Deploying bioinformatics tools with CloudBioLinux

Brad Chapman
Bioinformatics Core,
Harvard School of Public Health
https://github.com/chapmanb

30 June 2014

Outline

- Overview of CloudBioLinux
- Create custom set of tools
- Install from custom flavor
- Add a new tool

What is CloudBioLinux?

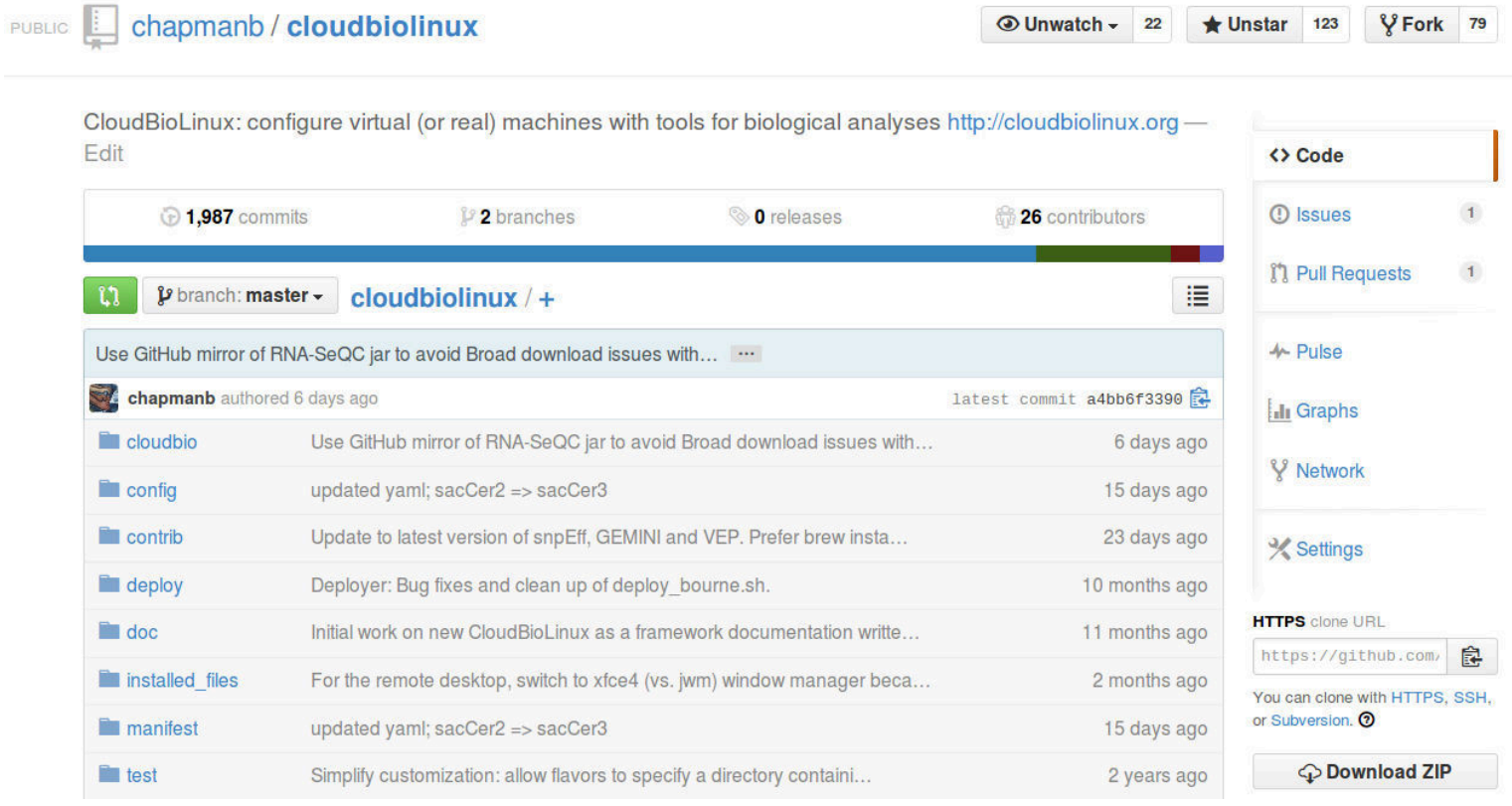
Infrastructure for installing biological software

