

View Point



Streamlining Drug Discovery Research by Leveraging Grid Workflow Manager*

Anirban Ghosh, Anirban Chakrabarti, Dheepak R.A., Shakeb Ali

Presently, pharmaceutical companies are under pressure to speed up time-to-market and reduce costs of producing novel drug molecules. One of the key enablers, which can be leveraged in various silos of this industry, to enhance productivity is grid computing. Grid allows linking up as many processors, storage and/or memory of distributed computers to make more efficient use of all available computing resources to solve large problems quickly. The benefits of grid computing include cost savings, decrease time to deliver results, enhance collaboration and harness existing computing resources. We have developed a workflow solution on grid called Discovery Research Workflow on Grid (DRWG) which helps automate and accelerate gene discovery research. DRWG enables the user to custom design a pipeline of compute intensive tasks which can be executed on heterogeneous platforms. The repetitive and complex set of tasks are efficiently managed by Grid Workflow Manager (GridWorM) which schedules, load levels and delegates tasks onto the available computing resources. This paper will discuss how high throughput functional annotation of genomic DNA sequences or profiling groups of protein sequences can leverage grid computing services.

**This work is partly supported by Life Sciences - Drug Discovery Informatics and Software Engineering and Technology Labs solution grant.*

1. INTRODUCTION

1.1. How Genomics Help Drug Discovery?

Presently, pharmaceutical companies are adopting targeted and rational drug discovery route. Traditional novel drug discovery programs were essentially based on the screening of a library of chemical compounds developed by a combinatorial chemistry approach. For this, a target protein molecule was essential for designing a drug. Only a small number of specific drug targets were available earlier. With the availability of human genome information, more targets have been identified, largely predicted by bioinformatics tools and later validated by molecular biology experiments. Moreover, the targets identified using the bioinformatics approach is more 'reliable'. This not only lead to a fewer number of drug candidates for pre-clinical testing, but also reduce the attrition rate of molecules in the clinical validation phase. The translation of genomics knowledge into drugs has conclusively established the importance of use of informatics in pharmaceutical research.

From the complete genome sequences of 261 organisms¹, there is immediate imperative to extract genuine new insights and discoveries from the genes and proteins that are coded for an organism. Bioinformatics has the potential to reduce cost and complexity of drug discovery projects and identify specific, selective targets and druggable leads for a therapeutic program.

1.2. Grid Services Opportunities in Drug Discovery

There are many examples in drug discovery which are currently leveraging data and compute grid services. Some of the salient examples are – a) high throughput screening for lead molecules against targets for cancer, b) fightaids@home by executing auto-dock to screen for suitable inhibitor to active site of HIV protease², c) identifying protein profiles from tandem mass spectrometry from serum samples of Thalassemia patients, d) simulating complex cellular models, e) determining statistical trends from profiles of micro-array gene expression, f) computational protein folding modeling, and g) predicting protein-protein interaction in a biochemical pathway.

Discovery research in biological and chemical sciences involves repeated execution of compute intensive applications like BLAST³, ClustalW⁴, or HMMER⁵ which form a part of the pipeline of tasks. In other cases, these research problems are solved by conducting a long computer simulation of a molecular system.

Each one of the research processes are executed on dedicated machines which results in long idle time and hence inflated budget. Either the manual intensive tasks are managed by trained specialists who are expensive, or scientists supervise the execution of the workflows resulting in ineffective use of the scientist time. In addition, there are overheads in terms of preparation time to initiate application and assimilation and presentation of scientific outcomes, which are all susceptible to human error. These challenges in discovery research can be suitably overcome by deploying grid services and help determine novel drug molecule fast and accurate.

1.3. Computational Workflows for Genomic Research

Quite unlike generic transactional workflows, computational workflow integrates a pipeline of scientific data management tasks. To accelerate and automate a complete research process, compute and data intensive workflow management provides speed, throughput, compute resource utilization, multi-application integration, diverse database access, and enable scientific collaboration. Shown in figure 1, high level research processes for target identification and lead optimization. The computational workflows to each can be formed by integrating the computational tasks underlying the process maps.

