

Weekly Presentation

14.10.2016
Ceren Güzelgün

$W^\pm \notin Z^0$ BOZONS

- Every interaction has its mediator(s).

EM \rightarrow Photon γ

Strong \rightarrow Gluons

Weak \rightarrow $W \neq Z$ bosons

Gravitational \rightarrow Gluons

GENERAL INFO

- $W \rightarrow$ can have ∓ 1 electric charge. $W^+ \not\equiv W^-$ are each others antiparticle.
- $Z^0 \rightarrow$ electrically neutral, its own antiparticle.
- All three have spin of 1. And they are very short lived particles.
- W boson \rightarrow named after "**Weak force**"
- Z boson \rightarrow "zero electric charge"
- They are very massive particles $\rightarrow \sim 100$ GeV
- First observed at Large e^-e^+ collider in CERN, 1983
- They are a huge triumph for SM, encouraged researchers to look further for Higgs.
- Both known to be created under Beta decay.

- $W \& Z$ bosons are intermediate vector bosons. They mediate the weak interaction.
- Weak force must be mediated by its own boson, just like electric force is carried by photons.
 - The weak boson must carry charge, in order to transfer it in weak interactions.
 - SO \rightarrow there must be at least 2 force carriers, namely : $W^+ \& W^-$
 - BUT : weak interactions doesn't always involve charge transfer. In theory there was a 3rd mediator needed. $\rightarrow Z$

- The problem: unlike photons, these carriers needed to have MASS.
 - This breaks the analogy with Electromagnetism.
 - Massless particles → a spin parallel or anti-parallel to their direction of motion.
 - Particles with mass can also have a spin PERPENDICULAR to their motion.
 - This caused all the problems. The symmetry between photons and $W\pm$ was 'broken.'



Abdus Salam



Steven Weinberg

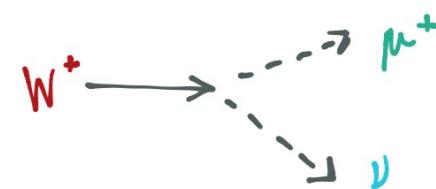
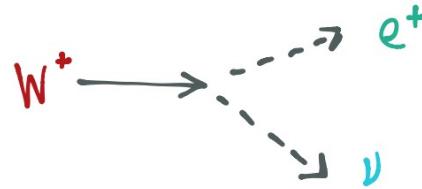
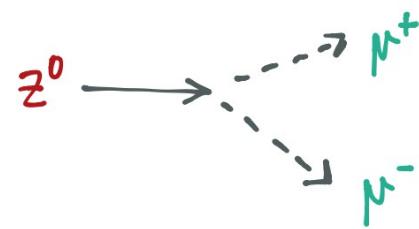
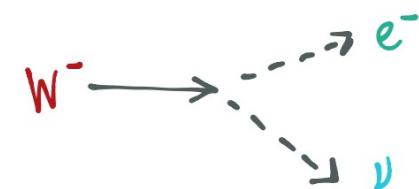
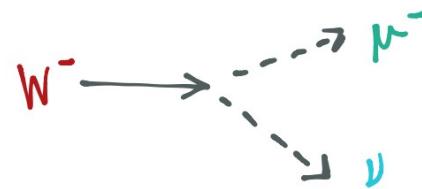
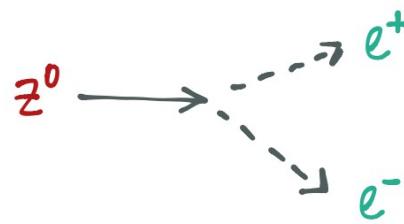


Sheldon Glashow

- Abdus Salam , S. Weinberg + Glashow \rightarrow symmetry breaking requires a Higgs Field.
 - EM + Weak Interactions \rightarrow symmetrical Gauge field : electroweak int. with massless mediating bosons.
 - This combined interaction only works under really high energy conditions , such as Big Bang.
 - at low energies \rightarrow EM \neq Weak interactions separate.

DECAYS

- In $W \& Z$ decays, two particles called **leptons** are produced.
- Z is 0-charged. It can decay to $\rightarrow 2$ leptons of opposite charges.
- W^\pm decays into \rightarrow a single electron, or muon + neutrino.



Basic W & Z decay examples.

In My Opinion:

- Why the W & Z bosons were interesting?
 - They were mediators that have MASS, and this was breaking their mathematical symmetry with photons, and EM interactions.
- Why the W & Z bosons were important?
 - Their theoretical existence were explained by the Higgs field. And after they were experimentally discovered, it must have been a great encouragement for the physicists to keep looking for the Higgs boson.

Project X

- a superconducting linear, proton accelerator.
- Headquarters : Fermilab
- high intensity , time flexibility, making multiple experiments simultaneously.
- with Project X:
 - Neutrino Experiments
 - Kaon physics
 - Muon Experiments
 - EDM measurements
 - Neutron - Antineutron Collisions
 - Hadronic structure & Spectroscopy
 - Lattice QCD calculations.

- The project consists of 4 stages: 3 main, 1 more in the long term.
 - Stage I : Upgrades to existing facilities
 - Stage II : Delivery of 3 concurrent beam levels
 - 2.9 MW → 3 GeV
 - 50-200 kW → 8 GeV
 - 2.3 MW → 60-120 GeV
 - Stage III : Building next level particle accelerators.

SOFTSUSY

- Read the README tutorial.
- Ran example files
 - ./softsusy.x ./rpvsoftsusy.x ./rpvneut.x
 - ./softsusy-nmssm.x ./higher.x
- Ran SUSY Les Houches Accord with the input files
 - LesHouchesInput
 - slha2Input
 - rpvHouchesInput

```
ceren@alaska ~/softsusy-3.7.3
[$ ls
CHANGES
COPYING
Config
GUIDELINES
INSTALL
Makefile
Makefile.am
Makefile.in
Makefile.nmssmtools
README
TODO
aclocal.m4
config.log
config.status
configure
configure.ac
doc
inOutFiles
libsoft.la
libtool
m4
rpvneut.x
rpvsoftsusy.x
setup_nmssmtools.sh
softpoint.x
softsusy-nmssm.x
softsusy.x
softsusy_nmssmtools.x.in
src
ver
```

```

ceren@alaska ~/softsusy-3.7.3
$ ./rpvneut.x
SOFTSUSY3.7.3 Ben Allanach, Markus Bernhardt 2009
If you use SOFTSUSY, please refer to B.C. Allanach, Comput. Phys. Commun. 143 (2002) 305, hep-ph/0104145;
For RPV aspects, B.C. Allanach and M.A. Bernhardt, Comp. Phys. Commun. 181 (2010) 232, arXiv:0903.1805.
For RPV neutrino masses, B.C. Allanach, M. Hanussek and S. Kom,
Comput. Phys. Commun. 183 (2012) 785, arXiv:1109.3735 [hep-ph]
# SOFTSUSY3.7.3 SLHA compliant output
# B.C. Allanach, Comput. Phys. Commun. 143 (2002) 305-331, hep-ph/0104145
# B.C. Allanach and M.A. Bernhardt, Comput. Phys. Commun. 181 (2010) 232,
# arXiv:0903.1805
# B.C. Allanach, M. Hanussek and C.H. Kom, arXiv:1109.3735
Block SPINFO          # Program information
  1   SOFTSUSY      # spectrum calculator
  2   3.7.3        # version number
Block MODSEL          # Select model
  1   1             # sugra
  4   1             # R-parity violating
Block SMINPUTS         # Standard Model inputs
  1   1.27916000e+02  # alpha_em^(-1)(MZ) SM MSbar
  2   1.16637000e-05  # G_Fermi
  3   1.18400000e-01  # alpha_s(MZ)MSbar
  4   9.11876000e+01  # MZ(pole)
  5   4.18000000e+00  # mb(mb)
  6   1.73500000e+02  # Mtop(pole)
  7   1.77699000e+00  # Mtau(pole)
 21  0.00000000e+00  # Mdown(2 GeV) MSbar
 22  0.00000000e+00  # Mup(2 GeV) MSbar
 23  0.00000000e+00  # Mstrange(2 GeV) MSbar
 24  0.00000000e+00  # Mcharm(Mcharm) MSbar
 11  0.00000000e+00  # Me(pole)
 13  0.00000000e+00  # Mu(pole)
Block VCKMIN          # input CKM mixing matrix parameters
  1   2.27200000e-01  # lambda_W
  2   8.18000000e-01  # A
  3   2.21000000e-01  # rhobar
  4   3.40000000e-01  # etabar (no phases used in SOFTSUSY yet though)
Block MINPAR           # SUSY breaking input parameters
  3   2.00000000e+01  # tanb, DRbar, Feynman gauge
  4   1.00000000e+00  # sign(mu)
  1   1.00000000e+02  # m0
  2   5.00000000e+02  # m12
  5   9.24000000e+02  # A0
Block EXTPAR            # scale of SUSY breaking BCs
  0   1.55399994e+16  # MX scale
Block RVLAMLQDIN        # input LLE couplings at MGUT
  1 1 1   3.95000000e-02  # lambda'_{111}
  1 3 3   3.00000000e-05  # lambda'_{133}
  2 1 1   -1.80000000e-02  # lambda'_{211}
  2 3 3   3.20000000e-05  # lambda'_{233}
  3 1 1   1.90000000e-02  # lambda'_{311}
  3 3 3   -3.50000000e-05  # lambda'_{333}
# SOFTSUSY-specific non SLHA information:
# MIXING=1 Desired accuracy=1.00000000e-04 Achieved accuracy=2.44313419e-05
# 3-loop RGE corrections are off. 2-loop Yukawa/g3 thresholds are off
# 2-loop SUSY QCD computation of squark/gluino pole masses are off
Block MASS              # Mass spectrum
# PDG code    mass      particle
  24  8.05052072e+01  # MW
  25  1.13403519e+02  # CP even neutral scalar
  35  6.20829671e+02  # CP even neutral scalar
  36  6.20727796e+02  # CP odd neutral scalar
  37  6.26124542e+02  # charged scalar

```

Partial screenshot of running the example file rpvneut

SUSYHIT

- A fortran package that calculates the decays of the Higgs bosons, and SUSY particles in the Minimal SUSY Standard Model
- It comes with the main program file sdecay.f and the subroutine hdecay.f
- There is a fortran compiler dependency. gfortran serves this purpose very well.
- After installing the SUSYHIT source files, compile with “make” and execute with “./run”

