Technologies for Genomic Medicine

MaPSeq, A Computational and Analytical Workflow Manager for Downstream Genomic Sequencing

A RENCI TECHNICAL REPORT

TR-14-03

renci

CONTACT INFORMATION

Jason Reilly
Telephone: 919.445.9686
Email: jdr0887@renci.org

List of Technical Terms and Websites

Apache™ ActiveMQ, activemq.apache.org

Apache™ CXF RESTful (Representational State Transfer), www.ibm.com/developerworks/webservices/library/ws-restful

Apache™ CXF SOAP (Simple Object Access Protocol), axis.apache.org/axis2/java/core/docs/soapmonitor-module.html

Apache™ karaf software, karaf.apache.org

Apache™ OpenJPA (Java Persistence API), openjpa.apache.org

APIs (application programming interfaces), www.webopedia.com/TERM/A/API.html

CASAVA (Consensus Assessment of Sequence and Variation), www.illumina.com/software/genome_analyzer_software.ilmn

CPU (Central Processing Unit)

daemons, en.wikipedia.org/wiki/Daemon_computing

fastq files, www.broadinstitute.org/gatk/gatkdocs/org_broadinstitute_sting_gatk_walkers_qc_FlagStat.htm

FlagStat files, www.broadinstitute.org/gatk/gatkdocs/org_broadinstitute_sting_gatk_walkers_qc_FlagStat.html

Glidein, www.cl.cam.ac.uk/manuals/condor-V6_8_3-Manual/5_4Glidein.html

GlideinWMS, www.uscms.org/SoftwareComputing/Grid/WMS/glideinWMS/doc.prd/index.html

HTCondor™, research.cs.wisc.edu/htcondor

JSON (JavaScript Object Notation), www.json.org

MaPSeq (Massively Parallel Sequencing system)

OSG (Open Science Grid), www.opensciencegrid.org

OSGi (Open Science Grid initiative), www.osgi.org/Main/HomePage

PostgreSQL (Structured Query Language) database, www.postgresql.org

SSH (Secure Shell) technology, www.webopedia.com/TERM/S/SSH.html

SOA (Service-Oriented Application), www.opengroup.org/soa/source-book/soa/soa.htm#soa_definition

TeraGrid™ Science Gateways Program, info.teragrid.org

Introduction

Genomic medicine is advancing at a remarkably fast past, with major technological achievements such as next-generation genomic sequencing producing large-scale genomic data sets within a reasonable timeframe and cost (Mardis, 2008; Horvitz and Mitchell, 2010; Koboldt et al., 2010; Kahn, 2011). Yet large-scale computation on the gigabyte- to petabyte-scale data sets that are generated from massively parallel genomic sequencing projects remains enormously challenging. Indeed, the National Consortium for Data Science (Ahalt et al., 2014), the Global Alliance to Enable Responsible Sharing of Genomic and Clinical Data (2013), and the BD2K Data and Informatics Working Group, National Institutes of Health BD2K Initiative (2012) have recognized computational and analytical challenges as significant barriers to the advancement of genomic medicine.

Herein, we describe the Massively Parallel Sequencing (MaPSeq) system—an open source, secure, centralized, grid-based SOA that facilitates, manages, and executes the complex, project-specific, computational and analytical downstream steps involved in high-throughput genomic sequencing.

The Team*

Jason Reilly, RENCI Senior Research Software Developer; **Stanley Ahalt**, PhD, RENCI Director and Professor, Department of Computer Science, UNC Chapel Hill; **Karamarie Fecho**, PhD, Medical and Scientific Writer for RENCI; **Corbin Jones**, Associate Professor, Department of Biology, UNC Chapel Hill; **John McGee**, RENCI Director of Cyberinfrastructure; **Jeff Roach**, Senior Scientific Research Associate, Research Computing, UNC Chapel Hill Information Technology Services; **Charles P. Schmitt,** PhD, RENCI Chief Technical Officer and Director of Informatics and Data Science; and **Kirk C. Wilhelmsen**, MD, PhD, RENCI Director of Biomedical Research, RENCI Chief Domain Scientist for Genomics, and Professor, Departments of Genetics and Neurology, UNC Chapel Hill.

