

---

# JBrowse

发布 *0.0.1*

2020 年 01 月 29 日



<b>1</b>	<b>JBrowse 中文文档</b>	<b>3</b>
<b>2</b>	<b>安装</b>	<b>5</b>
2.1	安装 JBrowse 之前需要做什么 . . . . .	5
2.2	导入 JBrowse 插件 . . . . .	5
2.3	下载 JBrowse 的发行版本 . . . . .	6
2.4	如果将 JBrowse 作为插件需要备用 JBrowse 设置 . . . . .	6
2.5	祝贺! . . . . .	7
<b>3</b>	<b>索引文件格式</b>	<b>9</b>
3.1	加载 FASTA 格式的索引 . . . . .	9
3.2	加载 Tabix GFF3 . . . . .	10
3.3	加载 BAM . . . . .	11
3.4	检查您的文件是否已经加载成功 . . . . .	11
3.5	祝贺! . . . . .	12
<b>4</b>	<b>经典快速入门指南</b>	<b>13</b>
4.1	参考序列 . . . . .	13
4.2	基因组注释和特征 . . . . .	13
4.3	bin/flatfile-to-json.pl . . . . .	13
4.4	bin/biodb-to-json.pl . . . . .	14
4.5	Next-gen reads (BAM) . . . . .	14
4.6	Next-gen read track 类型 . . . . .	14
4.7	Index Names . . . . .	15
4.8	定量轨道 (BigWig 和 Wiggle) . . . . .	15
4.9	结论 . . . . .	16
<b>5</b>	<b>Indices and tables</b>	<b>17</b>



sphinx-quickstart on Wed Jan 29 13:25:56 2020. You can adapt this file completely to your liking, but it should at least contain the root *toctree* directive.

---

**注解:** 本人在开发时阅读的 JBrowse 文档, 仅供参考学习

---



# CHAPTER 1

---

JBrowse 中文文档

---





一个最通常的情况是, 下载 JBrowse 之后, 将整个项目文件夹放在 web 服务器的目录下 (在 ubuntu 中通常是 /var/www/), JBrowse 就是一个静态站点, 它通过 index.html 中的 js 进行数据处理, 不需要后端处理.

## 2.1 安装 JBrowse 之前需要做什么

有一些前提条件可以帮助您进行 JBrowse 设置, 包括

- unix 操作系统系列- MacOSX, Linux, 或者 WSL on Windows(在 windows10 的应用商店中下载 ubuntu, 使用 cmd 输入命令 `bash` 即可进入 linux 环境)
- Web 服务器-JBrowse 是一组静态文件, 可与 Apache 或 Nginx 一起使用
- 命令行技能-熟悉命令行将帮助您更好的使用本教程
- Sudo 访问-sudo 是没有必要的, 除非您需要它来修改 Web 服务器文件, 例如在 / var / www 中, 修改权限

如果您不具备所有这些条件, 请考虑使用 JBrowse Desktop, 因为它不需要命令行, 并且易于在所有操作系统上使用:)

## 2.2 导入 JBrowse 插件

如果您使用的是 JBrowse 插件, 则还需要安装 Node.js 版本 6 或更高版本。请按照 <https://nodejs.org/en/download/package-manager/#debian-and-ubuntu-based-linux-distributions> 中的

步骤在 ubuntu 上安装 node.js

## 2.3 下载 JBrowse 的发行版本

```
curl -L -O https://github.com/GMOD/jbrowse/releases/download/1.16.7-release/JBrowse-1.16.7.zip
unzip JBrowse-1.16.7.zip
sudo mv JBrowse-1.16.7 /var/www/html/jbrowse
cd /var/www/html
sudo chown `whoami` jbrowse
cd jbrowse
./setup.sh
```

---

**注解:** 文档假设你已经安装了 web 服务器, 例如 apache. 在 ubuntu 上安装 apache 使用命令 `apt install apache2` 会在 /var 中自动生成 www 文件夹, 里面会有 index.html 初始文件. 访问 ubuntu 服务器 ip 即可查看该页面. 将 JBrowse 项目文件夹放入 /var/www 中, 配置 apache, apache 具体配置自行搜索

