cGENIE (Cupcake)
Mac OS X Instructions

Gregory J. L. Tourte

29th May 2015

These instructions have been tested on Mac OS X Yosemite (10.10) but should work on Mavericks (10.9) as well. However we recommend that you use the latest version of the OS available as we may not be able to reproduce errors and bugs you may report.

The following instructions will require you to use the Mac OS X Terminal application. This can be opened by using the ⌘ + Space key combination and searching for '*Terminal*', alternatively it can be found in the `Macintosh HD ▸ Applications ▸ Utilities` directory. We suggest you pin the application to your Dock for easy access if you haven't done so already.

1 System Requirements

To install and run cGENIE on Mac OS X you will need the following packages installed:

- Apple Xcode
- Xcode Command line Tools
- the MacPorts environment
- the HomeBrew packaging system
- python
- scons
- gfortran compiler
- wget
- NetCDF libraries

1.1 Apple Xcode

Apple Xcode can be downloaded for free from the Apple App Store. More details can be found at `https://developer.apple.com/xcode/downloads/`. XCode contains the GNU C Compiler (`gcc`) and most of the other libraries and tools to allow the compilation of the model.

In addition to Xcode, you will need to install the Xcode Command Line Tools. These used to be installed by default on older versions of Xcode but are now distributed separately. Once Xcode is installed, you will need to run the following command in the terminal:

```
$ xcode-select --install
```

You can check if Xcode is properly installed using the command:

```
$ xcode-select -p
```

which should contain the following line in the response:

```
/Applications/Xcode.app/Contents/Developer
```

While we are checking the environment, you can check that gcc is installed properly:

```
$ gcc --version
Configured with: --prefix=/Applications/Xcode.app/Contents/Developer/usr
 ↪  --with-gxx-include-dir=/usr/include/c++/4.2.1
Apple LLVM version 6.1.0 (clang-602.0.49) (based on LLVM 3.6.0svn)
Target: x86_64-apple-darwin14.3.0
Thread model: posix
```

1.2 Homebrew & MacPorts

Homebrew is a package manager for OS X which allows users to download and install packages found in other UNIX style environment such as Linux and keep them up to date in a managed way in the sense that one can update packages and manage dependencies. MacPort has a similar aim but with a slightly different philosophy. We will not compare these two here as it goes beyond the scope of this document but we have used both to install software required to run cGENIE on the Mac.

Homebrew and information about it can be found at `http://brew.sh/`. To install the environment, simply type the following line in the Terminal:

```
$ ruby -e "$(curl -fsSL
 ↪  https://raw.githubusercontent.com/Homebrew/install/master/install)"
```

MacPort and information about it can be found at `https://www.macports.org/` and instructions on how to install it are found at `https://www.macports.org/install.php`. Following the installation, you should make sure your installation is fully up to date by running:

```
$ sudo port -v selfupdate
```

1.3 Python

The install scripts for cGENIE are written in `python` and require version 2.7.9. Mac OS X comes with `python` installed by default since version 10.8 (Mountain Lion) but it is not the correct version (Yosemite comes with version 2.7.6). You will therefore need to install an other version of `python`. The homebrew package is the correct version so we suggest you install this one.

```
$ brew install python
```

To check which version of `python` you are running:

```
$ python -V
Python 2.7.9
```

1.4 SCons

`SCons` is a software construction tools built in `python` which role and functionality is similar to the `autotool`/`automake` and `make` tool chain. cGENIE Cupcake does not use `make` any more and is built on `SCons` instead.

As with previous tools, we will use the package provided by Homebrew:

```
$ brew install scons
```

You can then test your installations with:

```
$ scons --version
SCons by Steven Knight et al.:
   script: v2.3.4, 2014/09/27 12:51:43, by garyo on lubuntu
   engine: v2.3.4, 2014/09/27 12:51:43, by garyo on lubuntu
   engine path: ['/usr/local/Cellar/scons/2.3.4/libexec/scons-local/SCons']
Copyright (c) 2001 - 2014 The SCons Foundation
```

1.5 Fortran Compiler (gfortran)

We will get gfortran from homebrew. This is simply done by using the command:

```
$ brew install gfortran
```

it will download and install all the dependencies as well as the request software itself. At the time of writing, the version of gfortran available on the brew servers is 4.9.2.

You can check that gfortran is properly installed by issuing the command:

```
$ gfortran --version
GNU Fortran (Homebrew gcc 4.9.2_1) 4.9.2
Copyright (C) 2014 Free Software Foundation, Inc.

GNU Fortran comes with NO WARRANTY, to the extent permitted by law.
You may redistribute copies of GNU Fortran
under the terms of the GNU General Public License.
For more information about these matters, see the file named COPYING
```

1.6 wget

wget is a command line tool to download files hosted online. It is used by cGENIE install scripts to fetch files it requires. This programme is available as a Homebrew package:

```
$ brew install wget
```

1.7 NetCDF Libraries

For NetCDF we will use the version available from MacPorts (we have tried using Homebrew for this in order to minimise the requirements and not mix the packaging environment but unfortunately, the NetCDF brew package could not be installed on our test machine. If this changes, we will reevaluate our instructions.

Although previous versions of cGENIE required the C, C++ and fortran versions of the library, this is no longer the case and the C++ layer is not a requirement anymore. The following commands will install the necessary NetCDF libraries:

```
$ sudo port install netcdf
$ sudo port install netcdf-fortran
```

2 Installing and running cGENIE (cupcake)

With all the steps described in this document, you should be able to follow the instructions described in the `cupcake-config-build.pdf` file in this folder to install and run the model.

If you have any issues with running cGENIE (cupcake) on Mac OS X, please report them at `https://github.com/genie-model/cgenie/issues`.

3 Quirks

Because so much of the system depends on third party repositories, it may not always behave as expected. We are aware that MacPorts for example provide packages for most of the tools we get from Hombrew, however, we have had reports of failures on systems with MacPorts packages where installing the Homebrew package was a solution. This is especially valid for python.

We realised that making users change there tool chains is very much a annoyance. We are trying to test multiple install setup and try to understand why one tool chain fails where the other doesn't, but we currently lack the resources to do so quickly.

The information in this document is valid at the time of writing with the version of packages available at that time. Since we do not control the packages distributed on either Homebrew or MacPorts, we cannot tell whether updated versions of these packages will still work. We will endeavour to follow the releases of the packages with use and test any upgrade but it is not guaranteed happen in a timely manner. The main victim of that is python, the version of which is currently hardcoded in the code and is required to be 2.7.9.