Gene Database Testing Report - Heavy Metal HaterZ

From LMU BioDB 2015

Contents

Shewanella oneidensis

Our Gene Database Testing Report

Group Members

- Coder: Mary Alverson
- GenMAPP User & Project Manager: Ron Legaspi
- Quality Assurance: Josh Kuroda
- GenMAPP User: Emily Simso

Important Links

Our Files

Our Deliverables

Gene Database Project Links							
Overview	**Deliverables**	Reference Format (https://peerj.com/about/author-instructions/#reference-format)	**Guilds**	Project Manager	GenMAPP User	Quality Assurance	Coder
			Teams	Heavy Metal HaterZ	The Class Whoopers	GÉNialOMICS	Oregon Trail Survivors

Individual Journal Entries				
Mary Alverson	**Week 11**	**Week 12**	**Week 14**	**Week 15**
Emily Simso	**Week 11**	**Week 12**	**Week 14**	**Week 15**
Ron Legaspi	**Week 11**	**Week 12**	**Week 14**	**Week 15**
Josh Kuroda	**Week 11**	**Week 12**	**Week 14**	**Week 15**

Export Information

Version of GenMAPP Builder: **3 build 5**

Computer on which export was run: **HP Compaq 8300 Elite SFF FC**

Postgres Database name: **S. Oneidensis**

UniProt XML filename: **SOneidensisUNIPROT**

- UniProt XML version: **UniProt release 2015_10 - October 14, 2015**
- UniProt XML download link (http://www.uniprot.org/uniprot/?query=taxonomy:211586)
- Time taken to import: **3.18 minutes**
    - Note: *n/a*

GO OBO-XML filename: **go daily-termdb.obo-xml**

- GO OBO-XML version (The version information can be found in the file properties after the file downloaded from the GO Download page (http://beta.geneontology.org/page/download-ontology) has been unzipped):
- GO OBO-XML download link (http://geneontology.org/page/download-ontology#Legacy_Downloads)
- Time taken to import: **7.16 minutes**
- Time taken to process: **4.27 minutes**