- deb/rpm packages
- Bio-Linux
- Linuxbrew with homebrew-science
- Python, Ruby, R package management
- Conda + Binstar https://conda.binstar.org/
- Custom installation scripts

Key Features

- Community
- Curation

Community



https://github.com/chapmanb/cloudbiolinux

History

Integration of multiple efforts

- JCVI Cloud Bio-Linux
- Bioperl Max
- Infochimps machetEC2
- Bio-Linux
- DebianMed

Original goal

Overcome bare-metal problem with AWS images

- Ubuntu
- Single AMI with biological tools
- Automated build infrastructure
- Bring in developer community
- Ready to use for researchers

Biological data

- Genomes, organized and indexed
- Associated data files: dbSNP, reference transcripts
- S3 bucket
- Tools with organized data
- GEMINI: https://github.com/arq5x/gemini

Local installation

- Multiple platforms: Ubuntu, RedHat/CentOS, Debian, ScientificLinux
- Isolated installations: no sudo, non-VM environments
- Rapid turnaround for fixes

Flavors: customized installations

- Target specific use case
- Sub-collection of packages from full distribution
- Example: cloudbiolinux/contrib/flavor/biopython

Pjotr Prins

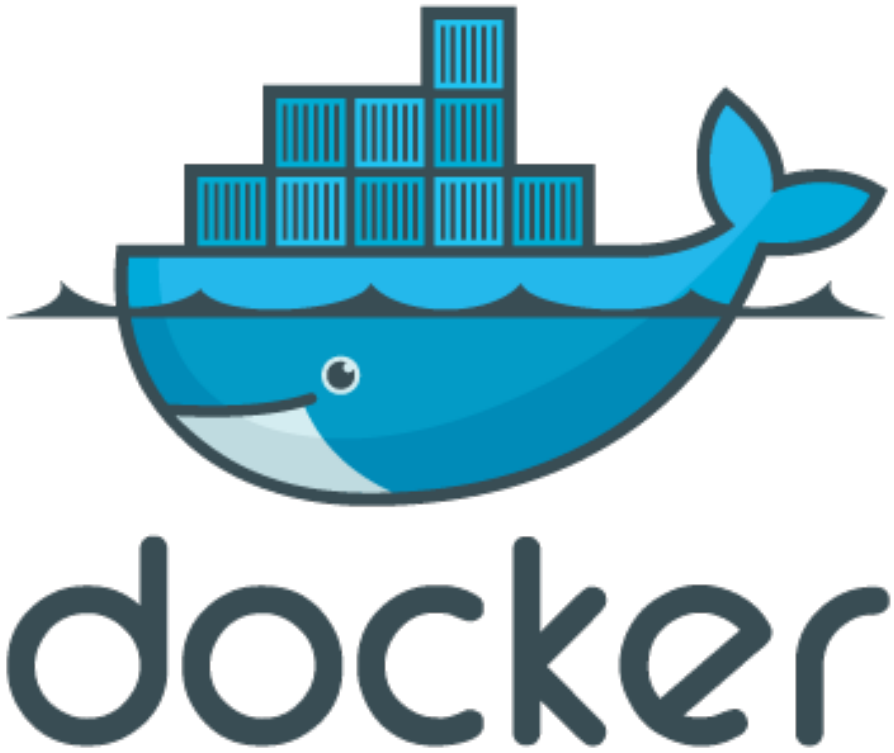
Hidden infrastructure

bcbio-nextgen

- CloudBioLinux drives fully automated installation
- Reproducible build scripts for docker migration

https://github.com/chapmanb/bcbio-nextgen

Containers



http://docker.io/

https://github.com/chapmanb/bcbio-nextgen-vm

Galaxy toolshed integration vision

- CloudBioLinux flavor to install tools
- Install in isolated Docker container
- Galaxy support for Docker

https://bitbucket.org/galaxy/galaxy-central/pull-request/401/allow-tools-and-deployers-to-specify

Manifest

- Full manifest of installed software
- Prioritize biological software
- YAML format for parsing and downstream queries

Outline

- Overview of CloudBioLinux
- Create custom set of tools
- Install from custom flavor
- Add a new tool

CloudBioLinux architecture

- YAML configuration
- Flavors
- Fabric scripts
- Documentation

What is a flavor?