---

## 2.4 如果将 JBrowse 作为插件需要备用 JBrowse 设置

JBrowse 现在在构建时捆绑了插件, 因此, 如果您使用插件或自己修改 jbrowse 源代码, 则必须下载源代码 <https://github.com/GMOD/jbrowse>

```
git clone https://github.com/gmod/jbrowse
cd jbrowse
git checkout 1.16.7-release # or version of your choice
./setup.sh
npm run start # starts a express.js dev server on port 8082, alternatively move the
entire jbrowse dir to /var/www/html as above
```

---

**注解:** 使用 `npm run watch` 自动获取对您所做的代码所做的更改 (但是, 如果添加或删除文件, 则需要重新启动) 对于中国的用户, 如果使用自定义配置, 推荐使用 npm 镜像

---

```
npm config set registry http://r.cnpmjs.org
npm config set puppeteer_download_host=http://cnpmjs.org/mirrors
export ELECTRON_MIRROR="http://cnpmjs.org/mirrors/electron/"
```

## 2.5 祝贺!

您应该看到一条消息, “Congratulations, JBrowse is on the web”, 并带有指向 “Volvox example data” 的链接。如果您没有看到此消息, 则可能是错过了设置步骤



在本节中，我们将使用“索引文件格式”，以纯文本配置为例

### 3.1 加载 FASTA 格式的索引

首先，我们假设正在建立 *Volvox mythicus* (*Volvox* 属中的一个神话物种) 的基因组。*Volvox* 基因组已由测序中心在 2018 年进行了测序，他们想立即设置 JBrowse。他们为我们提供了指向其 FASTA 文件的链接，我们将下载该文件

```
mkdir data
curl -L https://jbrowse.org/code/JBrowse-1.16.7/docs/tutorial/data_files/volvox.fa >data/volvox.fa
```

我们将使用 samtools 的 `faidx` 命令来创建“FASTA index”。FASTA index 能将非常大的 FASTA 文件“按需”下载到 JBrowse 中。例如，仅下载特定视图所需的序列。