  - Note: *n/a*

GOA filename: **ShewanellaOneidensisGOA**

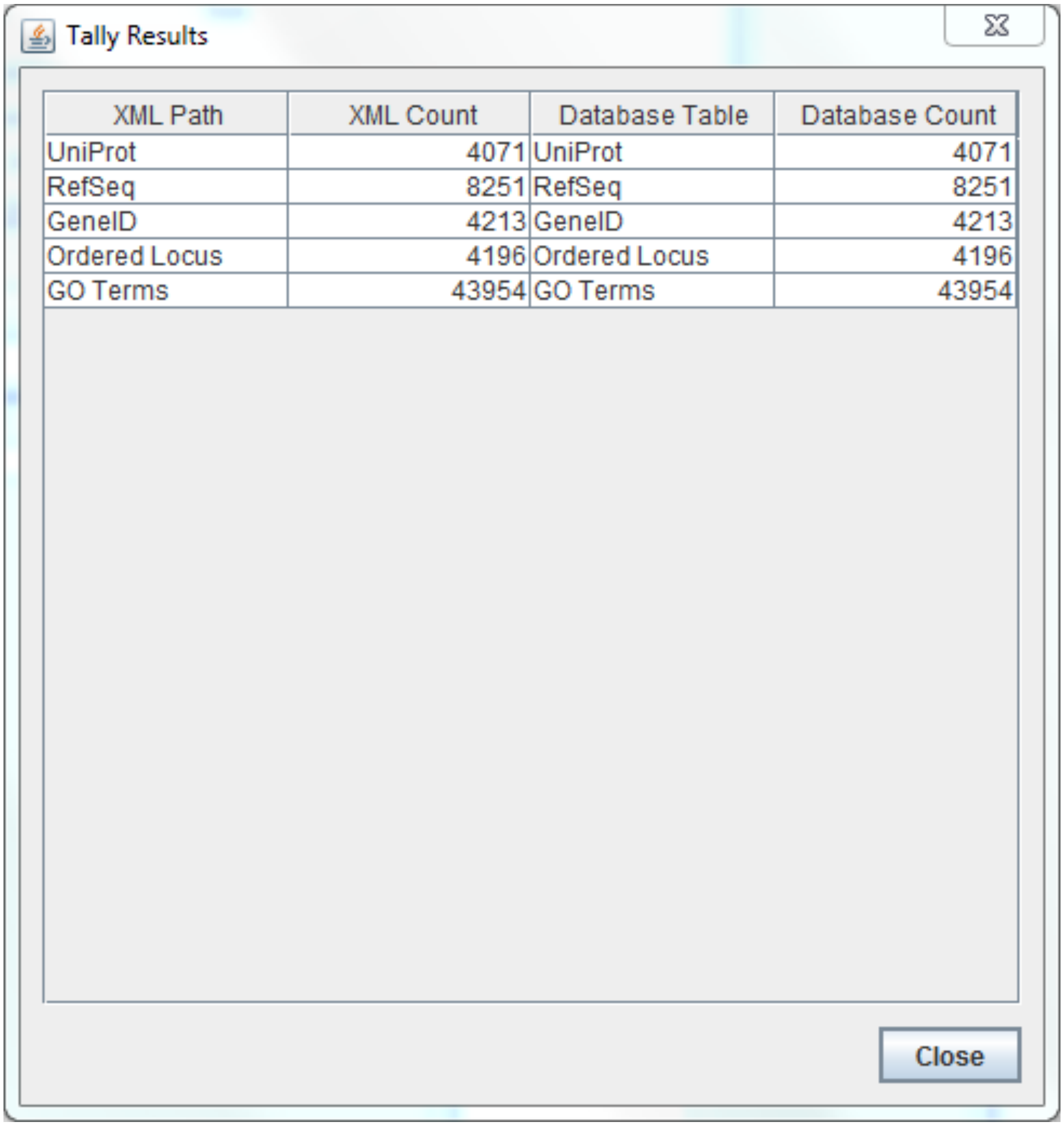
- GOA version (News on this page (http://www.ebi.ac.uk/GOA/) records past releases; current information can be found in the Last modified field on the FTP site (ftp://ftp.ebi.ac.uk/pub/databases/GO/goa/proteomes/)): **October 14, 2015 - GOA Proteome Sets 124**
- GOA download link (ftp://ftp.ebi.ac.uk/pub/databases/GO/goa/proteomes/106.S_oneidensis.goa)
- Time taken to import: **0.05 minutes**
  - Note: *n/a*

Name of .gdb file: **So-Std 20151119HMH.gdb**

- Time taken to export: **1 hour and 18 minutes**
  - Start time: **3:48pm**
  - End time: **5:06pm**
  - Note: *n/a*

TallyEngine

- Run the TallyEngine in GenMAPP Builder and record the number of records for UniProt and GO in the XML data and in the Postgres databases.
  - Choose the menu item Tallies > Run XML and Database Tallies for UniProt and GO...
  - Take a screenshot of the results. Upload the image to the wiki and display it on this page.
  - **4196** IDs
  - For more information, see this page.



XML Path	XML Count	Database Table	Database Count
UniProt	4071	UniProt	4071
RefSeq	8251	RefSeq	8251
GeneID	4213	GeneID	4213
Ordered Locus	4196	Ordered Locus	4196
GO Terms	43954	GO Terms	43954

Using XMLPipeDB match to Validate the XML Results from the TallyEngine

Follow the instructions found on this page to run XMLPipeDB match.

Are your results the same as you got for the TallyEngine? Why or why not?

- Initially, we got **4196** IDs for both XML and Postgres DB from TallyEngine but got **4079** IDs by using XMLPipeDB match. This result was from using the following command:

```
java -jar xmlpipedb-match-1.1.1.jar "SO_[0-9][0-9][0-9][0-9]" < SOneidensisUNIPROT
```

- After checking the .gdb file and looking through the Gene IDs, I found that some IDs were in the form "SO_A####" so I ran a new command accounting for this:

```
java -jar xmlpipedb-match-1.1.1.jar "SO_A?[0-9][0-9][0-9][0-9]" < SOneidensisUNIPROT
```

- This gave a total number of **4207** IDs.

Using SQL Queries to Validate the PostgreSQL Database Results from the TallyEngine

For more information, see this page.

