Contribution ID: **35** Type: **Presentations**

CloudQTL: Evolving a Bioinformatics Application into the Clouds

Tuesday, 9 April 2013 11:20 (20 minutes)

Impact

1990s to 2005 –Standalone Application to the World Wide Web.

Production and release of QTL Express [1], a user-friendly, web-accessible analysis tool, involved converting QTL mapping algorithms [2] initially written in Fortran into Java servlets. QTL Express allowed users to send data and receive output in series for simple QTL mapping analyses using moderately sized data of the order of kilobytes. It has seen wide use for the analysis of experimental data for QTLs, and it has received almost 500 citations.

2005-2010 - e-Science push - Grid Portal technologies

The advent of microarray technologies that produce high-density multiple trait gene expression datasets and the availability of dense gene marker maps for thousands of individuals increased the dimensionality and complexity of QTL analyses requiring computationally intensive and more advanced QTL mapping tools. This led to a push for more computational power, a need to develop more complex QTL algorithms as well as the ability to accommodate more users using larger data sets of the order of megabytes as the QTL community grew.

GridQTL [3] & [4] provided an expanded and improved QTL analysis tool from QTL Express in a user friendly web portal environment, harnessing Grid technologies to deal with these increased computational demands and offering data persistence, parallel submission and retrieval of data with access via a user login to a personal data space for reviewing results. Work started in 2005 and involved collaboration with the Institute of Evolutionary Biology (IEB), Roslin Institute, National e-Science Centre (NeSC), and EPCC. The web portal was based on GridSphere [5] that acted as a container to the QTL algorithms that had evolved once more into JSR 168 compliant Java portlets [6]. The portal uses the power of the NGS [7], ECDF [8] & [9] and, for very large data sets Hector [10] in the computational Grid. Grid middleware from the Globus Toolkit [11], and Enabling Grids for e-Science project, EGEE [12] were used initially for job-submission and querying methods as well as for management tools for the authentication and authorisation processes involved in the use of the Grid resources; qsub software with ssh key-pairs has since replaced the original middleware.

GridQTL was first released in the autumn of 2006 and demonstrated at the UK e-Science All Hands conference of that year [13]. To date nearly 500 individual users have performed near to 100000 analyses in their QTL studies and are now using around 2 cpu years of computation time on our Grid per year. Around 50 users a month use GridQTL in every continent of the world; a map detailing the location of our users who have cited GridQTL is available from our website [4].

QTL Studies performed with GridQTL to date have included: birth weight and fleece quality in sheep; growth in young cattle; fatness in pigs; harvest traits in salmon; domesticity studies in foxes; obesity in mice; growth in broiler chickens; wood quality of eucalyptus trees; scale quality in crocodiles and airway obstructions in thoroughbred racehorses.

2010 and onwards –up to the Clouds.

A further tranche of funding gave us the ability to include new QTL models in the portal as well as to investigate areas of Cloud computing. The GridQTL portal has so far given users access to the QTL algorithms and the computational resources free of charge; however, there is no way of sustaining this once the project funds run out.

Our view of Cloud Computing is in line with the view presented in [14]. Cloud Computing brings together Software as a Service (SaaS) and Utility Computing where Utility Computing is a service made available in a pay-as-you-go manner by the Cloud Provider. One can distinguish several classes of Utility Computing amongst the current Cloud computing offerings. The difference is based on the level of abstraction presented to the programmer wanting to access virtualised resources. For example the Google AppEngine [15] provides automatic scaling and load balancing but enforces the programmer to use a predefined application structure and a fixed API. On the other side of the coin is Amazon's EC2 [16] which allows the author to control nearly the entire software stack but at the same time is not providing any help in automatic scalability or fallover. There is also the middle ground represented by Microsoft's Azure platform [17] that supports general purpose computing but requires applications to be compiled to the specific runtime. GridQTL uses complex backend applications to perform calculations, and it was deemed to be too expensive to port these to new runtime environments. Only the fully virtualised model, similar to Amazon's EC2, was practical for moving the existing portal to Cloud infrastructure.

When developing CloudQTL we first sought the Amazon route via Eucalyptus [18] and OpenStack [19] middleware, both of which implement subsets of EC2 API, using a prototype local Cloud provided by the Edinburgh University ECDF Cloud; this would enable eventual Cloudbursting to similar Clouds implementing the EC2 API. Development of CloudQTL has however been considered with other Cloud scenarios in mind (e.g. OpenNebula [20] and OCCI [21]) so as not to tie ourselves to one specific access route to Cloud systems. In partnership with EPCC an initial version of CloudQTL, has been incorporated into the 3.1.0 release of GridQTL, which was released in December 2012. When the product proves robust a cost model accounting system based on EPCC's SAFE project [22] will be then considered for implementation.

References –on request.

URL

www.gridqtl.org.uk

Summary

A timeline is presented which shows the stages involved in converting a bioinformatics software application from a set of standalone algorithms through to a simple web based tool then to a web based portal harnessing Grid technologies (GridQTL) and on to its latest inception as a Cloud based bioinformatics web tool (CloudQTL). The nature of the software is discussed together with a description of its development at various stages and the resulting successful increase in the user base. A discussion is then made detailing the latest idea to achieve a paid for service using Cloud technologies.

Description

A quantitative trait is a phenotype or organism characteristic with continuous measurement such as product yield and quality in agricultural species or risk factors for disease in animal and human populations. It is usually complex in that it is influenced by the actions and interactions of many genes and environmental factors and geneticists are interested in identifying and understanding the role of the genes involved.

Quantitative trait locus mapping is a statistical modelling approach to identifying regions of the genome known as QTLs (Quantitative Trait Loci) that are involved in the control of the trait and is an essential tool for understanding the genetic basis of complex traits. It involves the use of molecular markers to follow inheritance of specific genome locations from parent to offspring and combines information from these with pedigree and trait records to look for associations between genotype and phenotype.

Primary author: Dr ALLEN, John (UE)

Presenter: Dr ALLEN, John (UE)

Session Classification: Cloud Platforms

Track Classification: Cloud Platforms (Track Lead: M Drescher and M Turilli)