

JBrowse - Modernizing a JavaScript Genome Browser for the Next Generation of Bioinformatics

Rob Buels¹, Eric Yao¹, Colin Diesh¹, Garrett Stevens¹, Shihab Dider¹, Andrew Duncan², Scott Cain², Lincoln Stein², Ian Holmes¹
1. Dept of Bioengineering, University of California, Berkeley, USA. 2. Ontario Institute for Cancer Research, Toronto, Canada.

Introduction

JBrowse is a widely used HTML5 genome browser with a thriving ecosystem of third-party contributed plugins. As one of the first JavaScript client genome browsers with a web API that can work directly from a static set of files, JBrowse can be run using any web server, including very lightweight web servers. The JBrowse code base has seen continued development to add new support for modernized ES6 and webpack based bundling, new data formats such as CRAM, BigBed, and CSI index formats, search dialogs, iframe-free embedding, and the JBrowse Desktop program. We have also developed JBrowse Connect, a backend server that can perform dynamic analysis that enables new workflows that are not possible purely client side.

Modernizing the codebase

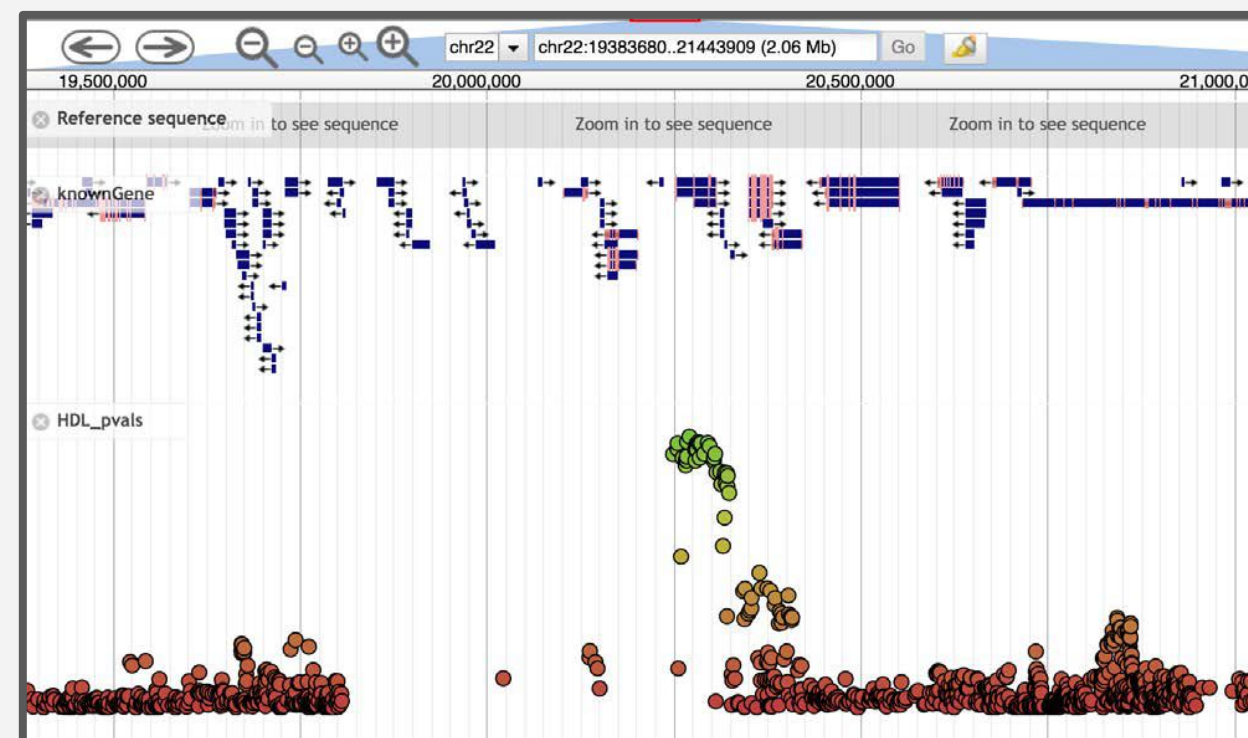
Recent JBrowse developments have included refactoring the JBrowse data parsers into separate NPM modules so that they can be used in new contexts. The NPM modules are generic enough to be used in the webbrowser or in the Node.js environment, and enable us to use it in new future applications. Meanwhile the current JBrowse codebase has been modernized to allow modern "ES6" style javascript using babel and webpack for compilation.

