Vire/trunk installation report on (X)ubuntu 16.04 LTS (64bits)

François Mauger, LPC Caen
mauger@lpccaen.in2p3.fr

2016-09-22

Version: 1.0

In this document we propose an installation procedure for the Vire/trunk library on top of the Bayeux/trunk and Cadfaelbrew on Xubuntu 6.04 LTS (Xenial Xerus) for a system (64-bits). By default, the build is done using the C++11 standard.

Contents

The target system

- Architecture:

  ```
  $ uname -a
  Linux mauger-laptop 4.4.0-34-generic #53-Ubuntu SMP Wed Jul 27 16:06:39 UTC 2016 x86_64 x86
  ```

- Processors:

  ```
  $ cat /proc/cpuinfo | grep "model name"
  model name    : Intel(R) Core(TM) i7-3540M CPU @ 3.00GHz
  model name    : Intel(R) Core(TM) i7-3540M CPU @ 3.00GHz
  model name    : Intel(R) Core(TM) i7-3540M CPU @ 3.00GHz
  model name    : Intel(R) Core(TM) i7-3540M CPU @ 3.00GHz
  ```

- Linux version:

  ```
  $ cat /etc/lsb-release
  DISTRIB_ID=Ubuntu
  DISTRIB_RELEASE=16.04
  DISTRIB_CODENAME=xenial
  DISTRIB_DESCRIPTION="Ubuntu 16.04.1 LTS"
  ```

- Dependencies:

Vire depends on Bayeux (and Cadfaelbrew). To install both packages, please follow the instructions from:

  - Bayeux (SuperNEMO Wiki)
  - Cadfael (SuperNEMO Wiki)
  - Cadfaelbrew repository (GitHub)
  - **Protobuf_** repository (GitHub)

- System packages:

Setup of Cadfaelbrew and Bayeux/trunk

You must have installed a standalone Bayeux/trunk on top of Cadfaelbrew.

Once you have installed Cadfaelbrew and Bayeux, you should be able to setup Bayeux:

```
$ brewsh            # Enter a *brew shell*
$ bayeux_dev_setup # Activate Bayeux
```

You can check the location and version of core software utilities:

```
$ which cmake
/opt/sw/SuperNEMO-DBD/Cadfaelbrew/bin/cmake

$ cmake --version
cmake version 3.6.1

$ g++ --version
 g++ (Ubuntu 5.4.0-6ubuntu1~16.04.1) 5.4.0 20160609

$ which bxquery
/opt/sw/Bayeux/Binary/Bayeux-trunk/Install-gcc-cxx11-Linux-x86_64/bin/bxquery
$ bxquery --version
3.0.0
```

Configuration and build of Vire/trunk

Working directory

Set the software base directory where there is enough storage capacity to host Vire (> 100 MB). Here we use a simple environment variable `SW_WORK_DIR` which points to a specific directory on the filesystem:

```
$ export SW_WORK_DIR=/opt/sw
```

You may adapt this base directory to your own system, for example:

```
$ export SW_WORK_DIR=${HOME}/Software
```

Then create a few working directories:

```
$ mkdir -p ${SW_WORK_DIR}
$ mkdir -p ${SW_WORK_DIR}/Vire          # base working directory for Vire
$ mkdir -p ${SW_WORK_DIR}/Vire/Binary  # hosts the build/installation directories
```

Download Vire

Download Vire/trunk source files:

```
$ export VIRE_SOURCE_BASE_DIR="${HOME}/Documents/Private/Software/Vire/Source"
$ export VIRE_DEV_SOURCE_DIR=${VIRE_SOURCE_BASE_DIR}/Vire-trunk
$ mkdir -p ${VIRE_SOURCE_BASE_DIR}
$ cd ${VIRE_SOURCE_BASE_DIR}
$ svn co https://nemo.lpc-caen.in2p3.fr/svn/Vire/trunk Vire-trunk
$ cd Vire-trunk
$ LANG=C svn info
Path: .
Working Copy Root Path: /home/mauger/Documents/Private/Software/Vire/Source/Vire-trunk
URL: https://nemo.lpc-caen.in2p3.fr/svn/Vire/trunk
Relative URL: ^/Vire/trunk
Repository Root: https://nemo.lpc-caen.in2p3.fr/svn
Repository UUID: 3e0f96b8-c9f3-44f3-abf0-77131c94f4b4
Revision: 17390
Node Kind: directory
Schedule: normal
Last Changed Author: mauger
Last Changed Rev: 17390
Last Changed Date: 2016-03-22 12:27:31 +0100 (mar., 22 mars 2016)
```

Configure Vire

1. Make sure Cadfaelbrew, Bayeux and Google Protocol Buffers are setup on your system. If you follow the Cadfaelbrew installation report available from the Bayeux wiki page, you just have to invoke:

```
$ brewsh
$ bayeux_dev_setup
$ protobuf_setup
```