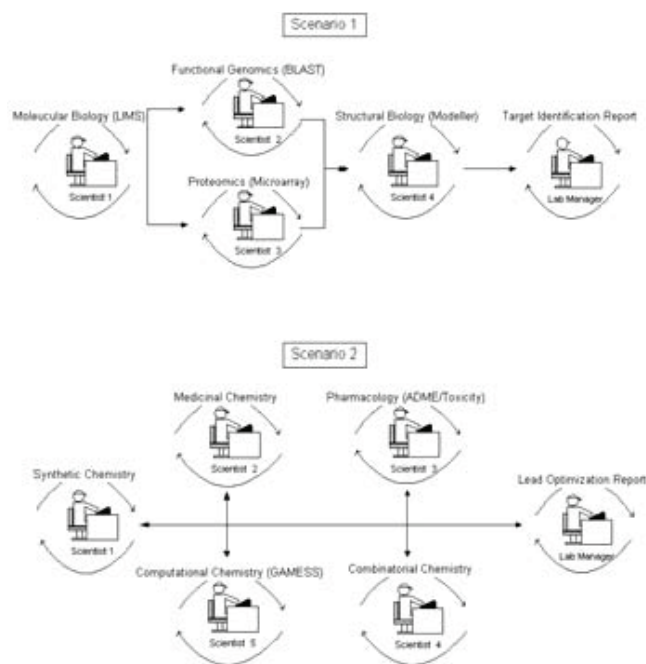


Figure 1: Representative target identification and lead optimization workflows

1.4. Grid Services for Computational Workflows

Grid Computing⁶ is defined as a mechanism to overcome heterogeneity of computing elements, operating systems, and also in terms of policy decisions and environment. A long-term vision of enterprise grid computing community is non dedicated seamless interoperability of different disparate systems which may be part of the same organization or different organizations. Grid computing is looked upon by many experts as a technology that can potentially change the world, like the Internet did. However, from the user's point of view, grid is nothing but a computer with huge amount of computing resource. There are workflows in the functional genomics⁷ domain and other silos of drug discovery research, which require grid services, to run compute intensive jobs through a workflow. Therefore, there is a need for process level workflow definitions to interact with the underlying heterogeneous grid infrastructure wherein the jobs comprising the workflow can be distributed across the infrastructure. This virtualization and load balancing results in improved efficiency, as the same infrastructure would support more load and hence lowering the overall total cost of ownership.

2. METHODS

2.1. Genomic Sequence Analysis for Target Identification

Sequencing projects obtain short nucleotide sequences or *Expressed Sequence Tags (ESTs)*⁸ which are mapped for its chromosomal location and putative gene function. ESTs have applications in the discovery of new human genes, mapping of the human genome, and identification of coding regions in genomic sequences. Determination of biological functions of each string of DNA sequence is important to understand its biological context e.g. whether they are involved in horizontal gene transfer, whether a cluster of protein sequence belong to microbial genomes, whether a collection of gene sequences belong to same biochemical pathway of the genome. All molecular findings are an important first step to identifying a drug target in any drug discovery research program.

The NCBI GenBank⁹, RefSeq BLAST¹⁰, SWISSPROT¹¹ and other data bases are updated with functional DNA, RNA and protein sequences. Most of these databanks grow at exponential rate. As of April 2004, there are over 44,575,745,176 base pair of DNA found in 40, 604, 319 sequences in the GenBank. Upon querying for similar nucleotide sequence from these databanks, the commonly occurring function description for all the matched sequences is studied. Once the function to the as yet unknown nucleotide sequence is assigned – the DNA sequence is annotated with a biological function. The same is true with protein sequences, one finds the similar protein sequences and all the homologous sequences are studied for common traits or a sequence profile is generated. Thus a whole suite of experiment and computational analysis is performed to annotate a gene or profile a protein sequences in the field of functional genomics.

