Re-compilation of gcc under openSUSE 11.1

Richard J. Mathar[*]
Max-Planck Institute of Astronomy, Heidelberg, Germany

(Dated: April 3, 2014)

Compilation instructions of upgrading the GNU compiler collection from 4.3 to 4.8.1 under a Linux (openSUSE) operating system are summarized.

I. OVERVIEW

Upgrading of the GNU compiler collection needs some time, because the lines in the source code are numerous. The avoid a major loss of time from multiple failures, I summarize one sucessful approach which started from an existing GCC release 4.3 and installed the release 4.8.1 in the local home directory of some anonymous user. “Anonymous” means that no root privileges will be needed for installation into some `/usr` directories, but all files will end up in the home directory. A nice side effect of such a side installation is that switching back to the system’s compiler is easy by uninstalling or disabling these local files with `chmod a-r`.

II. AUXILIARY PACKAGES

To avoid strange interferences from the installed packages, the two major environment variables were cleared first:

```
export PATH=/usr/bin:/usr/local/bin:/bin:/usr/lib64/mpi/gcc/openmpi/bin
export LD_LIBRARY_PATH=""
```

Due to mutual dependencies, the following subsections need basically to be done in the order described.

A. openssl

The following step was needed on a CentOS 6.3 system to let `serf` find a SSL library; it seems that `yum install openssl` does not install any header files. The installation of openssl was not needed under openSUSE. The source code was downloaded from www.openssl.org/source

```
gunzip openssl*.gz
tar xf openssl*.tar
cd openssl-1.0.1f
./config --prefix=$HOME -fPIC
make
make install_sw
```

The last line is not `make install` because `make install` reports errors of the form `Expected text after =item` for newer pod2man as for example installed with openSUSE 13.1. Alternatively, patch the ailing lines in the associated `smime.pod` and the other files by surrounding the numbers in these lines with `C<`*number*`>` and restart another `make install`, see http://lists.freebsd.org/pipermail/freebsd-ports-bugs/2013-August/260952.html.

B. APR

The following step was needed on a openSUSE 12.1 system to let `serf` find the apache runtime library; only the `APR-util` had been available under `/usr/lib64`, but that is insufficient to compile `serf`. The source code was downloaded from apr.apache.org/download.cgi

[*] URL: http://www.mpia.de/~mathar

```
gunzip apr-1.5.0.tar.gz
tar xf apr-1.5.0.tar
cd apr*1.5.0
./configure --prefix=$HOME
make
make install

gunzip apr-iconv-1.2.1.tar.gz
tar xf apr-iconv-1.2.1.tar
cd apr-iconv*
./configure --prefix=$HOME --with-apr=$HOME
make
make install

gunzip apr-util*.gz
tar xf apr-util*.tar
cd apr-util*
./configure --prefix=$HOME --with-apr=$HOME
make
make install
```

C. serf

`serf` had to be upgraded. The source code was downloaded from http://code.google.com/p/serf/downloads/detail?name=serf-1.2.1.tar.bz2. Note that the newer serf-1.3.1 has switched its packaging system to yet another unknown library that I've never heard of and on which I was unwilling to spend additional time.

```
bunzip2 *.bz2
tar xf serf-1.2.1.tar
cd serf-1.2.1
./configure --prefix=$HOME
# ./configure --prefix=$HOME --with-openssl=\$HOME # case of openssl rebuild
# ./configure --prefix=$HOME --with-apr=$HOME --with-apr-util=$HOME # case of APR rebuild
make "LDFLAGS=-L${HOME}/lib64 -L${HOME}/lib"
make install
```

On the said centOS system, an additional `--with-openssl=$HOME` was needed to snap the library build in Section II A.

D. gmp

The GNU multiprecision arithmetic library `gmplib` was upgraded next. The source code was downloaded from http://gmplib.org/#DOWNLOAD

```
bunzip2 *.bz2
tar xf gmp*.tar
cd gmp-6.*
./configure --prefix=$HOME
make
make install
```

E. mpfr

The multiprecision floating point library `mpfr` was upgraded next. The source code was downloaded from http://www.mpfr.org/mpfr-current/

```
gunzip *.gz
tar xf mpfr-3.1.2.tar
cd mpfr-3.1.2
./configure --prefix=$HOME --with-gmp=$HOME
make
make install
```

