Implementing Support for Multiple Species in XMLPipeDB’s GenMAPP Builder

Don Murphy1, John David N. Dionisio1, Kam D. Dahlquist2
1Department of Electrical Engineering and Computer Science, 2Department of Biology
Loyola Marymount University, 1 LMU Drive, Los Angeles, CA 90045 USA

Abstract
GenMAPP Builder is a Java program used for the creation of species-specific gene databases for use with GenMAPP, a bioinformatics program for viewing and analyzing DNA microarray data. Biological pathways and associated gene IDs are stored in a PostgreSQL database in the PostgreSQL .gdb format. To analyze DNA microarray data, GenMAPP Builder was extended to allow the creation of multiple-species databases. To allow species with larger genomes to be used with GenMAPP, imports can be performed with multiple subtypes of the GO annotation terms, biological pathways and function-only, creating smaller gene databases. Additionally, GO association terms are imported into the PostgreSQL database, allowing the export process and limiting the potential size of databases further. We have moved the import of the GOA linking file to the step where source data is brought into PostgreSQL, instead of the export step. This change allows the PostgreSQL database itself to be used for applications beyond GenMAPP alone. Finally, we have altered GenMAPP Builder so that it can handle importing and exporting data from multiple species at once. This functionality allows for cross-species comparison of microarray data in GenMAPP. These upgrades to GenMAPP Builder have been used in the creation of databases used in the analysis of microarray data for species such as Saccharomyces cerevisiae and Staphylococcus aureus.

Introduction
GenMAPP is a program used to view and analyze DNA microarray data.

• Graphical tools within GenMAPP are used to draw biological pathways and groupings of genes with similar functions as MAPPs.
• Expression Dataset files (.ges) store experimental data and are imported into PostgreSQL.
• A genomics database is used to store gene IDs, annotation, and pathways and groupings of genes with similar functions as MAPPs.
• The program was written in Visual Basic and reads Gene Database files (.gdb) from PostgreSQL.
• An underlying Gene Database stores gene IDs, annotation, and pathways and groupings of genes with similar functions as MAPPs.
• Expression Dataset files (.gex) store experimental data and are imported into PostgreSQL.
• After making associations within the database, GenMAPP Builder outputs the PostgreSQL database to a GenMAPP-compatible Gene Database.
• For full functionality, each species must have an export feature created in GenMAPP Builder’s source code.
• GenMAPP Builder has several problems, such as an inefficient export method and creating databases that are too large for GenMAPP to process.
• To resolve these issues, changes had to be made to the source code of the required species source files and how exports are performed.

Methods
GOA source files are imported to PostgreSQL database before export
• Three types of source files are used by GenMAPP Builder to create gene databases: UniProt XML (.xml), OBO-XML (.obo- xml), and OAE ontology (.gao).
• In previous versions of GenMAPP Builder, only the XML-formatted files were imported into the PostgreSQL database, while the tab-delimited GOA file was held in memory during export.
• An additional class was created to import GOA files into the PostgreSQL database, allowing the export process to only export genes for the species being exported.
• The new import process has found to be of use.
• Additional changes were needed to produce usable gene databases for well-documented species.
• To create smaller but functional gene databases that could be read by GenMAPP, several methods were attempted.
• Early attempts used GO Slim versions of source files, limiting the number of GO terms added to the gene database.
• To keep all terms while producing smaller databases, exporting was modified to allow for creation of separate partial gene databases dedicated to one of the three “aspects” of GO terms.

Additional changes were needed to produce usable gene databases for well-documented species.
• To create smaller but functional gene databases that could be read by GenMAPP, several methods were attempted.
• Early attempts used GO Slim versions of source files, limiting the number of GO terms added to the gene database.
• To keep all terms while producing smaller databases, exporting was modified to allow for creation of separate partial gene databases dedicated to one of the three “aspects” of GO terms.

Results
• Improvements to GenMAPP Builder that allowed the import of GOA files into the PostgreSQL database were effectively implemented.
• GO Slim was experimented with, but has been rejected for the time being as the resulting databases had too many GO terms removed to be of use.
• Single aspect databases were exported by performing the GOA source file to include only one GO term aspect, but was added as an option during export, alongside GOA importing.
• Gene databases created by the improved GenMAPP Builder have been used in studies submitted to the Loyola Marymount Undergraduate Research Symposium 2010 and SCCUR 2010.
• Multiple species import-export is still in development; recent changes which identify species by the unique taxonomy ID number as opposed to the varying species names for each species-specific gene databases possible.

Discussion
Challenges Encountered
• While developing the GOA import engine, a new format for GOA files was released, to support users of both the old and new formats, the engine was developed to identify the format being used and use different insert PostgreSQL statements for each format.
• Additional troubleshooting was necessary throughout the development process; for example, an export connection error originally thought to be attributed to Windows 7 was researched and found to be caused by the inability of the program to use JDBC-ODBC database connection drivers from a 64-bit Java environment and required a fix after a campus-wide upgrade to 64-bit Windows 7.

Future Work
• As all data needed for a gene database is now present in the PostgreSQL database, new GOA files are imported into GenMAPP Builder.
• Improvements to GenMAPP Builder that allowed the import of GOA files into the PostgreSQL database were effectively implemented.
• GO Slim was experimented with, but has been rejected for the time being as the resulting databases had too many GO terms removed to be of use.
• Single aspect databases were exported by performing the GOA source file to include only one GO term aspect, but was added as an option during export, alongside GOA importing.
• Gene databases created by the improved GenMAPP Builder have been used in studies submitted to the Loyola Marymount Undergraduate Research Symposium 2010 and SCCUR 2010.
• Multiple species import-export is still in development; recent changes which identify species by the unique taxonomy ID number as opposed to the varying species names for each species-specific gene databases possible.

Acknowledgments
We would like to thank the BiOL/CMSS/HRHS 398 Biological Databases Fall 2009 Class, especially Team Elite Three (Bernadette Pak, Kenny Rodriguez). We would like to thank the BIOL/CMSI/HNRS 398 Biological Databases Fall 2009 Class, especially Team Elite Three (Bernadette Pak, Kenny Rodriguez). We would like to thank the BIOL/CMSI/HNRS 398 Biological Databases Fall 2009 Class, especially Team Elite Three (Bernadette Pak, Kenny Rodriguez). We would like to thank the BIOL/CMSI/HNRS 398 Biological Databases Fall 2009 Class, especially Team Elite Three (Bernadette Pak, Kenny Rodriguez). We would like to thank the BIOL/CMSI/HNRS 398 Biological Databases Fall 2009 Class, especially Team Elite Three (Bernadette Pak, Kenny Rodriguez).

Literature Cited

Figure 1: GenMAPP can be used to create MAPPs of groups of genes and then be colored to show variations in the expression of the genes.

Figure 2: Components added to or edited in GenMAPP Builder for the use of GOA data within the PostgreSQL database are highlighted in blue in this component diagram.

Figure 3: GenMAPP Builder’s export interface lets the user choose to export either a full or single-aspect gene database.

Figure 4: When selecting a species to export to a gene database, the species is identified by its taxonomy ID alongside the species name.

Figure 5: GenMAPP can be used to create MAPPs of groups of genes and then be colored to show variations in the expression of the genes.