2.3. GRID WORKFLOW MANAGER (GridWorM)

The **Grid Workflow Manager (GridWorM)** allows the user to submit the jobs through a workflow. The workflow allows the integration of applications with enterprise entities like web services, relational data bases, and decision making.

2.3.1. Technical Features of GridWorM

GridWorM provides the user with the interface to provide relationships among the jobs. The features that GridWorM support are:

- a. **Jobs Submissions:** Users can submit jobs using a Graphical User Interface developed as part of the GridWorM application. The user can submit jobs based on the BPEL specifications, which has become an industry standard now.
- b. **Job Relationships:** Users can provide complicated conditional relationships among the different jobs.
- c. **QoS Support:** The GridWorM workflow manager provides variety of QoS support including availability, trust levels¹² etc.
- d. **Infrastructure Support:** GridWorM supports infrastructure level support in terms of making queries to multitudes of databases like MySQL, Oracle (different versions), Postgres and so on. Later versions of GridWorM will also support integration with enterprise level messaging services like JMS, MQ etc.
- e. **Web Services integration:** Jobs can either be standalone applications or web services. Standalone applications can also interact with Web Services seamlessly.

2.3.2. GridWorM Architecture

The inter-relationship between different components of GridWorM is shown in figure 3, and brief description of each is described below:

GWLang: The GridWorM language is specifically designed for the workflow in mind. It is based on XML and has properties of Business Process Execution Language (BPEL) and Grid Services Flow Language (GSFL). GWLang combines the advantages of both BPEL and GSFL in a scalable manner. It inherits the relationship and QoS models from BPEL and it inherits the grid services model from GSFL. In addition, it also supports standalone applications, file management, and infrastructure level support like native database queries, opening remote shell (rsh) or remote copy (rcp) facilities. The unified grid model necessitated the development of the GWLang language which provides features not available in any of the existing job flow or business flow languages.

GWLang Generator: Another important component of the GridWorM is the GWLang parser. It is responsible for converting the user requirements from the GUI to the XML based GWLang language. The language is not exposed to the external users; however the user may choose to enter their requirements through the native GWLang also. The parser also converts the user requirements specified through other workflow specification language like BPEL. The parser is developed using Apache XMLBeans¹³ which allows the developer to access the full power of XML in a Java friendly way.

GridWorm Pre-parser: Pre-parser is added because of mapping dynamic Web Services and JDBC calls in GWLang.

GridWorM Manager: GridWorM manager manages the different state machines within the GridWorM. GridWorM manager receives the jobs from the GWLang pre-parser and instantiates a GridWorM state machine. It also generates a unique workflow before giving it to the state machines.

GridWorM State Machine: The state machine in GridWorM manages the states and relationships among the different jobs within a particular workflow. It uses the Web Service provided by MAGI to submit the job. It continually polls for the status of the submitted job, based on the submission id returned by MAGI.

Guided User Interface: The user can use the tools provided on the interface to create the workflow and upload relevant input data. Each of the applications can be loaded with its associated parameters on the workflow. Once the computational workflows are saved, they can be resubmitted with minimum or no changes.

