

Instsllation of Gbrowser-Syn on a CentOS system

Last updated: 2015-09-16

Prerequisite

- A Physical/Virtual machine running on CentOS release 6.6

```
#To confirm the version of CentOS
cat /etc/*release*
```

bash

- Internet connection

Referenes

1. <http://gmod.org/wiki/GBrowse2.0HOWTO>
2. http://gmod.org/wiki/GBrowsesynTutorial2013#InstallingGBrowse_syn
3. <http://nix-bio.blogspot.jp/2013/10/installing-gbrowse2-on-centos-6-and-mac.html>

Installation

1. Apache

```
yum -y install httpd httpd-devel
```

bash

save a copy of original configuration file

```
sudo cp -i /etc/httpd/conf.d/welcome.conf /etc/httpd/conf.d/welcome.conf.org
sudo cp -i /etc/httpd/conf/httpd.conf /etc/httpd/conf/httpd.conf.org
```

bash

check the ip address of the machine and try access at the 80 port

```
ifconfig
```

bash

Additional tweak if a static IP is disrired

```
## Configure eth0
# add the static IP

sudo nano /etc/sysconfig/network-scripts/ifcfg-eth0

# change
BOOTPROTO="dhcp"
```

bash

```
# to
BOOTPROTO="static"

# add following lines
IPADDR=192.168.1.2
NETMASK=255.255.255.0
```

Configure Default Gateway

```
sudo nano /etc/sysconfig/network

# replace or add following lines
NETWORKING=yes
HOSTNAME=CentOS_6.6
GATEWAY=192.168.1.1
```

bash

Restart Network

```
/etc/init.d/network restart
```

2. MySQL

```
sudo yum -y install mysql mysql-server
```

bash

Start MySQL

```
/etc/init.d/mysqld start
```

bash

Setup the root password and other stuff

```
mysql_secure_installation

#memo the root password
```

bash

Enable MySQL and Apache service to run at bootup

```
chkconfig --levels 235 mysqld on
chkconfig --levels 235 httpd on
```

bash

3. PHP

```
sudo yum -y install php
sudo yum install php-mbstring
sudo yum install php-mysql
```

bash

```
sudo yum install php-devel
```

Additional: PhPMyAdmin

```
rpm --import http://dag.wieers.com/rpm/packages/RPM-GPG-KEY.dag.txt
yum -y install http://pkgs.repoforge.org/rpmforge-release/rpmforge-release-0.5.3-1.el6.rf.x86_64.rpm
yum -y install phpmyadmin
```

Edit the configuration file

```
nano /etc/httpd/conf.d/phpmyadmin.conf
```

change the following lines

```
#Deny from all
#Allow from 127.0.0.1
Allow from 0.0.0.0
```

Restart Apache

```
/etc/init.d/httpd restart
```

4. Install necessary Perl modules

```
sudo yum -y install make gcc gmp-devel wget git mod_fcgid fcgi-perl
sudo yum install perl-GD
sudo yum install perl-Module-Build
sudo yum install perl-CPAN
sudo yum install perl-IO-String perl-Capture-Tiny perl-CGI-Session
sudo yum install perl-JSON perl-JSON-Any
sudo yum install perl-libwww-perl perl-DBD-SQLite
sudo yum install perl-File-NFSLock perl-Net-SMTP-SSL
sudo yum install perl-Crypt-SSLeay perl-Net-SSLeay perl-Template-Toolkit
```

Final Installation

```
yum install perl-bioperl perl-Bio-Graphics inkscape
git clone https://github.com/bioperl/bioperl-live.git
cd bioperl-live
git checkout bioperl-release-1-6-901
perl Build.PL
./Build test
sudo ./Build install
sudo perl -MCPAN -e 'install Bio::Graphics::Browser2'
```

Configuration step

```
Directory for GBrowse's config and support files? [/etc/gbrowse2]
Directory for GBrowse's static images & HTML files? [/var/www/htdocs] --> [/var/www/html/gbrowse2]
Directory for GBrowse's temporary data [/var/tmp/gbrowse2]
Directory for GBrowse's sessions, uploaded tracks and other persistent data [/var/lib/gbrowse2]
Directory for GBrowse's example databases [/var/lib/gbrowse2/databases]
Directory for GBrowse's CGI script executables? [/var/www/cgi-bin] --> [/var/www/cgi-bin/gb2]
Internet port to run demo web site on (for demo)? [8000]
Apache loadable module directory (for demo)? [/etc/modules] --> [/etc/httpd/modules]
User account under which Apache daemon runs? [apache]
Automatically update Apache config files to run GBrowse? [y]
```

