



BAM Track Format

BAM is the compressed binary version of the [Sequence Alignment/Map \(SAM\)](#) format, a compact and index-able representation of nucleotide sequence alignments. Many [next-generation sequencing and analysis tools](#) work with SAM/BAM. For custom track display, the main advantage of indexed BAM over PSL and other human-readable alignment formats is that only the portions of the files needed to display a particular region are transferred to UCSC. This makes it possible to display alignments from files that are so large that the connection to UCSC would time out when attempting to upload the whole file to UCSC. Both the BAM file and its associated index file remain on your web-accessible server (http, https, or ftp), not on the UCSC server. UCSC temporarily caches the accessed portions of the files to speed up interactive display. If you do not have access to a web-accessible server and need hosting space for your BAM files, please see the [Hosting](#) section of the Track Hub Help documentation.

The typical workflow for generating a BAM custom track is this:

1. If you haven't done so already, [download](#) and build the [samtools](#) program. Test your installation by running `samtools` with no command line arguments; it should print a brief usage message. For help with `samtools`, please contact the [SAM tools mailing list](#).
2. Align sequences using a tool that outputs SAM directly, or outputs a format that can be converted to SAM. (See [list of tools and converters](#))
3. Convert SAM to BAM using the `samtools` program:

```
samtools view -S -b -o my.bam my.sam
```

If converting a SAM file that does not have a proper header, the `-t` or `-T` option is necessary. For more information about the command, run `samtools view` with no other arguments.

4. Sort and create an index for the BAM:

```
samtools sort my.bam my.sorted
samtools index my.sorted.bam
```

The sort command appends `.bam` to `my.sorted`, creating a BAM file of alignments ordered by leftmost position on the reference assembly. The index command generates a new file, `my.sorted.bam.bai`, with which genomic coordinates can quickly be translated into file offsets in `my.sorted.bam`.

5. Move both the BAM file and index file (`my.sorted.bam` and `my.sorted.bam.bai`) to an http, https, or ftp location. Note that the Genome Browser looks for an index file with the same URL as the BAM file with the `.bai` suffix added. If your hosting site does not use the filename as the URL link, you will have to specifically call the location of this `.bam.bai` index file with the `bigDataIndex` keyword. This keyword is relevant in Custom Tracks and Track Hubs. You can read more about `bigDataIndex` in [the TrackDb Database Definition page](#).
6. If the file URL ends with `.bam` and the BAM index file URL ends with `.bam.bai`, you can paste the URL directly into the [custom track](#) management page, click submit and view in the Genome Browser. The track name will then be the name of the file. If you want to configure the track name and descriptions or the URLs are not as described, you will need to create a [track line](#), as shown below in step 7.
7. Construct a [custom track](#) using a single [track line](#). The most basic version of the "track" line will look something like this:

```
track type=bam name="My BAM" bigDataUrl=http://myorg.edu/myLab/my.sorted.bam
```

Again, in addition to `http://myorg.edu/mylab/my.sorted.bam`, the associated index file `http://myorg.edu/mylab/my.sorted.bam.bai` must also be available at the same location. If not, you can specify the URL with the `bigDataUrl=http://myorg.edu/myLab/my.sorted.bam.bai`

8. Paste the custom track line into the text box in the [custom track management page](#), click submit and view in the Genome Browser.

Parameters for BAM custom track definition lines

All options are placed in a single line separated by spaces. In the example below, the lines are broken only for readability. If you copy/paste this example, you must remove the line breaks. Click [here](#) for a text version that you can paste without editing.

```
track type=bam bigDataUrl=http://...
  pairEndsByName=.
  pairSearchRange=N
  bamColorMode=strand|gray|tag|off
  bamGrayMode=aliQual|baseQual|unpaired
  bamColorTag=XX
  minAliQual=N
  showNames=on|off
```

```

name=track_Label
description=center_Label
visibility=display_mode
priority=priority
db=db
maxWindowToDraw=N
chromosomes=chr1,chr2,...