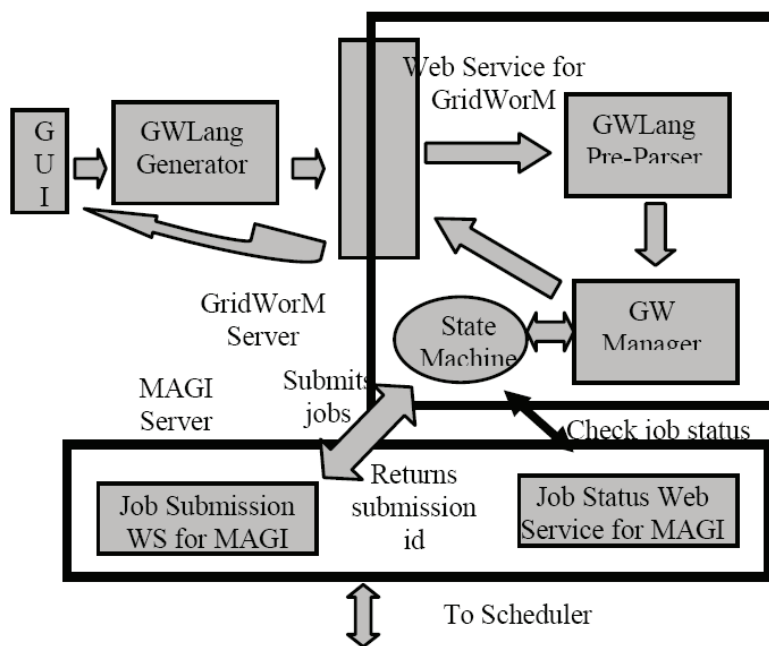


Figure 3: Schematic for GridWorM Architecture

3. RESULTS

3.1. Significance of the Case Study Reports

For each workflow submitted to the GridWorM a complete report is obtained as highlighted in Table 1. The abbreviated report on a fungal COX-2 protein sequence depicts that after searching the nr-db, it found 201 sequences to have exactly 51-60% sequence identity. The parser output identified 9 hits from another fungal family and selected all protein sequences better than 70% sequence for multiple sequence alignment. Upon pair-wise alignment of all sequences, a common profile is depicting in the report along with distances in the phylogenetic map. To summarize, this reporting facility can be tweaked to produce gene or protein family classification, in the way suitable for the scientists to derive significance from the raw data sets.

Table 1 Report highlighting the various data obtained executing various tasks of the workflow

Sections of the Report	Representative Data
Report ID: Date: User: Title	192.168.206.99: Mon Apr 11 2005 10:00; Dr. Anirban Ghosh; Report for annotating the Nucleic Acid/Protein Sequence
Details of the query nucleotide/protein sequence	sp P00411 COX2_NEUCR; Cytochrome c oxidase subunit2; EC 1.9.3.1; Neurospora crassa.
Program Detail: Package Used: Database	Blastall: BLASTp: nr-db
Closely lying sequence - Nucleotide ID: Protein ID: Length: Score: E-value: Identity: Start Seq: End Seq: Organism: Sequence	NP_074950.1:P20682: 250:476:e-133: 91%: 1: 250: Podospira anserine : MGLLFNNLIMNF
Simple Statistical Analysis: Total no. of HITS: Total no. of HITS between 71-80%: Quarter Percentile	250: 6: 0.0
Top three organism which has maximum number of hits	Candida Glabrata – 9
Sequence with identity greater than 70% included to generate a profile by ClustalW	gi 117030 sp P00411 COX2_NEUCR; gi 12408617 ref NP_074950.1;
Multiple Sequence alignment	MFFLINKLVMNLLNQVSVFINR ***.*.*** * :.*****

3.2. Performance

The system is tested using GridWorM 1.0 and MAGI 1.0¹². The prototype version of GridWorM has been developed which interacts with the MAGI 1.0 system to provide the desired end to end result to the user. The GridWorM system interacts with the MAGI system through Apache Tomcat 4.1. Final scheduling is handled by the CONDOR¹⁴ system.

We define the performance metrics for our study as Average Peak Utilization. We take the average of the utilizations of different machines for the period when all the machines were busy i.e. the grid system was running at its peak capacity. We needed to devise this metric as the traditional average CPU utilization will include the times when some machines were idle due to lack of jobs in the system.