New data formats and parsers

A JavaScript CRAM parser that parses CRAM client side is now available in JBrowse, giving bandwidth and storage size benefits over BAM. There is also a BigBed adapter that enables the display of genes, but can also be extended in other ways e.g. the BigMaf parser in mafviewer is simply a subtype of BigBed (Figure 1). The support for tabix based files such as GFF3, BED, and VCF have all been greatly improved too, avoiding the need for perl pre-processing using flatfile-to-json.pl. Finally, we also have a CSI parser which allows indexing files for large genomes e.g. wheat.



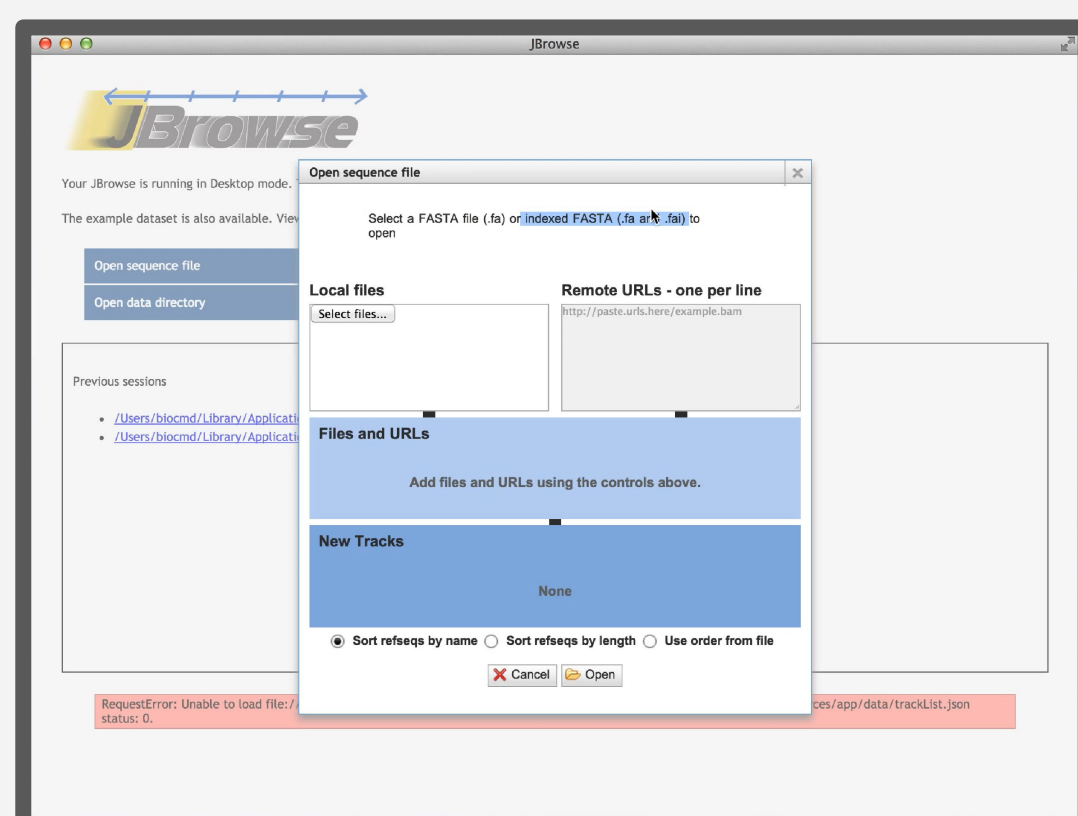
MAFViewer - This is a 100-way multiple sequence alignment of vertebrates from UCSC. It uses the BigBed subtype "BigMaf" for the data.



