensation')
+ SubClassOf(sensory perception of smell http://purl.obolibrary.org/obo/GO_0007600)

Class: http://www.ebi.ac.uk/efo/EFO_0004525
Label: bitter taste sensitivity
- SubClassOf(bitter taste sensitivity 'phenotype')
+ SubClassOf(bitter taste sensitivity http://purl.obolibrary.org/obo/GO_0050909)

Class: http://www.ebi.ac.uk/efo/EFO_0004356
Label: taste
- SubClassOf(taste 'sensation')
Repeatable release cycles
EFO Web [www.ebi.ac.uk/efo](http://www.ebi.ac.uk/efo)

Representing experimental variables with EFO

The **Experimental Factor Ontology** (EFO) provides a systematic description of experimental variables found in external projects such as the NHGRI GWAS catalogue. It combines parts of several existing ontologies related to chemicals, compounds. The scope of EFO is to support the annotation, analysis and visualization of experimental data. It also adds terms for external users when requested. If you are new to ontologies see a blog post by James Malone on what ontologies are for.[1]

**Browse**

Browse EFO at [NCBO BioPortal](http://www.bioportal.org) or in EBI's [OLS](http://www.ebi.ac.uk/ols). You can also search EFO using the search box, above.

**FAQ**

Read more about EFO or see the [Frequently Asked Questions](http://www.ebi.ac.uk/efo/faq). You can also read about ontologies in James Malone’s [blog](http://www.ebi.ac.uk/efo/blog).

**Submit**

Submit new terms or report bugs using our [JIRA ticket system](http://www.ebi.ac.uk/efo/jira). You can also join the [mailing list](http://www.ebi.ac.uk/efo/mail).
Which external resources to use?

- OBO Foundry has attempted to herd the cattle
- But we don’t restrict to only Foundry
- There are some clear winners, e.g. GO, ChEBI as they are widely used in science
- Objective analysis is hard but some things we check:
  1. Does it provide coverage?
  2. Is it actively developed?
  3. Does it have textual definitions?
  4. Do other data providers use it?
- We’ve studied 1, 2 & 3, 4 is more anecdotal
Activity Study PMID:22554701

- An example of why this is hard from human disease ontology.
- Lots of activity, yes, but lots of deletions
- Question: is this a stable resource?
Activity PMID:22554701

- Compare this to Gene Ontology with lots of edits but no deletions
Importing still needs human hand

- We need fuzziness which is hard to do programmatically

<table>
<thead>
<tr>
<th>Input from submission</th>
<th>Ontology class</th>
</tr>
</thead>
<tbody>
<tr>
<td>2’-deoxy-5-azacytidine</td>
<td>5-aza-2’-deoxycytidine</td>
</tr>
<tr>
<td>Ovarian Cancer</td>
<td>ovarian carcinoma</td>
</tr>
<tr>
<td>Anterior tibialis</td>
<td>tibialis anterios</td>
</tr>
<tr>
<td>Barret’s Esophagus</td>
<td>Barrett’s esophagus</td>
</tr>
<tr>
<td>Endothelium, Vascula</td>
<td>cardiovascular system endothelium</td>
</tr>
</tbody>
</table>
Adding synonyms helps enrich searching
Axiomatise as required by queries

- Find diff expressed genes for **cancer cell lines**
- Use a bit of OWL
Upper ontologies

• We use a very slimmed down version of BFO
• Will we ever need more? A proper cost-benefit analysis might be worth performing.
• Either way we need to hide upper ontologies:
  • Users should not start to type ‘gene’ and see ‘generically dependent continuant’
  • Upper level classes should not appear in visualisations
  • Community specific labels (for functional genomics) should be chosen over others for applications
Views still need work

- View technology is not well supported for ontologies
- Often ignored – we face problems reconciling the view and the ontology
- e.g. partonomy vs hierarchy
Summary

- EFO takes existing ontologies, glues them together
- Adds values with additional axiomatisation – but only to satisfy our questions (we are not modelling the world)
- Adds synonyms for search
- Adds QC with continuous integration
- Rely on external resources but is decoupled enough from potentially show-stopping changes
Summary

• People that use EFO tell me ‘we like it because it looks like our data’

• Formal ontology structures are not a panacea – do not meet all use cases

• But reusing is important for integration

• Sometimes the good stuff can be hidden by how its glued together – do not underestimate this in applications
Acknowledgements

- Those contributing to EFO are entire FG Production team, especially editors: Helen Parkinson, Tomasz Adamuisak, Ele Holloway, Eleanor Williams, Jon Ison, Dani Welter, Natalja Kurbatova, Simon Jupp. Also Tony Burdett on process work
- Gene Expression Atlas team [www.ebi.ac.uk/gxa](http://www.ebi.ac.uk/gxa)
- ArrayExpress team [www.ebi.ac.uk/arrayexpress](http://www.ebi.ac.uk/arrayexpress)
- External users and groups