Installing SNANA on SciNet & CentOS Linux 7

Anna O'Grady

August 3, 2017

1 Introduction

The SNANA program[1] requires some small fixes to the default installation for modern systems. I've installed SNANA both on SciNet[2] and on a computer with CentOS 7. Below I detail the order of installation and any fixes I applied to completely install the program. The sections are meant to be read independently, so there may be some repetition.

2 SciNet

2.1 Dependencies & Modules

From the SNANA Installation Guide[3], the dependencies `GSL`, `CFITSIO`, `libncurses`, and one of `CERNLIB` or `ROOT` are required. While `CERNLIB/ROOT` is not required to complete the installation of SNANA, it is required for the visualization of light curve fits through `mkfitplots.pl`, so this manual will include how to install them.

SciNet has a number of pre-installed 'modules', which you can load and unload within a cluster. Included in the SciNet modules are `GSL`, `CFITSIO`, and `ROOT`, along with the `gcc` compilers required to compile the codes.

For SciNet, since `ROOT` is an included module, I used this for the visualizations within SNANA. You can use `CERNLIB` instead, and I explain how to install that in Section 3.

2.2 General Installation Order

1. Download `ncurses6.0` and the SNANA source code and data files. Transfer them to your SciNet account directory, and then log on. Also download and transfer `CERNLIB` (see Section 3) if you wish to use that instead of `ROOT`.
2. Create 2 directories (`$SNANA` and `$SNDATA_ROOT`) to hold the SNANA source code + dependencies and the SNDATA_ROOT files receptively. Untar all of these.
3. Load the required modules in SciNet. See Section 2.3.2 for specifics.
4. Install the non-module dependencies (`ncurses6.0`, and possibly `CERNLIB`).
5. Set your environment variables. See Section 2.3.4 for specifics.
6. Edit the Makefile to ensure environment variables and compilers are correct. See Section 2.3.5 for specifics.
7. `make` SNANA

2.3 Detailed Instructions with Fixes

2.3.1 Downloads and Directories

Download the `ncurses6.0` library[4], as well as the source code and data files for SNANA (`SNANA` and `SNDATA_ROOT` respectively)[5]. Transfer these files to your SciNet account, and then log in.

```
> scp -C [yourfile] [USERNAME]@login.scinet.utoronto.ca:~[PathToFileInSciNet]
> ssh -Y [USERNAME]@login.scinet.utoronto.ca
> ssh -Y gpc[XX]
```

[1] http://snana.uchicago.edu/
[2] http://www.scinethpc.ca/
[3] http://snana.uchicago.edu/doc/snana_install.pdf
[4] https://www.gnu.org/software/ncurses/
[5] http://snana.uchicago.edu/downloads.php

Within your SciNet home directory, you should create 2 new directories: one for holding the SNANA source code and any non-module dependencies, and one for `SNDATA_ROOT`. Move and untar (using `tar -xzf [FILE]`) `SNDATA_ROOT` in the SNDATA_ROOT directory, and move an untar `SNANA.tar.gz` and `ncurses6.0` in the SNANA directory.

2.3.2 Modules

We will now load the modules required for the installation. The version numbers for these modules are accurate as of August 2, 2017. You may use the most updated version of any packages *except* `ROOT` and its requirements (`gcc`, `intel`, and `python`). You should be able to use any sub-version of `ROOT-5` and its associated dependencies, but `ROOT-6` will not work with SNANA.

```
> module load extras/64_6.4
> module load Xlibraries/X11-64
> module load emacs/24.4
> module load make/3.81
> module load dcap/2.47
> module load gsl/2.3-gcc
> module load cfitsio/3.380
> module load cmake/3.5.2
> module load gcc/4.6.1
> module load intel/12.1.3
> module load python/2.7.2
> module load ROOT/5.30.03
```

2.3.3 Non-module Dependencies

We will now install the one non-module dependency for SNANA, `ncurses6.0`. If you wish to use `CERNLIB` instead of `ROOT`, you would install that here. As detailed in Section 3, the `CERNLIB` distribution I have used requires `imake` to compile, which is contained in `gcc/4.4.6` or `gcc/4.6.1` on SciNet.

Go to the `/SNANA/ncurses-6.0/` directory and install via:

```
> ./configure
> make
> make install # optional
```

As you do not have root privileges in SciNet, `make install` will be unable to install things in the `/usr/` directory. However I have not run into any errors due to this, and many dependencies indicate that `make install` is optional anyway.