TR-14-03 in the RENCI Technical Report Series, May 2014

Renaissance Computing Institute
University of North Carolina
at Chapel Hill
919-445-9640
www.renci.org
@renci

About RENCI

RENCI is an institute of the University of North Carolina at Chapel Hill that develops and deploys advanced technologies to enable research discoveries and practical innovations. The institute was launched in 2004 as a collaborative effort involving UNC Chapel Hill, Duke University, and North Carolina State University. For more information, see www.renci.org.

*Jason Reilly serves as the technical lead on MaPSeq; Kirk Wilhelmsen serves as Principle Investigator and Director of RENCI's Biomedical Research division, which is leading the development of MaPSeq; all other team members are listed alphabetically.

MaPSeq

MaPSeq was developed by RENCI in collaboration with Information Technology Services (ITS) Research Computing at the University of North Carolina at Chapel Hill (UNC), the UNC High-Throughput Sequencing Center, and Lineberger Comprehensive Cancer Center to address a need for reliable and efficient high-throughput informatics processing of genomic data for both large and small research projects. The design of MaPSeq was informed by previous work with the OSG and TeraGrid™ Science Gateways Program. It includes a plugin architecture that provides researchers with a framework to facilitate the construction, deployment, and execution of sequence analysis workflows. MaPSeq is designed to make simultaneous, opportunistic use of multiple institution-wide and cloud-based computational resources from across administrative domains at UNC.

MaPSeq evolved from a Science Portal that RENCI had created to assist researchers with large-scale, high-throughput, computational science problems (McGee, 2010). The Science Portal served as a computational science platform, and it had several desirable features: (1) it was user-friendly and accessible via a web browser and a variety of APIs; and (2) it provided high throughput, in that it triaged job requests to multiple, high-capacity, computational clusters (namely, OSG, TeraGrid™, and clusters available at RENCI and UNC's Department of Computer Science). At the core of the Science Portal was HTCondor™, which is a high-throughput computing platform used extensively in academic research. HTCondor™ also is at the core of the OSG and services hundreds of millions of compute jobs, data transfers, and CPU hours per year (display.grid.iu.edu).

MaPSeq was developed using a strategy and structure similar to the Science Portal, as a plugin-based, centralized, SOA environment comprised of a database-backend, user-friendly web services, multiple programmatic interfaces, server-side applications, and persistence (Figure 1). As with the Science Portal, HTCondor™ provides the underlying high-throughput computing capabilities. MaPSeq utilizes SSH technology for authentication and data management and transfer.