You can also look for counts at the SQL level, using some variation of a *select count(*)* query. This requires some knowledge of which table received what data. Here's an initial tip: the *gene/name* tags in the XML file land in the *genenametype* table. A query on this table counting values from this table that were marked as *ordered locus* in the XML file matching the pattern *SO_[0-9][0-9][0-9][0-9]* would look like this:

```
select count(*) from genenametype where type = 'ordered locus' and value ~ 'SO_[0-9][0-9][0-9][0-9]';
```

- However, once I found that some IDs were in the form "SO_A####" I tweaked the pattern to account for those IDs:

```
select count(*) from genenametype where type = 'ordered locus' and value ~ 'SO_A?[0-9][0-9][0-9][0-9]';
```

In *pgAdmin III*, you can issue these queries by clicking on the pencil/SQL icon in the toolbar, typing the query into the *SQL Editor* tab, then clicking on the green triangular *Play* button to run.

Are your results the same as reported by the TallyEngine? Why or why not?

- Initially, we got a count of **4068** IDs using SQL, which differed from the **4196** IDs from TallyEngine.
- After tweaking the pattern to account for IDs with that extra *A*, we got a grand total of **4196** IDs, which matches with what TallyEngine gave us.

OriginalRowCounts Comparison

Within the .gdb file, look at the OriginalRowCounts table to see if the database has the expected tables with the expected number of records. Compare the tables and records with a benchmark .gdb file.

Benchmark .gdb file: 2010 benchmark file (http://sourceforge.net/projects/xmlpipedb/files/V.%20cholerae%20Gene%20Database/V.%20cholerae%2020101022/Vc-Std_External_20101022.zip/download)

Copy the OriginalRowCounts table from the benchmark and new gdb and paste them here:

Original Row Counts Table

2010 Benchmark Original Row Counts Table

See Analysis section for more on the comparison and the discrepancy found because of this comparison.

Note: Using Microsoft Access, we found *7664* IDs, which was actually double the number of IDs present because of duplicated IDs that did not have an underscore.

Visual Inspection

Perform visual inspection of individual tables to see if there are any problems.

- Look at the Systems table. Is there a date in the Date field for all gene ID systems present in the database?
    - No. For the current version, a good number of gene ID systems in the database do not have a value for the date field. Some systems that lack a date include: GenBank, UniGene, WormBase, and EcoGene.
- Open the UniProt, RefSeq, and OrderedLocusNames tables. Scroll down through the table. Do all of the IDs look like they take the correct form for that type of ID?
    - For the UniProt table, the IDs start with either *Q8* or *K4* and have some string of four letters/numbers trailing that. For the RefSeq tables, the IDs have forms that start with either *NP_* or *WP_*, with the *WP_* forms having 9 numbers afterwards and the *NP_* forms having 6 numbers afterwards. For the OrderedLocusNames table, the IDs either start with `SO_` or `SO_A`.

Note: *n/a*

Analysis

Consolidating the counts of gene IDs from the various methods, I got:

- 4196 IDs from Tally Engine
- *4207* IDs from xmlpipedb-match

```
java -jar xmlpipedb-match-1.1.1.jar "SO_A?[0-9][0-9][0-9][0-9]" < SOneidensisUNIPROT
```

- 4196 IDs from PostgreSQL

```
select count(*) from genenametype where type = 'ordered locus' and value ~ 'SO_A?[0-9][0-9][0-9][0-9]';
```

- 4196 IDs from Microsoft Access (Noted that there were 4068 IDs in the form `SO_`*####* and 128 IDs in the form `SO_A`*####*)

Notice that there is a small but significant discrepancy in that there seems to be eleven more IDs when using xmlpipedb-match. This is troubling because of the fact that the other three methods seemed to confirm a total count of 4196. So, I used Microsoft Excel to compare the list of gene IDs from the actual .gdb file and the list I got back from xmlpipedb-match. As you can see on this document, there are 11 IDs in the xmlpipedb-match column that are not found in the gdb column. This discrepancy was further pointed out by the use of some `match` functions to see where an ID was missing from either list. Below are the two match functions I used in the document:

```
=MATCH(A2, B$2:B$4208, 0)
=MATCH(B2, A$2:A$4208, 0)
```

