Map-Reduce Expansion of the ISGA Genomic Analysis Web Server

Chris Hemmerich[1], Adam Hughes[2], Yang Ruan[2,3], Aaron Buechlein[1], Judy Qiu[2,3], and Geoffrey Fox[2,3]
[1]The Center for Genomics and Bioinformatics
[2]Pervasive Technology Institute
[3]School of Informatics and Computing
Indiana University, Bloomington IN, USA
{ chemmeri, adalhugh, yangruan, abuechle, xqiu, gcf@indiana.edu}

***Abstract*— Biological sequence data can be subjected to a variety of analysis workflows to glean pertinent scientific insight. Recent advances in sequencing techniques have led to a deluge of biosequence data, which necessitates the use of high-performance computing resources in order to carry out analysis in a reasonable period of time. The tasks involved in creating and managing these computational jobs, though, can be daunting to typical biology researchers, which has lead to the emergence of portal software architectures that abstract many of the details in building and executing computational pipelines.**

This paper presents a brief overview of one of these genome annotation servers, Integrative Services for Genomics Analysis (ISGA), and then describes a simple extension to the underlying workflow system that leverages the powerful Twister Iterative Map-Reduce runtime for streamlined data-parallel job control and enhanced access to clusters, grids, and Cloud resources. The accompanying live demonstration will showcase ISGA's Workbench for submitting independent BLAST jobs as well as the use of the Twister interface to expand resource access for this utility.

Keywords-biosequence; bioinformatics; ISGA; BLAST; Twister; map-reduce; genome annotation

I. INTRODUCTION

Modern high-throughput biosequencing instruments are capable of generating millions of sequence reads in a single run spanning several days, leading to a proliferation of raw data which must be analyzed in order for scientists to draw meaningful conclusions [1]. The processes of biosequence analysis generally require the use of multiple software packages, executed in a particular order, to complete [2]. Many of these tools are computationally intensive, requiring the use of high-performance computing (HPC) resources in order to achieve acceptable performance in relation to the rate of new data production. In addition, tedious manual tasks are often required to set up and manage the jobs in a typical annotation pipeline. These circumstances have led to the development of portal architectures, which abstract many of the details of job creation and data management, freeing researchers to focus more on understanding the science of their results than on manipulating large data sets [4][5].

This paper presents a brief overview of one such portal for bioinformatics pipelines, Integrative Services for Genomics Analysis (ISGA) [3], and then describes a simple extension to the underlying workflow system (TIGR Workflow) that leverages the powerful Twister Iterative Map-Reduce runtime for streamlined data-parallel job control and enhanced access to clusters, grids, and Cloud resources. The accompanying live demonstration will showcase ISGA's Workbench that allows users to run BLAST jobs *a la carte*, independent of any pre-defined workflow, to perform the basic sequence alignment that is vital to many varied sequence analyses. We will also show how Twister can be invoked through existing ISGA interfaces to automate the distribution of input data for such BLAST jobs, which are inherently data-parallel, and to allow access to nearly any type of HPC platform.

.

II. ISGA

The Center for Genomics and Bioinformatics at Indiana University has developed a web-based server for bioinformatics pipeines, such as prokaryotic annotation, Integrative Services for Genomics Analysis (ISGA), which is built on the Ergatis workflow system [3]. While Ergatis is a flexible and powerful system for building and managing bioinformatics pipelines, its complexity makes it somewhat inaccessible to typical biologists [4]. ISGA reduces this technical barrier by providing access to the powerful analysis tools exposed by Ergatis through intuitive interfaces, allowing biologists to customize and execute workflows with a minimum of tedium.

