GeneCrunch: Experiences on the SGI POWER CHALLENGEarray with Bioinformatics applications

Reinhard Schneider*, Georg Casari*, Antoine de Daruvar*, Pam Bremer*, Michael Schlenkrich*, Richard Mercille*, Horst Vollhardt*, Chris Sander*

*EMBL Heidelberg, Meyerhofstr. 1, 69012 Heidelberg, Germany; * Silicon Graphics, Kaegenstrasse 17, 4153 Reinach, Switzerland; * Silicon Graphics, Chemin des Rochettes 2, 2016 Cortaillod, Switzerland; * EMBL-EBI, Hinxton Hall, Cambridge, CB10 1RQ, United Kingdom

E-Mail: schneider@embl-heidelberg.de , casari@embl-heidelberg.de, daruvar@embl-heidelberg.de, pam@basel.sgi.com, ms@basel.sgi.com, rim@neu.sgi.com, horstv@basel.sgi.com, sander@ebi.ac.uk

Abstract. Analyzing genomic data is a computationally intensive and complicated process in which scientists must typically choose among multiple databases and analysis methods and make expert judgements inspecting multiple results. GeneQuiz, an automated software system for large scale genome analysis developed at the EMBL/EBI, tackles this problem by using an automated, rigorous, rule-based system to select among the results of sequence analysis and database searches, builds informative annotation and aims at predicting the function of new genes. In a demonstration project more than 6000 proteins from the Baker's yeast, for which the complete genomic sequence was completed in 1996, were analyzed on a Silicon Graphics POWERCHALLENGEarray with 64 processors (R8000 @90 MHz) so that the analysis could be completed in 3 days. The results of the analysis were published on two web servers as they were computed.

1 Large-scale sequence analysis and the need for automatic tools

As genome sequence data is being produced at an accelerating pace, there is a need for faster and more reliable methods of large-scale sequence analysis. There exists a multitude of algorithms, a large number of sequence and bibliographic databases, and various single methods that can be useful in the prediction of protein function. From this large collection of tools, an optimal constellation must be chosen that satisfies the requirements for accurate and sensitive function prediction by

homology. Speed is also an important factor for the analysis, but accuracy should not be sacrificed.

The technical challenges are two-fold:

- how to quickly identify sequence similarities in molecular databases efficiently without losing sensitivity and
- how to integrate existing software and databases, annotate, evaluate and document the findings of experts in a multi-user interactive environment.

Large-scale sequence analysis differs from traditional practices in two basic respects:

- with the current and forseen growth of the biological databases computational efficiency using fast algorithms, certain heuristics as well as supercomputers are essential
- information support for expert users is becoming crucial, as the emerging gene and protein families from genome projects extend beyond the areas of expertise of a single individual.

Therefore, the development of a system is required which performs the necessary analytical steps for a large number of sequences as well as providing access to molecular and bibliographic databases.
The most compelling question in computational genome analysis is the identification of homologies in search of a function. However, the issue of function prediction for proteins is partly a problem of definition. We can define function prediction as any evidence towards the identification of various protein sequence characteristics indicative of substrate recognition and catalysis, interactions, localization and evolutionary relationships. Therefore, the characterization of a protein sequence (or an ORF) usually takes place at various levels of accuracy, for example, from prediction of cell membrane spanning regions to the derivation of a three-dimensional model, on the basis of homology to a well-characterized protein.

2 The GeneQuiz system

GeneQuiz [1, 2] is an integrated system for large-scale biological sequence analysis that goes from a protein sequence to a biochemical function, using a variety of search and analysis methods and up-to-date protein and DNA databases. Applying an "expert system" module to the results of the different methods, GeneQuiz creates a compact summary of findings. It focuses on deriving a predicted protein function, based on the available evidence, including the evaluation of the similarity to the

closest homologues sequences in the database. The analysis yields a great portion of the information that can possibly be extracted from the current databases, including three-dimensional models by homology, when the structure can be reliably calculated.
The principal design requirement is the complete automation of all repetitive actions: database updates, efficient sequence similarity searches, sampling of results in a uniform fashion and evaluation and interpretation of the results using expert knowledge coded in rules (Figure 1). For handling such a heterogenous set of tools and tasks in the GeneQuiz system we chose the *perl* script language [3].