- Subset of full CloudBioLinux packages
- Defined set of packages for a task

YAML configuration: directory

packages-debian.yaml	Remap debian packages not available from latest ubuntu list. Thanks t…
packages-homebrew.yaml	Update to latest version of snpEff, GEMINI and VEP. Prefer brew insta…
packages-nix.yaml	Disable default use of Nix Packages
packages-scientificlinux.yaml	Fill in some missing gaps between Ubuntu and CentOS/SL (base R). Add …
packages-yum.yaml	Ensure ruby installed for bootstrapping homebrew on bare machines dur…
packages.yaml	Avoid estscan install which is problematic on Ubuntu 14.04. Fixes #163
perl-libs.yaml	Mega patch of fixes to get installation working cleanly on localhost …
puppet_classes.yaml	Initial work LWR integration.
python-libs.yaml	fixed issues with libraries missing from pypi by adding --allow-unver…
r-libs.yaml	updated yaml; sacCer2 => sacCer3
ruby-libs.yaml	Mega patch of fixes to get installation working cleanly on localhost …

YAML configuration: example

```yaml
# Packages available in the Homebrew and Linuxbrew package manager
---
bio_nextgen:
  alignment:
    - bwa
    - bowtie2
    - novoalign
    - rna-star
  utilities:
    - bamtools
    - bedtools
    - cramtools
    - libmaus
    - biobambam
    - fastqc
    - fastx_toolkit
    - qualimap
    - sambamba
    - staden_io_lib
  analysis:
    #- cufflinks
    - samtools
    - htslib
    - bcftools
    #- tophat
  variant:
```

Example flavor

branch: **master** **cloudbiolinux** / contrib / flavor / **ngs_pipeline_minimal** / +

Update to latest version of snpEff, GEMINI and VEP. Prefer brew insta… ···

 **chapmanb** authored 22 days ago

..

custom.yaml	Remove pbgzip which fails to compile on Mac and is not currently used…
main.yaml	Add bioconductor libraries to bcbio-nextgen installation flavor to st…
packages-homebrew.yaml	Update to latest version of snpEff, GEMINI and VEP. Prefer brew insta…
r-libs.yaml	Add additional R libraries for cn.mops: snow and rtracklayer. Update …

Edit main.yaml

```
---
# Flavor containing with minimal instructions to install tools for
# running next-generation sequencing pipelines.
packages:
  - minimal
  - libraries
  - python
  - java
  - r
  - ruby
  - bio_nextgen
libraries:
  - r-libs
```

Edit set of brew installed packages

```
# Packages available in the Homebrew and Linuxbrew package mar
---
bio_nextgen:
  alignment:
    - bwa
#    - bowtie2       (2.2.0 doesn't work with Tophat, so use t
    - novoalign
    - rna-star
  utilities:
    - bamtools
    - bedtools
    - cramtools
    - libmaus
    - biobambam
    - fastqc
    - qualimap
    - sambamba
    - samblaster
    - seqtk==HEAD
    - speedseq
    - staden_io_lib
```

Edit fabricrc.txt

```
35 # Global installation directory for packages and standard programs
36 system_install = /usr/local
37
38 # Local install directory for versioned software that will not
39 # be included in the path by default
40 local_install = /usr/local/share
41
42 # Shell to be used by CBL scripts during runtime
43 shell_config = ~/.bashrc
44 shell = /bin/bash -i -c
45
46 # Global setting for using sudo; allows installation of custom packages
47 # by non-privileged users.
48 # *Note*: ``system_install`` needs to point to a user-writeable directory if
49 # ``use_sudo`` is set to ``False``
50 use_sudo = True
51
52 # -- Details about reference data installation
53
54 # Path where biological reference data files should be retrieved to
55 data_files = /mnt/biodata
56
```

Outline

- Overview of CloudBioLinux
- Create custom set of tools
- Install from custom flavor
- Add a new tool