5. check the Demo

```
http://localhost/gbrowse2/
```

http

6. Install GBrowse_syn

```
git clone https://github.com/GMOD/GBrowse.git
cd GBrowse/
perl Build.PL
./Build test
./Build install
```

bash

Confirm the installation

```
http://localhost/cgi-bin/gb2/gbrowse\_syn
```

http

7. Prepare the Human-Mouse pairwise alignment data

```
This directory contains alignments of the following assemblies:
- target/reference: Human (hg19, Feb. 2009 (GRCh37/hg19), GRCh37 Genome Reference Consortium Human Reference 37 (GCA_000001405.1))
- query: Mouse (mm10, Dec. 2011 (GRCm38/mm10), Genome Reference Consortium Mouse Build 38 (GCA_000001635.2))
```

```
http://hgdownload.cse.ucsc.edu/goldenPath/hg19/vsMm10/axtNet/
```

http

use rsync to fetch a copy

```
mkdir hg19vsMm10_alignment
cd hg19vsMm10_alignment
rsync -a -P rsync://hgdownload.cse.ucsc.edu/goldenPath/hg19/vsMm10/axtNet/* ./
```

```
cd ..
```

8. Convert the format

The AxtNet format looks like as below

```
1 chrY 150918 150993 chr5 110101042 110101117 + 4971
GACTGGATGTCGGCACTGTGTCCCCGGCTCTGGGATGTGCCCTCCACCACCTCTCCATCCCAGGTGAGGTTGGGG
GACTGGATGTCGCAGCTGTGCCCTCAGCTGTGGGATGTCCCCCTGCACCACCTGTCCATCCCAGGTGAGTACGGGG
```

The desired format that can be handled by `load_alignment_database.pl` is

```
#species1      seqid1  start1  end1   strand1  reserved  species2      seqid2      start2  end
2 strand2 reserved pos1-1 pos1-2 ... posn-1 posn-2 | pos1-2 pos1-1 ... posn-2 posn-1
c_briggsae     chrI    1583997 1590364 +      .      c_remanei     Crem_Contig24 631879 634
679 -          .      1584000 634676 1584100 634584 (truncated...) |      631900 1590333 632000
1590233 (truncated ...)
```

9. The GFF3 annotation file for Hg19 and Mm10

Basic information about the annotation

```
Human GFF3: Release 19 (GRCh37.p13)
http://www.gencodegenes.org/releases/19.html

Mouse GFF3: Release M6 (GRCm38.p4)
http://www.gencodegenes.org/mouse\_releases/current.html
```

http

Fetch the data

```
mkdir GFF3
cd GFF3
wget ftp://ftp.sanger.ac.uk/pub/gencode/Gencode_human/release_19/gencode.v19.annotation.gff3.gz -O
hg19.gff3.gz
wget ftp://ftp.sanger.ac.uk/pub/gencode/Gencode_mouse/release_M6/gencode.vM6.annotation.gff3.gz -O
mm10.gff3.gz

gunzip hg19.gff3.gz
gunzip mm10.gff3.gz
```

bash

10. Load to the MySQL database

```
mysql -uroot -p
```

Create database

```
create database hg19;
create database mm10;
quit;
```

sql

Parse and upload to the MySQL database

```
bp_seqfeature_load.pl -u root -p ***** -d hg19 -c -f hg19.gff3
bp_seqfeature_load.pl -u root -p ***** -d mm10 -c -f mm10.gff3
```

bash

Part of the log output

```
loading hg19.gff3...
Building object tree...62.91s02.94s

Loading bulk data into database.../tmp/feature.44907
/tmp/name.44907
/tmp/attribute.44907
/tmp/parent2child.44907
338.07s
load time: 2003.95s
Building summary statistics for coverage graphs...
2612000 features processed
coverage graph build time: 723.74s
total load time: 2727.69s

-----

loading mm9.gff3...
Building object tree...24.78s2.97s

Loading bulk data into database.../tmp/feature.45166
/tmp/name.45166
/tmp/attribute.45166
/tmp/parent2child.45166
206.92s
load time: 1124.70s
Building summary statistics for coverage graphs...
1537000 features processed
coverage graph build time: 469.13s
total load time: 1593.83s
```

Prepare hg19.conf

```
[GENERAL]
description    = Home sapiens
db_adaptor     = Bio::DB::SeqFeature::Store
db_args        = -dsn dbi:mysql:hg19

tmpimages     = /tmp/gbrowse2
```

```

[CG]
label          = 1
description    = 1
feature       = mRNA
category      = Genes
glyph         = processed_transcript
font2color    = blue
height        = 6
key           = Gene Models
bgcolor       = sub {
  my $flip = pop->panel->flip;
  my $strand = shift->strand;
  return $strand < 0 ? 'violet' : 'turquoise' if $flip;
  return $strand > 0 ? 'violet' : 'turquoise';
}

# draw genes differently for segments > 100Kb
[CG:100001]
label          = 0
description    = 0
glyph         = generic
strand_arrow  = 1

```

Prepare mm10.conf

```

[GENERAL]
description    = Mus musculus
db_adaptor    = Bio::DB::SeqFeature::Store
db_args       = -dsn dbi:mysql:mm10

tmpimages     = /tmp/gbrowse2

[CG]
label          = 1
description    = 1
feature       = mRNA
category      = Genes
glyph         = processed_transcript
font2color    = blue
height        = 6
key           = Gene Models
bgcolor       = sub {
  my $flip = pop->panel->flip;
  my $strand = shift->strand;
  return $strand < 0 ? 'violet' : 'turquoise' if $flip;
  return $strand > 0 ? 'violet' : 'turquoise';
}

# draw genes differently for segments > 100Kb
[CG:100001]
label          = 0

```