Ergatis itself is built upon the TIGR Workflow system, which supports access to HPC resources through one of two Distributed Computing Environments (DCE), namely Condor and Sun Grid Engine [4]. Consequently, ISGA installations are currently limited to HPC machines running one of these two DCEs, as shown in Fig. 1. Because of the large quantities of input data involved in sequencing

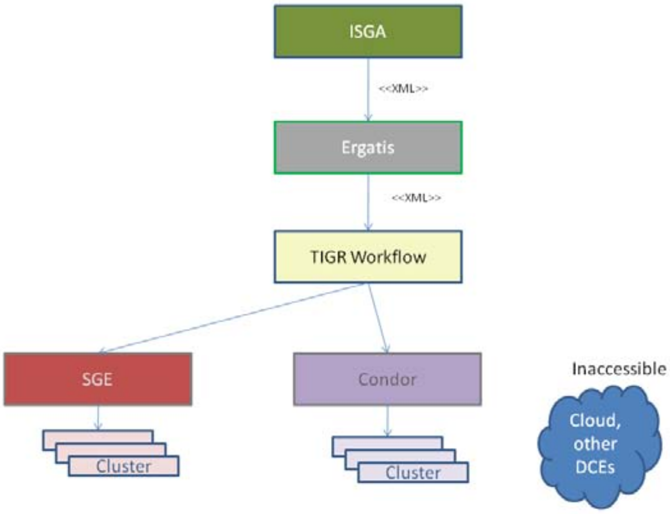


Figure 1. Schematic of current ISGA architecture, with limited access to HPC resources.

projects and the pleasingly parallel nature of many steps in a typical analysis pipeline, expanding access to other HPC flavors, especially Cloud platforms, holds the potential to significantly increase the throughput of current and future analysis pipelines. As a step toward bridging this gap, an interface between Twister and the underlying TIGR Workflow framework has been developed, as described below.

III. TIGR-TWISTER INTEGRATION

Twister is an iterative map-reduce runtime that uses the Narada brokering system to spawn and manage jobs on a variety of HPC platforms [6]. In the context of ISGA, Twister is advantageous in at least two respects. First, many of the most important analysis components in genome annotation pipelines are inherently data-parallel. That is, large data sets, such as FASTA files of sequence reads, can be partitioned and distributed among many compute nodes (on grids, clusters, or Clouds), and the same algorithm can be run independently on each partition of data. Then, when the jobs are finished, the individual results can simply be concatenated to form a complete result set. One typical example of such an execution scheme is the utilization of BLAST to carry out sequence alignment studies. This standard map-reduce pattern can be tedious to implement manually or through scripting languages, but is handled seamlessly by Twister.

In addition to providing automated map-reduce control, Twister also has the ability to spawn jobs on nearly any existing computer system, including various Cloud platforms. By embedding this capability into the the ISGA-Ergatis-TIGR Workflow stack, Twister can greatly expand the resource pool available to an ISGA installation.

In order to take advantage of these powerful features of Twister, an interface between Twister and the low-level TIGR-Worfklow system has been developed, as shown in Fig. 2. This interface presents a new Twister DCE specification that uses standard TIGR Workflow syntax, expanded to include some simple Twister configuration parameters. This DCE is supported by a generic map-only implementation of Twister, which accepts any command-line argument and a set of configuration parameters. Based on these parameters, Twister then distributes the associated input to the appropriate compute resources and executes the given software on each of the selected nodes. Upon job completion or failure, Twister notifies the parent TIGR Workflow system of its status, and Workflow can then proceed with post-processing steps.

This implementation allows consumers of TIGR Workflow functionality to take full advantage of both Twister's map-reduce functionality and its ability to access multiple computing platforms. Because the TIGR-Twister interface may be invoked through a standard TIGR DCE specification, minimal changes to existing workflows, such as those orchestrated through Ergatis, are required in order to leverage these enhanced functionalities.