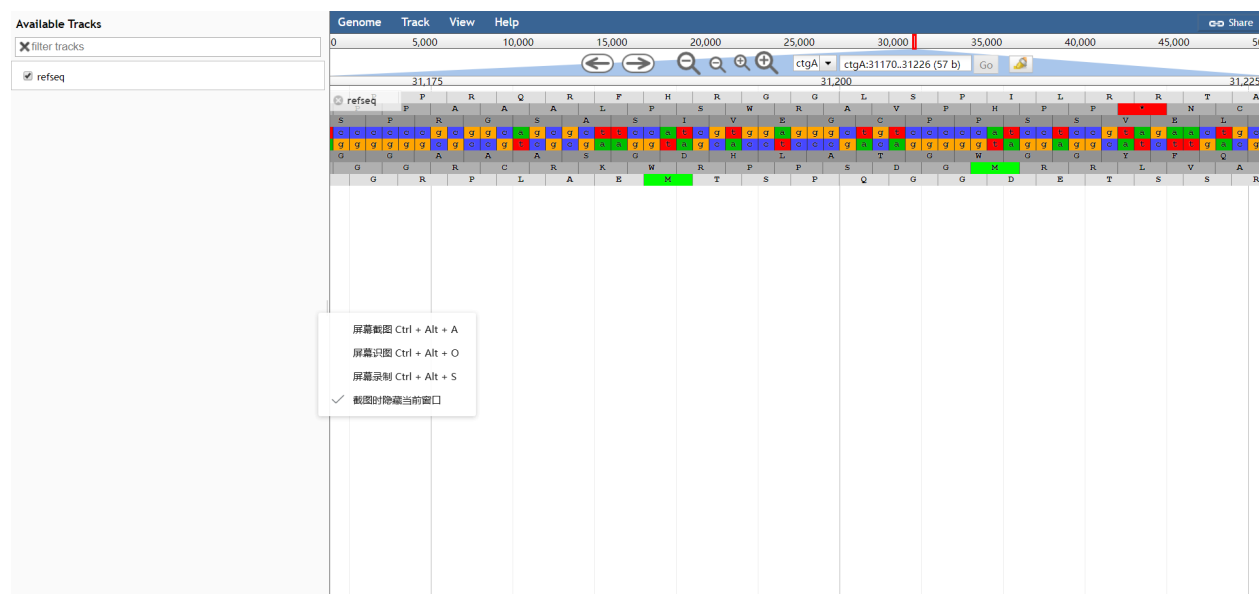
```
samtools faidx data/volvox.fa
```

FASTA index 将是一个名为 `volvox.fa.fai` 的文件。然后，我们将这些文件移到 JBrowse 可以使用的“数据目录”中

然后创建文件 `data / tracks.conf`，使用以下内容

```
[GENERAL]
refSeqs=volvox.fa.fai
[tracks.refseq]
urlTemplate=volvox.fa
storeClass=JBrowse/Store/SeqFeature/IndexedFasta
type=Sequence
```

此时，您应该可以打开 <http://localhost/jbrowse/?data=data>（或者打开 <http://localhost/jbrowse/>），您就会看到带有参考序列轨迹的基因组。示例图如下：



## 3.2 加载 Tabix GFF3

我们将使用的新生成的“基因注释”文件

```
curl -L https://jbrowse.org/code/JBrowse-1.16.7/docs/tutorial/data_files/volvox.gff3 > \
data/volvox.gff3
```

当我们处理 GFF3 以在 JBrowse 中使用，我们的目标是使用 GFF3Tabix 格式。Tabix 格式允许随机访问类似于 Indexed FASTA 的基因组区域。因此，我们必须首先对 GFF 进行排序以为 Tabix 做准备

```
sort -k1,1 -k4,4n data/volvox.gff3 > data/volvox.sorted.gff3
bgzip data/volvox.sorted.gff3
tabix -p gff data/volvox.sorted.gff3.gz
```

在 `data/tracks.conf` 中加入一下配置：

```
[tracks.genes]
urlTemplate=volvox.sorted.gff3.gz
storeClass=JBrowse/Store/SeqFeature/GFF3Tabix
type=CanvasFeatures
```

完成!

### 3.3 加载 BAM

如果已获得序列比对，也可以创建一个显示比对的“比对”轨迹对于 volvox，我们得到一个文件

```
curl -L https://jbrowse.org/code/JBrowse-1.16.7/docs/tutorial/data_files/volvox-sorted.
↪bam > data/volvox-sorted.bam
```

请注意，此 BAM 文件已经排序。如果您的 BAM 未排序，则必须对其进行排序以在 JBrowse 中使用。使用以下命令行：

```
samtools index data/volvox-sorted.bam
```

最后在 data/tracks.conf 中添加以下内容：

```
[tracks.alignments]
urlTemplate=volvox-sorted.bam
storeClass=JBrowse/Store/SeqFeature/BAM
type=Alignments2
```

### 3.4 检查您的文件是否已经加载成功

此时，如果 jbrowse 文件位于您的 Web 服务器中，则应具有目录布局，例如

```
/var/www/html/jbrowse
/var/www/html/jbrowse/data
/var/www/html/jbrowse/data/volvox.fa
/var/www/html/jbrowse/data/volvox.fa.fai
/var/www/html/jbrowse/data/volvox.sorted.gff3.gz
/var/www/html/jbrowse/data/volvox.sorted.gff3.gz.tbi
/var/www/html/jbrowse/data/volvox-sorted.bam
/var/www/html/jbrowse/data/volvox-sorted.bam.bai
/var/www/html/jbrowse/data/tracks.conf
```