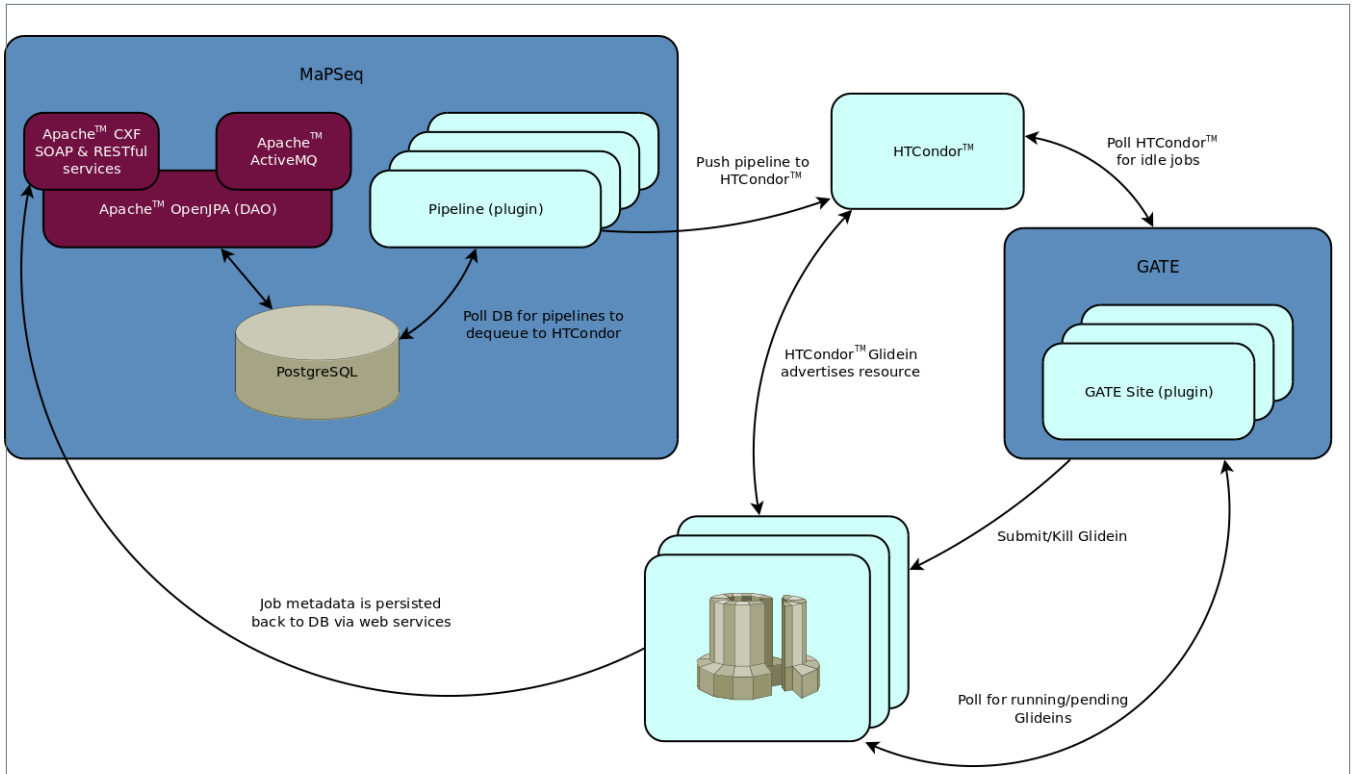

Figure 1. A schematic showing the architecture of MaPSeq, with arrows depicting the flow of information. DB = database; GATE = Gate Access Triage Engine; RESTful = Representational State Transfer; SOAP = Simple Object Access Protocol.

MaPSeq Operations Overview

To provide an overview of MaPSeq operations, suppose a client wishes to process the output from a high-throughput genomic sequencing run. The client will use a web service to send the job request via a JSON-formatted message to Apache™ ActiveMQ. The message specifies the data (e.g., format, size) and the workflow pipeline needed to process the data. ActiveMQ evokes SSH technology to determine if the object layer (i.e., the input raw sequencing data) in the message is legitimate for subsequent pipeline workflow parsing (e.g., Does the file contain sequencing metadata?). If so, then MaPSeq uses CASAVA software to transform the raw sequencing data (typically ~500 gigabytes/project) into compressed fastq files. In a single-end run, one compressed fastq file is produced per sample. In a paired-end run, two compressed fastq files are produced per sample.

Software sustainability promoted through workflow sub-pipelines

Example: Hello World

- Echo1 "Hello" → file1 (3rd party library)
- Echo2 "World" → file2
- Cat file1 file2

→ Hello World

Example: Genomic Research Project

Echo1 "Alignment sub-pipeline" → file1

Echo2 "Variant calling sub-pipeline" → file2

Cat file1 file2

→Alignment + Variant calling]

The two fastq files share the same base name and thus can be matched against each other. The metadata for the fastq files are stored in a PostgreSQL database. MaPSeq then activates the workflow run attempts, which persist in a queue in the PostgreSQL database via Apache™ OpenJPA. Note that the pipeline workflows are comprised of the detailed, often complex, computational steps required for tailored downstream analysis of sequencing data (e.g., sequence alignment, variant calling).

The pipeline workflows themselves continuously check the PostgreSQL database for pipeline run attempts. When pipeline run attempts are identified by the workflow pipelines, the pipeline is dequeued and sent to the HTCondor™ meta-scheduler. HTCondor™ sequentially reads each task in the dequeued pipeline, with a specific task being read only after the previous one has been completed. HTCondor™ reads both the job attributes (e.g., CPU and memory needs, transfer file needs, etc.) and the machine/cluster needs (e.g., availability, computing power, etc.) and matches the two using OSG assumptions/premises. HTCondor™ can opportunistically take advantage of multiple clusters. Of note, HTCondor™ does not execute jobs; rather, it catalogs job tasks and machine/cluster needs, matches the results, and then sits idle.