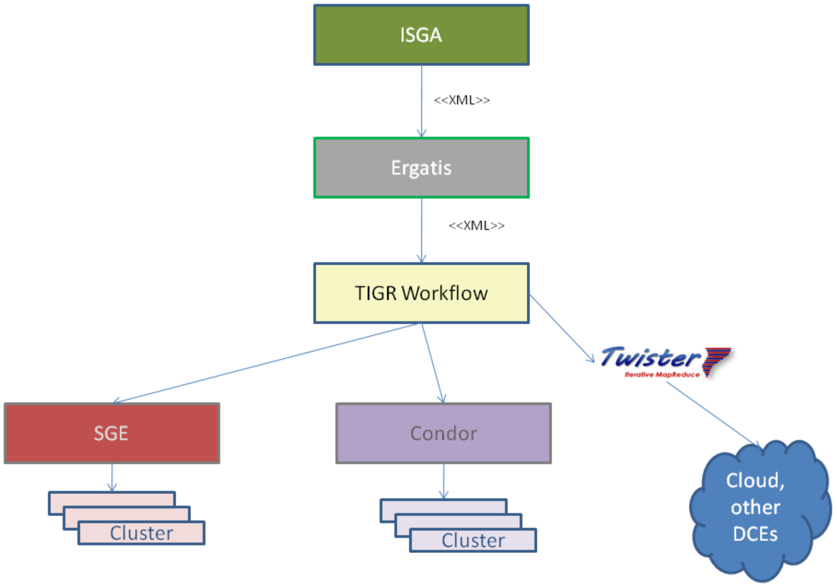


Figure 2. Schematic of proposed ISGA architecture, with Twister interface enabling access additional cluster, grid, and Cloud platforms.

IV. Results

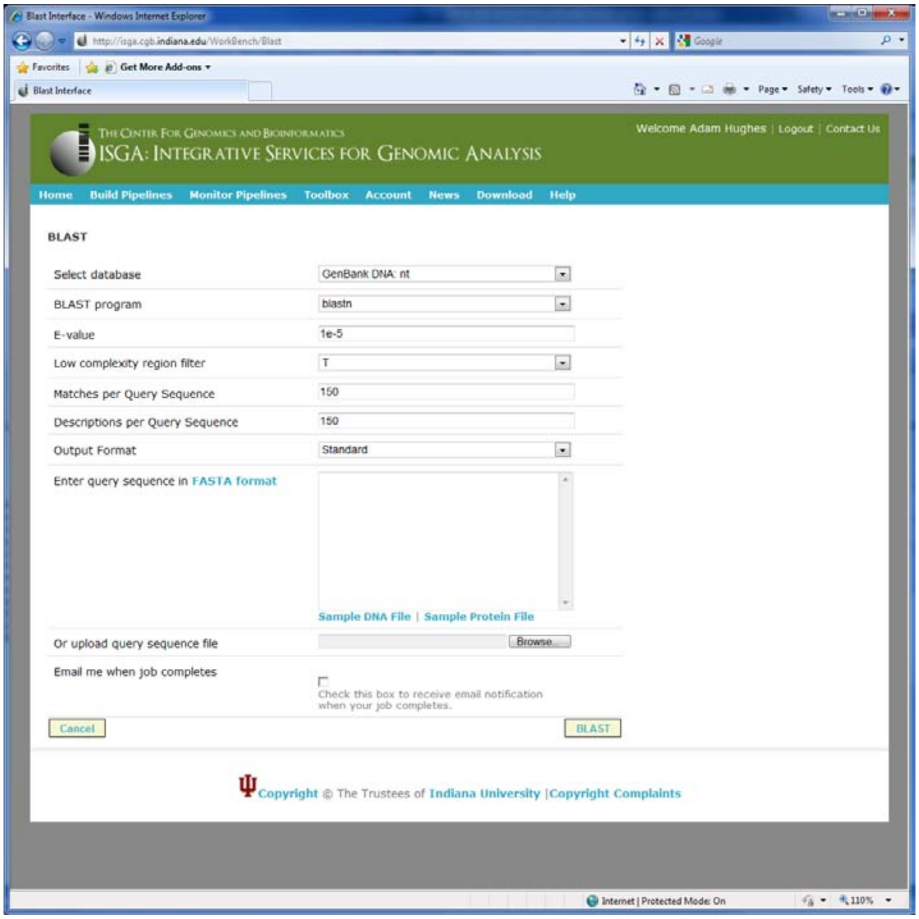
To this point, the TIGR-Twister interface has been tested for use with the ISGA Workbench BLAST utility, which provides researchers with a simple web interface, as shown in Fig. 3, for submitting BLAST jobs independent of any other analysis tools or pipeline implementations. BLAST was chosen for the initial implementation because it is widely used as a component in bioinformatics pipelines and as a standalone tool and because it is computationally expensive. This deployment has been tested on the ISGA development cluster, on a local high-memory cluster, and on Polar Grid machines. The accompanying live demonstration will showcase the BLAST Workbench tool using sample job submissions and walk through examples of using the TIGR-Twister configuration files to access multiple HPC platforms.

