HEP Software Installation

SERB SCHOOL ON EXPERIMENTAL HIGH ENERGY PHYSICS TIFR,

Mumbai, Jan 7-27, 2019

Outline

1. Installation Instructions for SL6
2. Installation of Pythia8
3. Installation of Pythia6
4. GSL (GNU Scientific Library) Installation
5. ROOT6 Installation
6. GEANT4 (CLHEP) Installation
7. GENIE (LHAPDF6, log4cpp, libxml2) Installation
8. EVTGEN (HepMC, PHOTOS, TAUOLA) Installation
9. HIJING Installation
10. RooUnfold: ROOT Unfolding Framework
11. Working With Genie

SL6 Instructions:

Installing root6, GEANT4 packages they require gcc 4.8 and above but SL6 will support upto GCC4.4.7 only. To avoid this we have to enable devtoolset provided by CERN in-order to install above packages.

Follow the instructions below to install and enable devtoolset-3.

```
#Scientific Linux 6 (SLC6)
#Save repository information as /etc/yum.repos.d/slc6-devtoolset
$ wget -O /etc/yum.repos.d/slc6-devtoolset.repo  \\
http://linuxsoft.cern.ch/cern/devtoolset/slc6-devtoolset.repo

# 2. Install the collection:
$ sudo yum install devtoolset-3

# 3. Start using software collections:
$ scl enable devtoolset-3 bash
```

NOTE: you have to enable devtoolset-3 in SL6 in-order to install packages mentioned in the following Sections.

Installation of Pythia8

- Download the latest Pythia (pythia8235)
  `http://home.thep.lu.se/~torbjorn/pythia8/pythia8235.tgz`
- Unzip and expand it with

  $ tar xvfz pythia8235.tgz
  $ cd pythia8235
  $ ./configure --enable-shared -fPIC --prefix=/<path>/products/pythia8235
  Use the -fPIC configure flag only on 64bit systems
  $ make -j n
  $ make install

- Environment variable setup

  $ export PYTHIA8=/<path>/products/pythia8235
  $ export PYTHIA8DATA=$PYTHIA8/share/Pythia8/xmldoc
  $ export PATH=$PYTHIA8/bin:$PATH
  $ export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$PYTHIA8/lib

Installation of Pythia6

- Download the file pythia6.tar.gz from the site `http://neutrino.ift.uni.wroc.pl/files/pythia6.tar.gz` and unpack.
- Download the latest Pythia (pythia6428) from the site `http://www.hepforge.org/archive/pythiasix/pythia-6.4.28.f.gz` and unpack
-
  ```
  $ mv pythia6/ pythia6428/
  $ rm pythia6428/pythia6416.f
  $ cp pythia6428.f pythia6428/
  $ cd pythia6428/
  $ ./makePythia6.linuxx8664 ;; it will create libPythia6.so in current folder.
  ```
  - Remove Objective files (*.o)
- Environment variable setup

  ```
  $ export PYTHIA6=/<path>/products/pythia6428
  $ export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$PYTHIA6428/lib
  ```

GSL (GNU Scientific Library) Installation

- The library provides a wide range of mathematical routines such as random number generators, special functions and least-squares fitting. There are over 1000 functions in total with an extensive test suite
- GSL can be found in the gsl subdirectory on your nearest GNU mirror `http://ftpmirror.gnu.org/gsl/`.
- Download GSL from `http://mirror.rasanegar.com/gnu/gsl/gsl-2.5.tar.gz`

```
$ tar xzvf gsl-2.5.tar.gz
$ cd gsl2.5
$ ./configure –prefix=/localhome/products/gsl25
$ make -j n
$ make install
```

- **Environment variable setup**

```
$ export GSLHOME=/localhome/products/gsl25
$ export PATH=$GSLHOME/bin:$PATH
$ export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$GSLHOME/lib
```

ROOT6 Installation

- Get the root software [version 6.14.00] from cern web page.
  `https://root.cern.ch/download/root_v6.14.00.source.tar.gz`
- **Required packages:**
  $ sudo yum install git cmake gcc-c++ gcc binutils libX11-devel libXpm-devel libXft-devel libXext-devel
- **Optional packages:**
  $ sudo yum install gcc-gfortran openssl-devel pcre-devel mesa-libGL-devel mesa-libGLU-devel glew-devel ftgl-devel mysql-devel fftw-devel cfitsio-devel graphviz-devel avahi-compat-libdns_sd-devel libldap-dev python-devel libxml2-devel gsl-static
- $ tar xvzf root_v6.14.00.source.tar.gz
  $ mkdir root_v6.14.00-build
  $ cd root_v6.14.00-build
- Now run the cmake3 command by enabling gdml, mathmore, gsl, pythia6 and pythia8 in root6 build directory to create the make file.