To promote software reusability, the pipeline workflows are typically broken into smaller computational sub-pipelines, with each sub-pipeline modified and/or reused as needed. Pipeline workflow tasks are dictated by the needs of a given project and defined by the researcher; thus, pipeline workflows can be tailored, revised, and reused.

GATE (Grid Access Triage Engine)

GATE serves as a sister technology for MaPSeq. It is also plugin-based and provides a dynamically expanding and contracting set of compute resources available to service MaPSeq computational work across a distributed set of computational clusters. GATE is conceptually modeled after GlideinWMS, which provides critical resource elasticity services for the OSGi and manages the vast majority of OSG job submissions.

GATE runs in the background to determine if a Glidein, or temporary addition of one or more grid resources, is needed for a given pipeline workflow. GATE continuously monitors HTCondor™ for idle jobs. If there are any idle jobs, then GATE profiles clusters for availability and, after identifying available clusters, submits an HTCondor™ instance on a compute node to a remote batch scheduler, which then registers back to the initial instance of HTCondor™. The local instance of HTCondor™ then executes the catalogued matchmaking and activates the pipeline jobs on the remote compute node. In this manner, GATE enables opportunistic use of all available clusters to ensure computational efficiency.

Of importance, metadata about a given job's progress persists from the compute node to the PostgreSQL database. This allows the client to send a message to the web-based Apache™ CXF SOAP and RESTful (Rodriguez, 2008) services, with a request to retrieve

information regarding the status of a job; the web services can then pull that information from the database and send it back to the client. In addition, the pipelines house their own web services via MaPSeq's web services (i.e., Apache™ CXF SOAP Monitor and RESTful). For example, FlagStat files collect metadata on a pipeline's analytical results, including simple statistics, such as the % complete reads, % missing data, and number of duplicate reads; this information persists back to the database, which allows the client to send a request to MaPSeq's web services to access the information.

Finally, MaPSeq was designed such that clients can create, modify, and deploy their own pipeline workflows without MaPSeq administrator involvement. In particular, MaPSeq and GATE use an instance of Apache™ karaf software to run lightweight plugin tools using the OSGi framework. SSH, running with daemons, can be used to access the karaf container for MaPSeq and GATE and deploy a new or modified pipeline for any particular project. Thus, the researcher or developer of a pipeline workflow can directly control the pipeline itself.

Conclusion

MaPSeq is an open source, secure, centralized, grid-based, SOA that facilitates, manages, and executes the complex, project-specific, downstream analytical steps involved in high-throughput genomic sequencing.

Key Features:

- Architecture is open source
- System requires minimal user intervention after system configuration
- Multiple, remote computational clusters are accessed opportunistically
- Software reusability is promoted through sub-pipelines
- Pipeline workflows can be tailored, modified, and updated as needed
- Pipeline workflows can house web services
- Pipeline workflows can be revised and deployed by clients, thus minimizing administrator burden

Underlying Software and Technologies:

MaPSeq/GATE:

- Apache™ ActiveMQ
- Apache™ CXF RESTful
- Apache™ CXF SOAP
- Apache™ karaf
- Apache™ OpenJPA
- CASAVA
- HTCondor™
- PostgreSQL
- Secure Shell Technology

Local Computational Clusters Currently Accessed by MaPSeq/GATE:

KillDevil (ITS Research Computing):

- Dell blade-based Linux Cluster
- 604 Compute Nodes: 48GB RAM
- 68 Compute Nodes: 96GB RAM
- 2 Compute Nodes: 1TB RAM
- 32 GPU compute nodes with 64 Nvidia Tesla GPU cards

Kure (ITS Research Computing):

- 2.2 PB Isilon system
- HP blade-based Linux Cluster
- 136 Compute Nodes: 48GB RAM
- 80 Compute Nodes: 72GB RAM
- 2 Compute Nodes: 96GB RAM
- 3 Compute Nodes: 192GB RAM
- Infiniband 4x QDR

BlueRidge (RENCI)

- Dell blade-based Linux Cluster
- 128 Dell PowerEdge m610 blades (1024 cores total)
- 32 Dell PowerEdge m610 blades (384 cores total)
- 2 NVidia Teslas s1070-500
- 2 Dell PowerEdge R910 4 x 2.00Ghz Intel Nehalum-EX, 8 core, 1TB 1066Mhz memory