2.3.4 Environment Variables

Now we will need to set environment variables for both the installation of SNANA and efficiency. These follow directly from the SNANA installation guide, but I note here where these variables need to be pointed for the SciNet modules. These are accurate as of August 2, 2017, and may vary slightly depending on which versions modules you use. Note that variables such as `$SCINET_GSL_BASE` should always point to the correct directory, and load with the module itself. `ROOT` does not have these pre-set environment variables if an older version, such as the one we are using, is loaded, so it must be pointed manually.

```
> ROOT_DIR=/scinet/gpc/Applications6/ROOT/root-5.30.03 # The last numbers, for example, may
    change if you use a different version of ROOT
> CFITSIO_DIR=$SCINET_CFITSIO_BASE
> GSL_DIR=$SCINET_GSL_BASE
> export
> PATH=$ROOT_DIR/bin:$PATH
> LD_LIBRARY_PATH=$ROOT_DIR/lib:$LD_LIBRARY_PATH
> export
> SNANA_DIR=/home/[PathToWorkingDirectory]/SNANA/SNANA_v10_50f # Your version number may be
    different
> SNDATA_ROOT=/home/[PathToWorkingDirectory]/SNDATA_ROOT
> export
> PATH=$SNANA_DIR/bin:$SNANA_DIR/util:$PATH
> export
```

You may wish to double check these pointings with `echo $VARIABLE`. It is also recommended to build a script to set these environment variables, so you don't need to manually do it each time you log in to SciNet.

2.3.5 Makefile Edits + Installing SNANA

With all the dependencies installed and environment variables pointed, we are nearly ready to install SNANA! There are some edits we must make to SNANA's Makefile in order for the installation to work properly. Go to `$SNANA_DIR/src` and enter `emacs Makefile &` (or your preferred text editor available in SciNet).

Below I will be indicating the sections that need changing and what the code should be changed *to*.

First, the library and include pointings for `GSL` and `CFITSIO` must be fixed:

```
# ------------------------------------------------
# define libraries # !The commented lines are the old pointings!

      LGSL     = $(SCINET_GSL_LIB)/libgsl.a $(SCINET_GSL_LIB)/libgslcblas.a  #$(GSL_DIR)/$(
    LIB_SDIR)/libgsl.a  \
       #$(GSL_DIR)/$(LIB_SDIR)/libgslcblas.a
      IGSL     = -I$(SCINET_GSL_INC)    #-I$(GSL_DIR)/include

      LCFITSIO = -L$(SCINET_CFITSIO_LIB) -lcfitsio    #-L$(CFITSIO_DIR)/$(LIB_SDIR) -lcfitsio
      ICFITSIO = -I$(SCINET_CFITSIO_INC)    #-I/$(CFITSIO_DIR)/include
```

We need to ensure all of the compiler commands are set to `gfortran`, `gcc`, and `c++`. There are two instances where `cc` is used instead of `gcc`:

```
# ==============================

FFC  = gfortran
CC   = gcc      #!change here!

   DBUGFLAGS = -O1
#   DBUGFLAGS = -O1 -Wall
#  DBUGFLAGS = -g -fbounds-check
```

```
# -----------------

# USE_ROOT flag affects the compilation of sntools_ouptut.c,
# and the -lstdc++ & root libraries for linking.

ifeq ($(USE_ROOT),0)
   CCoutput = gcc       #!change here!
   CPPLIB   =
   SNTOOLS_ROOT =
else
   CCoutput     = c++
   CPPLIB       = -lstdc++
   IROOT        = -I/$(ROOT_DIR)/include/root -I/opt/local/include/freetype2
   ROOT_LIBDIR = $(LIB_SDIR)
   ROOT_LIBDIR = lib
   LROOT = -L$(ROOT_DIR)/$(ROOT_LIBDIR)/root -lCore -lCint -lRIO -lNet -lHist -lGraf -lGraf3d -
    lGpad -lTree -lRint -lPostscript -lMatrix -lPhysics -lMathCore -lThread -ldl
   SNTOOLS_ROOT = $(SRC)/sntools_output_root.c
endif
```

You should now be ready to install SNANA! Inside `$SNANA_DIR/src`, type `make`. You may get some warnings through the installation, but so long as the process does not stop with an error the installation should have been completely successfully. Try the 'Getting Started' Sections (4.2 and 5.1) from the SNANA Manual (located here[6] or in `$SNANA_DIR/doc`) to test the simulation and fitting programs.

