

Database Management Systems Technologies for Computational Biology & Bioinformatics Applications

Knowledgebase R&D Pilot Project

Victor Markowitz, Keith Jackson, Ernest Szeto, Konstantinos Mavrommatis

Lawrence Berkeley National Laboratory (LBNL)

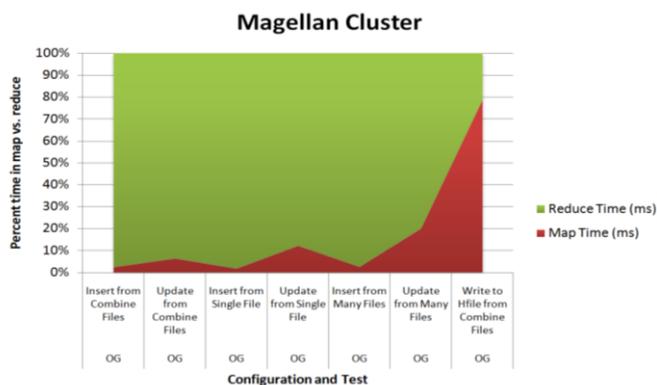
The aim of this project was to examine new **database management system technologies** for supporting efficient analysis of very large genome and metagenome sequence datasets.

Comparative analysis of genomic and metagenomic datasets is usually based on integrating these datasets in the context of databases implemented using relational commercial database management systems (DBMS) such as Oracle or open source DBMS such as MySQL. The rapid increase in the number and size of these datasets results in a decrease in performance of typical comparative analysis tools, such as examining putative operons across microbial genomes. A recent benchmark of relational DBMS¹ indicates that new database management technologies are better suited for scientific data management applications. We set out to evaluate the usage of cloud based data management technologies for handling large genome and metagenome datasets, in particular Hadoop data management components for data storage and querying. Hbase² is a distributed, column-oriented data store that supports real-time access to extremely large data.

Cloud based data management technologies can be potentially very useful for a wide variety of genome and metagenome data management applications. We used as a case study the Integrated Microbial Genomes (IMG) system. IMG currently stores the results of "all vs. all" pairwise gene comparisons in sequence similarity files. These files are tab-delimited files generated by NCBI's blastall program with the -m8 option, containing the identifiers of pairs of matching genes, scores pertaining the strength of match such as alignment percentage identity, regions of matches, bit score, and an evaluation of statistical significance through the expectation value (E-value). Storing the results in flat files has several disadvantages in comparison to tabular storage. Modifying individual entries is challenging, and queries are significantly harder than would be the case in tabular storage.

Our testing of HBase shows that distributed tabular storage has significant long term potential for the GTL Knowledgebase, but that current HBase versions are not ready for large-scale production use today. Issues with both stability and performance will need to be addressed before HBase can be used in a production Knowledgebase application.

We encountered significant difficulties in running HBase in a stable fashion. We encountered frequent crashes and performance problems while attempting to bulk load data. Some of these problems were surely caused by our inexperience in running Hadoop/Hbase in a production environment, but others are likely the result of the relative immaturity of the software. Both Hadoop and HBase are undergoing rapid development currently and we anticipate that many of these stability problems will be addressed over the next year or two.



¹ Stonebraker M. et al. One Size Fits All? – Part 2: Benchmarking Results. <http://www.cs.brown.edu/~ugur/osfa.pdf>.

² Hbase: <http://hadoop.apache.org/hbase/>