

SNANA Installation Guide

J. Bernstein, S. Jha, R. Kessler

May 16, 2012

Contents

1 Overview	1
2 Linux	2
2.1 Troubleshooting	2
3 Mac OS with Intel Processor	3
4 Data Samples in Download	6

1 Overview

The installation of SNANA involves two tarballs. First is the SNANA software that includes all of the source code and a `Makefile`. The second tarball, called “`SNDATA_ROOT`,” contains data ($\$4$), a simulation-output directory, K-correction tables, model parameters, filter responses, primary spectra (Vega, BD17), etc. Environment variables `$SNANA_DIR` and `$SNDATA_ROOT` must be defined to point to these two areas, and it is convenient to define these in a login script. You will also need to add `$SNANA_DIR/bin` and `$SNANA_DIR/util` to your path. It is recommended that `$SNANA_DIR` be write-protected, while `$SNDATA_ROOT` has write-access for all users. Once you have installed SNANA, see the `snana_manual` for instructions on running the programs.

At least 2 Gb of memory is needed to run the SNANA programs. The SNANA software tarball expands to 25 MB after the `Makefile` script has run, and the `SNDATA_ROOT` tarball expands to a few Giga-bytes.

2 Linux

To use the SNANA package, you need to have the 32-bit versions of CERNLIB, CFITSIO, libncurses, GSL, glibc, and libz installed. On 64-bit machines, if you get errors about incompatible libraries check to make sure you have the 32-bit versions installed and that they are visible. If your version of CERNLIB comes with nypatchy instead of ypatchy, then make a symbolic link named ypatchy pointing to nypatchy.

In the following, \$SOFTDIR refers to the top-directory of your software packages, and \$SCRATCHDIR refers to your working area or scratch disk.

After installing CERNLIB and CFITSIO in \$SOFTDIR/cern and \$SOFTDIR/cfitsio, respectively, and assuming you are using the tcsh shell, do, e.g.:

```
> setenv CERN_DIR $SOFTDIR/cern
> setenv PATH $CERN_DIR/bin:$PATH
> setenv LD_LIBRARY_PATH $CERN_DIR/lib:$LD_LIBRARY_PATH
> setenv CFITSIO_DIR $SOFTDIR/cfitsio
> setenv GSL_DIR $SOFTDIR/gsl
```

Download the most recent SNANA tarball to \$SOFTDIR/SNANA and do

```
> cd $SOFTDIR/SNANA
> tar xzf snana_v7_07.tar.gz
> cd snana_v7_07/src
> make
    (hold your breath ... )
> setenv SNANA_DIR $SOFTDIR/SNANA/snana_v7_07
> setenv PATH $SNANA_DIR/bin:$SNANA_DIR/util:$PATH
```

Once you have the SNANA software installed, download the most recent SNDATA_ROOT tarball to \$SCRATCHDIR/SNDATA_ROOT, and do

```
> setenv SNDATA_ROOT $SCRATCHDIR/SNDATA_ROOT
```

2.1 Troubleshooting

1. If you get errors similar to

```
$SOFTDIR/cern/2004/lib/libpacklib.a(cfclos.o)(.text+0xa):
In function `cfclos_': : undefined reference to `rfio_close'
```

then, in \$SNANA_DIR/src/Makefile, replace “-lkernlib -lpacklib” with “-lkernlib_noshift -lpacklib_noshift” if you have the latter with your CERNLIB distribution. If not, then try installing the appropriate version from the CERNLIB page.

2. If you have a 64-bit machine of type other than x86_64, then modify the BITNESS logical test in the Makefile so that the -m32 flag is used.

3 Mac OS with Intel Processor

2012.03.01 Updated for Mac OSX 10.6 by S.Rodney.

1. Make sure you have a working gfortran.

Here are three possible sources:

- (a) <http://hpc.sourceforge.net>:
unpack a tar ball, putting binaries into `/usr/local/bin`
- (b) <http://gcc.gnu.org/wiki/GFortranBinaries> :
.dmg installer puts gfortran in `/usr/local/gfortran/bin`
add that to your `$PATH` or make sure links in `/usr/local/bin` exist.
- (c) <http://r.research.att.com/tools/> :
.dmg installer puts binaries in `/usr/local/bin`
they say "most other binaries are either incomplete or broken (do not use compilers from HPC, they won't work correctly!)"
but that may be outdated.

2. g95 is needed for compiling the Sussex cosmology fitter `sncosmo_mcmc`. For example, the binary from <http://www.g95.org/downloads.shtml>.