ROOT6 Installation Contd ....

```
$ cmake3 -Dgdml=ON -Dbuiltin_gsl=ON -Dmathmore=ON -Dpythia6=ON
-Dpythia8=ON -Droofit=ON -DGSL_DIR=/localhome/products/gsl25
-DGSL_CONFIG_EXECUTABLE=/localhome/products/gsl25/bin/gsl-config
-DPYTHIA6_LIBRARY=/localhome/products/pythia6428/libPythia6.so
-DPYTHIA8_DIR=/localhome/products/pythia8235
DPYTHIA8_INCLUDE_DIR=/localhome/products/pythia8235/include
-DPYTHIA8_LIBRARY=/localhome/products/pythia8235/lib/libpythia8.so
-DCMAKE_INSTALL_PREFIX=/localhome/products/root-6.14.00 ../root-6.14.00
```

```
$ make -j n
```

```
$ make install
```

- **Path Setting:**
  open .bashrc file in your favorite editor (vi or emacs or gedit)
  $ vi .bashrc
  now add the path for root executing script *thisroot.sh* located in root installation folder at the end of the .bashrc.
  $. /localhome/products/root-6.14.00/bin/thisroot.sh

GEANT4 Installation

- **Build Prerequisites for GEANT4**
  - ROOT (version 6.14) (ROOT installation section) (version 6)
  - CLHEP (version 2.4.0.4)
  - GEANT4 (version 4.10.04)
- It is necessary to follow this order strictly or else it will cause problems while compiling GEANT.
- **CLHEP Installation:**
  - Download CLHEP from "`http://proj-clhep.web.cern.ch/proj-clhep/DISTRIBUTION/tarFiles/clhep-2.4.0.4.tgz`" and unzip it
  - Create a directory CLHEP under "products" folder and Rename 2.4.0.4 directory as clhep-source
  - Create a directory named clhep-build and go into this directory
    $ cmake3 -DCMAKE_INSTALL_PREFIX=/<path >/products/CLHEP/clhep2404 /<path >/products/CLHEP/clhep-source
    $ make -j n
    $ make install

GEANT4 Installation Contd..

- **Setting PATH variables for CLHEP**

  export CLHEP_BASE_DIR=/<path >/products/CLHEP/clhep2404
  export PATH= $PATH:$CLHEP_BASE_DIR/bin
  export
  LD_LIBRARY_PATH=$CLHEP_BASE_DIR/lib:$LD_LIBRARY_PATH

- **GEANT 4 Installation**
  - Download `http://geant4-data.web.cern.ch/geant4-data/releases/geant4.10.04.p02.tar.gz` from Geant4 site
  - Create a directory GEANT4 and go to the directory.
  - untar geant4.10.04.p02.tar.gz and rename it geant4.10.04.p02-source
  - Also create a directory named geant4.10.04.p02-build (Hereafter it will refered as geant build directory.)
  - Go to the geant build directory.
  - To build geant you need the following extra libraries.

    For SL7: libXmu-devel, libXi-devel, libconfig-devel, postgresql-devel, xerces-c-devel

    For Ubuntu: libxmu-dev, libxi-dev, libconfig++-dev, libpq-dev

GEANT4 Installation Contd..

```
$ cmake3
-DCMAKE_INSTALL_PREFIX=/localhome/products/GEANT4/geant4.10.04.p02_install
-DGEANT4_USE_SYSTEM_CLHEP=ON
-DCLHEP_ROOT_DIR=/localhome/products/CLHEP/clhep2404/
-DGEANT4_USE_GDML=ON -DGEANT4_USE_QT=ON
-DGEANT4_USE_OPENGL_X11=ON -DGEANT4_USE_RAYTRACER_X11=ON
-DGEANT4_INSTALL_DATA=ON /localhome/products/GEANT4/geant4.10.04.p02_source
```

- Also, the -DGEANT4_INSTALL_DATA=ON flag will download and install data required by the version of GEANT4. For this you should have working internet connection.
- If you have are doing it offline, then please download datafiles related to that patch of GEANT4 and copy it in a directory named **data** and untar them.