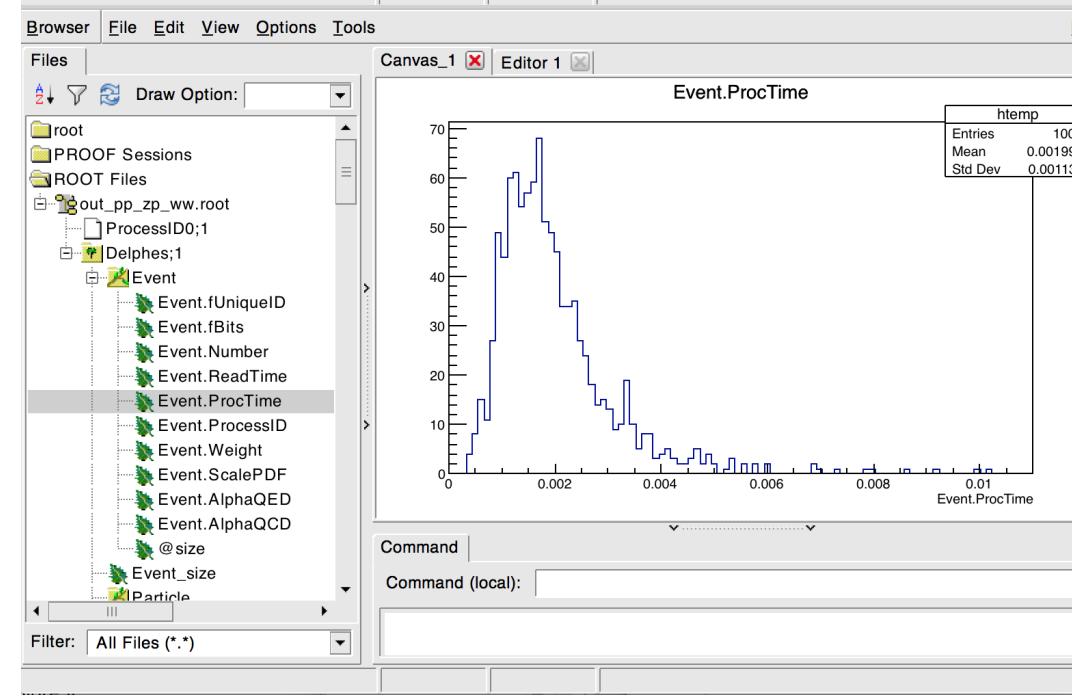
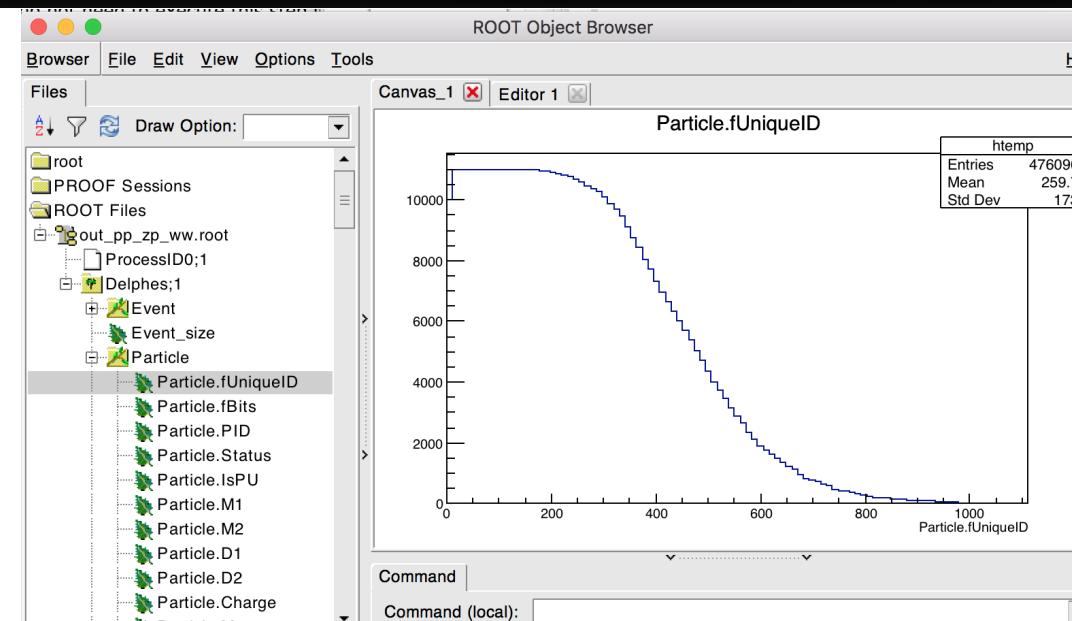
```
ceren@alaska ~susyhit
[$ ls
Xvegas.f          hgaga.f          makefile        suspect2.o      susylha.o
Xvegas.o          hgaga.o          run            suspect2.out    twooloophiggs.f
bsg.f             lightst3bod.f    sdecay.f       suspect2_lha.in  twooloophiggs.o
bsg.o             lightst3bod.o    sdecay.o       susyhit.in     susylha.f
hdecay.f          lightst4bod.f    slhaspectrum.in susyhit_slha.out
hdecay.o          lightst4bod.o    suspect2.f
```

Delphes

- Installed Delphes.
- Downloaded Z' to WW and dijet events from
 - wget
http://cp3.irmp.ucl.ac.be/downloads/delphes_tuto/pp_zp_ww.hep.gz
 - wget
http://cp3.irmp.ucl.ac.be/downloads/delphes_tuto/pp_jj.hep.gz
- Ran and explored the input file on ROOT.

```
ceren@alaska ~/Delphes-3.3.3
$ ls
CHANGELOG          FastJetDict_rdict.pcm      doc           pp_jj.hep
CMakeLists.txt     Makefile                  examples       pp_zp_ww.hep
COPYING            MinBias.pileup          external      python
CREDITS            ModulesDict_rdict.pcm    hepmc2pileup readers
ClassesDict_rdict.pcm README                 lhco2root   root2lhco
DelphesEnv.sh      README_4LHCb           libDelphes.so root2pileup
DelphesHepMC       VERSION                libDelphesDisplay.so stdhep2pileup
DelphesLHEF        cards                 libDelphesNoFastJet.so step_1.root
DelphesSTDHEP      classes              modules
DisplayDict_rdict.pcm configure           out_pp_zp_ww.root tmp
ExRootAnalysisDict_rdict.pcm converters        output.root
Example1           display
```

```
ceren@alaska ~/Delphes-3.3.3
$ ./DelphesSTDHEP cards/delphes_card_CMS.tcl out_pp_zp_ww.root pp_zp_ww.hep
ceren@alaska ~/Delphes-3.3.3
[$ root -l out_pp_zp_ww.root
[root [1] TBrowser t;
```



28.10.2016

Pythia

- Tried to solve the .hepmc output issue we've faced with Pythia.
- Adding

```
#include "HepMC/GenEvent.h"
#include "HepMC/IO_GenEvent.h"
```

- solved the issue in the workstation, as Onur and I both have confirmed. In my own computer there is still a problem. Might need to reconfigure hepmc.
- Therefore I worked on the workstation via ssh connection.

Pythia

- As for the “main24.cmnd” calling problem, I've found a way in the Pythia documentation to run a mainNN together with a mainNN.cmnd

```
./mymain.exe mymain.cmnd > mymain.out
```

- Tried to run the main41 with main24.cmnd accordingly. As for the main41.cc, I have tried multiple versions.

Pythia - 1st Try

- First I tried to generate 20000 events with the generator section, without calling main24.cmnd:

```
// Generator. Process selection. LHC initialization. Histogram.  
Pythia pythia;  
pythia.readString("Beams:eCM = 7000.");  
pythia.readString("HardQCD:all = on");  
pythia.readString("PhaseSpace:pTHatMin = 20.");  
pythia.init();  
Hist mult("charged multiplicity", 100, -0.5, 799.5);
```

- And got the .hepmc2 output:

```
HepMC::Version 2.06.09
HepMC::IO_GenEvent-START_EVENT_LISTING
E 0 -1 4.4340968265784852e+01 1.4678824080782335e-01 7.7541356235098068e-03 111 0 891 1 2 0 1 1.00000000
00000000e+00
N 1 "0"
U GEV CM
C 1.2342333032867470e+08 1.2342333032867470e+08
F 21 21 6.4037575195798738e-02 2.5609353448786121e-03 4.4340968265784852e+01 1.5912834486093796e+00 1.82
55737977917445e+01 0 0
V -1 0 0 0 0 0 0 2 0
P 3 21 0 0 2.2413151318529557e+02 2.2413151318529557e+02 0 21 0 0 -3 2 1 101 2 102
P 11 21 1.6043474888547685e+01 3.7520697520069257e+00 2.2254541119145742e+01 2.7689993036242058e+01 0 43
0 0 -15 2 1 105 2 101
V -2 0 0 0 0 0 0 1 0
P 4 21 0 0 -8.9632737070751425e+00 8.9632737070751425e+00 0 21 0 0 -3 2 1 103 2 104
V -3 0 0 0 0 0 0 2 0
P 5 21 1.3857841873089052e+01 4.2119849066298890e+01 9.0566648815463992e+01 1.0083867980299486e+02 0 23
0 0 -6 2 1 101 2 104
P 6 21 -1.3857841873089052e+01 -4.2119849066298890e+01 1.2460159066275631e+02 1.3225610708937572e+02 0 2
3 0 0 -7 2 1 103 2 102
V -4 0 0 0 0 0 0 1 0
P 7 21 1.7763568394002505e-15 2.2204460492503131e-16 3.5750032725166238e+02 3.5750032725166238e+02 0 41
0 0 -1 2 1 105 2 102
V -5 0 0 0 0 0 0 2 0
P 8 21 0 0 -8.9632737070751602e+00 8.9632737070751034e+00 0 42 0 0 -2 2 1 103 2 104
P 21 21 7.2933989235458494e+00 7.4163430435031374e+00 1.1258824205985299e+01 1.5328305010606279e+01 0 43
0 0 -21 2 1 104 2 110
V -6 0 0 0 0 0 0 1 0
P 9 21 7.0073848811582788e+00 4.0517740255993978e+01 1.3901025724621570e+02 1.4496427952424659e+02 0 44
0 0 -13 2 1 101 2 104
V -7 0 0 0 0 0 0 1 0
P 10 21 -2.3050859769705959e+01 -4.4269810008000903e+01 1.8727225517922568e+02 1.9380932839824837e+02 0
44 0 0 -14 2 1 103 2 102
```

- cakir/Programs/pythia1810/examples/hepmcoutCeren.hepmc2

Pythia - 2nd Try

- Then tried to run the same file, as the documentation suggested;
 - ./main41 main24.cmnd

```
HepMC::Version 2.06.09
HepMC::IO_GenEvent-START_EVENT_LISTING
E 0 -1 4.4340968265784852e+01 1.4678824080782335e-01 7.7541356235098068e-03 111
0 891 1 2 0 1 1.000000000000000e+00
N 1 "0"
U GEV CM
C 1.2342333032867470e+08 1.2342333032867470e+08
F 21 21 6.4037575195798738e-02 2.5609353448786121e-03 4.4340968265784852e+01 1.5
912834486093796e+00 1.8255737977917445e+01 0 0
V -1 0 0 0 0 0 2 0
P 3 21 0 0 2.2413151318529557e+02 2.2413151318529557e+02 0 21 0 0 -3 2 1 101 2 1
02
P 11 21 1.6043474888547685e+01 3.7520697520069257e+00 2.2254541119145742e+01 2.7
689993036242058e+01 0 43 0 0 -15 2 1 105 2 101
V -2 0 0 0 0 0 1 0
P 4 21 0 0 -8.9632737070751425e+00 8.9632737070751425e+00 0 21 0 0 -3 2 1 103 2
104
V -3 0 0 0 0 0 2 0
P 5 21 1.3857841873089052e+01 4.2119849066298890e+01 9.0566648815463992e+01 1.00
83867980299486e+02 0 23 0 0 -6 2 1 101 2 104
P 6 21 -1.3857841873089052e+01 -4.2119849066298890e+01 1.2460159066275631e+02 1.
3225610708937572e+02 0 23 0 0 -7 2 1 103 2 102
V -4 0 0 0 0 0 1 0
P 7 21 1.7763568394002505e-15 2.2204460492503131e-16 3.5750032725166238e+02 3.57
50032725166238e+02 0 41 0 0 -1 2 1 105 2 102
V -5 0 0 0 0 0 2 0
P 8 21 0 0 -8.9632737070751602e+00 8.9632737070751034e+00 0 42 0 0 -2 2 1 103 2
104
P 21 21 7.2933989235458494e+00 7.4163430435031374e+00 1.1258824205985299e+01 1.5
328305010606279e+01 0 43 0 0 -21 2 1 104 2 110
```

- By comparing the first couple of lines, this .hepmc2 output looks the same as the 1st version.
- This makes sense, for I haven't included the “ReadString:: "main24.cmnd" ” line in the generator section.