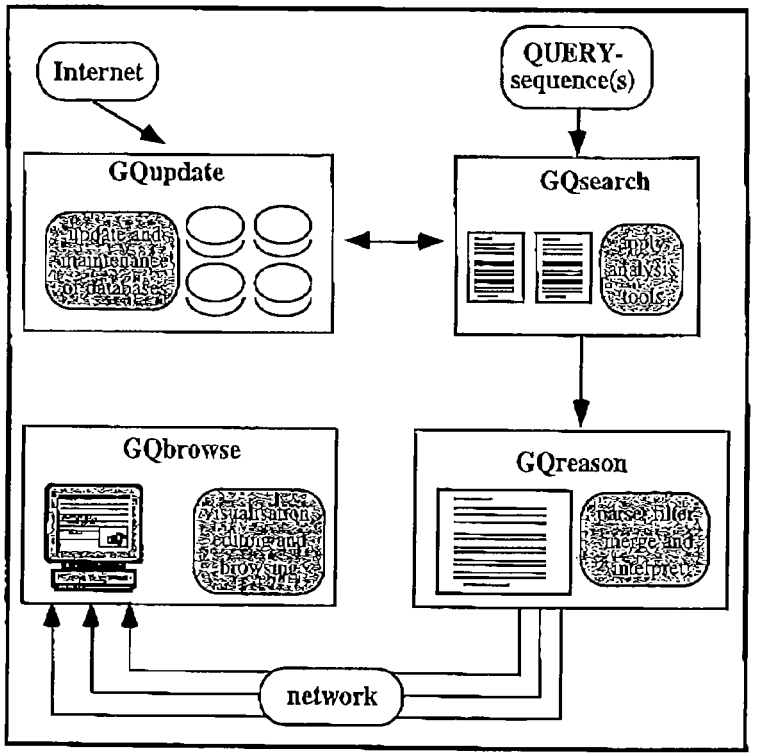

Figure 1: Schematic flow chart of the GeneQuiz system. Note: for the analysis done during the GeneCrunch project the databases were frozen and no update was performed during the runs.

2.1 Automated database updates and indexing

Parallel to the explosion of data production from genome projects, various databases have been created to accommodate the needs of specialized scientific communities. The generation of these databases is done locally, and computer networks with appropriate information retrieval systems may provide access. The exponential growth of these databases mandates frequent local updates, sometimes even during the analysis process.
For the purpose of automatic database updates we developed the GQupdate module, however for the GeneCrunch project we decided to use up to date, but „frozen", databases consisting of non-redundant protein as well as DNA sequences (see Table 1).

2.2 Automated database searches and sequence analysis

To accelerate a first scanning of all databases in the most efficient way, a hierarchical model for database searches was implemented. First, searching with the fastest available tool, presently BLAST [4], allows the identification of clear homologous sequences from which a possible function can be inferred. The search is by default performed against a non-redundant database, which also includes the proteins translated from genes in the DNA databases.
Additional characteristics of newly sequenced ORFs are of interest, especially when function by homology cannot be predicted. For example, structural features, indication for the subcellular location, or previously described sequence patterns can be of extreme importance for a further understanding of protein function. The computing time for these analysis is negligible therefore they are always. In addition to standard analyses, we use filters, pre- and postprocessing tools as well as additional methods for shorter and more meaningful output lists, multiple alignments, cluster analysis and secondary structure prediction.

The GQsearch control program also allows the distribution of jobs in a cluster of workstations or a parallel computer for a speed-up of the searching process. Due to the fact that each single sequence analysis can be seen as an independent task the problem becomes in principle „embarassingly parallel" and allows for a straightforward load balancing scheme. The program contains a checkpointing facility to ensure the recovery of jobs after hardware and software problems.

2.3 Parsing of search results, reasoning and function assignment

Various programs provide a wide range of output formats, usually a compromise between machine and human readability. The lack of syntax and a standard has necessitated the implementation of a variety of dedicated parsers for the output.

Parsers for all the database search programs and for most of the analysis tools have been implemented. In that respect, the system is independent of the search software. Organizing principles for the storage and the manipulation of results are necessary elements in this effort, since sequence database searches and other analyses provide us with a large amount of essentially unprocessed data. We decided to use a well-developed formalism in database design, the relational database model. The result parsers produce entity tables that are directly readable by a simple relational database (RDB), which is a simple yet powerful highly portable database system written in perl (developed by Walt Hobbs, RAND Corporation, Santa Monica, CA-USA).

The module GQreason for automated reasoning and function assignment, is the third, and in some ways most crucial, component. Instead of relying on experts for the interpretation of the performed search and analysis , this suite of programs controls the evaluation of findings with very high reliability and reproducibility.

The function assignment for a protein sequence is made on the basis of the documentation of the homologues sequences in the database. For each method or topic, we have an independent component containing a coded set of rules to treat this task. These components first check if the required information is available. If so, their rules code the expert knowledge about the interpretation of the specific results, like how to assess reliability based on scores from different methods, which results are worth reporting, and how to process the information to derive new facts (e.g. the total number of proteins found with significant homology).