Finally

```
$ make -j n
$ make install
```

GEANT4 Installation Contd..

If you are doing this offline

After the make install is run you have to copy data directory to /localhome/products/GEANT4/geant4.10.04.p02_ install/share/Geant4-10.4.2 directory.

Also you need write the following lines in your .bashrc or .bash profile

Environment Variable Setup

export G4INSTALL=/localhome/products/GEANT4/geant4.10.04.p02_install /share/Geant4-10.4.2/geant4make
export export G4WORKDIR=/home/user/G4WORK
source $G4INSTALL/geant4make.sh
source /localhome/products/GEANT4/geant4.10.04.p02_install/bin/geant4.sh

Or in an setenv.sh file. If you write it in this manner then everytime you start a GEANT session, you will have to source this file.

GENIE Installation

- **For a typical GENIE installation you need:**
  - ROOT (version 6.14) (ROOT installation section) (version 6)
  - GNU Scientific Library (GSL)
  - PYTHIA6
  - LHAPDF5 or LHAPDF6

    [Both of which are optional in GENIE v3.0.0 and above]
  - log4cpp
  - libxml2

- **L**HAPDF Installation:

a) Download from LHAPDF web page `https://lhapdf.hepforge.org/downloads/?f=LHAPDF-6.2.1.tar.gz`

b) Create directory LHAPDF and untar LHAPDF-6.2.1.tar.gz

```
$ cd LHAPDF ; $ tar xzvf LHAPDF-6.2.1.tar.gz
$ cd LHAPDF-6.2.1
$ ./configure --prefix=/localhome/products/LHAPDF/lhapdf621
$ make -j n ; $ make install
```

NOTE:
LHAPDF no longer bundles PDF set data in the package tarball. The downloadable PDF sets are packaged as tarballs, which must be expanded to be used. Here is an example of how to retrieve and install a PDF set from the command line:
$ wget http://www.hepforge.org/archive/lhapdf/pdfsets/6.2/CT10nlo.tar.gz -O- | tar xz -C /localhome/products/LHAPDF/lhapdf621/share/LHAPDF

GENIE Installation Contd ..

- **log4cpp Installation:**

a) Download from log4cpp from web page
`http://sourceforge.net/projects/log4cpp`

b) Create directory LOG4CPP and untar log4cpp-1.1.3.tar.gz

$ cd LHAPDF ; $ tar xzvf log4cpp-1.1.3.tar.gz

$ cd log4cpp

$ ./configure --prefix=/localhome/products/LOG4CPP/log4cpp

$ make -j n ; $ make install

- **libxml2 Installation:**

a) Get the libxml2 from site
ftp://xmlsoft.org/libxml2/libxml2-git-snapshot.tar.gz

$ tar xzvf libxml2-git-snapshot.tar.gz ; $cd libxml2-2.9.7

$ ./configure --prefix=/localhome/products/libxml2

$ make -j n ; $ sudo make install

GENIE Installation Contd..

- Create directory GENIE under products folder ; $ cd GENIE
- Download genie by using git repository

  $ git clone https://github.com/GENIE-MC/Generator.git

  $ cd Genrator ; Make sure that GSL Path is exported

  $ ./configure --prefix=/localhome/products/GENIE/genie --disable-profiler
  --disable-validation-tools --disable-cernlib --enable-lhapdf6
  --enable-flux-drivers --enable-geom-drivers --disable-doxygen --enable-test
  --enable-mueloss --enable-dylibversion --enable-t2k --enable-fnal
  --enable-atmo --enable-nucleon-decay --disable-masterclass --disable-debug
  --with-optimiz-level=O2
  --with-pythia6-lib=/localhome/products/pythia6428/libPythia6.so
  --with-lhapdf6-inc=/localhome/products/LHAPDF/lhapdf621/include
  --with-lhapdf6-lib=/localhome/products/LHAPDF/lhapdf621/lib
  --with-libxml2-inc=/localhome/products/libxml2/include/libxml2
  --with-libxml2-lib=/localhome/products/libxml2/lib
  --with-log4cpp-inc=/localhome/products/LOG4CPP/log4cpp/include
  --with-log4cpp-lib=/localhome/products/LOG4CPP/log4cpp/lib