Pythia - 3rd Try

- Changed the generator section of main41.cc, and faced the issue after running ./make41 alone

```
// Generator. Process selection. LHC initialization. Histogram.  
Pythia pythia;  
pythia.readString("main24.cmnd");  
//pythia.readString("Beams:eCM = 7000.");  
pythia.readString("HardQCD:all = on");  
//pythia.readString("PhaseSpace:pTHatMin = 20.");  
pythia.readString("Random::setSeed = on");  
pythia.readString("Random::seed = 0");  
pythia.init();  
Hist mult("charged multiplicity", 100, -0.5, 799.5);
```

```
PYTHIA Error: input string not found in settings databases::  
    main24.cmnd  
PYTHIA Abort from Pythia::init: some user settings did not make sense  
PYTHIA Abort from Pythia::next: not properly initialized so cannot generate events  
  
*----- PYTHIA Event and Cross Section Statistics -----*  
| Subprocess | Code | Number of events | sigma +- delta  
|           |     | Tried   Selected Accepted | (estimated) (mb)| | |
|---|---|---|---|---|---|
| sum | | 0 | 0 | 0 | 0.000e+00 0.000e+00 |  
*----- End PYTHIA Event and Cross Section Statistics -----*  
  
*----- PYTHIA Error and Warning Messages Statistics -----*  
| times message |  
| 1 Abort from Pythia::init: some user settings did not make sense  
| 10000 Abort from Pythia::next: not properly initialized so cannot generate events  
*----- End PYTHIA Error and Warning Messages Statistics -----*  
guzelgun@itu-cms:/home/cakir/Programs/pythia8210/examples$
```

Pythia - 4th Try

- Tried with another generator section, as a shot in the dark.

```
// Generator. Process selection. LHC initialization. Histogram.  
Pythia pythia;  
pythia.readString("main24.cmnd");  
pythia.readString("Beams:eCM = 7000.");  
pythia.readString("HardQCD:all = on");  
pythia.readString("PhaseSpace:pTHatMin = 20.");  
pythia.readString("Random::setSeed = on");  
pythia.readString("Random::seed = 0");  
pythia.init();  
Hist mult("charged multiplicity", 100, -0.5, 799.5);
```

- Got the same error. Tried with ./main41, and afterwards “ ./main41 main24.cmnd ” both.

11.10.2016

Reading Information From an External File with Pythia

- We had a problem for a few weeks, not being able to read information from “main24.cmnd” file.
- We were trying with the command line:
 - `pythia.readString("main24.cmnd");`
- And the error was:

```
PYTHIA Error: input string not found in settings databases::  
  main24.cmnd  
PYTHIA Abort from Pythia::init: some user settings did not make sense  
PYTHIA Abort from Pythia::next: not properly initialized so cannot generate events
```

Reading Information From an External File with Pythia

- The correct command should have been
 - `pythia.readFile("main24.cmnd", 0);`
- When fixed, .hepmc outputs can be successfully gotten from main41.
- If we want to read only a specific parameter from a file, that can be done as: (ex: LHEF)
 - `pythia.readString("Beams:LHEF = ttbar.lhe")`
- When to use `read"String"` or `read"File"` should not be mistaken.

main41

- I tried to read “main24.cmnd” within main41.

```
// Specify file where HepMC events will be stored.  
HepMC::IO_GenEvent ascii_io("hepmcout24C.hepmc", std::ios::out);  
  
// Generator. Process selection. LHC initialization. Histogram.  
Pythia pythia;  
pythia.readFile( "main24.cmnd", 0);
```

- There were no warnings, and my output “hepmcout24C.hepmc” was successfully created.

Main41 - output

```
HepMC::Version 2.06.09
HepMC::IO_GenEvent-START_EVENT_LISTING
E 0 -1 4.2909187395059448e+01 1.4765629989774828e-01 7.7512750449987323e-03 114 0 7 1
 2 0 1 1.0000000000000000e+00
N 1 "0"
U GEV MM
C 1.6970491506730981e+07 1.6970491506730981e+07
F -1 -2 2.3389801603964032e-03 6.4260525194150028e-02 4.2909187395059448e+01 9.201711
0363574717e-01 1.2132670416705740e-01 0 0
V -1 0 0 0 0 0 1 0
P 3 -1 0 0 8.1864305613874109e+00 8.1864305613874109e+00 0 21 0 0 -3 1 2 101
V -2 0 0 0 0 0 1 0
P 4 -2 0 0 -2.2491183817952509e+02 2.2491183817952509e+02 0 21 0 0 -3 1 2 102
V -3 0 0 0 0 0 2 0
P 5 -1 -3.1193033244450650e+01 2.9463267664992689e+01 -1.0791820496242724e+02 1.16135
85719665079e+02 3.300000000000002e-01 23 0 0 -6 1 2 102
P 6 -2 3.1193033244450650e+01 -2.9463267664992689e+01 -1.0880720265571036e+02 1.16962
41154426163e+02 3.300000000000002e-01 23 0 0 -7 1 2 101
V -4 0 0 0 0 0 1 4 0
P 1 2212 0 0 3.4999998742356274e+03 3.500000000000005e+03 9.382700000000005e-01 4 0
 0 -4 0
P 7 -1 4.3603148464931063e-01 -5.1492948653461135e-01 8.1711184650238060e+00 8.198929
9790375849e+00 0 61 0 0 -1 1 2 101
P 11 2101 -1.4466228041795454e-01 3.3746330712980999e-02 2.6589437357536008e+03 2.658
9438030151287e+03 5.793300000000001e-01 1 0 0 0 1 2 103
P 12 2 -9.8376869147663359e-02 4.0439591151480481e-01 6.7728729161866829e+02 6.772874
9988628363e+02 3.300000000000002e-01 1 0 0 0 1 1 101
P 13 1 -1.9299233508369018e-01 7.6787244306822836e-02 1.5558014931099191e+02 1.555806
3794043531e+02 3.300000000000002e-01 1 0 0 0 1 1 103
V -5 0 0 0 0 0 1 4 0
```

Reading .hepmc with Delphes

- I moved on to Delphes, to take my .hepmc file as input, and give a .root output.
- Went to my Delphes directory, and typed

```
ceren@alaska ~/Delphes-3.3.3
[$ ./DelphesHepMC cards/delphes_card_CMS.tcl outputhep.root hepmcout24.hepmc
```

```
** INFO: adding module ParticlePropagator ParticlePropagator
** INFO: adding module Efficiency ChargedHadronTrackingEfficiency
** INFO: adding module Efficiency ElectronTrackingEfficiency
** INFO: adding module Efficiency MuonTrackingEfficiency
** INFO: adding module MomentumSmearing ChargedHadronMomentumSmearing
** INFO: adding module MomentumSmearing ElectronMomentumSmearing
** INFO: adding module MomentumSmearing MuonMomentumSmearing
** INFO: adding module Merger TrackMerger
** INFO: adding module Calorimeter Calorimeter
** INFO: adding module Merger EFlowMerger
** INFO: adding module Efficiency PhotonEfficiency
** INFO: adding module Isolation PhotonIsolation
** INFO: adding module PdgCodeFilter ElectronFilter
** INFO: adding module Efficiency ElectronEfficiency
** INFO: adding module Isolation ElectronIsolation
** INFO: adding module Efficiency MuonEfficiency
** INFO: adding module Isolation MuonIsolation
** INFO: adding module Merger MissingET
** INFO: adding module Merger ScalarHT
** INFO: adding module PdgCodeFilter NeutrinoFilter
** INFO: adding module FastJetFinder GenJetFinder
** INFO: adding module Merger GenMissingET
** INFO: adding module FastJetFinder FastJetFinder
** INFO: adding module EnergyScale JetEnergyScale
** INFO: adding module JetFlavorAssociation JetFlavorAssociation
** INFO: adding module BTagging BTagging
** INFO: adding module TauTagging TauTagging
** INFO: adding module UniqueObjectFinder UniqueObjectFinder
** INFO: adding module TreeWriter TreeWriter
** INFO: initializing module Delphes
```

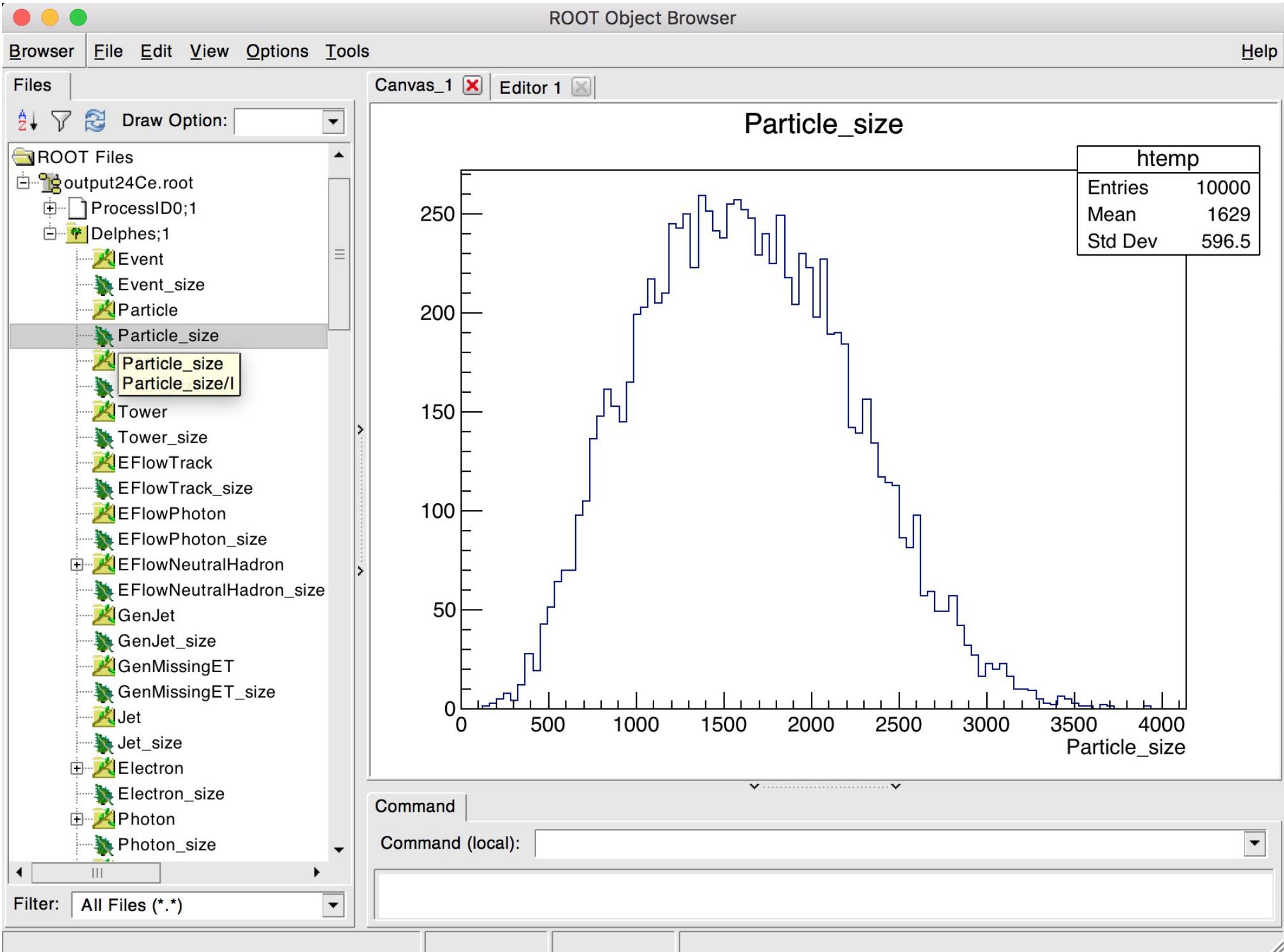
Looking at the .root output with ROOT

- The output from Delphes can be examined with ROOT.

