Genome Browsers for interacting with Genomic data

Many different browser environments

1. [NCBI Genome Maize Maize chromosome](#)
2. [Ensembl](#)
3. [FungiDB](#)
4. [Mycocosm](#)
5. [Wormbase](#)
6. [FlyBase](#)
7. [Saccharomyces Genome Database](#)
8. [TAIR](#)
9. [Gramene](#) - Plant Comparative Resources
10. [Phytozome](#) - Plant Comparative Genomics portal
11. [UCSC Genome Browser](#)

Setting up your own - JBrowse2 - Genome Browser

To visualize genome annotation combined with Epigenomic, Transcriptomic, or Variant data you you want to visualize them onto a genome browser. [JBrowse2](#) provides an easy to setup tool for this visuzalition. The [Quick Start](#) provides easy to use instructions.

The [FAQ](#) is also incredibly helpful.

Setup JBrowse on UCR [HPCC](#)

These steps will show you how to setup JBrowse on HPCC with some already installed systems to make it easier for you

Configure your HPCC account to be able to share via HTTP / Web

First you need to configure your account to be able to share data via the web.

Follow the [directions on the HPCC manual](#) so that you can configure your home folder `~/.html` to be able to serve up data.

All of these can be changed paths - the only critical part on UCR HPCC is in the ~/.html folder is where website serves up data from our server. On other local servers it might be ~/public_html or /var/www/html on your own host.

```
mkdir -p ~/bigdata/jbrowse2 # it is best to store data on the bigdata partition so you do n
mkdir -p ~/.html
```

```
cd ~/.html/
ln -s ~/bigdata/jbrowse2 .
cd jbrowse2 # now you will proceed to install browser sites in this folder
```

To add an assembly for this genome

```
jbrowse add-assembly NC_045512.fna.gz --load inPlace --type bgzipFasta
```

If you do not want to make everything in this folder public you can use some simple strategies to enable a password protected space by creating a `.htaccess` file. Generally if you want to protect the data, setup a `.htaccess` and a corresponding `.htpasswd` to require logging in.

As it is on the UCR system you need to setup an .htaccess file with at least these data. Create the `.htaccess` file in the ~/.html/jbrowse2 folder. You can also add additional directives in there for specifying password protected access. You can also put this file in the sub-folders you will have for each browser (eg. `~/.html/jbrowse2/SARS-CoV-2/.htaccess`) so different password settings can be defined for different folders and collaborations.

```
# This Apache .htaccess file is for
# allowing cross-origin requests as defined by the Cross-Origin
# Resource Sharing working draft from the W3C
# (http://www.w3.org/TR/cors/).  In order for Apache to pay attention
# to this, it must have mod_headers enabled, and its AllowOverride
# configuration directive must allow FileInfo overrides.
<IfModule mod_headers.c>
    AddType application/octet-stream .bam .bami .bai
    Header onsuccess set Access-Control-Allow-Origin *
    Header onsuccess set Access-Control-Allow-Headers X-Requested-With,Range
    Header onsuccess set Access-Control-Expose-Headers Content-Length,Content-Range
</IfModule>
```

On the web a user browsing will not have permission to see `https://cluster.hpcc.ucr.edu/~YOURUSERNAME/jb`

Setting up your own copy of JBrowse software

The next directions are specific to the UCR HPCC. These instructions use an already build conda environment which you can link to.

```
cd ~/.html/jbrowse
```

```
module load jbrowse/2 # UCR specific - otherwise if you installed jbrowse via npm ``
jbrowse create SARS-CoV-2
```

These instructions are UCR specific - otherwise if you installed jbrowse via `npm` you would just be able to specify `jbrowse` as is or follow their other download/install options on the JBrowse2 site.

If you are going to support multiple JBrowse environments you only need to have a custom data folder. So you can symlink to all the files within the jbrowse checkout and then make a separate data folder too. Otherwise you need to make sure you have a separate custom jbrowse checkout for each project you are supporting.

Download the SARS-CoV-2 genome and annotation from NCBI - this can be either in the folder you want to put the data or you can later symlink or copy from this folder

```
module load samtools
module load bcftools
cd SARS-CoV-2
# download and compress the genome with bgzip to save space (bgzip is different from gzip s
curl -o NC_045512.fna.gz https://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/009/858/895/GCF_009858
gunzip NC_045512.fna.gz
bgzip NC_045512.fna
# need to index the genome with faidx
samtools faidx NC_045512.fna.gz
# download the GFF file
curl -o NC_045512.gff.gz https://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/009/858/895/GCF_009858
# uncompress, sort and re-compress with bgzip
# to load GFF we need to ensure it is sorted
# sort
zgrep -v "^#" NC_045512.gff.gz  | sort -k1,1 -k4,4n >  NC_045512.sorted.gff
# compress and index
bgzip -i NC_045512.sorted.gff
# re-index with other index scheme
tabix NC_045512.sorted.gff.gz
```

To load genome you have already put in the `SARS-CoV-2` folder - you need to *GO INTO THE* `SARS-CoV-2` folder

```
cd SARS-CoV-2
# if downloaded the data into the folder
jbrowse add-assembly NC_045512.fna.gz  --load inPlace --type bgzipFasta
# if you had a different folder for this you might do something like this
# this example here assumes a) uncompressed file b) you also already ran samtools faidx GENU
# jbrowse add-assembly ../path/to/NC_045512.fna --load symlink
# or if you want to copy it
# jbrowse add-assembly ../path/to/NC_045512.fna --load copy

# if you forgot to create the index it will give you a message
# then you need to do
# samtools faidx NC_045512.fna
# if you created the gff and ran bgzip and tabix in this folder
jbrowse add-track  NC_045512.sorted.gff.gz --load inPlace
# if you had put this in another folder
```

```
#jbrowse add-track ../path/to/NC_045512.sorted.gff.gz --load symlink
```

To load VCF files (SNPs and variants)

```
jbrowse add-track SARS-CoV-2.vcf.gz --load inPlace
# if there are warnings you need to build an index you can srun
# module load bcftools
# tabix SARS-CoV-2.vcf.gz
# then re-run the add-track
# if VCF file is in this directory
# jbrowse add-track SARS-CoV-2.vcf.gz --load inPlace
```

To load BAM files, WIG files, or other gFF you can use same add-track. For BAM, CRAM, files they need to have been indexed

```
module load samtools
samtools index SRR11140748.bam
jbrowse add-track SRR11140748.bam --load inPlace
# or if the file was made IN this directory
# if it gives you a warning about index file run
# samtools index BAMFILE
```

Other file types that can be loaded include bigwig files.

Note that on the UCR HPCC to serve up BAM files properly you need to create a `.htaccess` file in the jbrowse folder (remember ours is called `SARS-CoV-2` in this example).

Now navigate to the web with you link based on your username and folder.

You can see a working version of JBrowse2 hosted on this example github hosted site for SARS-CoV-2.

Note you cannot really host large genomes or files here as github limits to files ~50Mb and smaller. So this is is merely an example.