- **Environmental variable setup for GENIE**
- Include following varibles in your .bashrc or genrate saperate source file genie-env.sh

```
export GENIEBASE=/localhome/products
export GENIE=$GENIEBASE/GENIE/genie
export PYTHIA6=$GENIEBASE/pythia6428
export ROOTSYS=$GENIEBASE/root-6.14.00
export GSLHOME=$GENIEBASE/gsl25
export GSL_INC=$GSLHOME/include
export GSL_LIB=$GSLHOME/lib
export LOG4CPP_INC=$GENIEBASE/LOG4CPP/log4cpp/include
export LOG4CPP_LIB=$GENIEBASE/LOG4CPP/log4cpp/lib
export LHAPATH=$GENIEBASE/LHAPDF/lhapdf621
export LHAPDF_INC=$GENIEBASE/LHAPDF/lhapdf621/include
export LHAPDF_LIB=$GENIEBASE/LHAPDF/lhapdf621/lib
export XSECSPLINEDIR=$GENIEBASE/data
export LD_LIBRARY_PATH=$LHAPDF_LIB:$LD_LIBRARY_PATH
export LD_LIBRARY_PATH=/usr/lib64:$LD_LIBRARY_PATH
export LD_LIBRARY_PATH=$LOG4CPP_LIB:$LD_LIBRARY_PATH
export LD_LIBRARY_PATH=$PYTHIA6:$LD_LIBRARY_PATH
export LD_LIBRARY_PATH=$ROOTSYS/lib:$LD_LIBRARY_PATH
export LD_LIBRARY_PATH=$GENIE/lib:$GSL_LIB/lib:$LD_LIBRARY_PATH
export PATH=$GENIE/bin:$ROOTSYS/bin:$GSLHOME/bin:$PATH
```

EVTGEN Installation

Recommended versions of the external packages.

1. **HepMC** Mandatory Package

$ curl -O http://lcgapp.cern.ch/project/simu/HepMC/download/HepMC-2.06.09.tar.gz

2. **Pythia8** See section for installing Pythia8
3. **PHOTOS**

$ curl -O http://photospp.web.cern.ch/photospp/resources/PHOTOS.3.61/PHOTOS.3.61.tar.gz

4. **TAUOLA**

$ curl -O http://tauolapp.web.cern.ch/tauolapp/resources/TAUOLA.1.1.6c/TAUOLA.1.1.6c.tar.gz

Before installing external packages create a directory called **external** and install all packages here.

Installation of HepMC

```
$ cd external ; tar xzvf HepMC-2.06.09.tar.gz
$ mkdir -p HepMC ; cd HepMC
$ cmake -DCMAKE_INSTALL_PREFIX=/localhome/products/EVTGEN/external/HepMC
/localhome/products/EVTGEN/external/HepMC-2.06.09 -Dmomentum:STRING=GEV
-Dlength:STRING=MM
```

```
$ make -j n ; make install
```

EVTGEN Installation Contd...

Installation of PHOTOS

```
$ cd external ; tar xzvf PHOTOS.3.61.tar.gz
$ cd PHOTOS
$ ./configure --with-hepmc=/localhome/products/EVTGEN/external/HepMC
$ make
```

Installation of TAUOLA

```
$ cd external ; tar xzvf TAUOLA.1.1.6c.tar.gz
$ cd TAUOLA
$ ./configure --with-hepmc=/localhome/products/EVTGEN/external/HepMC
$ make
```

Installation of EVTGEN

```
$ git clone -b R01-07-00 http://evtgen.hepforge.org/git/evtgen.git
$ cd /localhome/products/EVTGEN/EvtGen/R01-07-00
$ ./configure --hepmcdir=/localhome/products/EVTGEN/external/HepMC
--photosdir=/localhome/products/EVTGEN/external/PHOTOS --pythiadir=/localhome/products/pythia8235
--tauoladir=/localhome/products/EVTGEN/external/TAUOLA
$ make
```

Installation of EVTGEN Contd..

Environment variable setup for EVTGEN

export HEPMC=/localhome/products/EVTGEN/external/HepMC

export PHOTOS=/localhome/products/EVTGEN/external/PHOTOS

export TAULO=/localhome/products/EVTGEN/external/TAUOLA

export PYTHIA8=/localhome/products/pythia8423

export PYTHIA8DATA=$PYTHIA8/share/Pythia8/xmldoc

export EVTGEN=/localhome/products/EVTGEN/EvtGen/R01-07-00

export LD_LIBRARY_PATH=$HEPMC/lib:$PHOTOS/lib:
$PYTHIA8/lib:$TAULO/lib:$EVTGEN/lib:$LD_LIBRARY_PATH

