MySQL Implementation for a Database of Protein Structures: SupSQL

DEPARTMENT OF Scientific COMPUTING

Miguel A. Colón-Vélez[1], Gavin J. P. Naylor[1]

(1) Department of Scientific Computing, Florida State University

ABSTRACT

Computational biologists are increasingly interested in combining information from both sequence structure and sequence evolution. Historically databases have been generated to archive structural sequences or evolutionary trees but to date there have been few attempts to integrate this knowledge. Here we propose a MySQL database implementation (SupSQL) to explicitly incorporate structural and evolutionary information. The database will include useful information about the class of structure (globular/transmembrane), quaternary structure, genomic sequence, introns, taxonomy and protein family. All of these data are readily available on the internet but are spread out throughout several databases like UniProtKB, Pfam, RCSB Protein Data Bank, PDBTM, EMBL, NCBI Taxonomy, etc. The proposed implementation will let us combine all of this knowledge in a single database and by using a standardized language like SQL it would be easy to access and share with interested parties. With the information contained in this database, we would have a systematic method to select proteins that match certain criteria and therefore we don't depend on randomly selecting proteins based on past experience or biased knowledge.

INTRODUCTION

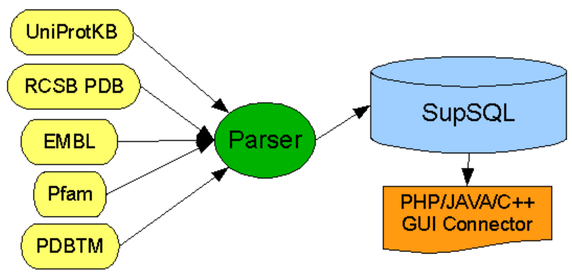
Important terms and definitions:

- **Computational Biology:** Development of algorithms and statistical models to analyze biological data.
- **Bioinformatics:** Development of computational methods for studying the structure function and evolution of genes proteins and whole genomes.
- **UniProt Knowledgebase (UniProtKB):** central hub for the collection of functional information on proteins, with accurate, consistent and rich annotation.
  - Swiss-Prot: reviewed, manually annotated.
  - TrEMBL: unreviewed, automatically annotated.
- **Pfam:** collection of multiple sequence alignment and hidden Markov models covering many common protein domains and families.
- **RCSB Protein Data Bank:** archive containing information about experimentally-determined structures of proteins, nucleic acids and complex assemblies.
- **PDBTM:** database created by scanning all PDB entries with a transmembrane detection algorithm.
- **EMBL Nucleotide Sequence Database:** European nucleotide sequence resource. Contains sequences submitted by individual researchers, genome sequencing projects and patent applications.
- **SupSQL:** Sequences in UniprotKB with a PDB entry stored in a SQL database.

GOALS

- Generate a database that combines structural and evolutionary information from various sources into a single place. This information would then be used in upcoming bioinformatic projects .
- Create an intuitive graphical interface or method to access the information contained in this database.

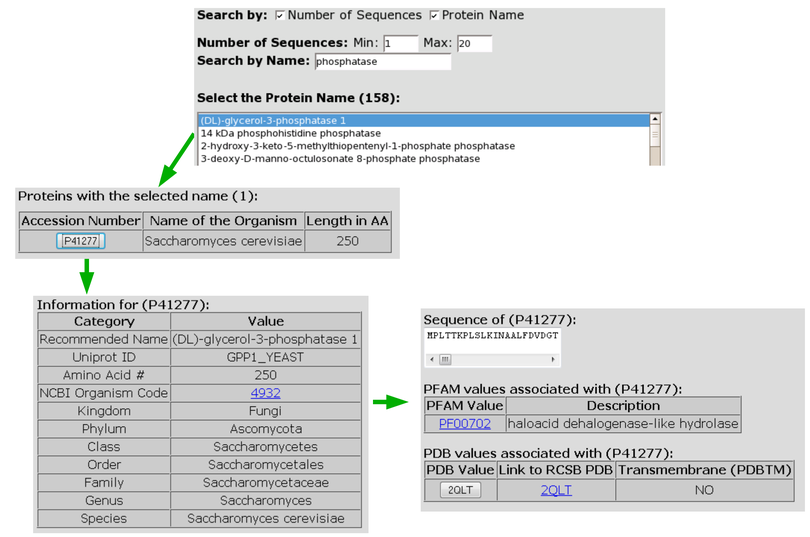
METHODS



- Using a bash script, I download all the entries on Swiss-Prot that contain PDB entries associated with them. In addition partial or complete downloads of the NCBI Taxonomy database, Pfam, PDBTM, RCSB PDB and others is done at this stage.
- The raw NCBI taxonomy flat files are parsed and a local Taxonomy database is created. With this we can generate the full or partial lineage of all the organisms found in UniProtKB.
- All the downloaded PDB files are read. Information about the quaternary structure of the proteins (if available) is extracted with a brief description of how this information was obtained.
- Information about the Pfam group and description of said group is extracted from the Pfam flat files. UniProtKB only contains the Pfam identifier therefore data is crucial to get the information related to the Pfam identifier.
- The PDBTM database is parsed to obtain a non redundant list of the PDB entries that have been found to be of transmembrane proteins.
- At this stage the Swiss-Prot flat file is parsed to obtain all the identifiers of the external databases and general information about the protein and its sequence.
- The output from the previous steps are loaded into MySQL. The loaded tables are for the most part in the third normal form which means that redundancies and other nuisances that could lead to a loss of data integrity are avoided.
- The process of updating or generating the database takes 2 hours at the most. The most time consuming process is the creation of the PDB table which can take over an hour. All of this has been automated and can be scheduled as a CRON job if desired.

USABILITY

- To access the information in the database the user can select from several connectors that are available to access data from MySQL. This makes the database more accessible and promotes the usage of a more standard method to compile and access bioinformatic data.
- As an option, an AJAX/PHP based website is currently in extensive development.



FUTURE WORK

- Improve the graphical interface since at the moment accessing information and security has been the priority. A more intuitive layout and presentation is planned.
- Provide a way to generate phylogenetic trees on demand from the sequences that the user selects.
- Find a more efficient way to obtain the genomic sequence data since the current method has problems with genes that experience splicing.

ACKNOWLEDGEMENTS

- I want to thank PhD student Clemens Lakner for the input he provided at various stages of this project.
- I want to thank Google for letting me learn PHP/AJAX faster than I thought it was possible and for helping me survive those moments in which I wanted to throw my computer out the window.