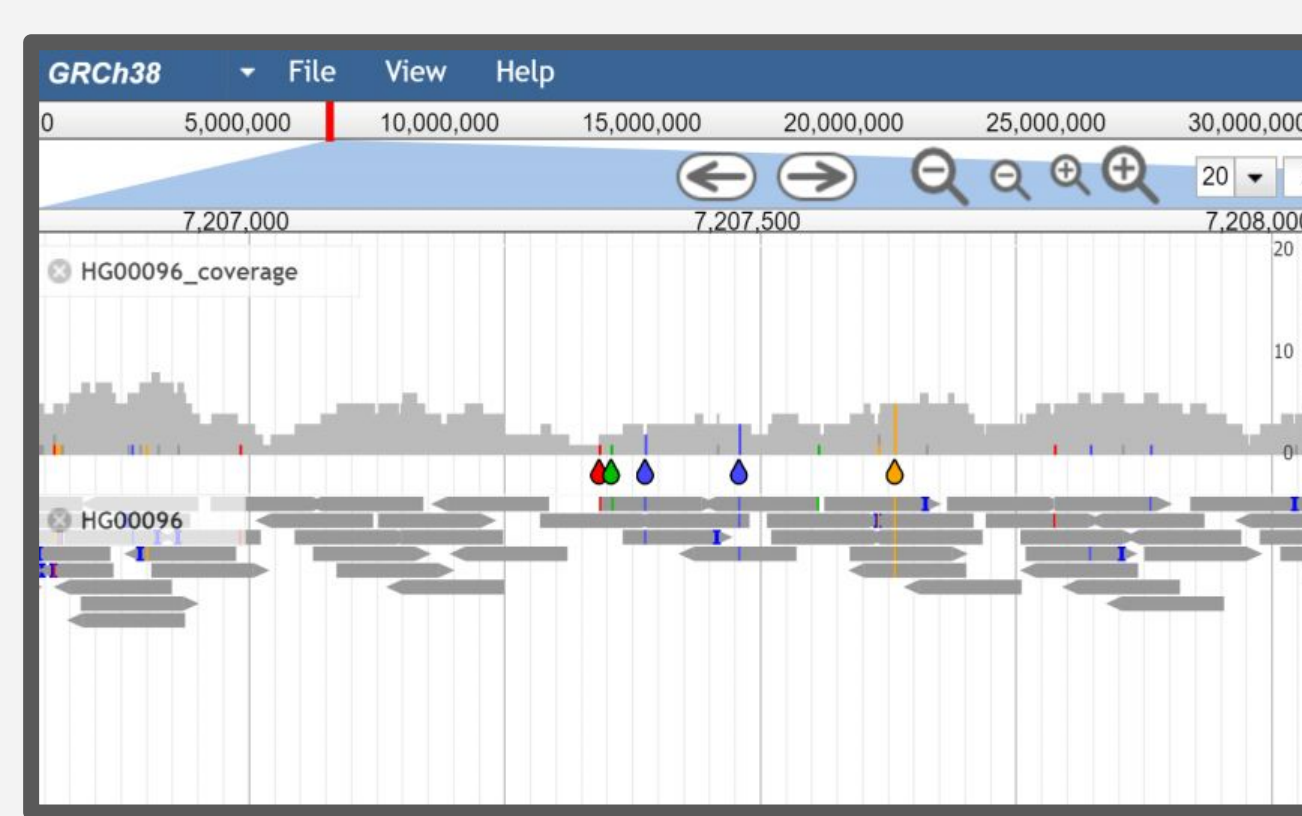
The GWASViewer plugin visualizes features with associated significance scores, and can use tabix files for efficient data access (source: elsiklab/gwasviewer)

JBrowse Desktop

The JBrowse desktop platform is a new way to run JBrowse that does not require a webserver and can be run privately on a user's computer. It can also save "session" with the users locally included files. New use cases could include variant inspection of private data that would be easy to set up in a lab setting, which can be done easily without any webserver setup.