Finally, in the last step, the extracted features are summarized at a higher level into a comprehensive table of the results. At this level, rules are very strict, and we report only clear results. In this way, the user can trust the derived facts and is relieved from time-consuming interactive checking. Ambiguous assignments are marked as such, and help the user to directly focus on those difficult cases which are not automatically resolved at present.

2.4 Viewing and browsing the results database using WWW-tools

The fourth module (GQbrowse), gives access to the result databases and allows for interactive evaluation and browsing of related sequences and other databases like bibliographic entries. The current solution is based on World Wide Web technology and dynamically provides HTML [5] documents that can be displayed with most of the Web browsers (like Netscape or Mosaic). With this technology, it is straightforward to make the results - and the whole browsing capacity - available to any user connected to the Internet.

2.4.1. Dynamic HTML pages

The pages provided by the viewer are dynamically generated from the result database generated during the run of the GQreason module. This way several user-specified views can be generated from the same set of data. The translation program is realized in the *perl* scripting language. All www-adresses are in fact calls to these 'CGI'-scripts [6]. They generate the HTML documents and even insert functional links to related information (like database entries of sequences) on the fly. For this purpose, we generate links to the SRS database retrieval system [7], that keeps many biologically relevant databases indexed and supplies rapid access to this information. Furthermore, SRS administers links among entries in different databases and helps to move easily to associated pieces of information. Besides this 'browsing' functionality we provide some 'zooming in' functionality. The result database keeps the sources of any information stored, and GQbrowse automatically generates links to these sources. Thus, more detailed information and inspection of the original results is just a mouse-click away.

2.4.2. Interactive Visualization of Protein Structures

The recently developed Virtual Reality Modeling Language (VRML) [8] provides new opportunities for the visualization of molecular structures over the World Wide Web. The basic elements (called nodes) of VRML can be used to describe the layout of a three-dimensional (3D) object or a 3D scene. Besides the 3D modeling elements, VRML has building blocks such as hyperlinks and inline nodes very similar to the tags provided by HTML.

With this new technique, the user does not need direct access to a structure database in order to visualize its content. Moreover, there is no need to install, configure and learn any visualization software specialized for molecular modeling. This way, the content provider can reach a larger audience by supplying 3D scenarios based on VRML [9]. These scenarios do not have to be predefined by the author wasting a lot of disk space and time for updates. The VRML model of the structure can be generated on demand and the representation can be customized as requested by the user. We demonstrated this new method with an example implemented on the GeneCrunch WWW server [10]. During the analysis three-dimensional models were generated for proteins with a clear homology to an already known 3D-structure. For this purpose the automatic model building procedure build in the WHATIF program [11] was used.

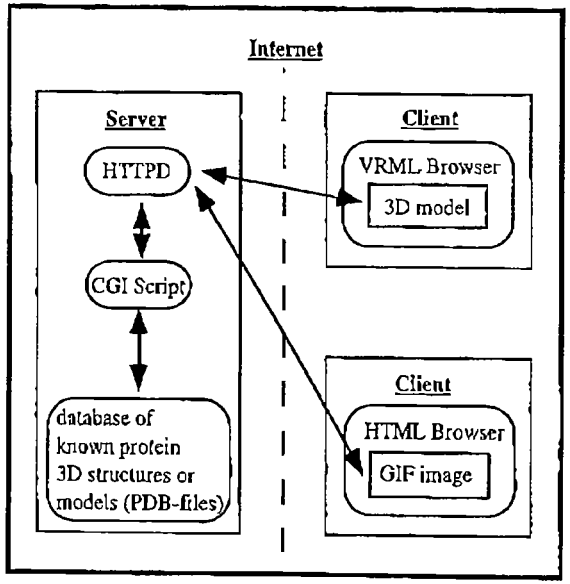

Figure 2: Schematic connection between the database on the WWW server and visualization tools on the WWW clients. With each client request a CGI script converts the original atomic coordinates in the VRML format. The VRML file can be converted into a GIF picture using an off-screen rendering program. Both steps require only a few seconds CPU time on the server.

3 GeneCrunch setup

The yeast genome sequence was completed and released in 1996 as a result of an international collaboration with teams from the European Union, North America and Japan. The genome consists of about 12.5 million base pairs on 16 chromosomes. It's content is estimated to about 6300 open reading frames (ORF's) which are regions of the genome identified as potentially encoding proteins. Complete analysis of the yeast genome represents a milestone in genomic research. It is the first eucaryotic genome fully sequenced and contains the complex subcellular organization typical of higher organisms.

