Installation report for Cadfaelbrew on SL6.7 at CCIN2P3

F.Mauger
mauger@lpccaen.in2p3.fr

2016-03-03

Version: 0.1

This note explains the Cadfaelbrew installation procedure on the `ccage` 64-bit cluster running SL6.7. This document concentrates on the installation of the basic layer of Cadfaelbrew and Cadfael.

Contents

Preparation

a. Set a minimal environmental because CCIN2P3 use a default setup with some third party software that *pollutes* the basic `PATH`:

```
bash$ export PATH="/usr/local/bin:/bin:/sbin:/usr/bin:/usr/sbin"
```

b. The working directory:
In this report, the `/sps/nemo/sw/dev` 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 /sps/nemo/sw/dev/Cadfaelbrew
bash$ cd /sps/nemo/sw/dev/Cadfaelbrew
```

b. About mandatory packages...
As we have no sysadmin priviledge on the CCIN2P3 cluster, we cannot install the dependency packages by ourselves (see below).

Download the Cadfael installer

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

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

Prepare the installation

Setup the HOMEBREW_CACHE and HOMEBREW_TEMP environmental variables:

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 /sps/nemo/sw/dev/Cadfaelbrew/_var
bash$ export HOMEBREW_CACHE=/sps/nemo/sw/dev/Cadfaelbrew/_var
```

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 6 GB of temporary files (mainly because compiling GCC uses a huge amount of disk space).

```
bash$ mkdir -p /sps/nemo/sw/dev/Cadfaelbrew/_tmp
bash$ export HOMEBREW_TEMP=/sps/nemo/sw/dev/Cadfaelbrew/_tmp
```

Build and install Cadfaelbrew in C++03 (legacy) mode

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

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++03. The installation prefix is set with the -p switch:

   /sps/nemo/sw/dev/Cadfaelbrew/supernemo/cxx03.

   Run:

   ```
   bash$ cd cadfael-installer.git/
   bash$ ./cadfael-installer -c -b -p /sps/nemo/sw/dev/Cadfaelbrew/supernemo/cxx03
   ```

   Note that we added the -c switch in the cadfael-installer script in order to skip the system check step that analyzes the missing packages on SL6.7 and abort in case of problem (here we cannot automatically install missing packages through yum or rpm because we haev no root proviledge). On the ccage system, some RPM packages are missing but we will request their installation through brew.

   The cadfael-installer builds and installs the following formulas:

   ```
   pkg-config
   homebrew/dupes/ncurses
   readline
   libyaml
   makedepend
   zlib
   openssl
   ruby
   ```

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

   /sps/nemo/sw/dev/Cadfaelbrew/supernemo/cxx03/Cadfael.git/

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

Install the m4 formula

Note : The following command does not work at CCIN2P3 because system startup scripts seems to discard the PATH after brew has defined one:

```
bash$ /sps/nemo/sw/dev/Cadfaelbrew/supernemo/cxx03/Cadfael.git/bin/brew sh
```

So we use this technique:

```
bash$ export PATH="/sps/nemo/sw/dev/Cadfaelbrew/supernemo/cxx03/Cadfael.git/bin:
bash$ export MANPATH="/sps/nemo/sw/dev/Cadfaelbrew/supernemo/cxx03/Cadfael.git/s
bash$ export INFOPATH="/sps/nemo/sw/dev/Cadfaelbrew/supernemo/cxx03/Cadfael.git/
bash$ echo $HOMEBREW_TEMP
/sps/nemo/sw/dev/Cadfaelbrew/_tmp/
bash$ echo $HOMEBREW_CACHE
/sps/nemo/sw/dev/Cadfaelbrew/_var/
```

Install the m4 formula (version 1.4.17):

```
bash$ brew install m4
```

This installs:

```
xz
m4
```

Install other core formulas

The following formulas are installed step by step through brew:

```
bash$ brew install binutils
bash$ brew install gcc49
bash$ brew install unzip
bash$ brew install sqlite
bash$ brew install gdbm
bash$ brew install python
bash$ brew install curl
bash$ brew install sphinx-doc
bash$ brew install cmake
bash$ brew install gettext
bash$ brew install flex
bash$ brew install bison
bash$ brew install doxygen
```

Install the Cadfael formulas

Now the core packages are available, we can install the formulas published by Cadfaelbrew:

```
bash$ brew install gsl
bash$ brew install boost
bash$ brew install camp
bash$ brew install clhep
bash$ brew install xerces-c
bash$ brew install libpng
bash$ brew install geant4 --with-opengl-x11
bash$ brew install freetype
bash$ brew install root5 --HEAD
bash$ brew install cadfael
```

Note that Root was installed from a specific branch of the Git source repository because the 5.34 release (through a tarball) failed to build.