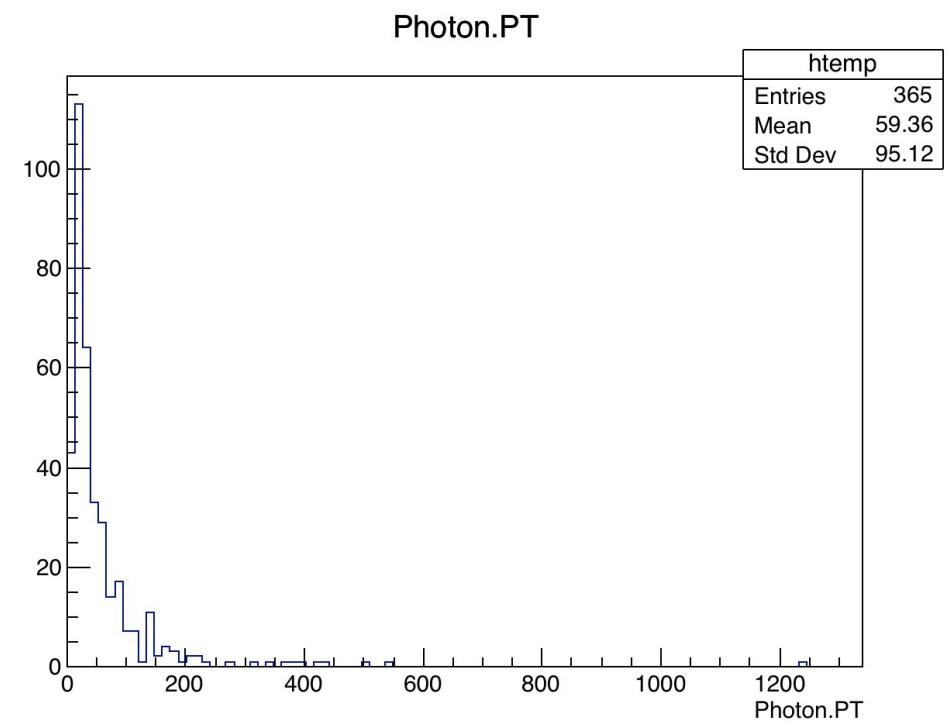
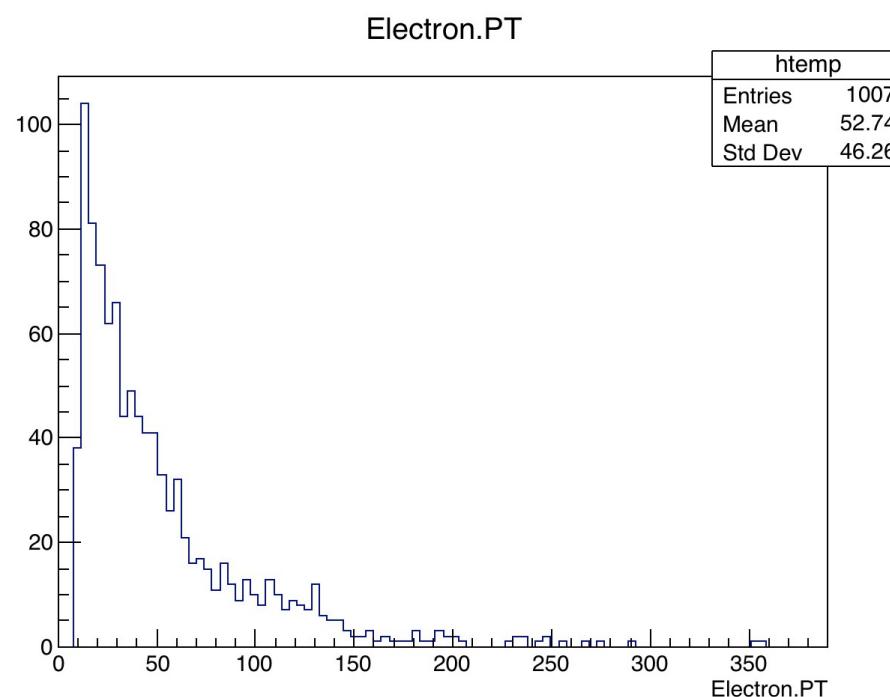
```
ceren@alaska ~/Delphes-3.3.3
[$ root
-----
| Welcome to ROOT 6.06/08                      http://root.cern.ch |
| (c) 1995-2016, The ROOT Team                  |
| Built for macosx64                            |
| From heads/v6-06-00-patches@v6-06-06-30-g3bae07b, Sep 01 2016, 14:28:05 |
| Try '.help', '.demo', '.license', '.credits', '.quit'/.q' |
-----
```

```
[root [0] TFile::Open("outputhep24.root")
```

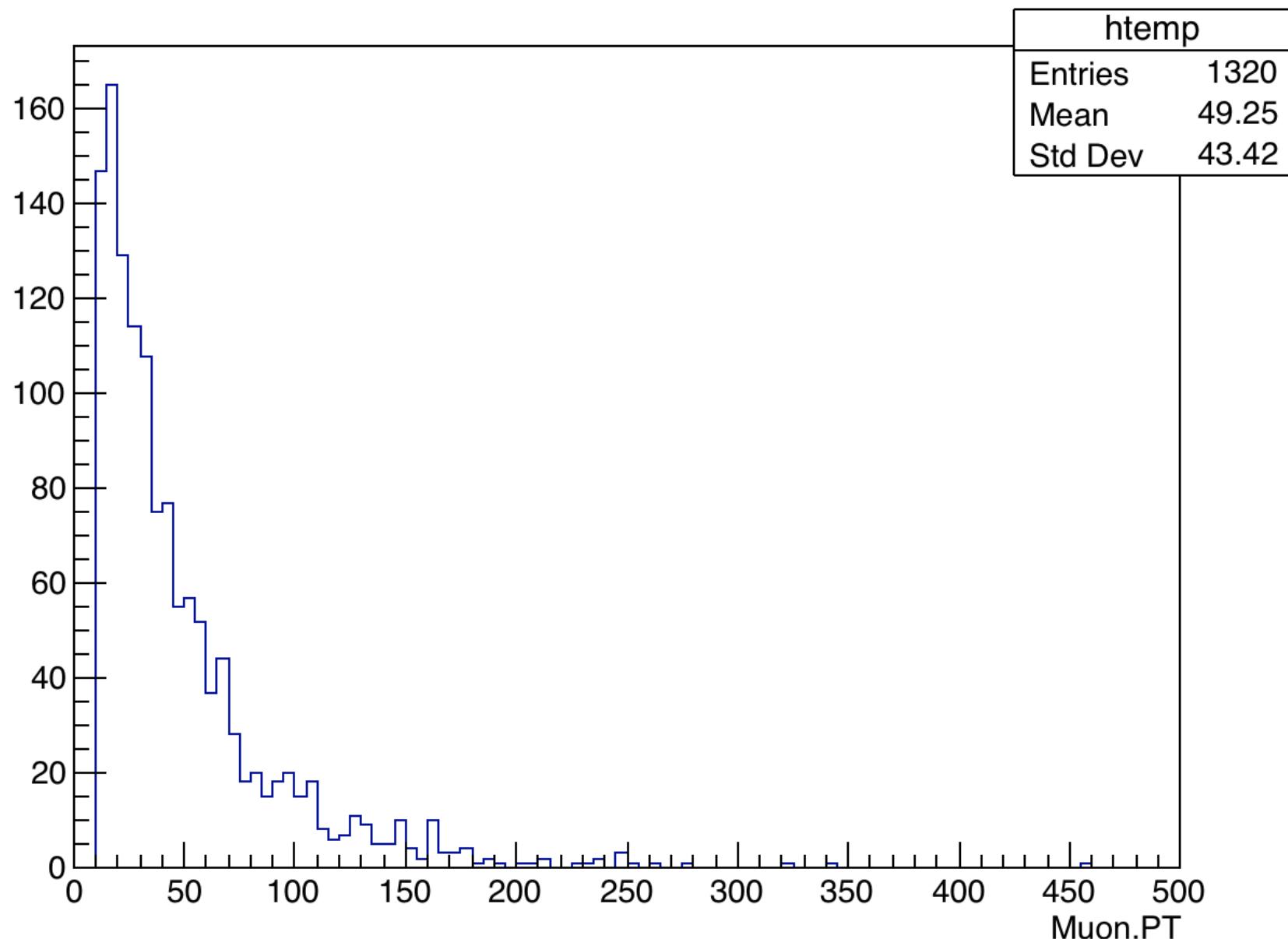
```
[root [1] TBrowser browser;
```



Examples from the .root File:



Muon.PT



Summarizing What I've Done

- Generated an event file in .hepmc format with Pythia8
- Fed this file as input to Delphes, and converted it to .root format.
- Examined the .root file by Tbrowser of ROOT.