While many parts of the yeast genome have been previously analyzed over the history of the biological studies and the sequencing effort, a complete re-analysis using the most up-to-date versions of the databases and the newest search and analysis methods was necessary. For the GeneCrunch project we extracted more than 6000 yeast protein sequences from the publicly available databases. The

analysis of these proteins represents a large computational effort that would require months to complete on standard workstations or servers.

To redress this computational bottleneck, the analysis was performed on a Silicon Graphics POWER CHALLENGEarray providing 23.04 GFlops of compute power on 4 POWER CHALLENGE nodes with 16 R8000 90MhzCPUs and 2 GB RAM each. POWER CHALLENGES use the concept of shared resource parallelism which permits applications with different memory, I/O, compute and visualization needs to run while the hardware and operating system ensure data consistency among the parallel threads and dynamically allocate the resources among the different programs. This concept makes it very easy to run the GeneQuiz system in which applications vary greatly from each other in computational resource needs.

A HIPPI internet providing 100MB/sec data transfer speeds, with a sustained bisectional bandwidth of 200Mb/sec connected the 4 nodes. The sustained bandwidth within each node is 1.2GB/sec. A separate CHALLENGE acted as a file server for the array, with 20GB of data stored in one XFS volume covering 5 4GB physical disks. One node of the array also served as the repository for 13GB of data generated during the computation and provided internal Web service on the Silicon Graphics Intranet.

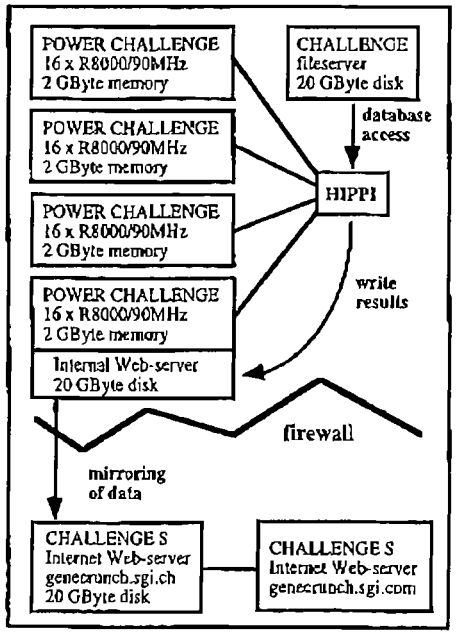

Figure 3: Schematic drawing of the setup for the GeneCrunch project

The external web sites were provided with two Challenge S systems located outside the firewall to provide Internet connections located in Switzerland and California. Data as it was generated was copied across the firewall to a 20GB RAID system on the Switzerland server and was NFS mounted to the server for the US site. While both machines were physically located in Switzerland, the US server was connected to the Silicon Graphics Mountain View site through a 128k dedicated line and thereby provided external internet access with good response times.

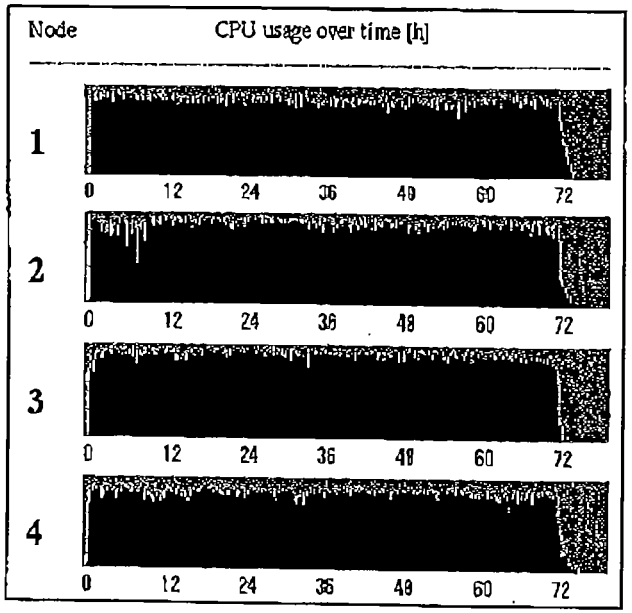

Figure 4: CPU usage over the 3 days run time of the GeneCrunch project. Each „Node" represents one POWER CHALLENGE with 16 * R8000 @ 90MHz.