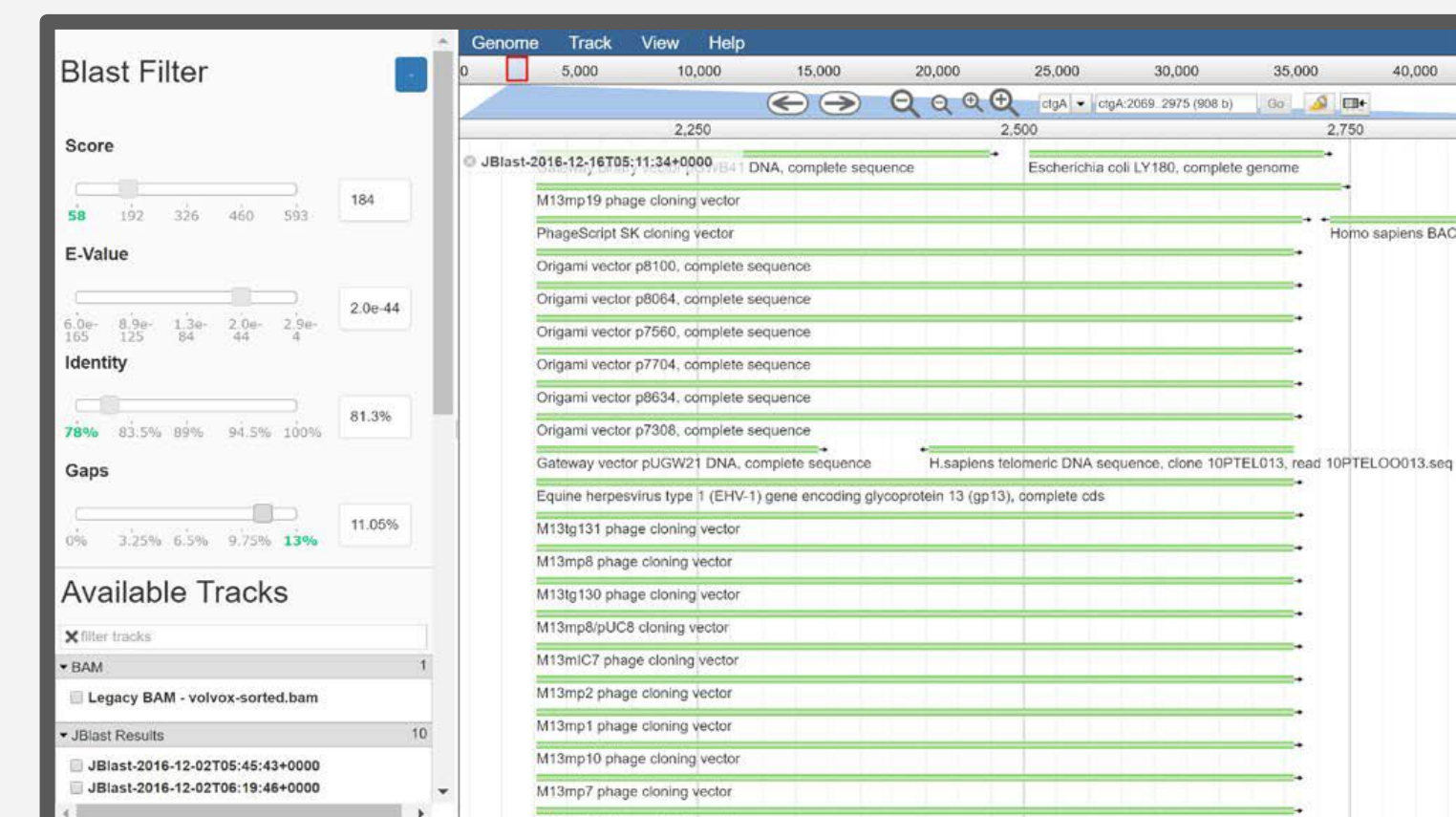
JBrowse Desktop allows users to run JBrowse as a native app without setting up a webserver. JBrowse Desktop has a home screen for opening a sequence file or a data directory, and has a list of saved sessions.