你的 tracks.conf 文件夹中的内容如下:

```
[GENERAL]
refSeqs=volvox.fa.fai

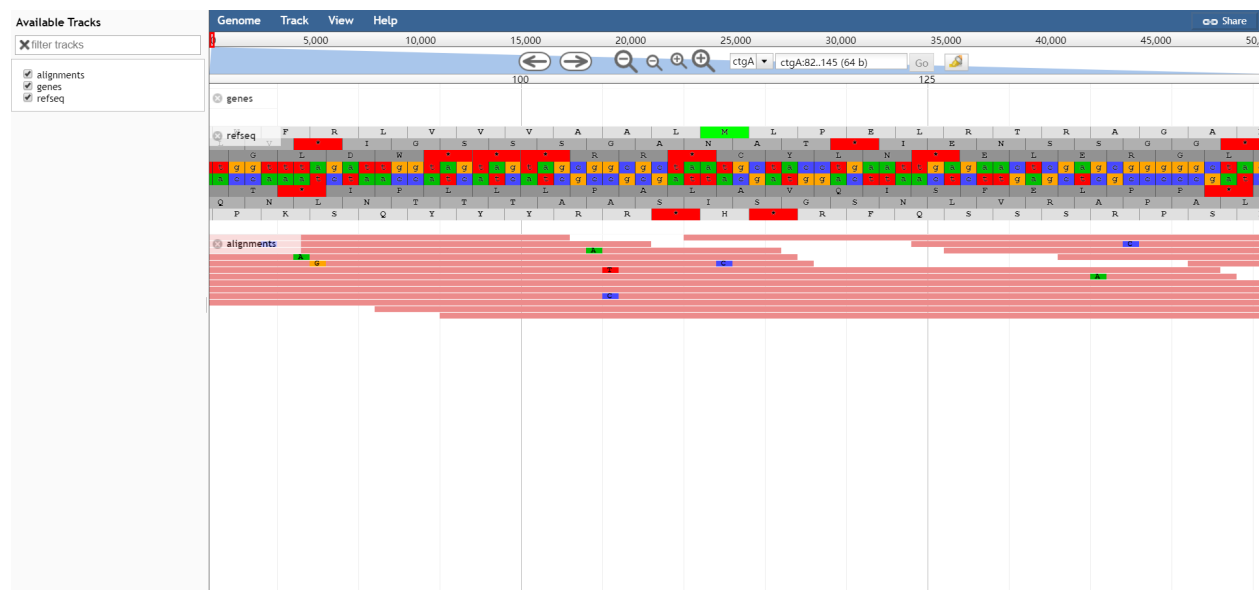
[tracks.refseq]
urlTemplate=volvox.fa
storeClass=JBrowse/Store/SeqFeature/IndexedFasta
type=Sequence

[tracks.genes]
urlTemplate=volvox.sorted.gff3.gz
storeClass=JBrowse/Store/SeqFeature/GFF3Tabix
type=CanvasFeatures

[tracks.alignments]
urlTemplate=volvox-sorted.bam
storeClass=JBrowse/Store/SeqFeature/BAM
type=Alignments2
```

然后, 您可以访问 <http://localhost/jbrowse/> 将自动加载 “data” 目录。

示例图如下:



### 3.5 祝贺!

您现在成功配置了 JBrowse.



## 4.1 参考序列

在加载注释数据之前，请使用 `bin/prepare-refseqs.pl` 格式化 JBrowse 的参考序列。对于 FASTA 格式序列文件

```
bin/prepare-refseqs.pl --fasta docs/tutorial/data_files/volvox.fa
```

对于已经存储在 Bio :: DB :: \*数据库中的序列，可以将 `prepare-refseqs.pl` 与 `biodb-to-json.pl` 配置结合使用

```
bin/prepare-refseqs.pl --conf docs/tutorial/conf_files/volvox.json
```

## 4.2 基因组注释和特征

JBrowse 可以从平面文件或具有 Bio :: DB :: \* Perl 接口的数据库中导入数据。

## 4.3 bin/flatfile-to-json.pl

如果您有平面文件，例如 GFF3 或 BED，通常最好使用 `bin / flatfile-to-json.pl` 导入它们。`bin / flatfile-to-json.pl` 接受许多不同的命令行设置，这些设置可用于自定义新 track 的外观。运行 `bin / flatfile-to-json.pl -help` 查看可用设置的描述。

```
bin/flatfile-to-json.pl --gff path/to/my.gff3 --trackType CanvasFeatures --trackLabel_
↪mygff
```

## 4.4 bin/biodb-to-json.pl

如果您拥有基因组注释数据库，例如 Chado, Bio :: DB :: SeqFeature :: Store 或 Bio :: DB :: GFF, 则可以使用 JBrowse 的 biodb-to-json.pl。您将 bin / biodb-to-json.pl 与配置文件一起使用（其格式在此处记录）。

```
bin/biodb-to-json.pl --conf docs/tutorial/conf_files/volvox.json
```