Installation of HIJING

- Download code: hijing1411.tgz
  [http://atomfizika.elte.hu/haladolabor/HEP/hijing1411.tgz]
- create directory HIJING/hijing1411 under products folder

  $ cd HIJING/hijing1411

  $ tar xzvf hijing1411.tgz

  $ make

- RUN
  - Edit test.f to perform the desired simulation. The last part does the event printing, after CALL HIJING.
  - Compile main program via make test.exe
  - Edit test.in to achieve desired settings. Here the first number on each line is just a placeholder. The subsequent strings after the first comma are important, and they set the following: random seed; coordinate system, collision energy; colliding particle or nucleus; nucleus A,Z parameters; number of events; print warnings; print info of decayed particles;
  - Run via ./test.exe $<$ test.in $>$ test.out

- ANALYZE
  - A test.out text file has been created. It contains a string BEGINNINGOFEVENT at the beginning of each event. After that, the number of event, number of particles and total energy of particles is printed. After that, particles are listed with their number, id, parent particle number (if decays are remembered), status, four-momentum coordinates.

Installation of RooUnfold: ROOT Unfolding Framework

- The latest version of RooUnfold (version-1.1.1) is compatible only with ROOT version 5.34/12. For installing this version with ROOT 6 compatibility we have to install Development package of RooUnfold.
- **Building the Library**
  - Make sure that ROOT is set up correctly: This can be most easily achieved using ROOT's *thisroot.(c)sh* setup script.
  - The latest development version of RooUnfold is maintained in the "Unfolding Framework" Project Subversion code repository, It can be checked out using:

$ svn co https://svnsrv.desy.de/public/unfolding/RooUnfold/trunk RooUnfold

$ cd RooUnfold

$ make

- **Post Installation - Set environment in ~/.bashrc**

#RooUnfold

export ROOUNFOLD=/<path>/products/RooUnfold/

export LD_LIBRARY_PATH=$ROOUNFOLD:$LD_LIBRARY_PATH

Working With Genie

Introduction

We require $\sim 10^7$ differential cross section evaluations are required just in order to select an interaction channel for a given initial state. If we have been simulating in realistic detector geometry with $\sim 10^2$ different isotopes, the number of differential cross section evaluations, before even starting simulating the event kinematics, would rise to $\sim 10^9$. It is therefore advantageous to pre-calculate the cross section data. The event generation drivers can be instructed to to load the pre-computed data and estimate the cross section by numerical interpolation, rather than by performing numerous CPU-intensive differential cross section integrations. The cross section data are written out in XML format and, when loaded into GENIE, they are used for instantiating Spline objects.

- Generating cross-section splines is a CPU-intensive task as a large number of processes and numerical integration of steeply peaked differential cross-sections over extended, multi-dimensional kinematical phase spaces.
- When cross-section calculations are needed for multiple targets, it is often impractical to generate all splines in a single job. The task is typically split into smaller jobs which can be run on parallel in a batch farm.
- The multiple XML outputs of all the gmkspl jobs can be merged into a single XML file using GENIE's gspladd utility.

Working With Genie Contd..

Generating cross-section splines

- gmkspl - A GENIE utility for generating the cross section splines for a specified set of modeled processes for a specified list of initial states. The cross section splines are written out in an XML file in the format expected by all other GENIE programs.

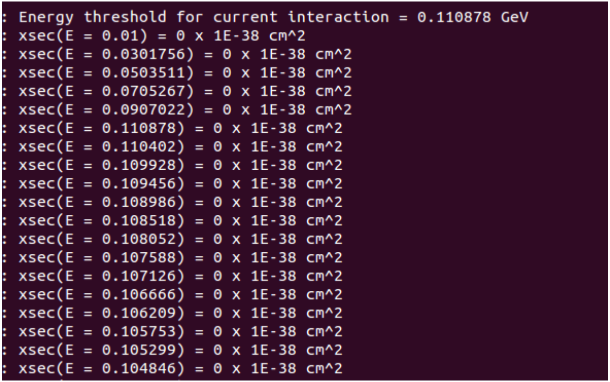
To calculate cross-sections for ν_μ (PDG code: 14) scattering off Fe^{56} (PDG code: 1000260560), and build splines with 10 knots in the energy range up to 0.1 GeV

```
gmkspl -p 14 -t 1000260560 -n 10 -e 0.1
```

If we want for antineutrino also we can use -14 separated by a comma.

