**Updating the imports**

**\*\*\* Note: the following changes should be made on the ReO google code site. \*\*\***

**Some preliminary steps:**

1. Be sure to update the google code directory to its latest version in SVN.
2. Under /src/tools/ you will find the python scripts needed for generating the import files.
3. Make sure you have set the permissions on these files to execute (chmod 755 \*.py)
4. Make sure that you have pycurl installed
   1. For Mac users with OSX Snow Leopard and up: you’ll need to have the DevTools installed first.
      1. They can be found here: <http://developer.apple.com/xcode/>
      2. Once unpacked, you’ll need to run the following command to install pycurl:

sudo env ARCHFLAGS="-arch x86\_64" easy\_install setuptools pycurl==7.19.0

* 1. From a Windows machine, Cygwin did not have the right version of curl or have a pycurl module/package, so I will have to do this from the cmd terminal. Make sure you have python and pycurl installed, and make sure python is set in your path.

1. Assuming you are in your local folder [/trunk/tools/]

You can run the following commands: (eventually, a batch script that parses all the import files and automatically generate .owl files will be generated, but for now the process is manual)

(From a Windows machine, run this from Cygwin terminal, or run from cmd terminal . . . run from trunk/tools/ as indicated above)

* CARO2 is only import files currently created by hand because not in ontofox

python ontofoxInputFilePropGenerator\_v2.py ../src/ontology/imports/bfo2\_relations\_import.owl ontofox/ontofox\_input\_files/ontofox\_bfo2\_relations\_input.txt BFO

python ontofoxInputFilePropGenerator\_v2.py ../src/ontology/imports/caro\_import.owl ontofox/ontofox\_input\_files/ontofox\_caro\_input.txt CARO

python ontofoxInputFilePropGenerator\_v2.py ../src/ontology/imports/chebi\_import.owl ontofox/ontofox\_input\_files/ontofox\_chebi\_input.txt CHEBI

python ontofoxInputFilePropGenerator\_v2.py ../src/ontology/imports/cl\_import.owl ontofox/ontofox\_input\_files/ontofox\_cl\_input.txt CL

python ontofoxInputFilePropGenerator\_v2.py ../src/ontology/imports/clo\_import.owl ontofox/ontofox\_input\_files/ontofox\_clo\_input.txt CLO

python ontofoxInputFilePropGenerator\_v2.py ../src/ontology/imports/doid\_import.owl ontofox/ontofox\_input\_files/ontofox\_doid\_input.txt DOID

python ontofoxInputFilePropGenerator\_v2.py ../src/ontology/imports/ero\_import.owl ontofox/ontofox\_input\_files/ontofox\_ero\_input.txt ERO

python ontofoxInputFilePropGenerator\_v2.py ../src/ontology/imports/go\_import.owl ontofox/ontofox\_input\_files/ontofox\_go\_input.txt GO

python ontofoxInputFilePropGenerator\_v2.py ../src/ontology/imports/iao\_main\_import.owl ontofox/ontofox\_input\_files/ontofox\_iao\_main\_input.txt IAO

python ontofoxInputFilePropGenerator\_v2.py ../src/ontology/imports/mod\_import.owl ontofox/ontofox\_input\_files/ontofox\_mod\_input.txt MOD

python ontofoxInputFilePropGenerator\_v2.py ../src/ontology/imports/mpath\_import.owl ontofox/ontofox\_input\_files/ontofox\_mpath\_input.txt MPATH

python ontofoxInputFilePropGenerator\_v2.py ../src/ontology/imports/ncbi\_taxon\_import.owl ontofox/ontofox\_input\_files/ontofox\_ncbi\_taxon\_input.txt NCBITaxon

python ontofoxInputFilePropGenerator\_v2.py ../src/ontology/imports/obi\_import.owl ontofox/ontofox\_input\_files/ontofox\_obi\_input.txt OBI

python ontofoxInputFilePropGenerator\_v2.py ../src/ontology/imports/ogms\_import.owl ontofox/ontofox\_input\_files/ontofox\_ogms\_input.txt OGMS

python ontofoxInputFilePropGenerator\_v2.py ../src/ontology/imports/pato\_import.owl ontofox/ontofox\_input\_files/ontofox\_pato\_input.txt PATO