Impact

- Three installations of MaPSeq have been deployed at UNC, with ~30 active pipelines, ~21,000 samples processed in 2013 alone, and 250–300 TB of data in MaPSeq's PostgreSQL database.
- Currently supports numerous research programs, including: (1) National Institute on Drug Abuse–funded NIDASeq, "Deep Sequencing Studies for Cannabis and Stimulant Dependence" (Dr. Kirk Wilhelmsen, PI), which is conducting whole genome sequencing of ~5,500 patient samples; (2) National Human Genome Resource Institute–funded NCGENES, "North Carolina Clinical Genomic Evaluation by NextGen Exome Sequencing" (Dr. James Evans, PI), which is conducting whole exome sequencing of >2,000 patient samples drawn from multiple disease categories; (3) National Institute of Child Health and Development–funded NC Nexus, "North Carolina Newborn Exome Sequencing and Newborn Screening Disorders" (Dr. Cynthia Powell, PI), which aims to conduct whole exome sequencing on 400 patient samples; and (4) CALGB, Cancer and Leukemia Group B (Dr. Charles Perou, PI), which comprises >900 samples.

Acknowledgements

This project was conceptualized and funded by RENCI and the UNC High-Throughput Sequencing Facility, in collaboration with UNC's Information Technology Services Research Computing and the Lineberger Comprehensive Cancer Center, and with additional funding from the National Institutes of Health (1R01-DA030976-01, 1U01-HG006487-01, 5UL1-RR025747-03, 1U19-HD077632-01, 1U01-HG007437-01).

Karen Green provided editorial and design support for the preparation of this technical report.

How to reference this technical report:

Reilly, J., Ahalt, S., Fecho, K., Jones, C., McGee, J., Roach, J., Schmitt, C., and Wilhelmsen, K. (2014): Technologies for Genomic Medicine: MaPSeq, A Computational and Analytical Workflow Manager for Downstream Genomic Sequencing. RENCI, University of North Carolina at Chapel Hill. http://dx.doi.org/10.7921/G0VD6WCF

renci
RESEARCH \ ENGAGEMENT \ INNOVATION

References and Resources

Ahalt S, Bizon C, Evans J, Erlich Y, Ginsberg G, Krishnamurthy A, Lange L, Maltbie D, Masys D, Schmitt C, Wilhelmsen K. Data to Discovery: Genomes to Health. A White Paper from the National Consortium for Data Science; 2014. RENCI, University of North Carolina at Chapel Hill. dx.doi.org/10.7921/G03X84K4. [Accessed Febuary 4, 2014]

Apache™ ActiveMQ. (An open source message and integration patterns server.) activemq.apache.org. [Accessed January 6, 2014]

Apache™ karaf. (An OSGi-based runtime application that provides a lightweight container onto which various plugin applications can be deployed.) karaf.apache.org. [Accessed January 6, 2014]

Apache™ OpenJPA (Java Persistence API). (A software program that provides a persistence layer within any JAVA-compliant container and many other lightweight frameworks.) openjpa.apache.org. [Accessed January 6, 2014]

Apache™ CXF RESTful (Representational State Transfer). (A lightweight, web-based utility to provide a messaging service between a remote client and a database.) axis.apache.org/axis2/java/core/docs/rest-ws.html. [Accessed January 6, 2014]

Apache™ CXF SOAP (Simple Object Access Protocol). (A web-based utility to provide a comprehensive messaging service between a remote client and a database.) axis.apache.org/axis2/java/core/docs/soapmonitor-module.html. [Accessed January 6, 2014]

APIs (Application Programming Interfaces). (A set of protocols for assembling software to create graphical user interfaces.) www.webopedia.com/TERM/A/API.html. [Accessed January 6, 2014]

CASAVA (Consensus Assessment of Sequence and Variation). (A software program that converts raw sequencing data into a variety of formats used for downstream analysis.) www.illumina.com/software/genome_analyzer_software.ilmn. [Accessed January 6, 2014]

Daemon. (A software program that runs as a background application disconnected from the user.) en.wikipedia.org/wiki/Daemon_%28computing%29. [Accessed January 6, 2014]

Data and Informatics Working Group, National Institutes of Health BD2K Initiative. NIH Request for Information: Management, integration, and analysis of large biomedical datasets. Analysis of public comments, 2012. NOT-OD-12-032. acd.od.nih.gov/DIWG_RFI_FinalReport.pdf. [Accessed October 31, 2013]