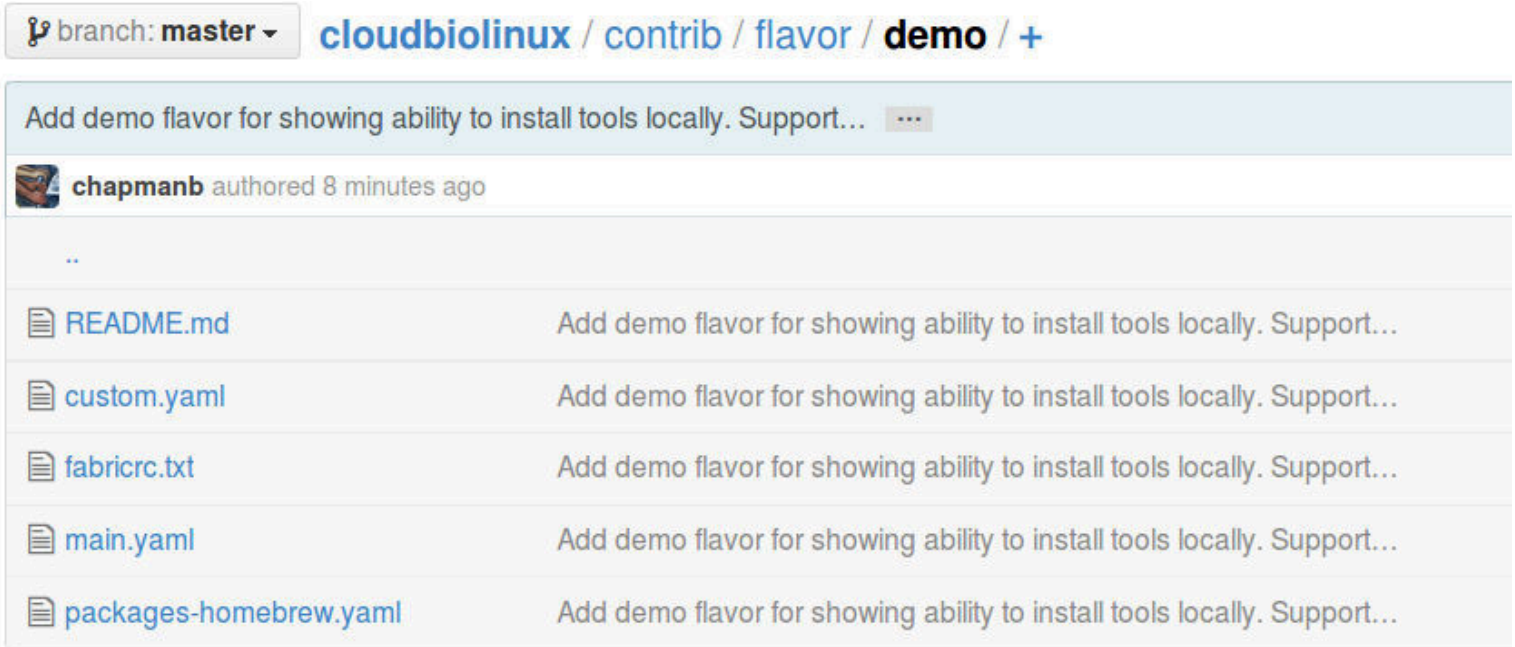
Setup: get CloudBioLinux and Fabric

Retrieve source and fabric for execution

```
$ git clone https://github.com/chapmanb/cloudbiolinux.git
$ pip install fabric
```

https://github.com/fabric/fabric

Short demonstration flavor



contrib/flavor/demo

Install

Single command

```
$ cd cloudbiolinux
$ fab -H localhost install_biolinux:flavor=demo
```

Isolated install directory

```
$ tree -d -L 2 ~/tmp/cbl_demo/
/home/chapmanb/tmp/cbl_demo
|-- bin
|-- Cellar
|   |-- bedtools
|   |-- bwa
|   |-- gatk-framework
|   `-- samtools
|-- include
|   `-- bam -> ../Cellar/samtools/0.1.19/include/bam
|-- lib
|   `-- pkgconfig
|-- Library
|   |-- Aliases
|   |-- Contributions
|   |-- ENV
|   |-- Formula
|   |-- Homebrew
|   |-- LinkedKegs
|   `-- Taps
|-- opt
|   |-- bedtools -> ../Cellar/bedtools/2.19.1
|   |-- bwa -> ../Cellar/bwa/0.7.9a
|   |-- gatk-framework -> ../Cellar/gatk-framework/3.1-1
|   `-- samtools -> ../Cellar/samtools/0.1.19
`-- share
    |-- doc
    |-- java -> ../Cellar/gatk-framework/3.1-1/share/java
    |-- man
    `-- samtools -> ../Cellar/samtools/0.1.19/share/samtools
```