F. mpc

The source code was downloaded from http://www.multiprecision.org/index.php?prog=mpc&page=download. The following steps suffer from some variability of the installation targets of the previous steps. Newer Linux versions tend to place libraries always into `/lib` whereas older versions put them into `/lib64` subdirectories if they found a 64bit version.

```
gunzip mpc-1.0.1.tar.gz
tar xf mpc-1.0.1.tar
cd mpc-1.0.1
./configure --prefix=$HOME --with-mpfr=$HOME
# if the libmprf ended up in $HOME/lib64 instead $HOME/lib:
# ./configure --prefix=$HOME --with-mpfr-lib=$HOME/lib64 --with-mpfr-include=$HOME/include
make
make install
```

G. gcj

If the JAVA compiler `gcj` is ever used, and to avoid errors of the type `error trying to execute ecj1`, we want to compile `ecj1`. The source code stems from ftp://sourceware.org/pub/java/ecj-4.9.jar and is put into `$HOME/share/java/ecj.jar`. The actual compilation follows in Section III.

III. COMPILING GCC

A. Configuration

The main compiler was taken from http://gcc.gnu.org/mirrors.html and compiled with

```
bunzip2 *.bz2
tar xf gcc*.tar
cd gcc-4.8.*
mkdir gcc-build
cd gcc-build
export LD_LIBRARY_PATH=$HOME/lib64:$HOME/lib:$HOME/libexec
../configure --build=x86_64-suse-linux --disable-multilib \
   --prefix=$HOME --with-mpfr=$HOME --with-mpc=$HOME --with-gmp=$HOME \
   --with-ecj-jar=$HOME/share/java/ecj.jar --with-x --enable-java-awt=gtk,xlib
```

For openSUSE 12.1 a more precise statement was needed, because the previous libraries had not been installed in `$HOME/lib` but in `$HOME/lib64`:

```
../configure --build=x86_64-suse-linux --disable-multilib \
   --prefix=$HOME --with-mpfr-include=$HOME/include \
   --with-mpfr-lib=$HOME/lib64 --with-mpc-include=$HOME/include \
   --with-mpc-lib=$HOME/lib64 --with-gmp-include=$HOME/include \
   --with-gmp-lib=$HOME/lib64 --with-ecj-jar=$HOME/share/java/ecj.jar \
   --with-x --enable-java-awt=gtk,xlib
```

The definition `x86_64-suse-linux` was changed to `x86_64-centos-linux` for the CentOS machine.

B. Compilation

```
make
make install
```

C. gdb

For program development an upgrade of gdb is probably also needed, http://ftp.gnu.org/gnu/gdb/gdb-7.6.2.tar.gz.

```
gunzip *.gz
tar xf gdb*.tar
cd gdb-7.2.*
./configure --prefix=$HOME
make
make install
```

IV. ENVIRONMENT VARIABLES

If installed as above, the user needs to ensure that the new libraries and executables take priority over the versions that are in the standard /usr path. So typically in a $HOME/.bashrc one would set

```
export LD_LIBRARY_PATH=$HOME/lib:${HOME}/lib64:$LD_LIBRARY_PATH
export PATH=$HOME/bin:$PATH
```

Appendix A: svnversion upgrade

The Subversion client was overloaded in addition, moving it from version 1.6.9 to 1.7.10. This started with an upgrade of the sqlite library of http://www.sqlite.org/download.html

```
gunzip sql*.gz
tar xf sql*.tar
cd sqlite-autoconf-3071700
./configure --prefix=$HOME
make
make install
```

This is followed by subversion of http://subversion.apache.org/download/#recommended-release,

```
gunzip *.gz
tar xf *.tar
cd subversion-1.7.10
./configure --prefix=$HOME --with-serf=$HOME --with-sqlite=$HOME
make
```

Appendix B: heatools upgrade

The installation of gcc has the nice effect of providing a Fortran compiler (which is not included with some openSUSE installations, even if all developer tools habe been check-marked at installation time). This allows to compile the HEASoft tools of http://heasarc.nasa.gov/lheasoft/download.html. We selected the Source code distribution, and the Genral-USE FTOOLS, downloaded the tar ball, and compiled as follows:

```
gunzip heasoft*.gz
tar xf heasoft*.tar
cd heasoft-6.15.1/BUILD_DIR
```

```
./configure --x-libraries=/usr/lib --x-includes=/usr/include > config.out 2>&1
make > build.log 2>&1
make install > install.log 2>&1
```

The `$HOME/.bash_login` was ammended by

```
export FVTMP=$TMPDIR/.fv
export HEADAS=${HOME}/heasoft-6.15.1/x86_64-unknown-linux-gnu-libc2.17
. $HEADAS/headas-init.sh
```