3. Install SNANA dependencies : CFITSIO, GSL, CERNLIB .

There are three principal dependencies that snana needs. Download the tar ball from the SNANA web site, then unpack it directly into `/usr/local`:

```
sudo tar -xvzf snanadep.osx-10.6.tgz -C /usr/local/
```

This tar ball unpacks to a directory `snanadep` containing three SNANA prerequisites: GSL, CFITSIO, and CERNLIB.

- (a) The Gnu Scientific Library (GSL):

This comes as pre-compiled universal libraries (i.e. including 32-bit) in `include` and `lib` (from Mark Dayel: <http://www.dayel.com/2009/09/building-gsl-universal-binary>)
You shouldn't have to do anything for this except make sure that you've got the environment variable set correctly (see below).

- (b) The CFITSIO libraries :

This comes as pre-compiled 32-bit libraries in `include` and `lib` (compiled by Steve Rodney). This is untested, and may not link properly. If so, you've got the source code in `CFITSIO-32`, and you can simply compile this in 32-bit:

```
cd /usr/local/snanadep/CFITSIO-32
./configure CFLAGS=''-m32 -O2'' --prefix=/usr/local/snanadep
make
make install
```

You may need to use `sudo make` if you don't have write permission in `/usr/local`. Ignore warnings like:

```
ranlib: for architecture: i386 file: libcfitsio.a(drvrgsiftp.o) has no symbols
ranlib: for architecture: x86_64 file: libcfitsio.a(drvrgsiftp.o) has no symbols
```

(c) CERNLIB :

The CERN libraries, pre-compiled, from Saurabh Jha, are in CERN.2005. It used to be possible to use a shell script to fetch and compile cernlib. In the far-off mythic days of yore one could even compile it from source. Now, the best option seems to be simply unpacking the tar ball of a recently successful build, and then kludging together the necessary dynamic library links.

Check if your version of `libgfortran.3.dylib` will work with CERNlib's `nypatchy`. Check the architecture of this dynamically linked library:

```
file /usr/local/lib/libgfortran.3.dylib
```

If you get a response that doesn't indicate i386 architecture, like:

```
libgfortran.3.dylib: Mach-O 64-bit dynamically linked shared library x86_64
```

then `nypatchy` cannot work with this `dylib`. You'll have to copy the i386 version (included in the `snanadep` tar ball) into `/usr/local/lib`:

```
cd /usr/local/lib
sudo cp /usr/local/snanadep/lib/libgfortran.3.dylib libgfortran.3.i386.dylib
sudo mv libgfortran.3.dylib libgfortran.3.x86_64.dylib
sudo ln -s libgfortran.3.i386.dylib libgfortran.3.dylib
```

4. Download the latest SNANA release and unarchive it somewhere. This directory MUST be referred to as `SNANA_DIR`, as shown below.
5. Download the `SNDATA_ROOT` tarball, and unpack it somewhere else. This directory should be referred to as `SNDATA_ROOT`, as shown in the next step.
6. Set environment variables. Adjust as necessary and then add these to your `.tcshrc`, or put the equivalent in `.profile` or `.bash_profile`. For example (from a `tcsh` shell):

```
setenv CFITSIO_DIR /usr/local/snanadep
setenv GSL_DIR      /usr/local/snanadep
setenv CERN_DIR     /usr/local/snanadep/CERN.2005
```

```
setenv SNANA_DIR /usr/local/snana
setenv SNDATA_ROOT /usr/local/SNDATA_ROOT
setenv NONIA_ROOT /usr/local/SNDATA_ROOT
setenv NON1A_ROOT /usr/local/SNDATA_ROOT
```

```
setenv g77 gfortran
setenv FC gfortran
```

```
setenv PATH ${PATH}:${CERN_DIR}/bin
```

7. Finally, compile SNANA :

```
% cd $SNANA_DIR/src
% make
```

or `sudo make`. Note that if you are installing some place that requires `sudo` then you'll have to use `sudo setenv` for all the requisite environment variables in order for the Makefile to pick them up.

If you encounter linker errors complaining about duplicate variables, then use `ranlib` to update the library's table of contents. e.g:

```
ranlib /usr/local/CERN.2005/lib/libkernlib.a
```

The snana binaries are installed into `$SNANA_DIR/bin`. Make sure this is on your `PATH` or set up the necessary `sym-links`.

4 Data Samples in Download

The downloads include SN data versions in `$SNDATA_ROOT/lcmerge`.¹ Each SN data version corresponds to a published data set that has been converted into the format needed for the SNANA light curve fitter. To see a summary of the available data samples,

```
> cd $SNDATA_ROOT/lcmerge/  
> ls *.README
```

and then “more” any `README` file for details.

¹“lcmerge” refers to the merging of data and meta-data such as PSF, skynoise, moon, etc ...