Installing Pythia & HepMC

- Downloaded the Pythia & HepMC tarballs to my own computer.
- Uploaded them to my directory in our workstation with:

```
scp <file> <username>@<IP address or hostname>:<Destination>
```

- scp “pythia8219.tgz”
guzelgun@160.75.20.56:/home/guzelgun/programs
- Unzipped Pythia with
 - tar xvfz pythia8219.tgz

Installing Pythia & HepMC

- Created the directory where I'll install HepMC.
 - mkdir hepmc
 - cd hepmc
- Unzipped the hepmc tarball.
- Created two other directories
 - mkdir build install
 - cd build
- Configured.

```
... 13  
475 cd build/  
476 ./HepMC-2.06.09/configure -prefix=/home/guzelgun/programs/hepmc/install -with-momentum=GEV -with-length=CM  
477 make  
478 make check  
479 make install  
... 14
```

Installing Pythia & HepMC

- Went back to the `pythia2819` directory, and configured it with `hepmc`.

```
545 ./configure --with-hepmc2=/home/guzelgun/programs/hepmc/install
```

- If it works, you should get the following feedback.

```
-----|-----|-----  
| PYTHIA Configuration Summary |  
-----|-----|-----  
Configured for LINUX with the following options:  
  
The following optional external packages will be used:  
* HEPMC2
```

- As a final step, compile with “make”

Installing Delphes

- Download the latest version of Delphes from
 - <https://cp3.irmp.ucl.ac.be/projects/delphes>
- Unpack the tarball with
 - Tar xzvf Delphes-3.4.0.tar.gz
- Change directory to the Delphes-3.4.0 folder.
 - cd Delphes-3.4.0
- Compile with
 - make

Progress Report

18.11.2016

Looking at Input Files

- First, looked at softsusy_nmssmtools.x.in and studied the bash commands within.
 - NMSSM: Next-to-Minimal Supersymmetric Standard Model
- Let's divide the input file into sections, and study it one “if/fi” at a time.
- I added explanations as green commands next to each line

```

# directory of this script
BASEDIR=$(dirname $0)

# path to the NMSSMTools directory
NMSSMTOOLS_PATH="@NMSSMTOOLS_PATH@" //NMSSM: Next to Minimal Supersymmetric Standard Model

if test $# -ne 1 -o "x$1" != "xleshouches"; then //If the number of arguments are not 1, or the first positional
                                                //argument is not xleshouches
    cat <<EOF > /dev/stderr
                                                //Redirect the lines of the input file
Usage: ./`basename $0` leshouches < slha-input-file //output formatting
                                                //end of file
                                                //unsuccessful termination
EOF
    exit 1
fi

```

- **test:** it's a bash command that allows us to perform tests, and sets an exit code of 0 (True) or 1 (false) It has some various arguments. (Some of them are logical) Can be used to test files.
 - -eq: equal to
 - -ne: not equal to
 - -gt: greater than
 - -ge: greater than or equal to
 - -lt: less than
 - -le: less than or equal to
 - -o: logical or
 - -a: logical and
 - -z: true if null
 - -x: true if executable
 - -r: true if readable
 - -d: true if a directory
 - -e: true if exists
 - -f: true if a regular file
 - -c: true if a character special file
 - -b: true if a block special file
 - -p: true if a named pipe
 - -S: true if socket file
 - -s: true if not empty
 - -L: true if a symbolic link
 - -w: true if writable

```
if ! test -d "${NMSSMTOOLS_PATH}"; then          // if NMSSMTOOLS directory does not exist,
  echo "# Error: Cannot find NMSSMTools directory \\"${nmssmtools_dir}\\"      //display error
  exit 1                                         //unsuccessful termination
fi

if ! test -x "${NMSSMTOOLS_PATH}/main/nmh_slhainp"; then    // if nmh_slhainp is not executable
  echo "# Error: Cannot find ${NMSSMTOOLS_PATH}/main/nmh_slhainp" //display error
  echo "  Please run ./setup_nmssmtools.sh --nmssmtools-dir=/path/to/NMSSMTools --compile" //display hint
  exit 1                                         //unsuccessful termination
fi
```

- Second and third if loops.
- First loop controls if the NMSSMTOOLS exists in its directory with “! test -d “\$Path”; displays an error message if can't be found. Terminates unsuccessfully with “exit 1”
- Second loop tests if the nmh_slainp file is executable with “! test -x “\$Path”; and displays an error message if it's not. Again, terminates with “exit 1”

```

nmssmtools_input_file="${NMSSMTOOLS_PATH}/main/inp"

${BASEDIR}/softpoint.x leshouches < /dev/stdin | tee ${nmssmtools_input_file} //redirect output to multiple files

if test $? -ne 0; then                                //if the most recent termination is unsuccessful
    echo "# Error: invalid point in Softsusy"        //display error
    exit 1                                            //unsuccessful termination
fi

(cd ${NMSSMTOOLS_PATH}/main/ && ./nmh_slhainp)      //go to main and start ./nmh_slhainp

if test $? -ne 0; then                                //if the most recent termination is unsuccessful
    echo "# Error: NMSSMTools could not calculate the decays" //can't calculate decays, display error
    exit 1
fi

```

- Fourth and fifth if loops.
- First loop checks the recent exit status with “\$? -ne 0” which means “if the last exit is not 0 (TRUE or successful), display the error message “Invalid point in Softsusy”
- After then, the directory is changed and ./nmh_slhainp is runned.
- Recent exit status is checked again in the following loop, and an error message “NMSSMTools couldn't calculate the decays” is shown.

- Let's have a quick look at the other versions of “\$N” command we've seen in the previous page.
 - \$1, \$2, ... are the positional parameters.
 - \$*: internal field separator (IFS) expansion of all pos. parameters.
 - \$# : number of pos. parameters
 - \$@ : array like construct of all positional parameters
 - \$- : current options set
 - \$\$: pid of the current shell
 - \$_ : most recent parameter
 - \$? : most recent pipeline exit status
 - \$! : PID of the most recent background command
 - \$0 : name of the shell, or shell script

```

decay_output_file="${NMSSMTOOLS_PATH}/main/decay"
omega_output_file="${NMSSMTOOLS_PATH}/main/omega"

echo ""

if test -r "${decay_output_file}"; then          //if the decay output file is readable,
    cat "${decay_output_file}"                  //redirect to output
else
    echo "# Error: cannot open decay output file \\"${decay_output_file}\\" " //if not, display error.
fi

if test -r "${omega_output_file}"; then          //if the omega output file is readable
    cat "${omega_output_file}"                  //redirect to output
else
    echo "# Error: cannot open micrOmegas output file \\"${omega_output_file}\\" " //if not, display error.
fi

```

- Check if `decay_output_file` is readable, redirect if it is. If not, display error “cannot open decay output file”
- Check if `omega_output_file` is readable, redirect if it is. Else, display error.

suspect2_lha.in

```
Block MODSEL      # Select Model
  * General MSSM    # Minimal SUSY model
  * SUGRA           # Supergravity
  * GMSB            # Gauge Mediated Symmetry Breaking
  * AMSB            # Anomaly Mediated Symmetry Breaking
  * Bottom-up RGE   # RGE: Renormalization Group Eqns
  * mSUGRA          # Gravity Mediated Symmetry Model
Block SU_ALGO     # special unitary algorithms
Block SMINPUTS    # Standard Model Inputs
```

ps: Summarized the code block by block.

Block MINPAR # Specific Model input parameters

* input for SUGRA models

* " " GMSB "

* " " AMSB "

Block EXTPAR # General MSSM output, when uncommented,
replaces MINPAR.

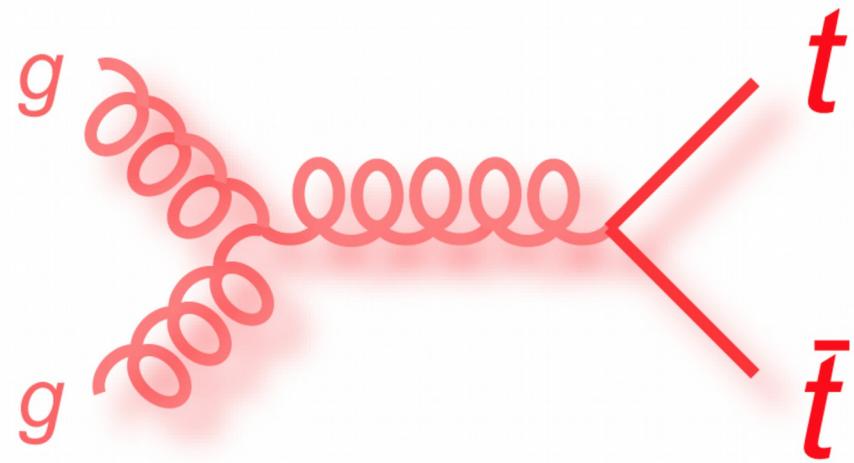
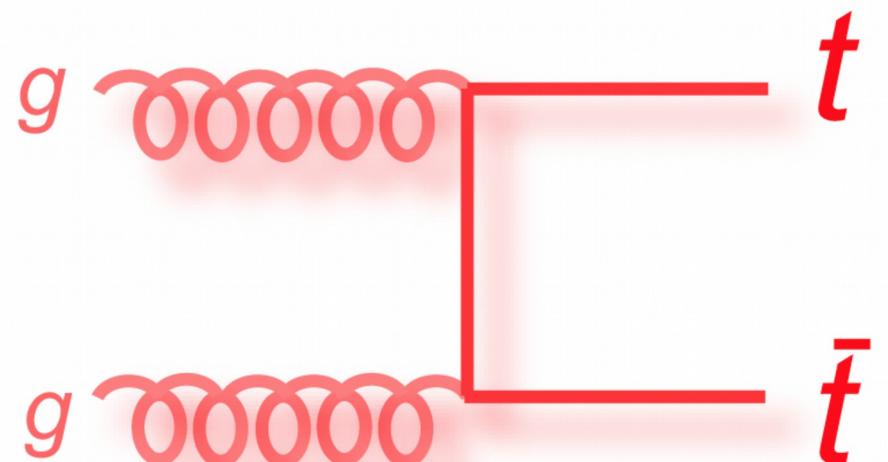
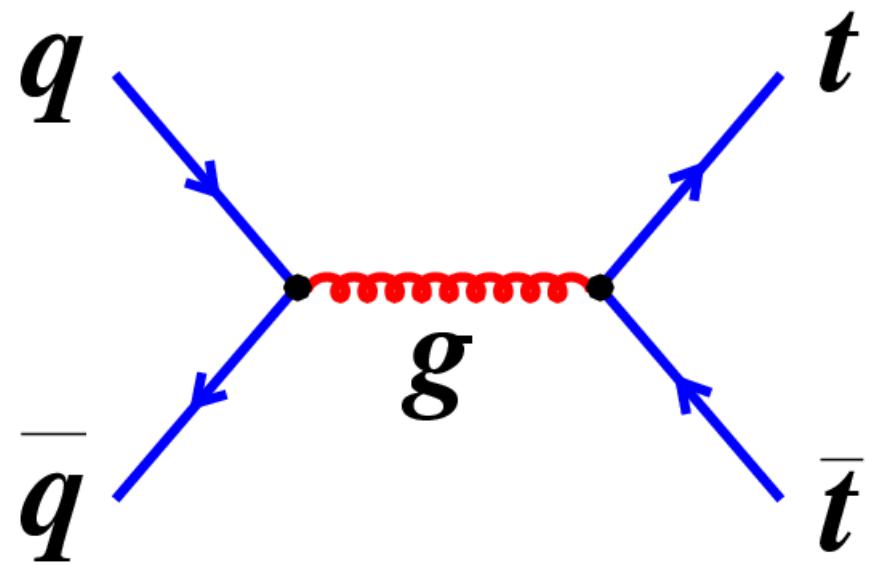
Rest of suspect2_lha.in

$g\ g \rightarrow q\ q\bar{}$ and $q\ q\bar{}$ $\rightarrow t\ t\bar{}$ with Pythia