There is one last, small fix that should be applied. When trying to run the simulation (`snlc_sim.exe`), SNANA was unable to create two new directories inside of `SNDATA_ROOT`: `$SNDATA_ROOT/SIM/[GENVERSION]`, where `GENVERSION` is the name you give your simulation in the `.input` file. This can be simply amended by doing `mkdir SIM` inside of `$SNDATA_ROOT` - the simulation can then create the `GENVERSION` directory just fine. This appears to be a permissions issue.

3 CentOS 7

3.1 Dependencies

From the SNANA Installation Guide[7], the dependencies `GSL`, `CFITSIO`, `libncurses`, and one of `CERNLIB` or `ROOT` are required. While `CERNLIB/ROOT` is not required to complete the installation of SNANA, it is required for the visualization of light curve fits through `mkfitplots.pl`, so this manual will include how to install them.

[6] http://snana.uchicago.edu/doc/snana_manual.pdf
[7] http://snana.uchicago.edu/doc/snana_install.pdf

You will also need gcc compilers with gcc, gfortran, c++, and possibly cc (though cc can be avoided by editing the Makefile as done in Section 2.3.5). You will also require imake if you want to install the same CERNLIB distribution I use.

I chose to install CERNLIB, but the installation should be the same if you choose to use ROOT instead. Simply replace instance of CERNLIB with ROOT, $CERNLIB_DIR with $ROOT_DIR, etc.

3.2 General Installation Order

1. Download GSL, CFITSIO, ncurses6.0, CERNLIB (or ROOT), and the SNANA source code and data files.
2. Create 2 directories ($SNANA and $SNDATA_ROOT) to hold the SNANA source code + dependencies and the SNDATA_ROOT files receptively. Untar all of these.
3. Install the dependencies (GSL, CFITSIO, ncurses6.0, and CERNLIB or ROOT).
4. Set your environment variables. See Section 3.3.3 for specifics.
5. Edit the Makefile to ensure environment variables and compilers are correct. See Section 3.3.4 for specifics.
6. make SNANA

3.3 Detailed Instructions with Fixes

3.3.1 Downloads and Directories

Download GSL[8], CFITSIO[9], ncurses6.0 library[10], as well as the source code and data files for SNANA (SNANA and SNDATA_ROOT respectively)[11].

There are a few different ways you can get CERNLIB. I used this[12] GitHub repository, as it was specifically designed to make installation on modern Unix systems simple. This is the version I will explain how to install, but feel free to use your own method to acquire CERNLIB - just ensure that paw works after installation

Within your working, you should create 2 new directories: one for holding the SNANA source code and any dependencies, and one for SNDATA_ROOT. Move and untar (using tar -xzf [FILE]) SNDATA_ROOT in the SNDATA_ROOT directory, and move an untar SNANA.tar.gz and all other dependencies in the SNANA directory.

3.3.2 Installing Dependencies

I will be using tcsh as my terminal. Using bash should be fine, just note that the controls to set environment variables change - it will be the way described in Section 2.3.4 instead of using setenv.

You need gcc and imake to perform these installations, so install these via:

```
> sudo yum install gcc
> sudo yum install imake
```

You can do > which gcc/cc/gfortran/imake to double check then have been installed.

The order of installation should not matter here, and GSL, CFITSIO, and ncurses6.0 are all installed in the same way. Go to the respective dependency directory in /SNANA/[DIR], and install via:

```
> ./configure
> make
> make install
```

If you have root privileges then make install should install things to the /usr/ directory. Many of the dependency README's indicate that make install is optional, but I believe that at least for GSL and CFITSIO it is necessary for SNANA.

Once you have GSL, CFITSIO, and ncurses6.0 installed, the last dependency is CERNLIB. If you are using the same resource I used, then enter the /SNANA/cernlib-master directory and run the script sh build.sh. This should build CERNLIB. You will then want to set some environment variables for CERNLIB, either by running sh setup.sh or by setting the variables manually:

```
> setenv CERN /home/[PathToWorkingDirectory]/SNANA/cernlib-master/cernlib_2005
> setenv CERN_LEVEL 2005
> setenv CERN_ROOT $CERN/$CERN_LEVEL
> setenv PATH $CERN_ROOT/bin:$PATH
```

[8] https://www.gnu.org/software/gsl/
[9] https://heasarc.gsfc.nasa.gov/fitsio/fitsio.html
[10] https://www.gnu.org/software/ncurses/
[11] http://snana.uchicago.edu/downloads.php
[12] https://github.com/davehadley/cernlib

CERNLIB should now be set up - to double check, run paw_setup.cmd and then paw and see if the visualization program starts.