All the software will be installed in `/sps/nemo/sw/dev/Cadfaelbrew/supernemo/cxx11/Cadfael.git`. It takes approximatively 3 GB.

For example you will be able to locate the `gsl-xxx` executable as well as the `clhep-config` utility.

Setup Cadfaelbrew for your environment

To activate Cadfaelbrew in your environement, a specific Bash function has been defined in the `~/.bashrc` startup file:

```
function do_cadfaelbrew_dev_setup()
{
   if [ -n "${CADFAELBREW_DEV_INSTALL_DIR}" ]; then
       echo "ERROR: Cadfaelbrew/dev is already setup !" >&2
```

```
        return 1
    fi
    export CADFAELBREW_DEV_INSTALL_DIR=\
      "/sps/nemo/sw/dev/Cadfaelbrew/supernemo/cxx11/Cadfael.git"
    export PATH="${CADFAELBREW_DEV_INSTALL_DIR}/bin:${PATH}"
    export MANPATH="${CADFAELBREW_DEV_INSTALL_DIR}/share/man:${MANPATH}"
    export INFOPATH="${CADFAELBREW_DEV_INSTALL_DIR}/share/info:${INFOPATH}"
    echo "NOTICE: Cadfaelbrew/dev is now setup !" >&2
    return 0
}
export -f do_cadfaelbrew_setup
```

Cadfaelbrew is thus activated from the bash with :

```
bash$ do_cadfaelbrew_setup
```

Issues

m4

Because of some problem at the configuration step for `bison`, `m4-1.4.17` was first installed manually to check the build of `bison`. Finally, the `m4` formula was found available from Cadfaelbrew so it was used to install a recent version of `m4` before to install the other formulas.

root5

The `root5` formula initially failed. Root 5.34 was thus installed manually to check possible issues on the system. After some tests, it was found that only the source code from the Git repository was working on this system, using the `v5-34-00-patches` branch.

Procedure:

```
$ mkdir /sps/nemo/sw/dev/root5
$ cd /sps/nemo/sw/dev/root5
$ git clone https://github.com/root-mirror/root.git
$ cd root
$ git checkout v5-34-00-patches
$ git status
```

Edit "cmake/modules/RootBuildOptions.cmake" and replace:

```
if(NOT opt MATCHES "thread|cxx11|cling|builtin_llvm|builtin_ftgl|explicitlink")
```

by:

```
if(NOT opt MATCHES "thread|cxx11|cling|builtin_llvm|builtin_ftgl|explicitlink|gnu
```

```
$ cd ..
$ mkdir _build.d
$ cd _build.d
$ cmake  \
  -DCMAKE_C_FLAGS_RELEASE= \
  -DCMAKE_CXX_FLAGS_RELEASE= \
  -DCMAKE_INSTALL_PREFIX=/sps/nemo/sw/dev/root5/install-5.34.30 \
  -DCMAKE_BUILD_TYPE=Release \
  -DCMAKE_FIND_FRAMEWORK=LAST \
  -DCMAKE_VERBOSE_MAKEFILE=ON \
  -Wno-dev -DCMAKE_INSTALL_RPATH_USE_LINK_PATH=1 \
  -Dgnuinstall=ON \
  -DCMAKE_INSTALL_SYSCONFDIR=etc/root \
  -Dgminimal=ON \
  -Dcxx11=OFF \
  -Dfortran=OFF \
  -Dpython=OFF \
  -Drpath=ON \
  -Dsoversion=ON \
  -Dasimage=ON \
  -Dbuiltin_asimage=ON \
  -Dbuiltin_freetype=ON \
  -Dcastor=OFF \
  -Dgdml=ON \
  -Doracle=OFF \
  -Drfio=OFF \
  ../root
$ make
$ make install
```

This procedure was succesful and proved that it was possible to install `root5` through `brew` using the `--HEAD` switch:

```
$ brew install root5 --HEAD
```