Update paths to include automatically

```
export PATH=~/tmp/cbl_demo/bin:$PATH
export LD_LIBRARY_PATH=~/tmp/cbl_demo/lib:$LD_LIBRARY_PATH
export PERL5LIB=~/tmp/cbl_demo/lib/perl5:
                ~/tmp/cbl_demo/lib/perl5/site_perl:${PERL5LIB}
```

Run

```
$ bedtools
$ samtools
$ gatk-framework
```

Outline

- Overview of CloudBioLinux
- Create custom set of tools
- Install from custom flavor
- Add a new tool

Tool add options

- Add to any existing packaging community
  - DebianMed
  - Bio-Linux
  - Homebrew
- Custom python code
- Will show example with brew recipe

Homebrew/Linuxbrew

Linuxbrew

A fork of Homebrew for Linux

Install Linuxbrew (tl;dr)

Paste at a Terminal prompt:

```
ruby -e "$(wget -O- https://raw.github.com/Homebrew/linuxbrew/go/install)"
```

See Dependencies and Installation below for more details.

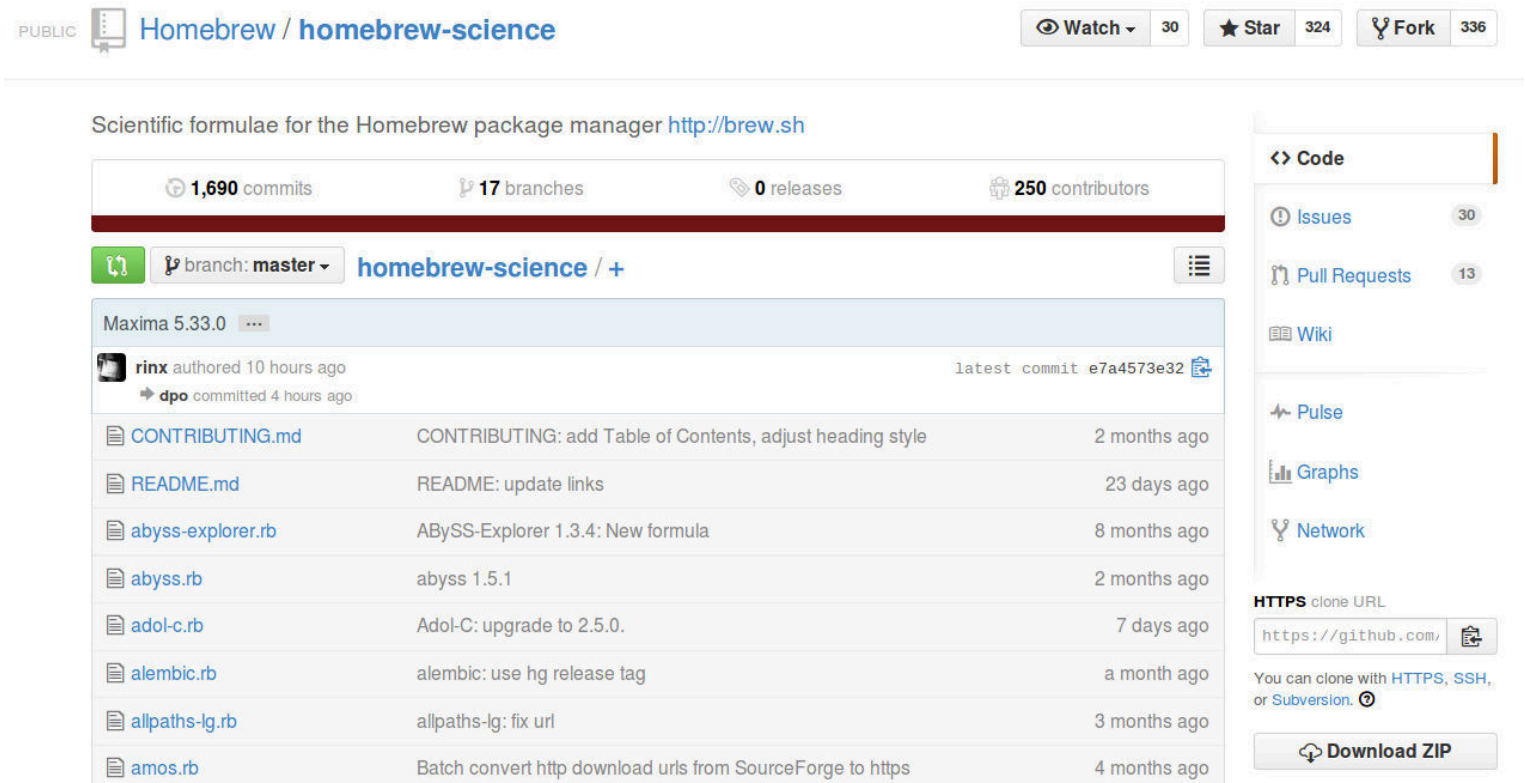
Features

- Can install software to a home directory and so does not require sudo
- Install software not packaged by the native distribution
- Install up-to-date versions of software when the native distribution is old
- Use the same package manager to manage both your Mac and Linux machines