2. Create a build directory and cd in it:

```
$ export VIRE_DEV_BIN_DIR="${SW_WORK_DIR}/Vire/Binary/Vire-trunk"
$ export VIRE_DEV_BUILD_DIR=${VIRE_DEV_BIN_DIR}/Build-gcc-cxx11-ninja-Linux-x86_64
$ mkdir -p ${VIRE_DEV_BUILD_DIR}
$ cd ${VIRE_DEV_BUILD_DIR}
$ pwd
/opt/sw/Vire/Binary/Vire-trunk/Build-gcc-cxx11-ninja-Linux-x86_64
```

3. Configure the Bayeux build with CMake and using Ninja and GCC :

```
$ echo ${CADFAELBREW_INSTALL_DIR}
/opt/sw/SuperNEMO-DBD/Cadfaelbrew
$ gsl-config --prefix
/opt/sw/SuperNEMO-DBD/Cadfaelbrew/Cellar/gsl/1.16
$ bxquery --prefix
/opt/sw/Bayeux/Binary/Bayeux-trunk/Install-gcc-cxx11-Linux-x86_64
$ which protoc
/opt/sw/GoogleProtocolBuffers/install-3.0.0/bin/protoc


$ export VIRE_DEV_INSTALL_DIR="${VIRE_DEV_BIN_DIR}/Install-gcc-cxx11-Linux-x86_64"
$ cmake \
 -DCMAKE_BUILD_TYPE:STRING=Release \
 -DCMAKE_INSTALL_PREFIX:PATH="${VIRE_DEV_INSTALL_DIR}" \
 -DCMAKE_FIND_ROOT_PATH:PATH="$(bxquery --prefix);${CADFAELBREW_INSTALL_DIR}" \
 -DVIRE_COMPILER_ERROR_ON_WARNING=ON \
 -DVIRE_CXX_STANDARD="11" \
 -DVIRE_ENABLE_TESTING=ON \
 -DVIRE_WITH_DOCS=ON \
 -DVIRE_WITH_DEVELOPER_TOOLS=ON \
 -DVIRE_WITH_PLUGINS=OFF \
 -DVIRE_WITH_SANDBOX=OFF \
 -DBoost_DIR:PATH="${CADFAELBREW_PREFIX_DIR}/lib/cmake" \
 -DVIRE_WITH_PROTOBUF_JAVA=ON \
 -DPROTOBUF_ROOT:PATH="/opt/sw/GoogleProtocolBuffers/install-3.0.0" \
 -GNinja \
 ${VIRE_DEV_SOURCE_DIR}
```

Build

Using 4 processors to go faster (depends on your machine):

```
$ time ninja -j4
```

Test programs

Before to do the final installation, we run the test programs:

```
$ ninja test
...
```

Installation

Run:

```
$ ninja install
...
```

Setup your environment for Vire

Here we explicitely *load/setup* the Vire environment from a Bash shell with a dedicated function defined in my `~/.bashrc` startup file:

```
# The base directory of all the software (convenient path variable):
export SW_WORK_DIR=/data/sw
export VIRE_DEV_BIN_DIR="${SW_WORK_DIR}/Vire/Binary/Vire-trunk"

# The Vire/trunk setup function:
function do_vire_trunk_cxx11_setup()
{
 do_bayeux_dev11_setup # Automatically setup the Bayeux dependency
 if [ -n "${VIRE_DEV_INSTALL_DIR}" ]; then
   echo "ERROR: Vire/trunk is already setup !" >&2
   return 1
 fi
 export VIRE_DEV_INSTALL_DIR=${VIRE_DEV_BIN_DIR}/Install-gcc-cxx11-Linux-x86_64
 export PATH=${VIRE_DEV_INSTALL_DIR}/bin:${PATH}
 echo "NOTICE: Vire/trunk is now setup !" >&2
 return;
}
export -f do_vire_trunk_cxx11_setup

# Special alias:
alias do_vire_dev11_setup="do_vire_trunk_cxx11_setup"
alias do_vire_dev_setup="do_vire_trunk_cxx11_setup"
```

When I want to use pieces of software from Vire, I run:

```
$ brewsh
$ do_vire_dev_setup
$ vire-query --help
```

Appendices

Rebuild Vire

In case of problem, you can discard the build directory and retry:

```
$ rm -fr ${SW_WORK_DIR}/Vire/Binary/Vire-trunk/Build-gcc-cxx11-ninja-Linux-x86_64
$ mkdir ${SW_WORK_DIR}/Vire/Binary/Vire-trunk/Build-gcc-cxx11-ninja-Linux-x86_64
$ cd ${SW_WORK_DIR}/Vire/Binary/Vire-trunk/Build-gcc-cxx11-ninja-Linux-x86_64
```

then re-configure and build.

Docutils System Messages

system-message

ERROR/3 in `vire-trunk_xubuntu16.04_report.rst`, line 67

Unknown target name: "protobuf".