The metric was tested on 2 dual proc XeonTM (2.8 GHz, 1 GB) and a single proc (3.2 GHz, 1 GB) Dell server connected within a 100Mbps LAN. The results were based on the 25 workflows, each consisting of applications like BLAST, ClustalW and their parsers and report applications. The average peak utilization (in percentages) is 99.5% on machine 1, 96% on machine 2 and 88% on machine 3, as shown in figure 4. This figure shows that the underlying grid infrastructure is utilized maximally and uniformly. The unequal utilization of different machines is because of the wide difference in execution times of the applications run through the GridWorM workflow manager. On a single processor 3.2 Ghz machine, the time taken to run 25 workflows is around 8.5 hours. While, it took a total time of 2.55 hours to execute 25 workflows submitted to the three machines. Therefore, the GridWorM along with MAGI shows a gain of 67% in terms of elapsed time.

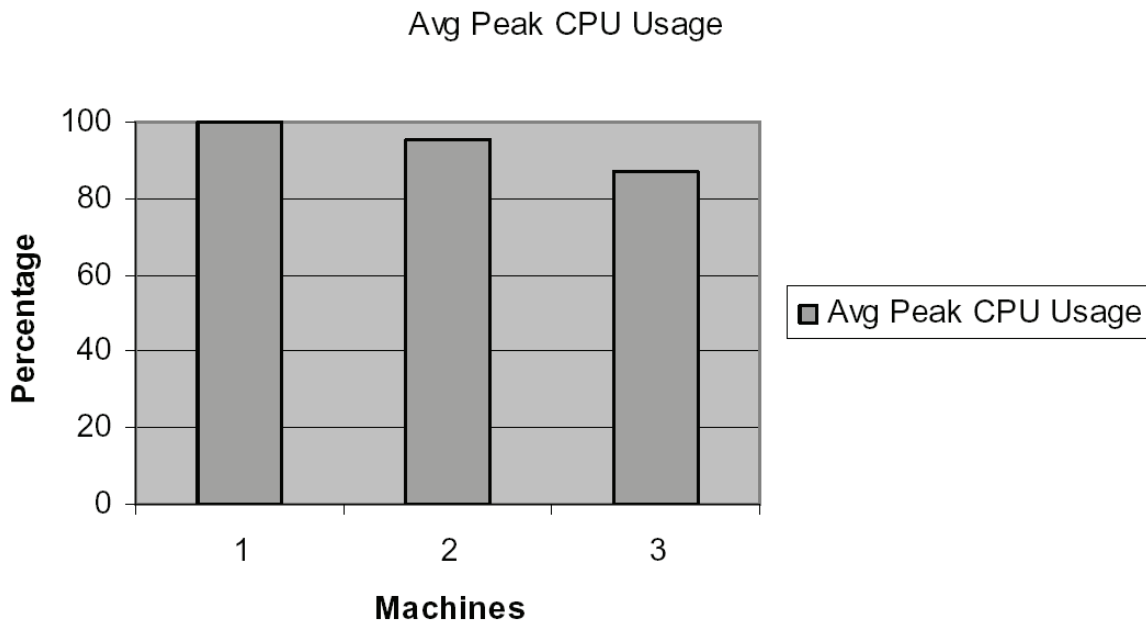


Figure 4: Plot of Average Peak usage of CPU of test machines

Another metric for performance of a GRID based workflow solution is latency which is the total time elapsed between job submission and obtaining the results. Table 2 shows the latency for various bulk sizes of the workflow when submitted on three machines as against only one.

Table 2 Latency of workflows submitted to one machine and three machine grid

No. of Workflows	Time Taken on 1 Machine	Time Taken on 3 Machines
10	3 hours 20 minutes	1 hour 28 minutes
20	6 hours 40 minutes	2 hours 5 minutes
25	8 hours 20 minutes	2 hours 33 minutes

3.3. Benefits of the Workflow

Workflow development and management enables faster execution of biological research activity such as functional annotation of genes or proteins and identifying homologous sequences. The ability to automate and accelerate the process without loss of quality in scientific output is the key benefit of this application. Considering the number of tasks whose execution is to be coordinated and completed, it is nearly impossible without the automation afforded by workflow management. The build, execution and reporting capabilities of the monolithic workflow manager can support large number of tasks over a distributed environment involving multiple heterogeneous platforms and multiple laboratories. The current capabilities of the workflow manager will be enhanced with increasing scientific challenges and demanding technological requirements in the near future.