The SNPcoverage and alignments views in JBrowse help variant inspection, and small direction indicators on the reads have been added recently to make forward and reverse strand reads easier to see.

JBrowse Connect

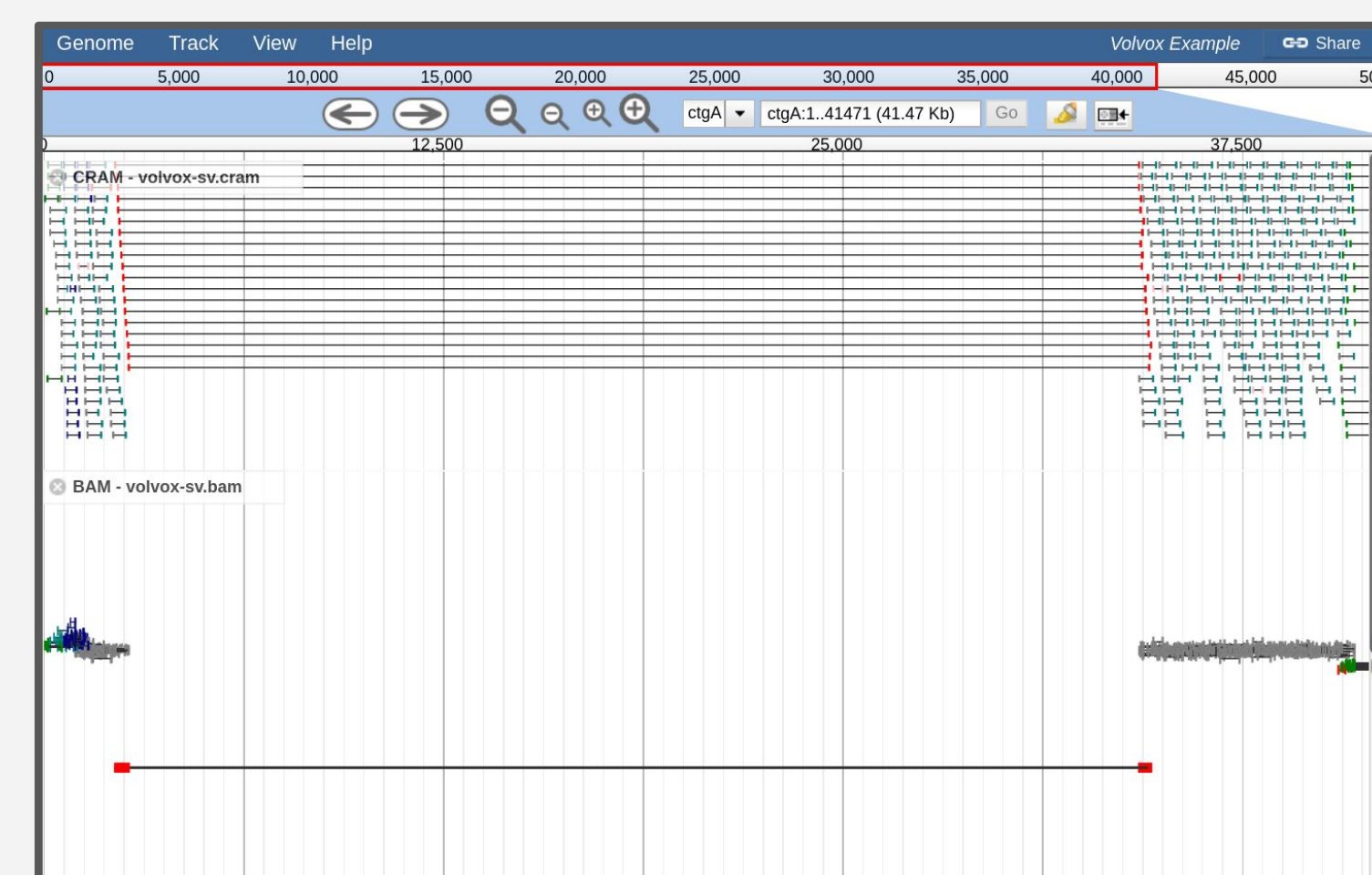
We have developed JBrowse Connect, an optional server extension to JBrowse that runs simultaneously as a JBrowse plugin (on the client) and as an extensible back-end (on the server). BLAST is implemented as an example analysis task, demonstrating the API for developing a JBrowse plugin that triggers, tracks, and then collects the results of an analysis workflow in Galaxy (or another workflow manager).