4 Results and Conclusion

The analysis was performed between March 4-7 on the POWER CHALLENGEarray placed at the Silicon Graphics' European Supercomputing Technology Center in Cortaillod, Switzerland. The results of the analysis were made available via two web servers while the computations were running. This procedure - quite unusual in science where data are not shared prior to publication in journals - was one of the novel features of the GeneCrunch project. The results are available at the following web-addresses:

- http://genecrunch.sgi.ch
- http://genecrunch.sgi.com
- http://www.sander.embl-ebi.ac.uk/genequiz/

Given the extremely heterogeneous „production environment" for the GeneCrunch project, the POWERCHALLENGEarray was stable during the live event with no single hardware or system software problem (see also Figure 4). The only problem during the project was a disk crash on the outside WWW-server which caused a backlog on the results for a few hours.
With the biological sequence databases used the analysis required more than $1.90 * 10^{10}$ sequence comparisons just for the database scanning. The complete analysis could be completed in 72 hours which is equivalent of more than 73000 sequence comparisons per second (Table I). In addition to the raw database scans the GeneQuiz system fired off all the other tools and analysis programs to produce things like multiple sequence alignments for protein families, predictions of structural features up to the generation of full 3D atomic coordinate sets for model structures. In total more than 4200 multiple sequence alignments were made and a few hundred 3D model were generated. Just the generation of these additional results would keep a fast workstation busy for a few days.

6613	Yeast sequences analyzed
185,688	entries in NR-DB (non-redundant protein database)
54,435,055	amino acids
438,305	entries in NR-EST (non-redundant EST database)
151,663,018	bases
13226	BLAST runs
2221	additional FASTA runs against NR-DB
$1.90*10^{10}$	number of sequence comparisons
72	hours run time
~73,700	sequence comparisons / second
~ 13 GByte	of results

Table 1: Short summary of the computational effort for the yeast genome analysis. Note: the number of sequence comparisons given here are only the comparisons during the database scan, not included here are the additional performed comparisons done for multiple sequence alignments and for internal repeat searches.

During the three-day event, scientist located at more than 1000 sites worldwide immediately accessed the gene analysis results using the World Wide Web servers. with continuos and increasing access from the scientific community.
The results represent a unique consistent snapshot of the function prediction of yeast protein sequences, which will take a few month to analyze in detail. A first rough analysis however already showed new functional predictions for a few hundred proteins.
The use of powerful supercomputers in genome analysis permits the processing of huge amounts of raw data in days instead of months. It also allows to keep up with the current information growth by performing frequent re-analysis. The sheer compute power necessary to do all this is enormous. And it will continue to grow exponentially as the databases containing the raw data are doubling in size every 12 months.

References

[1] G. Casari, M. A. Andrade, P. Bork, J. Boyle, A. Daruvar, C. Ouzounis, R. Schneider, J. Tamanes, A. Valencia, and C. Sander, Challenging times for bioinformatics, Nature, vol. 376, pp. 647-648, 1995.

[2] M. Scharf, R. Schneider, G. Casari, P. Bork, A. Valencia, C. Ouzounis, and C. Sander, GeneQuiz: a workbench for sequence analysis, presented at ISMB-94 Seconds International Conference on Intelligent Systems in Molecular Biology, Stanford, California, USA, 1994.

[3] L. Wall and R. L. Schwartz, Programming perl. Sebastopol, CA: O'Reilly & Associates, Inc., 1990.

[4] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman, Basic local alignment search tool, J. Mol. Biol., vol. 215, pp. 403-410, 1990.

[5] T. Berners-Lee and D. Connolly, Hypertext Markup Language 2.0, http://www.w3.org/hypertext/WWW/MarkUp/html-spec/html-spec_toc.html, 1995.

[6] The Common Gateway Interface Specification, National Center for Supercomputing Applications at the University of Illinois at Urbana - Champaign , IL, USA., URL: "http://hoohoo.ncsa.uiuc.edu/cgi/interface.html".

[7] T. Etzold and P. Argos, SRS - an indexing and retrieval tool for flat file data libraries, Comput. Appl. Biosci., vol. 9, pp. 0-0, 1993.

[8] VRML Architecture Group, URL: "http://vag.vrml.org".

[9] H. Vollhardt, C. Henn, G. Moeckel, M. Teschner, and J. Brickmann, Virtual Reality Modeling Language in Chemistry, J. Mol. Graphics, vol. 13, pp. 368-372, 1995.

[10] GeneCrunch - Genome Supercomputing, SGI, Cortaillod, Switzerland, URL: "http://genecrunch.sgi.ch".

[11] G. Vriend, WHAT IF: a molecular modelling and drug design program, J. Mol. Graphics, vol. 8, pp. 52-56, 1990.

[12] PDB WWW Home Page, Protein Data Bank, Brookhaven National Laboratory, URL: "http://pdb.pdb.bnl.gov".