V. Future Directions

The use of TIGR-Twister for BLAST Workbench jobs serves as a proof-of-concept for a generic, map-only Twister implementation as a means for accessing multiple computing environments to conduct biosequence analysis. In order to fully exploit these capabilities, this paradigm will be further investigated as a means for accessing other HPC systems, notably distributed grids (TeraGrid, FutureGrid, etc.) and Cloud platforms. In addition, the use of TIGR-Twister in existing and future ISGA pipelines will be explored.

VI. Conclusions

The proliferation of biosequence data necessitates the use of high-performance computing resources to perform analyses in a timely fashion. Managing hundreds or thousands of computational jobs and data files associated with such analyses is a daunting task, which has spawned the creation of automated pipeline systems such as ISGA. Integrating the powerful Twister iterative map-reduce runtime with the existing ISGA framework promises to expand HPC access and enable more seamless map-reduce job control, ultimately bringing more computational power to biologists.


Blast Interface - Windows Internet Explorer
http://isga.cgb.indiana.edu/WorkBench/Blast
Google
Favorites
Get More Add-ons
Blast Interface
Page
Safety
Tools
THE CENTER FOR GENOMICS AND BIOINFORMATICS
ISGA: INTEGRATIVE SERVICES FOR GENOMIC ANALYSIS
Welcome Adam Hughes | Logout | Contact Us
Home
Build Pipelines
Monitor Pipelines
Toolbox
Account
News
Download
Help
Sample DNA File | Sample Protein File
Browse...
Cancel
BLAST
Copyright © The Trustees of Indiana University |Copyright Complaints
Internet | Protected Mode: On
110%

BLAST

Select database	GenBank DNA: nt
BLAST program	blastn
E-value	1e-5
Low complexity region filter	T
Matches per Query Sequence	150
Descriptions per Query Sequence	150
Output Format	Standard
Enter query sequence in FASTA format	
Or upload query sequence file	
Email me when job completes	□ Check this box to receive email notification when your job completes.

Figure 3. Screenshot of ISGA Workbench BLAST interface.

ACKNOWLEDGMENT

This work is supported by the National Institutes of Health under grant 5 RC2 HG005806-02.

REFERENCES

[1] M. Marguilies et al., “Genome sequencing in microfabricated high-density picolitre reactors,” *Nature*, vol. 441, no. 7089, . pp. 376-380, Sep. 2005.

[2] B. Richter and D. Sexton, “Managing and Analyzing Next-Generation Sequence Data,” *PLoS Computational Biology*, vol. 5 no. 6, Jun. 2009.

[3] C. Hemmerich et al., “An Ergatis-based prokaryotic genome annotation web server,” *Bioinformatics*, vol. 26, no. 8, pp. 1122-1124, Apr 2010.

[4] J. Orvis, et al., “A web interface and scalable software system for bioinformatics workflows,” *Bioinformatics*, vol. 26, no. 12, pp, 1488-1492, Jun. 2010.

[5] O. Kaiser, et al., “Whole genome shotgun sequencing guided by bioinformatics pipelines – an optimized approach for an established technique,” *Journal of Biotechnology*, vol. 106, no. 2-3, pp. 121-133, Dec. 2003.

[6] J. Ekanayake, et al., “Twister: a runtime for iterative mapreduce,” Proceedings of the First International Workshop on MapReduce and its Applications of ACM HPDC 2010, June 20-25, 2010, Chicago, Illinois.