# ADAM Enables Distributed Analyses Across Large Scale Genomic Datasets

Frank Austin Nothaft[1,2], Arun Ahuja[3], Timothy Danford[1,4], Michael Heuer[1], Jey Kottalam[1], Matt Massie[1], Audrey Musselman-Brown[5], Beau Norgeot[5,6], Ravi Pandya[7], Justin Paschall[1], Jacob Pfeil[5], Hannes Schmidt[5], Eric Tu[1], John Vivian[5], Ryan Williams[3], Carl Yeksigian[8], Michael Linderman[3], Jeff Hammerbacher[3], Uri Laserson[3,9], Gaddy Getz[10], David Haussler[5], Benedict Paten[5], Anthony D. Joseph[1], David A. Patterson[1,2]

Website: http://bdgenomics.org
Repositories: https://github.com/bigdatagenomics/adam and https://github.com/BD2KGenomics/toil-scripts
License: Apache 2 License

The detection and analysis of rare genomic events requires integrative analysis across large cohorts with terabytes to petabytes of genomic data. Contemporary genomic analysis tools have not been designed for this scale of data-intensive computing. This abstract presents recent updates to ADAM, an Apache 2 licensed library built on top of the popular Apache Spark distributed computing framework. We are using ADAM and the Toil workflow management system (Apache 2 licensed) to recall the Simons Genome Diversity project dataset against the GRCh38 build of the human reference genome. Because ADAM is designed to allow genomic analyses to be seamlessly distributed across large clusters, we achieve a 3.5× improvement in end-to-end variant calling latency and a 66% cost improvement over current toolkits, without sacrificing accuracy.

On top of our previous results [1], we have achieved an additional 2–3× improvement in performance by modifying the schemas that ADAM uses. In addition to improving performance, these modifications address the problem of storing and processing metadata (e.g., information about a reference genome build) inline with an analysis. To run our workflow at large scale, we use the Toil workflow system. Toil is a system for running arbitrary computation that can be structured as a directed acyclic graph across various different schedulers. Toil provides fault tolerance, and supports computational reproducibility and portability. We build upon Toil to enable dynamic scaling of Spark clusters using the AWS spot market. Our end-to-end variant calling pipeline allocates machines on an as-needed basis to improve cost effectiveness, and allows the use of Apache Spark in a workflow with traditional, single node bioinformatics tools.

## References

[1] F. A. Nothaft, M. Massie, T. Danford, Z. Zhang, U. Laserson, C. Yeksigian, J. Kottalam, A. Ahuja, J. Hammerbacher, M. Linderman, M. Franklin, A. D. Joseph, and D. A. Patterson. Rethinking data-intensive science using scalable analytics systems. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data (SIGMOD '15)*. ACM, 2015.

[1] AMPLab, University of California, Berkeley, CA
[2] ASPIRE Lab, University of California, Berkeley, CA
[3] Icahn School of Medicine at Mount Sinai, New York, NY
[4] Tamr, Cambridge, MA
[5] Genome Informatics Lab, University of California, Santa Cruz, CA
[6] University of California, San Francisco, CA
[7] Microsoft Research, Redmond, WA
[8] GenomeBridge, Cambridge, MA
[9] Cloudera, Inc., San Francisco, CA
[10] The Broad Institute of MIT and Harvard, Cambridge, MA
Correspondence should be addressed to fnothaft@berkeley.edu.