- **-p** Specifies the neutrino PDG codes.
- **-t** Specifies the target PDG codes.
- **-n** Specifies the number of knots per spline.
- **-e** Specifies the maximum neutrino energy in the range of each spline.

The spline creation in terminal looks like in the following figure.



You can download the xml file from `https://www.hepforge.org/archive/genie/data/2.12.8/`. The first option "DefaultPlusMECWithNC" contains both charged current and neutral current interaction cross-sections for muon neutrino with Fe^{56}.

https://www.hepforge.org/archive/genie/data/2.12.8/README_for_non-ups_use

```
What splines are available for R-2_12_6?

There are 4 pre-generated configurations:

${cfgname}                use w/ --event-generator-list
------------------------------------------------------------------------------
DefaultPlusMECWithNC        "Default"            default set of models
                            "Default+CCMEC"           + CC-only Empirical MEC model
                            "Default+CCMEC+NCMEC"     + CC & NC Empirical MEC models
------------------------------------------------------------------------------
DefaultPlusValenciaMEC      "Default"            default set of models
                            "Default+CCMEC"           + CC-only Valencia MEC model
------------------------------------------------------------------------------
EffSFTEM                    "Default"            Effective Structure Function
                                                 (replace FGMBodekRitchie)
                                                 + Transverse Enhancement Model
------------------------------------------------------------------------------
LocalFGNievesQEAndMEC       "Default+MEC"        Local Fermi Gas + Nieves (Valencia)
                                                 model for QE + CC MEC
------------------------------------------------------------------------------
AltPion                     "Default"            use BergerSehgal instead of
                                                 ReinSehgal + add in DFR-CC + DFR-NC
------------------------------------------------------------------------------
ValenciaQEBergerSehgalCOHRES "Default+CCMEC+1K" use BergerSehgal
                                                   Valencia + add single Kaon

------------------------------------------------------------------------------

using the command:

   cd ${TOPPATH}    # your chosen area in which to unpack the files
   tar xjf genie_xsec-2.12.8-noarch-${cfgname}.tar.bz2

results in the hierarchy:

genie_xsec
|__v2_12_8.version
|    |__NULL_${cfgname}
|__NULL
   |__${cfgname}
        |__ups
```

The xml file looks like the following figure.


```xml
<?xml version="1.0" encoding="ISO-8859-1"?>

<!-- generated by genie::XSecSplineList::SaveSplineList() -->

<genie_xsec_spline_list version="2.00" uselog="1">

<spline name="genie::AhrensNCELPXSec/Default/nu:-12;tgt:1000000010;N:2112;proc:Weak[NC],QES;" nknots="500">
        <knot> <E>    0.01000 </E> <xsec> 3.5935638244e-15 </xsec> </knot>
        <knot> <E>    0.01021 </E> <xsec> 3.7960303813e-15 </xsec> </knot>
        <knot> <E>    0.01043 </E> <xsec> 4.0068096158e-15 </xsec> </knot>
        <knot> <E>    0.01066 </E> <xsec> 4.2262367883e-15 </xsec> </knot>
        <knot> <E>    0.01089 </E> <xsec> 4.4546598821e-15 </xsec> </knot>
        <knot> <E>    0.01112 </E> <xsec> 4.6924400523e-15 </xsec> </knot>
        <knot> <E>    0.01136 </E> <xsec> 4.9399520870e-15 </xsec> </knot>
        <knot> <E>    0.01160 </E> <xsec> 5.1975848826e-15 </xsec> </knot>
        <knot> <E>    0.01185 </E> <xsec> 5.4657419317e-15 </xsec> </knot>
        <knot> <E>    0.01211 </E> <xsec> 5.7448418240e-15 </xsec> </knot>
        <knot> <E>    0.01237 </E> <xsec> 6.0353187615e-15 </xsec> </knot>
        <knot> <E>    0.01263 </E> <xsec> 6.3376230869e-15 </xsec> </knot>
        <knot> <E>    0.01290 </E> <xsec> 6.6522218255e-15 </xsec> </knot>
        <knot> <E>    0.01318 </E> <xsec> 6.9795992416e-15 </xsec> </knot>
        <knot> <E>    0.01346 </E> <xsec> 7.3202574083e-15 </xsec> </knot>
        <knot> <E>    0.01375 </E> <xsec> 7.6747167911e-15 </xsec> </knot>
        <knot> <E>    0.01405 </E> <xsec> 8.0435168461e-15 </xsec> </knot>
        <knot> <E>    0.01435 </E> <xsec> 8.4272166315e-15 </xsec> </knot>
        <knot> <E>    0.01466 </E> <xsec> 8.8263954332e-15 </xsec> </knot>
        <knot> <E>    0.01497 </E> <xsec> 9.2416534044e-15 </xsec> </knot>
        <knot> <E>    0.01529 </E> <xsec> 9.6736122182e-15 </xsec> </knot>
```