python ontofoxInputFilePropGenerator\_v2.py ../src/ontology/imports/pro\_import.owl ontofox/ontofox\_input\_files/ontofox\_pro\_input.txt PR

python ontofoxInputFilePropGenerator\_v2.py ../src/ontology/imports/ro\_import.owl ontofox/ontofox\_input\_files/ontofox\_ro\_input.txt RO

python ontofoxInputFilePropGenerator\_v2.py ../src/ontology/imports/so\_import.owl ontofox/ontofox\_input\_files/ontofox\_so\_input.txt SO

python ontofoxInputFilePropGenerator\_v2.py ../src/ontology/imports/uberon\_import.owl ontofox/ontofox\_input\_files/ontofox\_uberon\_input.txt UBERON

python ontofoxInputFilePropGenerator\_v2.py ../src/ontology/imports/uo\_import.owl ontofox/ontofox\_input\_files/ontofox\_uo\_input.txt UO

--------

For getting initial iao import set up:

Command to create original iao import input file pulling from classes in iao-main.owl:

python ontofoxInputFilePropGenerator\_v2.py ../src/ontology/imports/iao/iao-main.owl ontofox/ontofox\_input\_files/ontofox\_iao\_main\_input.txt IAO

Subsequently, once I rearrange the imports appropriately so that iao is not a full import but an external>iao\_main\_imports.owl file, I can target this script to this import owl file as follows to generate all subsequent ontofox input txt files for iao\_main:

python ontofoxInputFilePropGenerator\_v2.py ../src/ontology/external/iao\_main\_import.owl ontofox/ontofox\_input\_files/ontofox\_iao\_main\_input.txt IAO

1. From the same directory as above, run ontofoxCurl (python script) to generate import .owl files:

(From a Windows machine, Cygwin did not have the right version of curl or have a pycurl module/package, so I will have to do this from the cmd terminal. Make sure you have python and pycurl installed, and make sure python is set in your path. Open a cmd terminal, and navigate to the tools directory in the cmd terminal, and run the commands below)

python ontofoxCurl.py ontofox/ontofox\_input\_files/ontofox\_bfo2\_relations\_input.txt ontofox/ontofox\_generated\_imports/ontofox\_bfo2\_relations\_import.owl

python ontofoxCurl.py ontofox/ontofox\_input\_files/ontofox\_caro\_input.txt ontofox/ontofox\_generated\_imports/ontofox\_caro\_import.owl

python ontofoxCurl.py ontofox/ontofox\_input\_files/ontofox\_chebi\_input.txt ontofox/ontofox\_generated\_imports/ontofox\_chebi\_import.owl

python ontofoxCurl.py ontofox/ontofox\_input\_files/ontofox\_cl\_input.txt ontofox/ontofox\_generated\_imports/ontofox\_cl\_import.owl

python ontofoxCurl.py ontofox/ontofox\_input\_files/ontofox\_clo\_input.txt ontofox/ontofox\_generated\_imports/ontofox\_clo\_import.owl

python ontofoxCurl.py ontofox/ontofox\_input\_files/ontofox\_doid\_input.txt ontofox/ontofox\_generated\_imports/ontofox\_doid\_import.owl

python ontofoxCurl.py ontofox/ontofox\_input\_files/ontofox\_ero\_input.txt ontofox/ontofox\_generated\_imports/ontofox\_ero\_import.owl

python ontofoxCurl.py ontofox/ontofox\_input\_files/ontofox\_go\_input.txt ontofox/ontofox\_generated\_imports/ontofox\_go\_import.owl

python ontofoxCurl.py ontofox/ontofox\_input\_files/ontofox\_iao\_main\_input.txt ontofox/ontofox\_generated\_imports/ontofox\_iao\_main\_import.owl

python ontofoxCurl.py ontofox/ontofox\_input\_files/ontofox\_mpath\_input.txt ontofox/ontofox\_generated\_imports/ontofox\_mpath\_import.owl

python ontofoxCurl.py ontofox/ontofox\_input\_files/ontofox\_mod\_input.txt ontofox/ontofox\_generated\_imports/ontofox\_mod\_import.owl