https://github.com/Homebrew/homebrew

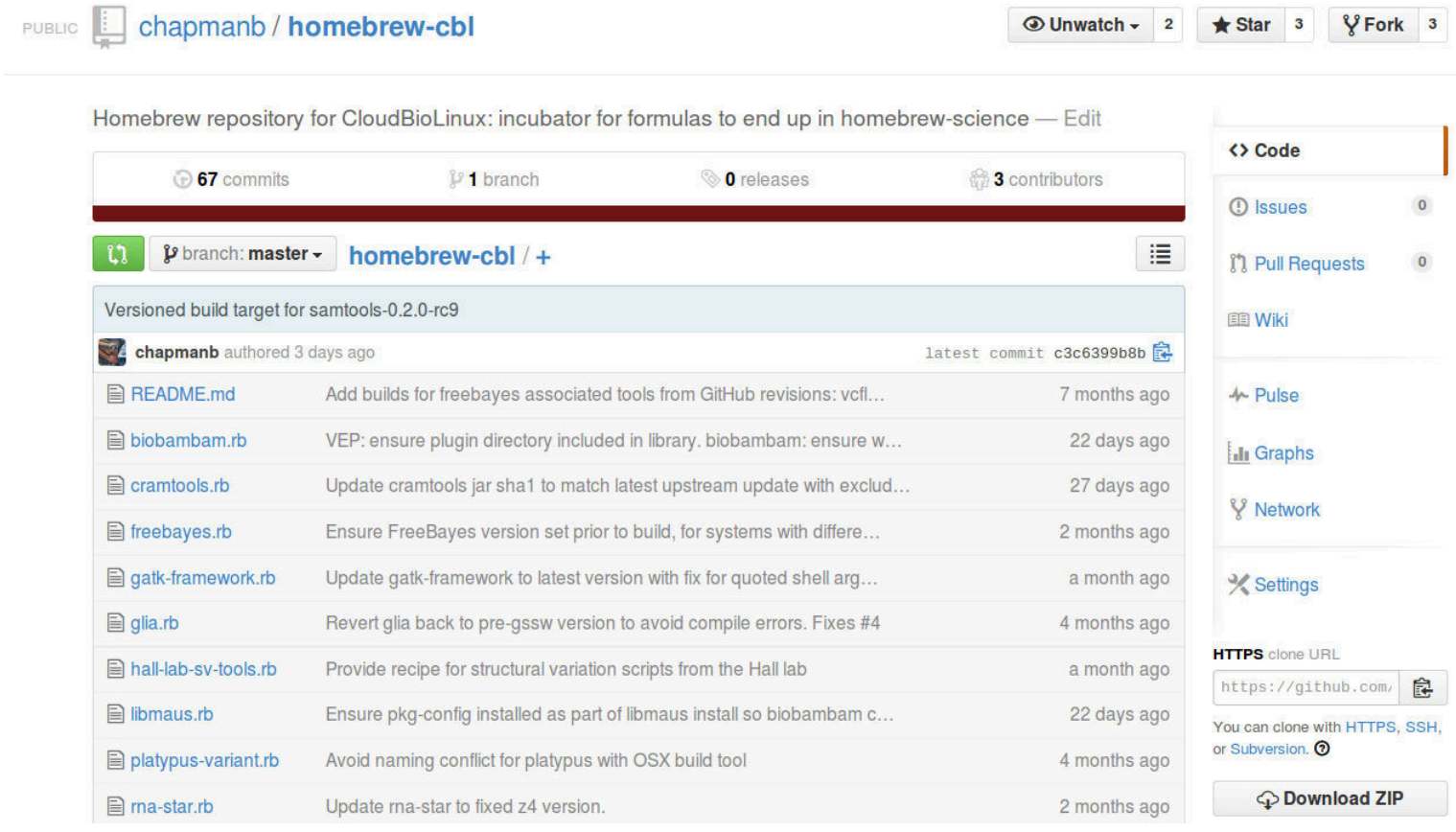
https://github.com/Homebrew/linuxbrew

homebrew-science



https://github.com/Homebrew/homebrew-science

homebrew-cbl



https://github.com/chapmanb/homebrew-cbl

Simple recipe

```ruby
require 'formula'

class Vt < Formula
  homepage 'https://github.com/atks/vt'
  version '2014-04-23'
  url 'https://github.com/atks/vt.git', :revision => '22894f949a'

  def install
    system 'make'
    bin.install 'vt'
  end

  test do
    system 'vt'
  end
end
```

https://github.com/chapmanb/homebrew-cbl/blob/master/vt.rb

Complex recipe

```ruby
require 'formula'

class Vep < Formula
  homepage 'http://ensembl.org/info/docs/variation/vep/index.html'
  version '75_2014-06-12'
  url 'https://github.com/Ensembl/ensembl-tools/archive/771dfa1016c357145be7016c91e1155ae7c021f2.zip'
  sha1 '141a8c639c442bf02d15846fab534dbe58dec3e4'

  resource "plugins" do
    url "https://github.com/ensembl-variation/VEP_plugins/archive/2c123faff2deef07ee094984fc44e19c48975af4.zip"
    sha1 "0569239ed8255d277db034d838d6ec51b90481a8"
  end

  resource 'loftee' do
    url 'https://github.com/konradjk/loftee/archive/545cf9ac5f25b6a6872984dd1a3197a7e7caf000.zip'