Event Generation in GENIE

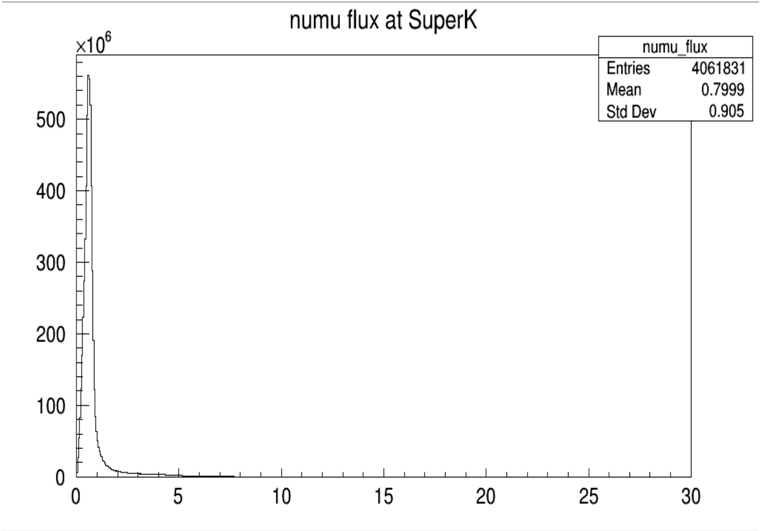
- gevgen - A generic GENIE event generation application for simple event generation cases. The application handles event generation for neutrinos scattered off a given target (or 'target mix'). It doesn't support event generation over ROOT/Geant4-based detector geometries. It handles mono-energetic flux neutrinos or neutrino fluxes described in simple terms (either via a functional form, a vector file or a ROOT TH1D histogram).

```
gevgen [-h][-r run#] -n nev -p neutrino_pdg -t target_pdg
 -e energy [-f flux][-w] [-seed random_number_seed]
 [--cross-section xml_file] [--event-generator-list
  list_name] [--message-thresholds xml_file] [--unphysical
  -event-mask mask][--event-record-print-level level]
[--mc-job-status-refresh-rate rate][--cache-file root_file]
```

The neutrino events generated was ν_μ (PDG code: 14) scattering off Fe^{56} (PDG code: 1000260560). This was done using the following command

```
gevgen -n 10000 -p 14 -t 1000260560 -e 0,10 --run 10 -f
/opt/GENIE-Generator_v2.12.8/data/flux/t2ksk.root,numu_flux
2989819 --cross-sections /home/jim/genie/genie_xsec/
v2_12_8/NULL/DefaultPlusMECWithNC/data/gxspl-FNALsmall.xml
```

The -f tag is for flux generator spectrum. Here t2ksk.root corresponds to the flux spectrum in superK which is as shown below.



The gevgen interaction summary is shown below. Also the same can be found in output status file.

```
--------------------------------------------------------------------------------------------------------
GENIE Interaction Summary
--------------------------------------------------------------------------------------------------------
[-] [Init-State]
 |--> probe        : PDG-code = 14 (nu_mu)
 |--> nucl. target : Z = 26, A = 56, PDG-Code = 1000260560 (Fe56)
 |--> hit nucleon  : PDC-Code = 2112 (neutron)
 |--> hit quark    : no set
 |--> probe 4P     : (E =     0.701141, Px =     0.000000, Py =     0.000000, Pz =     0.701141)
 |--> target 4P    : (E =    52.089777, Px =     0.000000, Py =     0.000000, Pz =     0.000000)
 |--> nucleon 4P   : (E =     0.927820, Px =    -0.196354, Py =     0.074695, Pz =    -0.109214)
[-] [Process-Info]
 |--> Interaction : Weak[CC]
 |--> Scattering  : QES
[-] [Kinematics]
 |--> *Selected* Bjorken x = 0.815494
 |--> *Selected* Inelasticity y = 0.281620
 |--> *Selected* Momentum transfer Q2 (>0) = 0.333974
 |--> *Selected* Hadronic invariant mass W = 0.938270
[-] [Exclusive Process Info]
 |--> charm prod.  : false |--> strange prod.  : false
 |--> f/s nucleons : N(p) = 0 N(n) = 0
 |--> f/s pions    : N(pi^0) = 0 N(pi^+) = 0 N(pi^-) = 0
 |--> resonance    : [not set]
--------------------------------------------------------------------------------------------------------
```

