Genomics Interlocking: An architecture that takes laboratory computing close to the data

Yousef Kowsar1, Andrew Lonie2

1The University of Melbourne, Melbourne, Australia, Kowsar.yousef@unimelb.edu.au
2 The University of Melbourne, Melbourne, Australia, alonie@unimelb.edu.au

INTRODUCTION-PROBLEM

Modern genomics has become a data/computing-intensive task [1]. Sequencing technologies such as Illumina X-10 are approaching the promise of sequencing the entire human genome with less than $1000, which will inevitably result in the routine use of sequencing data in clinical settings. However, the amount of data produced as a result of advances in sequencing technologies has opened problems in storing, retrieving, and processing their outputs effectively.

Advances in cloud computing provide a path to resolving these infrastructure problems in a more practical and tangible way. Very scalable, highly available computing and storage capacity, which once were only available through massive high performance computing facilities, are now accessible to everyone for renting on a hourly basis. Moreover, storing, maintaining and retrieving data is well supported through sophisticated cloud-based data services, which significantly reduce the cost of manipulating the data. However, this comes with the cost of transferring data to and from the cloud.

Despite the advances in digital communication speed, the size of the files used in genomics and in particular human and plant sequences [2], can make it impractical to transfer them on-demand. The size of these files can range from a few gigabytes for human genome to tens of gigabytes for plants. Therefore, it is clear that, given limited network bandwidth, commuting the files between local and remote compute resources will soon become a bottleneck. In addition, high performance computing centres often stipulate that they do not provide a data storage service, requiring their users to move their files by the time their contract or allocated resources are finished. This is a fundamental tension in data-driven disciplines, a consequence of computing on data being a transient resourcing requirement, while the data is a permanent resource that needs to be maintained.

One solution is to focus the genomics analysis environment on the data, moving the computing close to the data so that analysis tools may access the data on-demand. From this point of view, cloud providers are providing facilities for accessing computing resource close to (in a bandwidth and latency sense) the datacentres where files have been stored.

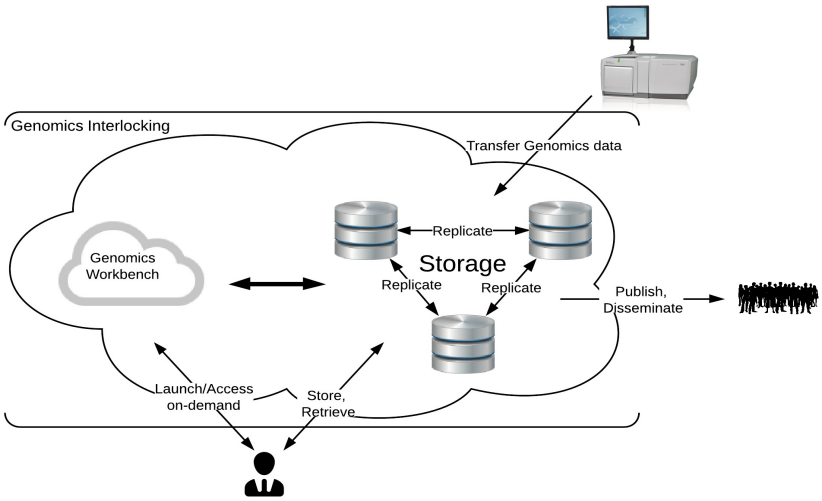

Figure 1: The Genomics Interlocking architecture overview

GENOMICS INTERLOCKING[1]

We define an interlocking as the environment that integrates facilities for basic file handling and manipulation of data such as storing, uploading, deleting, sharing together with some form of computational platform such as a workflow platform or analysis platform to the data. Genomics interlocking, then, defines an interlocking solution that is designed for working on genomics data. In our data-centric cloud architecture users store their data on permanent and reliable

[1] In railway system interlocking is the system for controlling and directing trains from the platforms.

storage on the cloud. Users can access, share and publish their data on the cloud through RESTful API with no user management overhead. In this architecture, the virtual laboratory can be seen as a workbench where pre-installed tools and referenced data are the two main utilities provided in a controlled way to the end users for analysing their data. Users may launch their own virtual laboratory or share an organizational one on the same domain to perform their process. An overview of the genomics interlocking architecture can be seen in Figure 1.

We have realized an implementation of our proposed architecture in two major components:

1. GenomeSpace[3]: A lightweight, cloud based middleware that facilitates accessing cloud-based storage while allowing tools to share data. GenomeSpace is designed as a middleware over the cloud storage which can provide a homogeneous and secure method for accessing and sharing the cloud storage such as AWS S3, OpenStack Swift, Dropbox, etc.
2. Genomics Virtual Laboratory: A workbench that provides tools and referenced data for creating genomics workflows as well as data analysing and browsing. This platform is build upon the use of the following platforms:
    i. Galaxy [4]: A genomics workflow platform for managing tools and referenced data over the cloud.
    ii. Ipython notebook: An interactive computing framework for developing, documenting, and executing code, as well as communicating the results.
    iii. UCSC-Genome-browser [6]: The UCSC genome browser is visualising genome sequence data from a variety of vertebrate and invertebrate species and major model organisms, integrated with a large collection of aligned annotations.