Below are the eleven IDs in question:

```
SO_3699 NO-KD
SO_1312 NO-KD
SO_4269 NO-KD if they are all part of "Protein-protein interaction databases", then you can safely leave them
SO_2875 NO-MA
SO_4532 NO-MA
SO_4580 NO-MA
SO_2662 NO-MA
SO_4423 NO-MA
SO_3156 NO-MA
SO_2967 NO-MA
SO_2024 NO-MA
//I looked these up on www.uniprot.org and all were found to be only in "Protein-protein interaction database
```

Look up the IDs at UniProt web site (http://www.uniprot.org) and then search for them on the UniProt record web page. If they are part of the "STRING" protein-protein interaction database, then you can safely leave them out. — *Kdahlquist (talk) 15:55, 3 December 2015 (PST)*

None of these IDs are in our MOD. We searched for these IDs in the microarray statistical analysis sheet and did not find the following IDs:

```
SO_4269, SO_2875, SO_4580
```

As of 12/01/15, we are waiting on input from Professor Dahlquist to see if we will adjust our GenMAPP Builder to account for these 11 IDs.

- A manual inspection was done on the **SOneidensisUNIPROT** XML file and it looks like these 11 IDs are contained within entries that are missing a gene tag, which explains why the other methods only picked up 4196 IDs.

.gdb Use in GenMAPP

While the above sections perform quality assurance on the exported Gene Database via verifying ID counts, the "proof in the pudding" is to actually use the Gene Database in GenMAPP. You can follow the instructions in Part 2 of the *Vibrio cholerae* Microarray Data Analysis (http://www.openwetware.org/wiki/BIOL367/F10:GenMAPP_and_MAPPFinder_Protocols) to verify that the Gene Database works in GenMAPP. In this case, the emphasis is not on the findings of the data analysis itself, but that the Gene Database functions appropriate in GenMAPP.

For assistance with using the GenMAPP program, the GenMAPP Help is very extensive. To access it within GenMAPP, go to the menu item Help > GenMAPP Help and either browse or search for your topic of interest.

Note: *n/a*

Putting a gene on the MAPP using the GeneFinder window

- In the main GenMAPP Drafting Board window, left-click on the icon for "Gene" in the upper left corner of the window. Click on the Drafting Board to place the Gene on the MAPP. Now, right-click on the gene to access the GeneFinder window. Type or paste a gene ID into the Gene ID field. Select the appropriate Gene ID system from the drop-down menu and click the Search button. For example, for *Vibrio cholerae*, you could search for the ID "VC0028", which is an OrderedLocusNames ID. Once the ID has been found, click the OK button to return to the Drafting Board window.
  - For the Final Project, you will need to try a sample ID from each of the gene ID systems, not just OrderedLocusNames.
- Open the Backpage by left-clicking on the gene box on the Drafting Board to see if all of the cross-referenced IDs that are supposed to be there are there.

Note: I tried out the search for a gene ID and was able to bring up the Backpage for that ID. The cross-referenced IDs that were supposed to show up were indeed on the page.

Creating an Expression Dataset in the Expression Dataset Manager

- How many of the IDs were imported out of the total IDs in the microarray dataset? How many exceptions were there? Look in the EX.txt file and look at the error codes for the records that were not imported into the Expression Dataset. Do these represent IDs that were present in the UniProt XML, but were somehow not imported? or were they not present in the UniProt XML?

Note: The Expression Dataset Manager reported that there were 1441 errors during the conversion. From looking over the error codes, I found that these genes were the ones we expected to ignore, like the IDs with an added 'F.'

Coloring a MAPP with expression data

Note: I was able to successfully color the MAPP by coloring the increased and decreased Log Fold Changes.

Running MAPPFinder

Note: After the results had been calculated, a Gene Ontology browser opened showing my results. All of the Gene Ontology terms that have at least 3 genes measured and a p value of less than 0.05 were highlighted yellow. A term with a p value less than 0.05 is considered a "significant" result. Browsed through the tree to see the results.

Documents produced from this run-through can be found here: Gene Database Testing docs

Retrieved from "https://xmlpipedb.cs.lmu.edu/biodb/fall2015
/index.php?title=Gene_Database_Testing_Report_-_Heavy_Metal_HaterZ&oldid=7692"

Categories: Group Projects | Heavy Metal HaterZ

- This page was last modified on 12 December 2015, at 16:16.
-