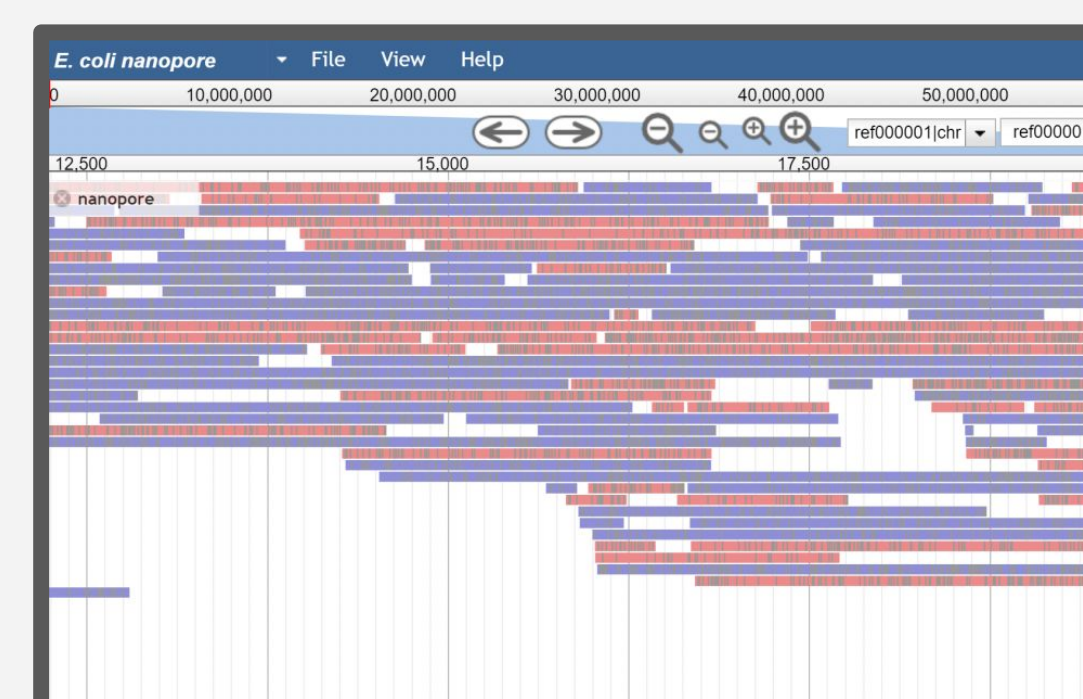
An example of using JBrowse Connect and BLAST to visualize alignments and control BLAST parameters.