To 'bring compute to the data', we have designed and developed techniques for launching the genomics workbench on-demand, using CloudMan [5] which is a tool to orchestrate the cloud. The user can easily request a compute cluster on the cloud through CloudMan and deploy the virtual laboratory environment on-demand in the same region as the data. Upon completion, the user can release the workbench cluster back to the resource pool (again via CloudMan). If required, CloudMan makes it possible to store the current state of the workbench back to the cloud and retrieve it for reusability purposes. An overview of the whole process for building a Genomics Interlocking solution can be seen in Figure 2.

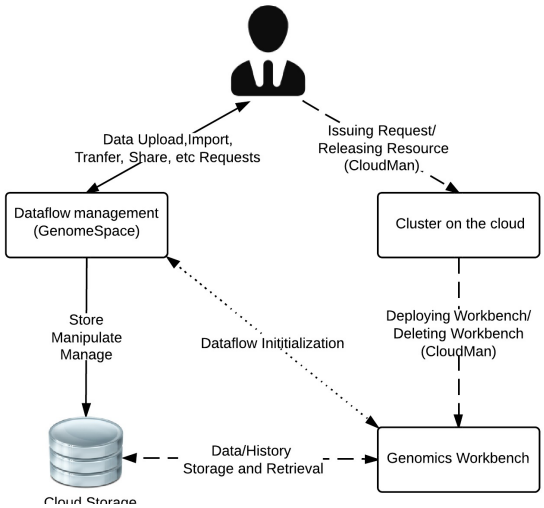

Figure 2: The process of launching a Genomics Interlocking

CONCLUSION

We believe that ultra-scalable permanent storage as available on the cloud, co-located with scalable, highly available compute resources, can address potential bottlenecks in the genomics analysis lifecycle. Furthermore, we demonstrate the capability of the cloud on handling these kind of problems by designing and developing the concept of Genomics Interlocking; a virtual environment where data and process can both be controlled and monitored, implemented using the available federated cloud technologies provided by NeCTAR.

REFERENCES

1. Schadt, Eric E., Michael D. Linderman, Jon Sorenson, Lawrence Lee, and Garry P. Nolan. "Computational solutions to large-scale data management and analysis." *Nature Reviews Genetics* 11, no. 9 (2010): 647-657.
2. Kahn, Scott D. "On the future of genomic data." *Science(Washington)* 331, no. 6018 (2011): 728-729.
3. Reich, Michael, John Liefeld, Helga Thorvaldsdottir, Marco Ocana, Eliot Polk, D. K. Jang, and Jill Mesirov. "GenomeSpace: An environment for frictionless bioinformatics." In *Proceedings of the 103rd Annual Meeting of the American Association for Cancer Research*, vol. 72, p. 3966. 2012. Available from: http://genomespace.org
4. Giardine B, Riemer C, Hardison RC, Burhans R, Elnitski L, Shah P, Zhang Y, Blankenberg D, Albert I, Taylor J, Miller W, Kent WJ, Nekrutenko A. "Galaxy: a platform for interactive large-scale genome analysis." *Genome Research*. 2005 Oct; 15(10):1451-5.
5. Afgan, Enis, Dannon Baker, Nate Coraor, Brad Chapman, Anton Nekrutenko, and James Taylor. "Galaxy CloudMan: delivering cloud compute clusters."*BMC bioinformatics* 11, no. Suppl 12 (2010): S4.
6. Karolchik D, Kent WJ. The UCSC Genome Browser. *Curr Protoc Bioinformatics*. 2002;Chapter 1:Unit 1.4.

ABOUT THE AUTHOR(S)

Yousef Kowsar is an expert in cloud and distributed systems. He is currently working as a Scientific Software Developer to solve the data-centric aspects of a genomics platform, while collaborating with the Broad Institute of Harvard and MIT to adopt GenomeSpace into the OpenStack cloud environment. Yousef obtained his Master of Science (Computer Science) with distinction from The University of Melbourne. Before pursing his masters, he was working on large-scale distributed system's projects such as stock market exchange and railway train systems in industry. His research includes the problems of storing, archiving, and retrieving data on the cloud.

A/Prof. Andrew Lonie is a faculty member of the department of Computing and Information Systems and coordinator of the MSc (Bioinformatics) at the University of Melbourne, and foundation head of the Life Sciences Computation Centre, a cross-institutional center of bioinformatics and computational biology expertise and infrastructure support within the Victorian Life Sciences Computation Initiative. Genomics, molecular modeling and biomedical image analysis experts within the LSCC collaborate with and support life sciences researchers in a variety of research projects across Victoria; the center is also responsible for implementing and disseminating best practice methods and techniques, advising on experimental design and interpretation, and resourcing and maintaining informatics analysis platforms.