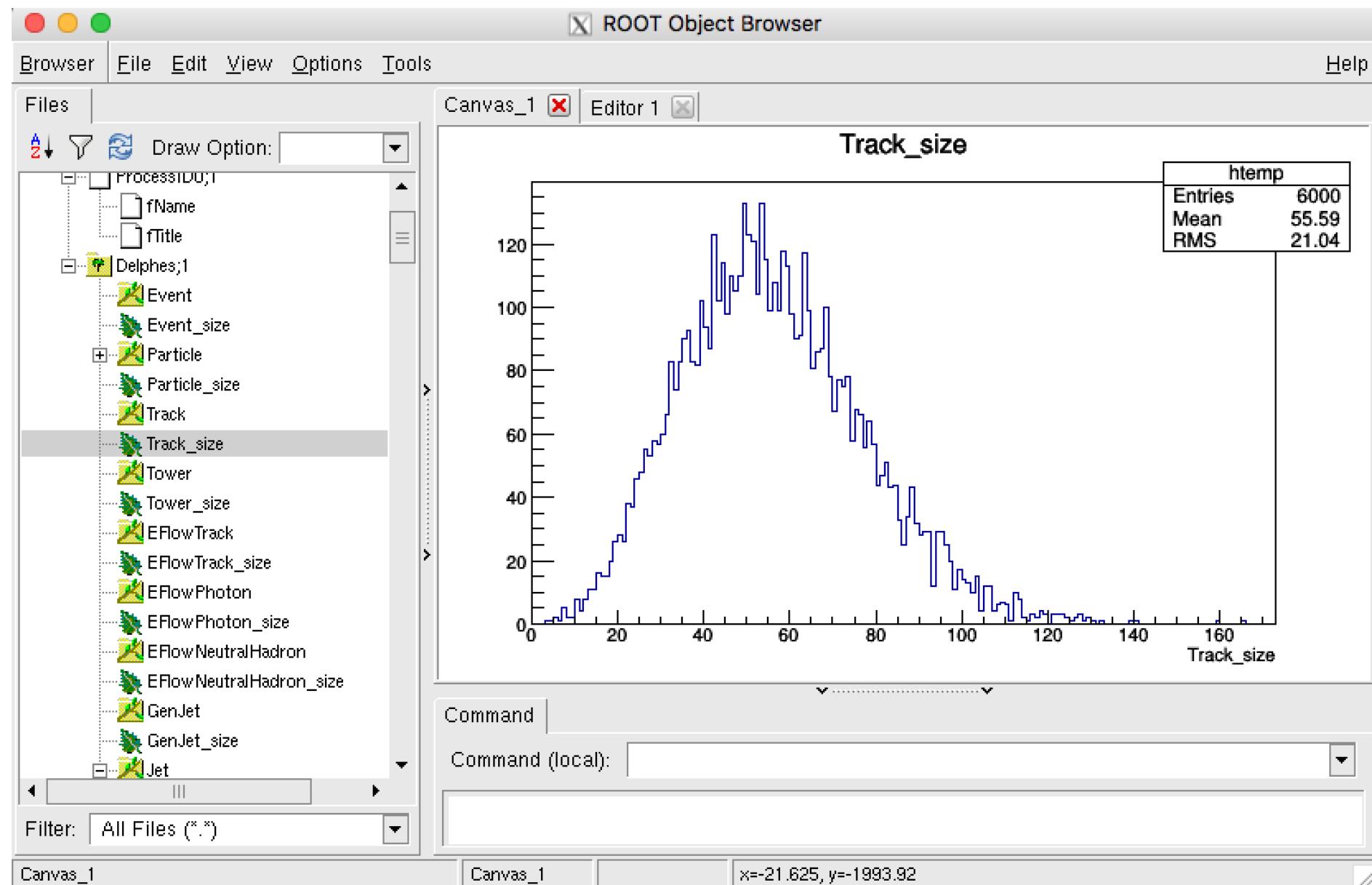
```
! Example 2: t-tbar production.  
Top:gg2ttbar = on                                ! g g -> t tbar  
Top:qqbar2ttbar = on                            ! q qbar -> t tbar
```

In main03.cmnd the following choices were activated. In main41.cc, main03.cmnd was imported as:

```
// Interface for conversion from Pythia8::Event to HepMC event.  
HepMC::Pythia8ToHepMC ToHepMC;  
  
// Specify file where HepMC events will be stored.  
HepMC::IO_GenEvent ascii_io("hepmcoutop.hepmc", std::ios::out);  
  
// Generator. Process selection. LHC initialization. Histogram.  
Pythia pythia;  
pythia.readFile("main03.cmnd",0);  
pythia.readString("Random::setSeed = on");  
pythia.readString("Random::seed = 0");  
//pythia.readString("Beams:eCM = 8000.");  
//pythia.readString("HardQCD:all = on");  
//pythia.readString("PhaseSpace:pTHatMin = 20.");  
pythia.init();  
Hist mult("charged multiplicity", 100, -0.5, 799.5);
```

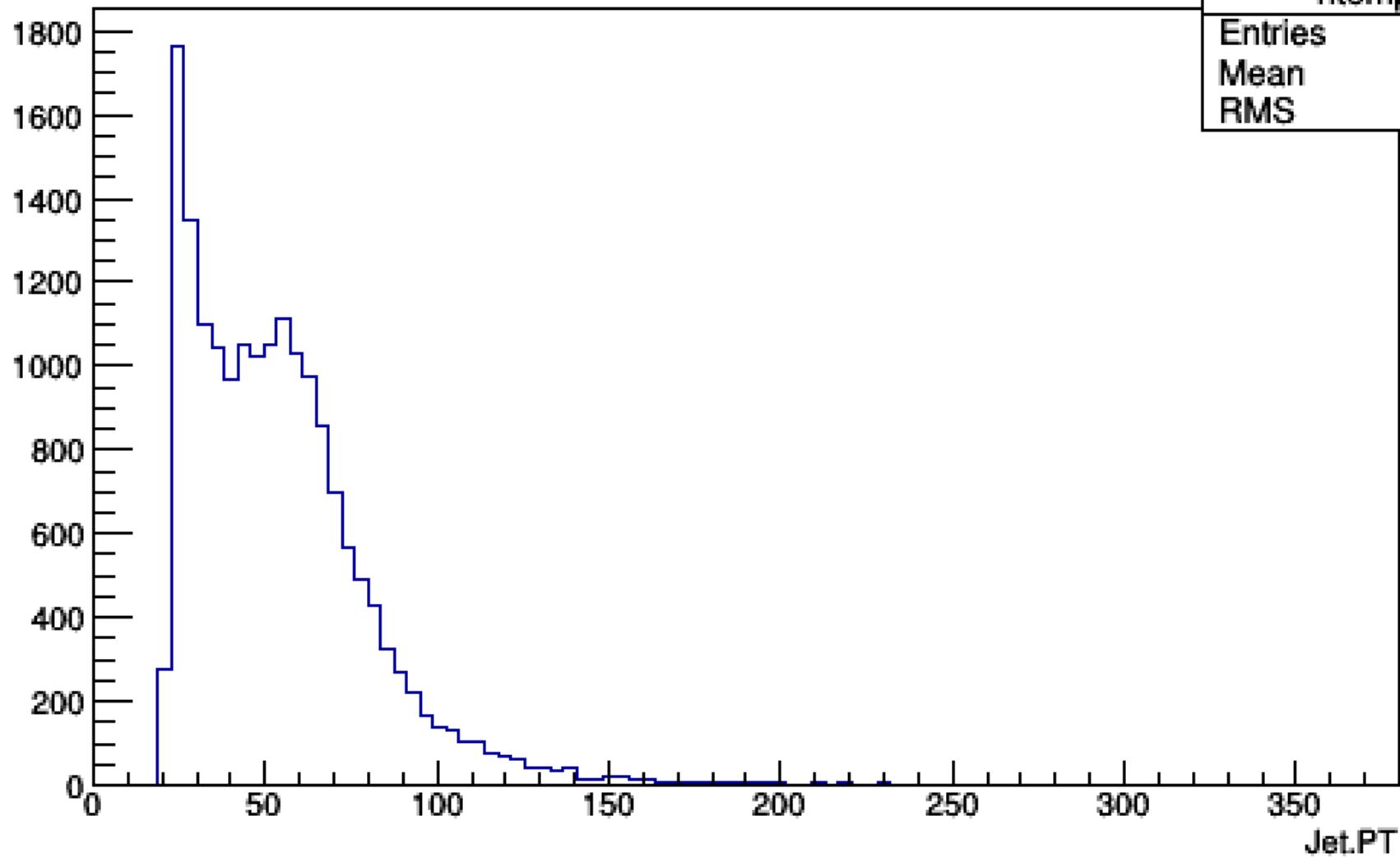


Output as outputop.root:



