

Installation report for Cadfaelbrew on Xubuntu Linux 14.04 (LTS)

F.Mauger
mauger@lpccaen.in2p3.fr

2016-03-09

Version: 0.1

This note explains the Cadfaelbrew installation procedure on Xubuntu Linux 14.04 (LTS) (64-bits). Cadfaelbrew is a package manager, derived from the Linux/Homebrew projects. It allows you to install various software *formulas*. Thanks to the generic `brew` utility, you will be able to install the *Cadfael* formula which includes Boost, CAMP, CLHEP, GSL, ROOT, Geant4... that is all the software you need to run Bayeux and the SuperNEMO software. Other formulas can also be added (Bayeux, Falaise...). This is out of the scope of this document which concentrates on the installation of the basic layer of Cadfaelbrew and Cadfael formulas.

Cadfaelbrew is hosted at <https://github.com/SuperNEMO-DBD/>.

Ben Morgan (Ben.Morgan@warwick.ac.uk) is the developer and maintainer of Cadfaelbrew for the SuperNEMO collaboration.

Contents

Preparation	1
Download the Cadfael installer	2
Prepare the installation	2
Build and install Cadfaelbrew in C++11 mode	3
Install the <code>cadfael</code> formula	3
Setup Cadfaelbrew for your environment	4
Additional operations after installation of the <code>cadfael</code> formula	4
Geant4	4

Preparation

a. Requirements:

- A working network connection
- Bash shell

b. The working directory:

In this report, the `/data3` directory will be used as the base working directory to build and install the software. You are free to choose any other location on your system, provided there is enough space on it (~10 GB).

```
bash$ mkdir -p /data3/sw/Cadfaelbrew
bash$ cd /data3/sw/Cadfaelbrew
```

c. Install mandatory packages:

```
bash$ LANG=C sudo apt-get install build-essential curl git m4 \
    ruby texinfo libbz2-dev libcurl4-openssl-dev \
    libexpat-dev libncurses-dev zlib1g-dev
```

Download the Cadfael installer

The Cadfael installer utility is available from the SuperNEMO-DBD git repository (GitHub). You must have a recent version of git installed on your system. The installer will first install Cadfaelbrew before the Cadfael software itself.

- a. Clone the default Git branch to your local filesystem:

```
bash$ LANG=C git clone https://github.com/SuperNEMO-DBD/cadfael-installer.git \
    cadfael-installer.git
Cloning into 'cadfael-installer.git'...
remote: Counting objects: 295, done.
remote: Total 295 (delta 0), reused 0 (delta 0), pack-reused 295
Receiving objects: 100% (295/295), 55.80 KiB | 0 bytes/s, done.
Resolving deltas: 100% (156/156), done.
Checking connectivity... done.
```

- b. Explore what you have downloaded:

```
bash$ LANG=C tree -L 1 -F cadfael-installer.git/
cadfael-installer.git/
|-- LICENSE
|-- README.md
|-- cadfael-installer*
`-- containers/
```

Prepare the installation

The following steps are not mandatory. It depends on the available storage on your filesystem. By default, Cadfaelbrew uses the `~/ .cache/Homebrew` directory as the download cache directory. If your home directory has not enough space left, you will need to define the `HOMEbrew_CACHE` environmental variable to another location of your filesystem.

Also Cadfaelbrew uses the `/tmp` directory as the temporary location to build the software. If the disk partition hosting this directory has not enough space left, you will need to define the `HOMEbrew_TEMP` environmental variable.

Examples:

- a. Here we create a directory that will be used as the cache directory to download source tarballs. This directory must be large enough to endure at least 500 MB of downloaded files (source tarballs, patch files, Geant4 datasets...).

```
bash$ mkdir -p /data3/var/cache/Homebrew
bash$ export HOMEbrew_CACHE=/data3/var/cache/Homebrew
```

- b. Here we create a directory that will be used as the temporary build directory for all software packages (formulas). This directory must be large enough to endure at least 4 GB of temporary files (mainly because compiling GCC uses a huge amount of disk space).

```
bash$ mkdir -p /data3/var/tmp
bash$ export HOMEbrew_TEMP=/data3/var/tmp
```

Build and install Cadfaelbrew in C++11 mode

The `cadfael-installer` script is provided to automate the installation of Cadfaelbrew and also some software packages (formulas) used by the SuperNEMO software (example: tagged version of Bayeux...).

- a. You can print some basic online help with:

```
bash$ cd cadfael-installer.git/  
bash$ ./cadfael-installer -h
```

There is only a very small set of options for now.

- b. Here we request a bare installation (`-b`) of Cadfaelbrew without installing specific formulas for SuperNEMO software (like Bayeux). We build the software with C++11 (`-e`). The installation prefix is set with the `-p` switch:

```
/data3/sw/Cadfaelbrew/supernemo/cxx11.
```

Run:

```
bash$ cd cadfael-installer.git/  
bash$ ./cadfael-installer -b -e -p /data3/sw/Cadfaelbrew/supernemo/cxx11
```

At the end of the process, the Cadfaelbrew environment will be installed in :

```
/data3/sw/Cadfaelbrew/supernemo/cxx11/Cadfael.git/
```

You will find the `brew` script in the `bin/` subdirectory.