## 4.5 Next-gen reads (BAM)

JBrowse 可以直接从 BAM 文件显示对齐方式，而无需进行预处理。要使用该文件，请将 BAM 文件和 BAM 索引下载到您的数据文件夹中，然后您可以将文本片段手动编辑到 tracks.conf 中：

```
[tracks.alignments]
storeClass = JBrowse/Store/SeqFeature/BAM
urlTemplate = myfile.bam
category = NGS
type = JBrowse/View/Track/Alignments2
key = BAM alignments from SEQ1654
```

然后，你需要在 data 文件夹中存在以下文件：

```
data/myfile.bam
data/myfile.bam.bai
data/tracks.conf
```

在 tracks.conf 包含上述配置的地方，然后 myfile.bam 将相对于 tracks.conf 的位置进行定位，因此由于它们位于同一目录中，因此不需要任何其他路径限定。请注意，urlTemplate 可以是完整的 URL 路径。

另请注意：BAM 文件需要排序和索引（即具有对应的.bai 文件或.csi 文件。如果为.csi，则可以使用 csiUrlTemplate。如果为.bai，则将自动定位它作为 <yourbamfile.bam>，最后加上.bai）。

## 4.6 Next-gen read track 类型

JBrowse 有两种主要的轨道类型，这些轨道类型专门设计用于 BAM 数据：

Alignments2 显示 BAM 文件中的各个比对，以及 BAM 的 MD 或 CIGAR 字段中编码的插入，删除，跳过的区域和 SNP。

SNPCoverage 显示覆盖度直方图，彩色条显示碱基水平不匹配和读数中可能的 SNP 的位置。

要使用它们，请在轨道配置中设置 `type = Alignments2` 或 `type = SNPCoverage`。

## 4.7 Index Names

加载要素数据后，要让用户通过在自动完成搜索框中键入要素名称或 ID 来查找要素，必须使用 `bin/generate-names.pl` 生成要素名称的特殊索引。

```
bin/generate-names.pl -v
```

注意：每次使用任何 `*-to-json.pl` 脚本向 JBrowse 添加新注释时，都需要重新运行 `bin/generate-names.pl` 将新功能名称添加到索引。

还要注意：`bin / generate-names.pl` 索引的轨道类型包括：

- GFF, GBK, BED 通过 `flatfile-to-json.pl` 加载。默认情况下，对“ID”，“名称”和“别名”建立索引。请注意，`-nameAttributes` 可用于索引其他字段
- Features from `biodb-to-json.pl`
- VCF tabix 文件和 GFF3 tabix 文件（对它们从 GFF 的“ID”和“名称”字段建立索引，并且仅对来自 VCF 的 ID 进行索引）

BAM 的 reads 和 BigWigs 不会由 `generate-names.pl` 编制索引

## 4.8 定量轨道 (BigWig 和 Wiggle)

JBrowse 可以直接从 BigWig 文件显示对齐方式，而无需进行预处理。只需将带有文件相对 URL 的节添加到您的 `data / tracks.conf` 文件中，格式为：

```
[ tracks.my-bigwig-track ]
storeClass = JBrowse/Store/SeqFeature/BigWig
urlTemplate = myfile.bw
type = JBrowse/View/Track/Wiggle/XYPlot
key = Coverage plot of NGS alignments from XYZ
```

JBrowse 有两种专门用于定量数据的 track 类型：

- JBrowse/View/Track/Wiggle/XYPlot

将定量数据显示为条形图。有关配置选项，请参见 JBrowse Wiki。

- JBrowse/View/Track/Wiggle/Density

将定量数据显示为“热图”图，默认情况下，其具有正分数的区域绘制为逐渐变深的蓝色，而具有负分数的区域绘制为逐渐变深的红色。有关配置选项，请参见 JBrowse Wiki，包括如何更改颜色改变点 (bicolor\_pivot) 和颜色。

要使用它们，请在轨道配置中设置 `type = JBrowse / View / Track / Wiggle / XYPlot` 或 `type = JBrowse / View / Track / Wiggle / Density`。

## 4.9 结论

祝您好运，并享受 JBrowse!

## CHAPTER 5

---

### Indices and tables

---

- search