Fastq format. (A file format developed by the Sanger Institute for efficientl coding of sequencing data.) www.maq.sourceforge.net/fastq.shtml. [Accessed January 6, 2014]

FlagStat. (A software tool that continuously collects basic statistics on a given data set without user intervention.) www.broadinstitute.org/gatk/gatkdocs/org_broadinstitute_sting_gatk_walkers_qc_FlagStat.html. [Accessed January 6, 2014]

Glidein. (A tool to enable the temporary recruitment of one or more remote grid resources to a local HTCondor™ pool.) www.cl.cam.ac.uk/manuals/condor-V6_8_3-Manual/5_4Glidein.html. [Accessed January 6, 2014]

GlideinWMS (Workflow Management System). (A WMS designed to work on top of the HTCondor™ WMS to provide easy access to grid resources.) www.uscms.org/SoftwareComputing/Grid/WMS/glideinWMS/doc.prd/index.html. [Accessed February 18, 2014]

Global Alliance to Enable Responsible Sharing of Genomic and Clinical Data. Creating a Global Alliance to enable responsible sharing of genomic and clinical data. A white paper. Global Alliance; 2013. www.broadinstitute.org/files/news/pdfs/GAWhitePaperJune3.pdf. [Accessed June 6, 2013]

Horvitz E, Mitchell T. From data to knowledge to action: a global enabler for the 21st century. Computing Community Consortium, v. 11. September 11, 2010. www.cra.org/ccc/docs/init/From_Data_to_Knowledge_to_Action.pdf. [Accessed March 15, 2013]

HTCondor™. (A workload management system for computationally intensive jobs, with meta-scheduling capability.) research.cs.wisc.edu/htcondor. [Accessed January 6, 2014]

IDE (Integrated Development Environment). (A software application that provides a software programming environment.) www.webopedia.com/TERM/I/integrated_development_environment.html. [Accessed January 6, 2014]

JSON (JavaScript Object Notation) format. (A lightweight, Java-based, user friendly, data-interchange format.) www.json.org. [Accessed January 6, 2014]

Kahn SD. On the future of genomic data. Science. 2011;331(6018):728–728.

Koboldt DC, Ding L, Mardis ER, Wilson RK. Challenges of sequencing human genomes. Brief Bioinf. 2010;11(5):484–498.

Mardis ER. Next-generation DNA sequencing methods. Annu Rev Genomics Hum Genet. 2008;9:387–402.

McGee J. Effectively utilizing US national cyberinfrastructure resources. Presented to the Center for Theoretical & Mathematical Sciences at Duke University, June 1, 2010. www.phy.duke.edu/~muller/ctms/Lectures/McGee_100601.pdf. [Accessed January 7, 2014]

OSG (Open Science Grid). (A global community of computational experts whose mission is to provide opportunistic use of existing software and service resources to users and resource providers.) www.opensciencegrid.org. [Accessed January 6, 2014]

OSGi (Open Science Gateway initiative) technology. (A technology designed to facilite the componentization and intraoperability of remote software modules and applications.) www.osgi.org/Main/HomePage. [Accessed January 6, 2014]

PostgreSQL (Structured Query Language) database. (An open source, enterprise-class, object-relational database system.) www.postgresql.org. [Accessed January 6, 2014]

Rodriguez A. RESTful (Representational State Transfer) web services: the basics. November 6, 2008. (An architectural style that can be used to design web services that are client-specific and focused on a given system's available resources.) www.ibm.com/developerworks/webservices/library/ws-restful.

SSH (Secure Shell) Technology. (A software program that enables a user to log in to a remote computer to execute commands and move files between machines via strong authentication and secure communications over insecure network channels.) www.webopedia.com/TERM/S/SSH.html. [Accessed January 6, 2014]

TeraGrid™. (A community of integrated, high-performance computers, data resources, and tools.) info.teragrid.org. [Accessed January 6, 2014]

The Open Group. Service oriented architecture. What is SOA? 2013. (An architectural style that can be used to design client-specific service orientation.) www.opengroup.org/soa/source-book/soa/soa.htm#soa_definition. [Accessed January 7, 2014]