Install the `cadfael` formula

Now Cadfaelbrew is available, we can install the formula for the Cadfael software:

```
bash$ /data3/sw/Cadfaelbrew/supernemo/cxx11/Cadfael.git/bin/brew \  
install cadfael
```

This is a rather long process, particularly because the GCC compiler (version 4.9) is built and installed (~35 min on this system). Geant4 and Root are also rather long to build. All the software will be installed in `/data3/sw/Cadfaelbrew/`. It takes approximatively 3 GB.

The following report gives you an idea of the time needed to build the `cadfael` formula on a system with four i7-3540M processors @ 3 GHz:

```
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/binutils/2.25.1: 3031 files, 194M, built in 2.6 minutes  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/gcc49/4.9.2_2: 1111 files, 649M, built in 35.2 minutes  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/pkg-config/0.28: 10 files, 644K, built in 28 seconds  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/ncurses/6.0.1: 1977 files, 19M, built in 47 seconds  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/readline/6.3.8: 40 files, 1,7M, built in 9 seconds  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/unzip/6.0: 14 files, 432K, built in 2 seconds  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/sqlite/3.8.11.1: 9 files, 2,7M, built in 45 seconds  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/gdbm/1.11: 27 files, 828K, built in 6 seconds  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/makedepend/1.0.5: 7 files, 112K, built in 10 seconds  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/zlib/1.2.8: 9 files, 416K, built in 2 seconds  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/openssl/1.0.2d_1: 464 files, 16M, built in 2.8 minutes  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/python/2.7.10_2: 5374 files, 112M, built in 95 seconds  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/curl/7.45.0: 355 files, 3,4M, built in 44 seconds  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/sphinx-doc/1.3.1: 2618 files, 36M, built in 4 seconds  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/cmake/3.4.0: 1980 files, 31M, built in 4.7 minutes  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/xz/5.2.2: 65 files, 1,9M, built in 17 seconds  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/gettext/0.19.6: 1918 files, 23M, built in 2.5 minutes  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/flex/2.5.39: 45 files, 2,3M, built in 11 seconds  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/bison/3.0.4: 117 files, 3,3M, built in 34 seconds  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/doxygen/1.8.10: 8 files, 13M, built in 64 seconds  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/gsl/1.15: 250 files, 9,6M, built in 72 seconds  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/boost/1.55.0_2: 9957 files, 130M, built in 3.4 minutes  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/camp/0.8.0: 110 files, 1,2M, built in 19 seconds  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/clhep/2.1.3.1: 328 files, 12M, built in 101 seconds  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/xerces-c/3.1.2: 493 files, 22M, built in 99 seconds  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/geant4/9.6.4_3: 28278 files, 1,4G, built in 23.3 minutes  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/libpng/1.6.19: 17 files, 1,3M, built in 44 seconds  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/freetype/2.6.1: 60 files, 2,8M, built in 13 seconds  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/root5/5.34.30: 4607 files, 174M, built in 13.6 minutes  
/data3/sw/CadfaelBrew/supernemo/cxx11/Cadfael.git/Cellar/cadfael/2016.01: 4 files, 20K, built in 2 seconds
```

Setup Cadfaelbrew for your environment

To activate Cadfaelbrew in your environment, you have two possibilities:

- a. You can use the `brew.sh` script provided by Cadfaelbrew. In your `~/ .bashrc` startup file, define:

```
alias brewsh='/data3/sw/Cadfaelbrew/supernemo/cxx11/Cadfael.git/bin/brew sh \
--cc=gcc-4.9'
```

Then, each time you need to use Cadfaelbrew or some software managed by it, you just type in your shell:

```
bash$ brewsh
bash$ # ... you can now work with brewed software ...
```

This will start a dedicated shell with all Cadfaelbrew software activated. To leave this environment, type:

```
bash$ exit
```

- b. Alternatively, you can define a specific Bash function in your `~/ .bashrc` startup file:

```
function do_cadfaelbrew_setup()
{
    if [ -n "${CADFAELBREW_INSTALL_DIR}" ]; then
        echo "ERROR: Cadfaelbrew is already setup !" >&2
        return 1
    fi
    export CADFAELBREW_INSTALL_DIR=\
        "/data3/sw/Cadfaelbrew/supernemo/cxx11/Cadfael.git"
    export PATH="${CADFAELBREW_INSTALL_DIR}/bin:${PATH}"
    export MANPATH="${CADFAELBREW_INSTALL_DIR}/share/man:${MANPATH}"
    export INFOPATH="${CADFAELBREW_INSTALL_DIR}/share/info:${INFOPATH}"
    echo "NOTICE: Cadfaelbrew is now setup !" >&2
    return
}
export -f do_cadfaelbrew_setup
```

You can activate thus Cadfaelbrew from your shell with :

```
bash$ do_cadfaelbrew_setup
```

Additional operations after installation of the `cadfael` formula

Geant4

At the time I have installed this version of Cadfaelbrew, the Geant4 version shipped with Cadfaelbrew was not including by default the Geant4 OpenGL/X11 visualization library module on Linux. Ben has added an option (`--with-opengl-x11`) and I was able to rebuild Geant4 with this feature explicitly activated. This allows interactive Geant4 session with visualization.

To rebuild Geant4 with brew:

```
bash$ export HOMEBREW_CACHE=/data3/var/cache/Homebrew
bash$ export HOMEBREW_TEMP=/data3/var/tmp
bash$ brewsh
bash$ brew reinstall geant4 --with-opengl-x11
```

Unfortunately, this operation rebuilds Geant4 from scratch, so it is rather long.