The status file shows the mother daughter information for each event the following shows the corresponding information for the event number 9950

```
|------------------------------------------------------------------------------------------------------------------|
|GENIE GHEP Event Record [print level:   3]                                                                        |
|------------------------------------------------------------------------------------------------------------------|
| Idx |          Name | Ist |          PDG |   Mother  | Daughter   |      Px |      Py |      Pz |       E |      m  |
|------------------------------------------------------------------------------------------------------------------|
|   0 |         nu_mu |   0 |           14 |  -1 |  -1 |   4 |   4 |   0.000 |   0.000 |   4.591 |   4.591 |   0.000 |
|   1 |          Fe56 |   0 |   1000260560 |  -1 |  -1 |   2 |   3 |   0.000 |   0.000 |   0.000 |  52.090 |  52.090 |
|   2 |        proton |  11 |         2212 |   1 |  -1 |   5 |   5 |  -0.179 |  -0.090 |  -0.124 |   0.928 | **0.938 | M = 0.897
|   3 |          Mn55 |   2 |   1000250550 |   1 |  -1 |  12 |  12 |   0.179 |   0.090 |   0.124 |  51.162 |  51.162 |
|   4 |           mu- |   1 |           13 |   0 |  -1 |  -1 |  -1 |   0.384 |  -0.874 |   2.641 |   2.810 |   0.106 | P = 
(-0.137,0.311,-0.940)
|   5 |      HadrSyst |  12 |   2000000001 |   2 |  -1 |   6 |   8 |  -0.563 |   0.784 |   1.826 |   2.709 | **0.000 | M = 1.752
|   6 |        proton |  14 |         2212 |   5 |  -1 |   9 |   9 |  -0.551 |   0.140 |   0.787 |   1.350 |   0.938 | FSI = 3
|   7 |           pi+ |  14 |          211 |   5 |  -1 |  10 |  10 |  -0.124 |   0.611 |   0.982 |   1.171 |   0.140 | FSI = 1
|   8 |           pi0 |  14 |          111 |   5 |  -1 |  11 |  11 |   0.112 |   0.034 |   0.058 |   0.188 |   0.135 | FSI = 3
|   9 |        proton |   1 |         2212 |   6 |  -1 |  -1 |  -1 |  -0.304 |   0.236 |   0.892 |   1.350 |   0.938 |
|  10 |           pi+ |   1 |          211 |   7 |  -1 |  -1 |  -1 |  -0.124 |   0.611 |   0.982 |   1.171 |   0.140 |
|  11 |           pi0 |   1 |          111 |   8 |  -1 |  -1 |  -1 |   0.073 |   0.027 |   0.104 |   0.188 |   0.135 |
|  12 |      HadrBlob |  15 |   2000000002 |   3 |  -1 |  -1 |  -1 |  -0.029 |   0.001 |  -0.027 |  51.162 | **0.000 | M = 51.162
|------------------------------------------------------------------------------------------------------------------|
|       Fin-Init:                                                 |   0.000 |   0.000 |   0.000 |   0.000 |         |
|------------------------------------------------------------------------------------------------------------------|
|       Vertex:         nu_mu @ (x =     0.00000 m, y =     0.00000 m, z =     0.00000 m, t =    0.000000e+00 s)   |
|------------------------------------------------------------------------------------------------------------------|
| Err flag [bits:15->0] : 0000000000000000    |  1st set:                                                   none |
| Err mask [bits:15->0] : 1111111111111111    |  Is unphysical:    NO |   Accepted:   YES                        |
|------------------------------------------------------------------------------------------------------------------|
| sig(Ev) =       3.17399e-37 cm^2  | d2sig(x,y;E)/dxdy =       8.14213e-37 cm^2       | Weight =           1.00000 |
|------------------------------------------------------------------------------------------------------------------|
```

The output root file gntp.0.ghep.root is converted into a readable file using the "gntpc" command. Here "gst" stands for the standard GENIE Summary Tree format.

```
gntpc -i gntp.0.ghep.root -f gst -n 10000 -o output.root
```