Science Highlight

Supercomputing for the parallelization of whole genome analysis

Project Duration: April 2012-Present

Megan Puckelwartz, PhD, Lorenzo Pesce, PhD, Elizabeth McNally, MD, PhD

Summary:

The cost of sequencing an entire human genome is now moving into the range where it is being broadly applied in both the research and clinical setting. Whole genome analysis (WGA) requires the alignment and comparison of raw sequence data, with approximately 125GB of data generated per individual genome. While the cost of generating such data has declined dramatically, a computational bottleneck exists limiting accuracy and efficiency to analyze multiple genomes simultaneously. We have adapted Beagle, the Computation Institute's Cray XE6 supercomputer housed at Argonne National Laboratory, to achieve the parallelization necessary for concurrent multiple genome analysis and named this workflow **Megaseq**.

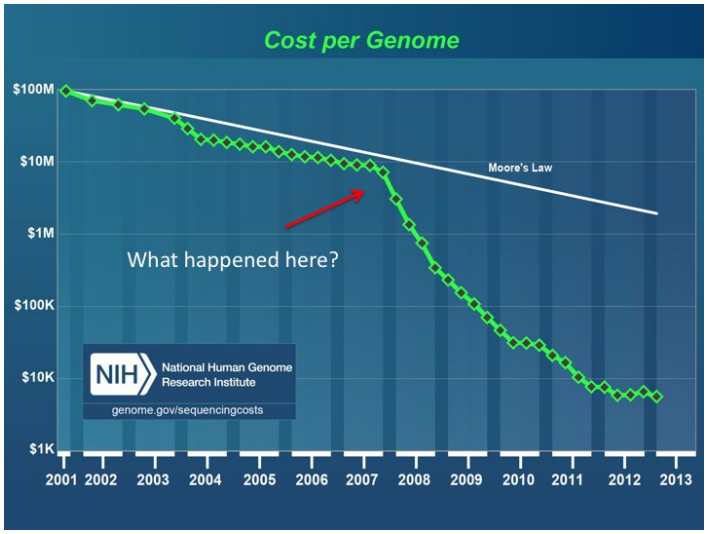

Figure 1. Cost of sequencing. Prior to 2007, the cost of sequencing (green line) followed the same scaling as Moore's law (white line). The advent of next generation sequencing technology has caused a dramatic shift in that the bottleneck for whole genome analysis is no longer sequencing cost/time, but computational efficiency.

Additional information on Beagle can be found at: Beagle Website

DAY OF THE BEAGLE
APRIL 23RD 2013
Scientific Discovery on the Computation Institute/BSD Cray "Beagle" Supercomputer

In a "Day of the Beagle" symposium on April 23rd, the innovative research made possible to date by Beagle was discussed by speakers representing the fields of genomics, neurobiology, molecular biology and medicine...

Trainings/ Workshops

The Beagle team will be offering **Beagle Intro** on May 28th, 2013 at 2:00 pm, room 240A, Searle building.
To learn more go to: Trainings and Events The training is meant for new and existing users of the Beagle system, and will cover topics such as:

- Beagle hardware overview
- Swift as a parallel scripting language
- How to easily write Swift scripts
- How to run Swift on Beagle
- Applications of Swift

Resources

- *Beagle Wiki* Get detailed usage information from the Beagle team
- *Beagle Support* Contact the Beagle experts for help
- *Globus Online* Get started moving files fast and reliably to/from Beagle using this data transfer service
- *Other CI resources* Learn about other computing resources available at the Computation Institute

Scheduling Policy and Access to Beagle

- For normal priority jobs, there are no limits to the number of nodes, submitted or running jobs, or walltime used. To find more details about specific queue type: qstat -q
- For the new low priority settings, the restrictions are a walltime of four hours or less, and nothing more than 10 nodes. Basically, we're scheduling normal priority jobs first, then filling in the gaps with **lower priority jobs**. The smaller the job, the easier it'll be to schedule, so the quicker it will move from queued to running.
- Nothing needs to be specified in the submit script. Low priority configurations are handled on the server side on a per-project basis. Any project that has been designated as having a low-priority allocation will automatically have those settings assigned when Moab reads the project code/name when the job is submitted.
- Scheduling is based on a fair-share system, with four queues to meet the varying needs of our users. Details can be found at: Beagle Scheduling Policy

Work done during the last maintenance period:

During this last maintenance period, we upgraded the software on Beagle's management workstation to SMW version 7.0.UP01(SLES11SP2) and the system software on Beagle itself to CLE version 4.1.UP01. Most of the changes involved in these updates will be transparent to users, but there are a few changes of note:

- With PADS being decommissioned, we are no longer mounting its GPFS filesystem on Beagle's login nodes (/gpfs).
- The /tmp directory should now be writable on compute nodes.
- The default compiler and related environment has changed to the native Cray compiler. Check our wiki for more details on using this and switching between available compilers.
- The cray-mpich2 module is now loading automatically for all users as part of the default modules profile.

Beagle Related Publications

Supercomputing for the parallelization of whole genome analysis

* Megan J. Puckelwartz, Lorenzo Pesce, Viswateja Nelakuditi, Sharlene Day, Thomas Cappola, Euan Ashley

Molecular autopsy for sudden death using whole genome sequencing

* Megan J. Puckelwartz, Lisa Dellefave-Castillo, Don Wolfgeher, Viswateja Nelakuditi, Mark T. Campbell, Jessica R Golbus

Validation of bootstrap-based comparison of ROC curves on partially-paired data sets

* Y. Peng, L. Pesce, Y. Jiang in SPIE (2013)

Computational Modeling of $\alpha\beta$ T-Cell Receptor Loop Conformations

*James Crooks, Ridgway Scott, Erin Adams in 57th Annual Meeting of the Biophysical Society (2013) Scaling Properties of Two Parallel Neuronal Network Simulation Algorithms Lorenzo L. Pesce, Hyong C. Lee Lee, Mark Hereld, Sid Visser, Rick L Stevens, Wim van Drongelen in TBD (2013)

Determinants of Fluid-Like Behavior and Effective Viscosity in Cross-Linked Actin Networks

* Taeyoon Kim, Margaret L. Gardel, Ed Munro in PLOS Comput Biol (2013) Numerical Solution of Dynamic Portfolio Optimization with Transaction Costs Y Cai, K.L. Judd, R. Xu (2013)

Searle Chemistry Laboratory
5735 South Ellis Avenue
Chicago, IL 60637

Help E-Mail:
beagle-support@ci.uchicago.edu
Beage CI Website:
beagle.ci.uchicago.edu