```

The track type and bigDataUrl are REQUIRED:

```
type=bam bigDataUrl=http://myorg.edu/myLab/my.sorted.bam
```

The remaining settings are OPTIONAL. Some are specific to BAM:

```

pairEndsByName  any value      # presence indicates paired-end alignments
pairSearchRange N              # max distance between paired alignments, default 20,000 bases
bamColorMode    strand/gray/tag/off # coloring method, default is strand
bamGrayMode     aliQual/baseQual/unpaired # grayscale metric, default is aliQual
bamColorTag     XX              # optional tag for RGB color, default is "YC"
minAliQual      N              # display only items with alignment quality at least N, default 0
showNames       on/off         # if off, don't display query names, default is on

```

Other optional settings are not specific to BAM, but relevant:

```

name            track_Label      # default is "User Track"
description      center_Label    # default is "User Supplied Track"
visibility       squish/pack/full/dense/hide # default is hide (will also take numeric values 4|3|2|1|0)
bigDataUrl       https://your.bam.bai.com # default is the bigDataUrl with .bai added to the end
priority         N               # default is 100
db              genome database  # e.g. hg18 for Human Mar. 2006
maxWindowToDraw N               # don't display track when viewing more than N bases
chromosomes      chr1,chr2,...   # track contains data only on listed reference assembly sequences
doWiggle         on/off         # if on, show data as density graph, default is off

```

The [BAM track configuration](#) help page describes the BAM track configuration page options corresponding to `pairEndsByName`, `minAliQual`, `bamColorMode`, `bamGrayMode` and `bamColorTag` in more detail.

`pairSearchRange` applies only when `pairEndsByName` is given. It allows for a tradeoff of display speed vs. completeness of pairing the paired-end alignments. When paired ends are split or separated by large gaps or introns, but one is viewing a small genomic region, it is necessary to search a large number of bases upstream and downstream of the viewed region in order to find mates of the alignments in the viewed region. However, searching a very large region can be slow, especially when the alignments have deep coverage of the genome. To ensure that all properly paired mates will be found, `pairSearchRange` should be set to the largest genomic size of a mapped pair. However, it can be set to a smaller size if necessary to speed up the display, at the cost of some items being displayed as unpaired when the mate is too far outside the viewed window.

Example #1

In this example, you will create a custom track for an indexed BAM file that is already on a public server — alignments of sequence generated by the [1000 Genomes Project](#).

You can paste the URL `http://genome.ucsc.edu/goldenPath/help/examples/bamExample.bam` directly into the [custom track management page](#) for the human assembly hg18 (May 2006), then press the *submit* button. On the following page, press the *chr21* link in the custom track and navigate to position chr21:33,038,946-33,039,092 to see the reads in the new BAM track.

Alternatively, you can specify more visualization options by creating a "track" line. The line breaks inserted here for readability must be removed before submitting the track line:

```

track type=bam name="BAM Example One" description="Bam Ex. 1: 1000 Genomes read alignments (individual NA1287
pairEndsByName=. pairSearchRange=10000 chromosomes=chr21 bamColorMode=gray maxWindowToDraw=200000
db=hg18 visibility=pack
bigDataUrl=http://genome.ucsc.edu/goldenPath/help/examples/bamExample.bam

```

Include the following "browser" line to view a small region of chromosome 21 with alignments from the .bam file:

```
browser position chr21:33,038,946-33,039,092
```

Note if you copy/paste the above example, you must remove the line breaks (or, click [here](#) for a text version that you can paste without editing).

Paste the "browser" line and "track" line into the [custom track management page](#) for the human assembly hg18 (May 2006), then press the "submit" button. On the following page, press the *chr21* link in the custom track listing to view the BAM track in the Genome Browser.

Example #2

In this example, you will create indexed BAM from an existing SAM file. First, save this SAM file [samExample.sam](#) to your machine. Perform steps 1 and 3-7 in the workflow described above, but substituting `samExample.sam` for `my.sam`. On the [custom track management page](#), click the "add custom tracks" button if necessary and make sure that the genome is set to Human and the assembly is set to Mar. 2006 (hg18) before pasting the track line and submitting. This track line is a little nicer than the one shown in step 6, but remember to remove the line breaks that have been added to the track line for readability (or, click [here](#) for a text version that you can paste without editing):

```
track type=bam name="BAM Example Two"
  bigDataUrl=http://myorg.edu/myLab/my.sorted.bam
  description="Bam Ex. 2: Simulated RNA-seq read alignments" visibility=squish
  db=hg18 chromosomes=chr21

browser position chr21:33,037,317-33,038,137
browser pack mrna
```

Sharing Your Data with Others

If you would like to share your BAM data track with a colleague, learn how to create a sharable URL by looking at [this page](#).

[UCSC HOME](#)
[BSOE HOME](#)
[GENOMICS INSTITUTE HOME](#)

[LICENSES](#)
[TRAINING](#)
[CITE US](#)
[FAQS & SEARCH](#)

[STORE](#)
[CONTACT](#)
[SUGGESTIONS](#)
[JOBS](#)



© 2000-2022 THE REGENTS OF THE UNIVERSITY OF CALIFORNIA. ALL RIGHTS RESERVED.
[CONDITIONS OF USE](#)