    sha1 'a61c6196964526becfbe7afa9838d0495d7996e9'
  end

  def install
    # VEP
    inreplace 'scripts/variant_effect_predictor/variant_effect_predictor.pl' do |s|
      s.sub! 'use lib $Bin;', "use lib $Bin;\nuse lib '#{prefix}/lib';\nuse lib '#{prefix}/lib/Plugins';\n;"
      s.sub! "my $default_dir = join '/', ($ENV{'HOME'}, '.vep')", "my $default_dir = '#{prefix}/lib';"
    end
    inreplace 'scripts/variant_effect_predictor/INSTALL.pl' do |s|
      s.sub! "$DEST_DIR ||= '.';", "$DEST_DIR ||= '#{prefix}/lib';"
      s.sub! "$CACHE_DIR    ||= $ENV{HOME} ? $ENV{HOME}.'/.vep' : 'cache';", "$CACHE_DIR    ||= '#{share}/data';"
    end
    inreplace 'scripts/variant_effect_predictor/convert_cache.pl' do |s|
      s.sub! 'use strict;', "use strict;\nuse lib '#{prefix}/lib';"
    end
    prefix.install Dir['scripts/variant_effect_predictor/*.pl']
    bin.install_symlink prefix / 'variant_effect_predictor.pl'
    bin.install_symlink prefix / 'filter_vep.pl'
    bin.install_symlink prefix / 'INSTALL.pl' => 'vep_install.pl'
```

https://github.com/chapmanb/homebrew-cbl/blob/master/vep.rb

Recap

- Overview of CloudBioLinux
- Create custom set of tools
- Install from custom flavor
- Add a new tool