3.3.3 Environment Variables

Now we will need to set environment variables for both the installation of SNANA and efficiency. These follow directly from the SNANA installation guide. These are accurate as of August 2, 2017, and may vary slightly depending on which versions of dependencies you use.

```
> setenv SOFTDIR /home/[PathToWorkingDirectory]/SNANA  # This makes it easier to do the
    following variables
> setenv CERN_DIR $SOFTDIR/cernlib-master/cernlib_2005/2005
> setenv CFITSIO_DIR $SOFTDIR/cfitsio
> setenv GSL_DIR $SOFTDIR/gsl-2.4    # This number, for example, may be different if you use a
    different version of GSL
> setenv PATH $CERN_DIR/bin:$PATH
> setenv LD_LIBRARY_PATH $CERN_DIR/lib:$LD_LIBRARY_PATH
#NOTE: An error may occur here. A workaround is to do the following instead, which does not
    seem to effect installation
> setenv LD_LIBRARY_PATH $CERN_DIR/lib:LD_LIBRARY_PATH
> setenv SNANA_DIR $SOFTDIR/SNANA_v10_50f # Your version number may be different
> setenv SNDATA_ROOT /home/[PathToWorkingDirectory]/SNDATA_ROOT
> setenv PATH $SNANA_DIR/bin:$SNANA_DIR/util:$PATH
```

You may wish to double check these pointings with `echo $VARIABLE`. It is also recommended to build a script to set these environment variables (*important:* this includes the variables for CERNLIB in Section 3.3.2!), so you don't need to manually do it each time you start a new session.

3.3.4 Makefile Edits + Installing SNANA

With all the dependencies installed and environment variables pointed, we are nearly ready to install SNANA! There is one small edit we must make to SNANA's Makefile in order for the installation to work properly. Go to $SNANA_DIR/src and enter `emacs Makefile &` (or your preferred text editor).

Below I will be indicating the sections that need changing and what the code should be changed *to*.

First, the library and include pointings for GSL and CFITSIO must be fixed:

```
# ------------------------------------------------
# define libraries #!the commented pointings are the originals!


  LGSL     =  /usr/local/lib/libgsl.a /usr/local/lib/libgslcblas.a
                          #$(GSL_DIR)/$(LIB_SDIR)/libgsl.a  \
                        #$(GSL_DIR)/$(LIB_SDIR)/libgslcblas.a
  IGSL     =  -I/home/[PathToWorkingDirectory]/SNANA/gsl-2.4/gsl/
                          #-I$(GSL_DIR)/include

  LCFITSIO =  -L/home/[PathToWorkingDirectory]/SNANA/cfitsio/lib/ -lcfitsio
                          #-L$(CFITSIO_DIR)/$(LIB_SDIR) -lcfitsio
  ICFITSIO =  -I/home/[PathToWorkingDirectory]/SNANA/cfitsio/include/
                          #-I/$(CFITSIO_DIR)/include
```

You can also choose to use environment variables like `-L$(CFITSIO_DIR)/lib` instead of the long directory path. Either should be fine.

Note: If you have gcc, gfortran, and c++ but *not* cc, then please follow the second set of Makefile edits in Section 2.3.5 to change instances of cc to gcc. This should not usually be a problem though - installation of gcc usually come with cc.

You should now be ready to install SNANA! Inside $SNANA_DIR/src, type make. You may get some warnings through the installation, but so long as the process does not stop with an error the installation should have been completely successfully. Try the 'Getting Started' Sections (4.2 and 5.1) from the SNANA Manual (located here[13] or in $SNANA_DIR/doc) to test the simulation and fitting programs.

There is one last, small fix that should be applied. When trying to run the simulation (snlc_sim.exe), SNANA was unable to create two new directories inside of SNDATA_ROOT: $SNDATA_ROOT/SIM/[GENVERSION], where GENVERSION is the name you give your simulation in the .input file. This can be simply amended by doing mkdir SIM inside of $SNDATA_ROOT - the simulation can then create the GENVERSION directory just fine. This appears to be a permissions issue.

[13] http://snana.uchicago.edu/doc/snana_manual.pdf