Enabling long read and paired-read views

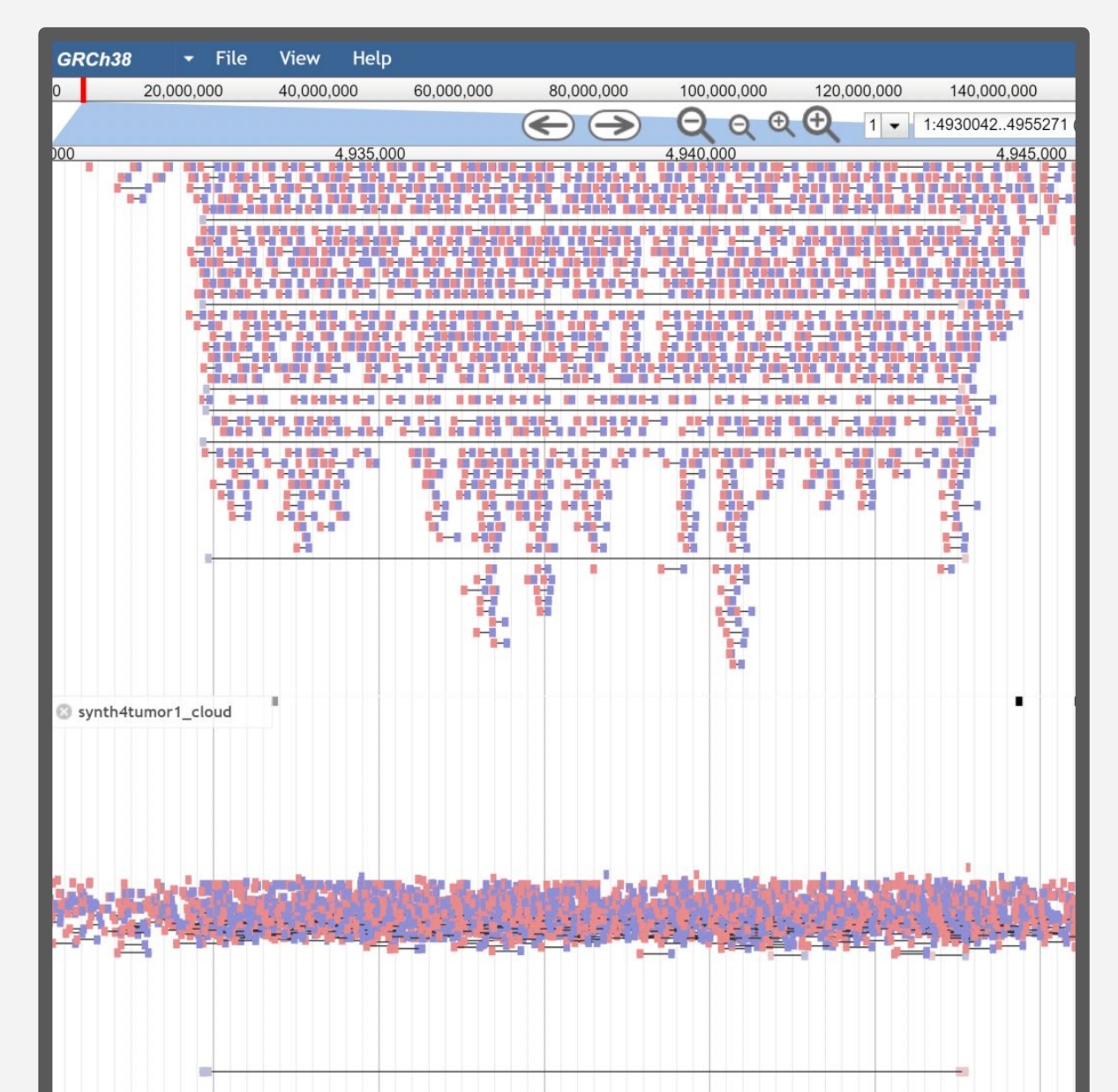
JBrowse has been enhanced recently to render long-reads by caching the calculations that render CIGAR/MD tags. This allows very long reads to be displayed with minimal overhead. The ability to display the paired-reads as a linked entity has also been added and is pending release. Paired read viewing incorporates a new method to redispach requests to resolve the paired reads and a caching layer to keep pairs in-tact while browsing in the middle of regions where long-distance pairs exist (since normally, BAM/CRAM queries to these regions would return nothing). We also developed a "read cloud" view that stratifies the reads according to their insert size so that long distance structural variants stand out more easily.



Paired read viewing of a simulated structural variants using the pileup view (above) and the "read cloud" view (below). The reads are also colored according to pair orientation and insert size abnormalities to help indicate structural variants.



JBrowse can display long reads from BAM and CRAM by doing efficient caching of the parsed tags e.g. CIGAR/MD. This is a nanopore sequencing dataset from *E. coli*.



Another paired read view for simulated structural variants, rendered again with the pileup and read cloud views