Jet.PT

htemp	
Entries	17783
Mean	54.58
RMS	26.88

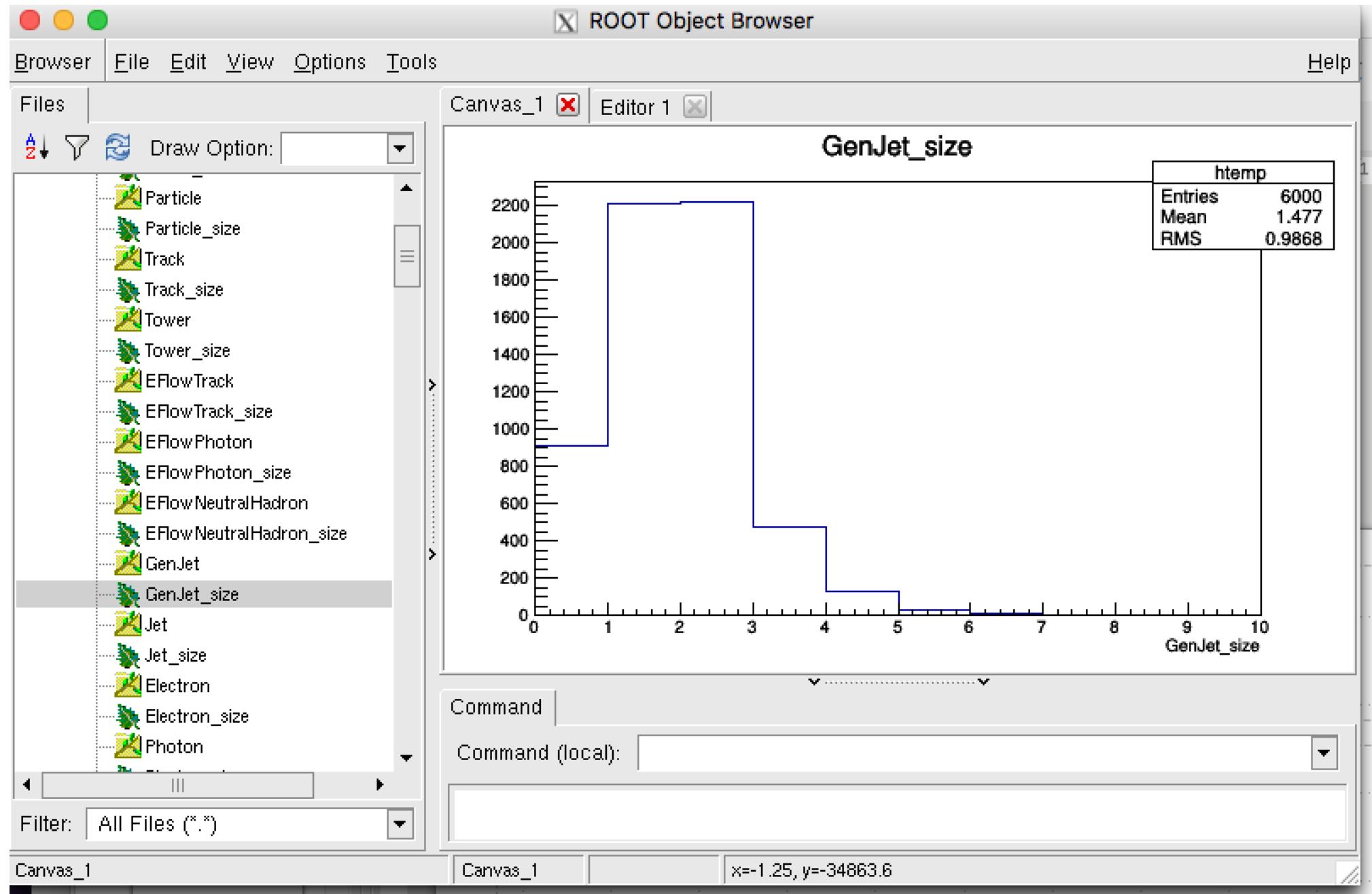


$q \bar{q} \rightarrow Z\bar{0} W^{+-}$ and $q \bar{q} \rightarrow W^+ W^-$

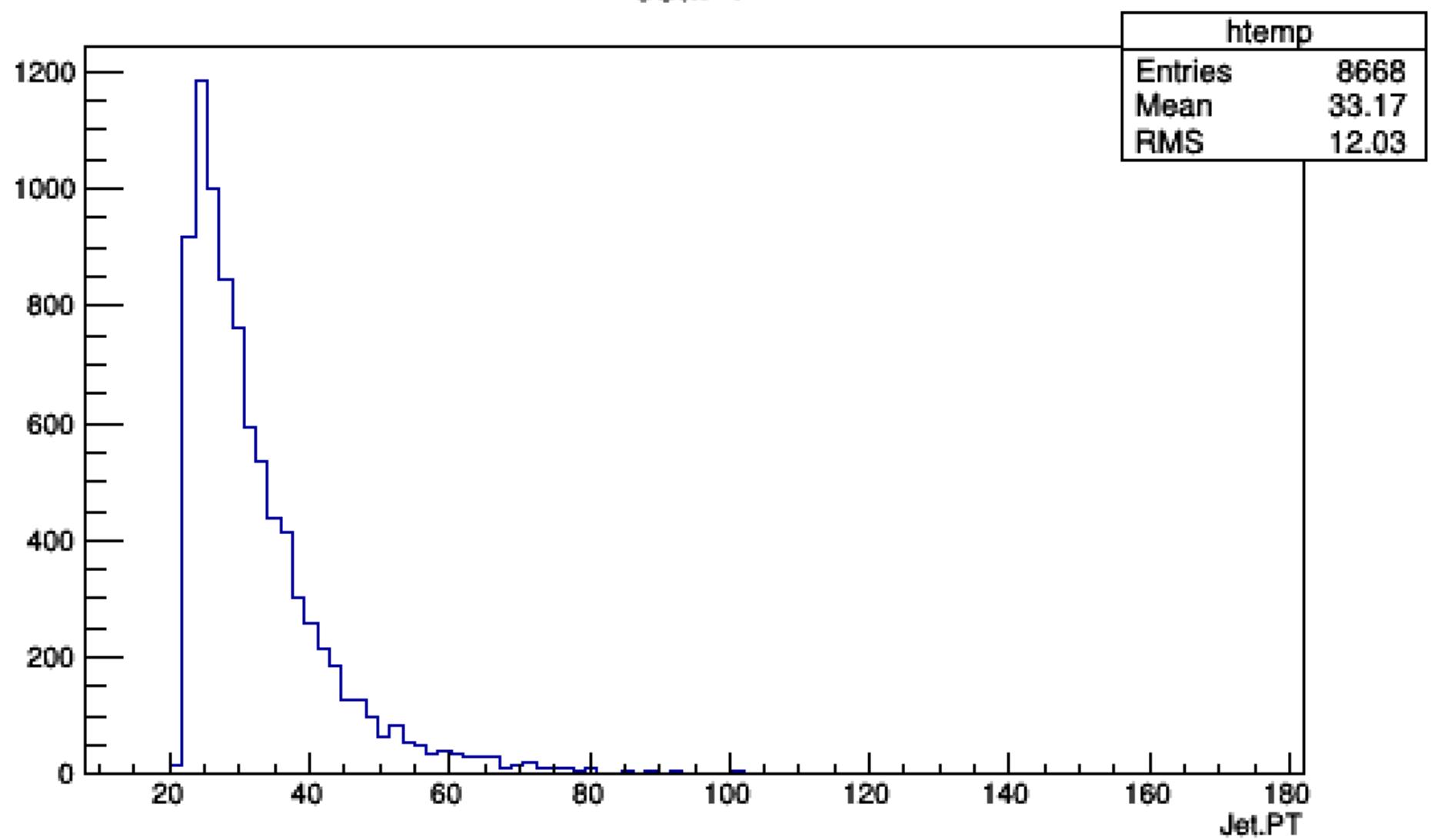
```
! Example 4: gauge boson pair production; set pTmin. Not yet complete.  
WeakDoubleBoson:ffbar2ZW = on           ! q qbar -> Z0 W+-  
WeakDoubleBoson:ffbar2WW = on           ! q qbar -> W+ W-  
PhaseSpace:pTHatMin = 20.                ! minimal pT scale in process
```

In main03.cmnd, the events were activated.

```
// Specify file where HepMC events will be stored.  
HepMC::IO_GenEvent ascii_io("hepmcoutW.hepmc", std::ios::out);  
  
// Generator. Process selection. LHC initialization. Histogram.  
Pythia pythia;  
pythia.readFile("main03.cmnd",0);  
pythia.readString("Random::setSeed = on");  
pythia.readString("Random::seed = 0");  
//pythia.readString("Beams:eCM = 8000.");  
//pythia.readString("HardQCD:all = on");  
//pythia.readString("PhaseSpace:pTHatMin = 20.");  
pythia.init();  
Hist mult("charged multiplicity", 100, -0.5, 799.5);
```



Jet.PT



Ongoing:

- Installed Herwig7 and started reading the user manual, studied the examples.

Progress Report

10.02.2017

For e-

```
using namespace Pythia8;

int main() {

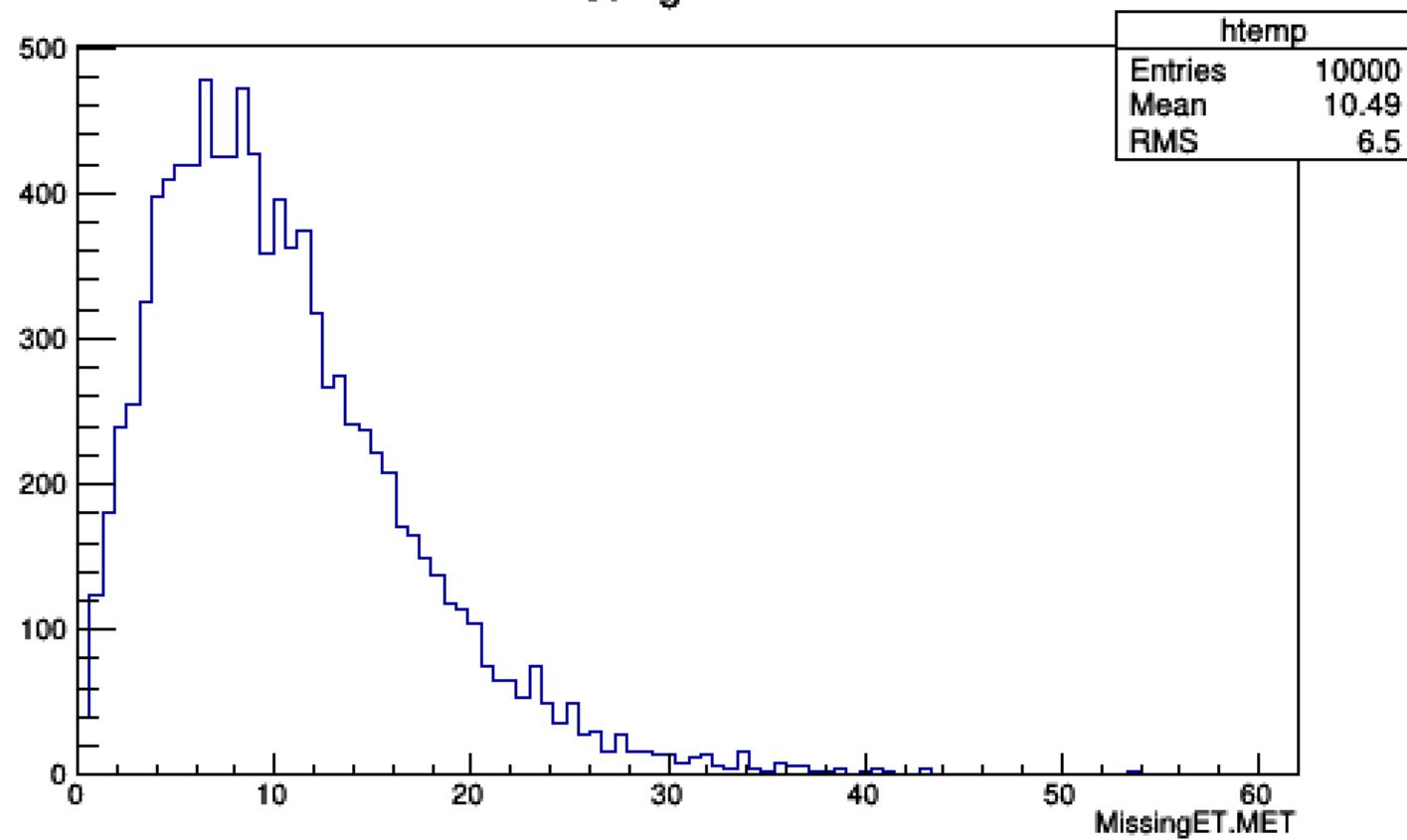
    // Interface for conversion from Pythia8::Event to HepMC event.
    HepMC::Pythia8ToHepMC ToHepMC;

    // Specify file where HepMC events will be stored.
    HepMC::IO_GenEvent ascii_io("leptonProd2.hepmc", std::ios::out);

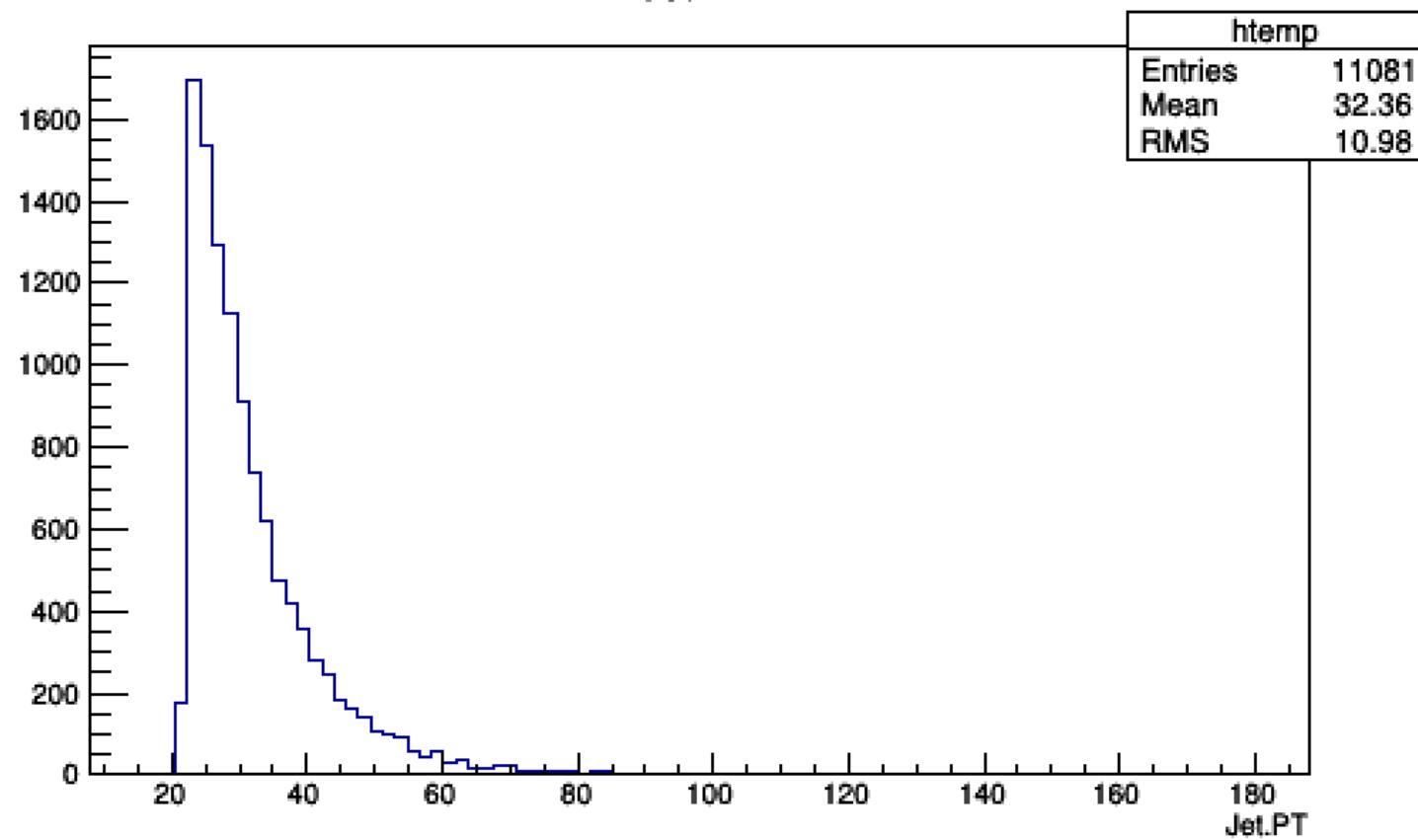
    // Generator. Process selection. LHC initialization. Histogram.
    Pythia pythia;
    pythia.readFile("main03.cmnd",0);
    pythia.readString("Random::setSeed = on");
    pythia.readString("Random::seed = 0");
    pythia.readString("Beams:eCM = 7000.");
    pythia.readString("HardQCD:all = on");
    pythia.readString("PhaseSpace:pTHatMin = 20.");
    pythia.readString("WeakBosonAndParton:qqbar2gmZg = on");
    pythia.readString("WeakBosonAndParton:qg2gmZq = on");
    pythia.readString("23:onMode = off");
    pythia.readString("23:onIfAny = 11");
    pythia.init();
    Hist mult("charged multiplicity", 100, -0.5, 799.5);

    // Begin event loop. Generate event. Skip if error.
    for (int iEvent = 0; iEvent < 10000; ++iEvent) {
        if (!pythia.next()) continue;
    }
}
```