python ontofoxCurl.py ontofox/ontofox\_input\_files/ontofox\_ncbi\_taxon\_input.txt ontofox/ontofox\_generated\_imports/ontofox\_ncbi\_taxon\_import.owl

python ontofoxCurl.py ontofox/ontofox\_input\_files/ontofox\_obi\_input.txt ontofox/ontofox\_generated\_imports/ontofox\_obi\_import.owl

python ontofoxCurl.py ontofox/ontofox\_input\_files/ontofox\_ogms\_input.txt ontofox/ontofox\_generated\_imports/ontofox\_ogms\_import.owl

python ontofoxCurl.py ontofox/ontofox\_input\_files/ontofox\_pato\_input.txt ontofox/ontofox\_generated\_imports/ontofox\_pato\_import.owl

python ontofoxCurl.py ontofox/ontofox\_input\_files/ontofox\_pro\_input.txt ontofox/ontofox\_generated\_imports/ontofox\_pro\_import.owl

python ontofoxCurl.py ontofox/ontofox\_input\_files/ontofox\_ro\_input.txt ontofox/ontofox\_generated\_imports/ontofox\_ro\_import.owl

python ontofoxCurl.py ontofox/ontofox\_input\_files/ontofox\_so\_input.txt ontofox/ontofox\_generated\_imports/ontofox\_so\_import.owl

python ontofoxCurl.py ontofox/ontofox\_input\_files/ontofox\_uberon\_input.txt ontofox/ontofox\_generated\_imports/ontofox\_uberon\_import.owl

python ontofoxCurl.py ontofox/ontofox\_input\_files/ontofox\_uo\_input.txt ontofox/ontofox\_generated\_imports/ontofox\_uo\_import.owl

1. Once the ontofox import files have been generated, they should be checked for terms that no longer exist in the source ontology. This can be done by opening each imports file with a text editor and scrolling to the commented out section at the very end of the file. Look for terms in the section titled - “The following terms do not have any related axioms in the source ontology:”. If the terms listed in this section **are from the ontology for which the file is named**, take note. These terms have been removed from their source ontology and their usage by eagle-i data must be checked in the repo via SPARQL. Only when these missing terms have been checked and dealt with satisfactorily, should you proceed to the next step.

\*\*\*\*\*\*FROM HERE ON IS NOT YET ReO READY\*\*\*\*\*\*

1. **If** terms have been removed, it is necessary clean out their annotations from the correct \*-imports-app.owl file. This can be achieved by searching for the removed term URIs in the correct text file (the app file corresponding to the imported ontology from which the term has been removed). Save the edited text file, check the changes in SVN, and recommit.
2. The Java class ImportsUpdater compares the new ontofox-generated files with the previous import files. The program reports classes that are not present and merges the annotations of the new file into the old one (when used in “write” mode).

The program is housed in /src/tools/Release Tools/ and is called ImportsUpdater.jar.

The usage is as follows:

ImportsUpdater original\_imports\_file ontofox\_imports\_file output\_file [true/false]

The true/false option chooses write and report modes. When “true” is selected, the program generates a merged copy of the two target .owl files. When “false” is selected, the program simply generates a report that characterizes the differences between the classes and annotations in both files. The program generates two text files with the changes in classes and annotations.

\*\*\* Note: the OBI imports are the only set for which a merge is currently required. Special edits have been made to OBI annotations for use in eagle-i and the files must be merged to carry these edits forward. Reports should be generated and checked for each set of imports, however. \*\*\*

The code can be called from the same directory as above using the following command.

java -jar ImportsUpdater.jar ../../ontology/imports/obi-imports.owl ../ontofox-files/ontofox-output-files/OBI-imports-new.owl ../ontofox-files/ontofox-output-files/merged\_obi-imports.owl true

1. Carefully swap **all** contents (**with the exception of the header from the original file – leave this alone**) of the old imports file for those of the new one using a text editor. Before committing, be sure to refresh the SVN client and check the changes line by line for irregularities.
2. When everything is perfect, commit the changes and watch the progress of the build here: <http://infra.search.eagle-i.net:8085/browse/DATAMODEL>

Keep your fingers crossed that the acceptance tests pass.