Following are the benefits to the clients once the solution is deployed: (i) Workflow can be created and saved for future use or re-engineer a new variant, (ii) Automated data conversions between components of the workflow, (iii) Reusable life science informatics application components, (iv) Promote collaboration activity, (v) Resource utilization by efficient task distribution and re-routing, (vi) Enhance experimental and computational research efficiencies.

4. CONCLUSIONS

Custom build computational workflows builds, executes, manages and reports complex compute intensive workflows. Such workflows can support specific functional areas of discovery research e.g. gene identification, proteomics, compound screening, toxicology studies and pharmacogenetics. The workflow solution described in the paper (called DRWG) enables collaboration among various specialists, automates a pipeline of compute or data intensive work, and optimizes resource utilization. It can manage repeated high throughput tasks, supports heterogeneous platform and reduces the cycle time of the process and together enhance productivity of scientific research.

Acknowledgments

The authors are grateful to Shubhashis Sengupta, Sandeep Raju, Sanjay Martis and Deependra Moitra for discussion and help.

References

1. Bernal A, Ear U, Kyrpides, N. Genomes OnLine Database (GOLD): a monitor of genome projects worldwide. *Nucleic Acids Res* 2001; 29: 126-127.
2. <http://fightaidsathome.scripps.edu/index.html>
3. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ. Basic local alignment search tool. *J. Mol. Biol.* 1990; 215: 403-410.
4. Thompson JD, Higgins DG, Gibson TJ. CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res.* 1994 Nov 11; 22: 4673-80.
5. Durbin R, Eddy S, Krogh A, and Mitchison G. Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids. Cambridge University Press, 1998.
6. Foster I, Kesselman C, Tuecke S. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. *International J. Supercomputer Applications* 2001; 15(3): 327-344.
7. Hieter P, Boguski M. Functional genomics: it's all how you read it. *Science* 1997; 278(5338):601-602.
8. Adams MD, Kelley JM, Gocayne JD, Dubnick M, Polymeropoulos MH, Xiao H, Merril CR, Wu A, Olde B, Moreno RF, et al. Complementary DNA sequencing: expressed sequence tags and human genome project. *Science* 1991 Jun 21; 252(5013):1651-6.
9. Benson DA, Karsch-Mizrachi I, Lipman DJ, Ostell J, Wheeler DL. GenBank: update. *Nucleic Acids Res.* 2004 Jan 1; 32(Database issue):D23-26.
10. Pruitt KD, Tatusova T, Maglott DR. NCBI Reference Sequence (RefSeq): a curated non-redundant sequence database of genomes, transcripts and proteins. *Nucleic Acids Res.* 2005; 33(1):D501-D504.
11. Apweiler R, Bairoch A, Wu CH. Protein sequence databases. *Curr. Opin. Chem. Biol.* 2004; 8:76-80.
12. Gor K, Dheepak RA, Ali S, Alves L, Arurkar N, Gupta I, Chakrabarti A, Sharma A, Sengupta S. Scalable enterprise level workflow and infrastructure management in a grid computing environment. *CCGrid*, May 2005.
13. Apache XML Beans, xmlbeans.apache.com
14. Tannenbaum T, Wright D, Miller K, and Livny M. Condor - A Distributed Job Scheduler. The MIT Press, 2002.



For more information, contact askus@infosys.com

About Infosys

Many of the world's most successful organizations rely on Infosys to deliver measurable business value. Infosys provides business consulting, technology, engineering and outsourcing services to help clients in over 30 countries build tomorrow's enterprise.

For more information about Infosys (NASDAQ:INFY), visit www.infosys.com.