MissingET.MET

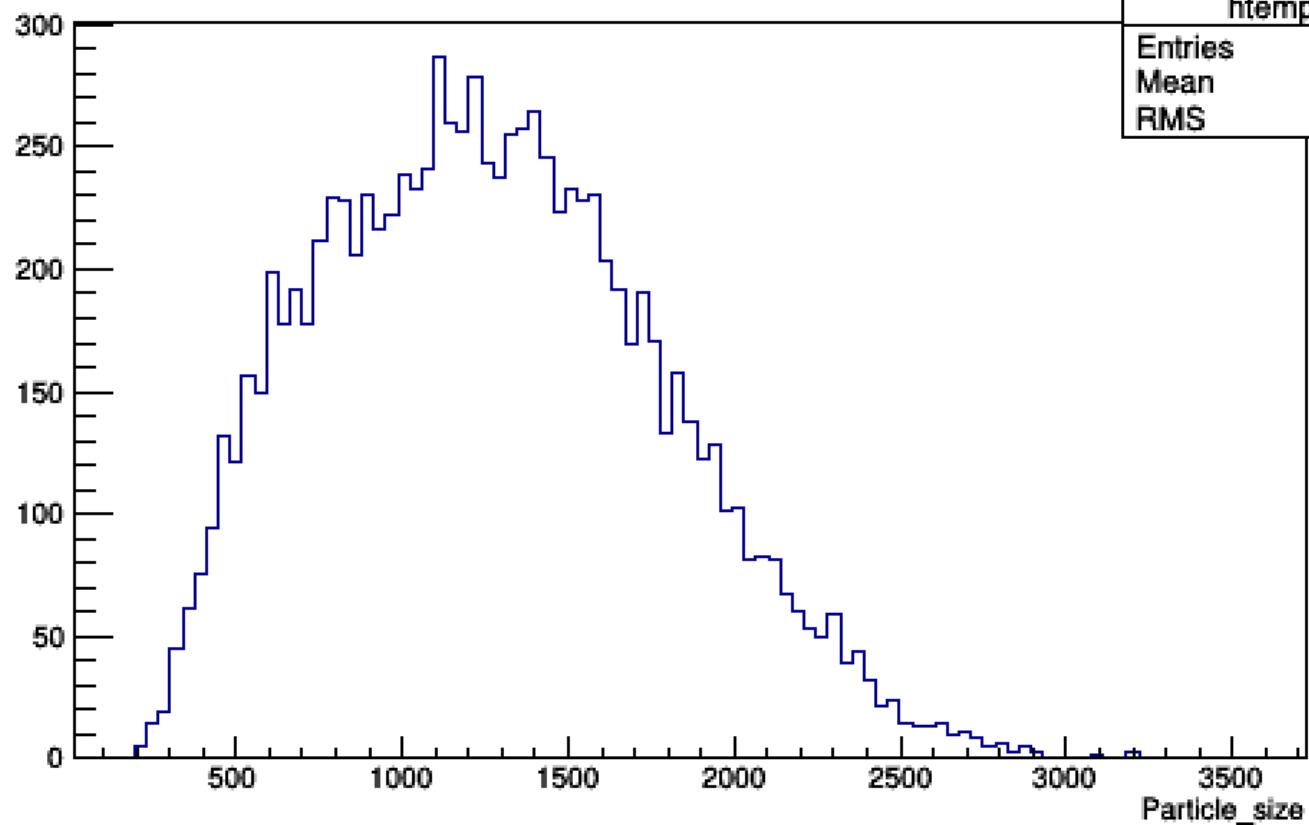


Jet.PT



Particle_size

htemp	
Entries	10000
Mean	1274
RMS	512.6



for muon

```
using namespace Pythia8;

int main() {

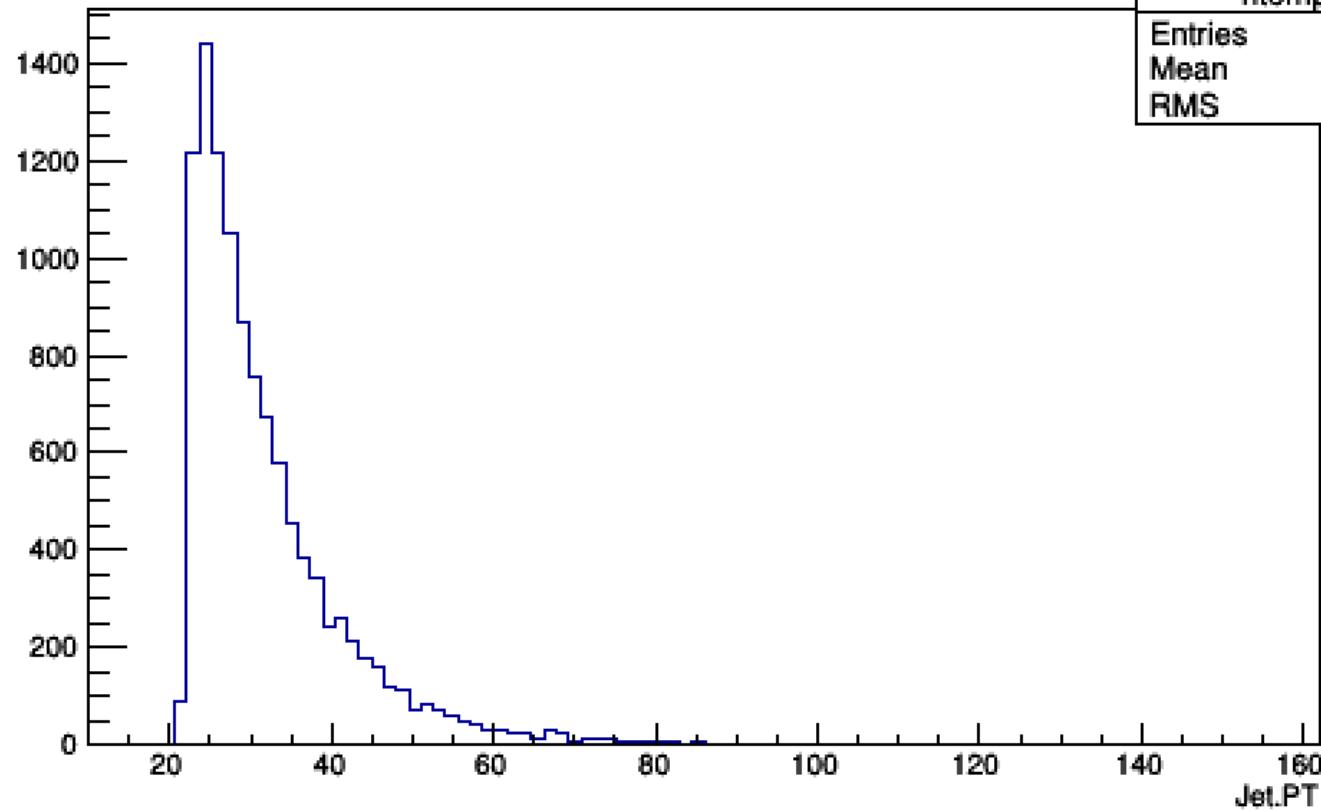
    // Interface for conversion from Pythia8::Event to HepMC event.
    HepMC::Pythia8ToHepMC ToHepMC;

    // Specify file where HepMC events will be stored.
    HepMC::IO_GenEvent ascii_io("leptonProd.hepmc", std::ios::out);

    // Generator. Process selection. LHC initialization. Histogram.
    Pythia pythia;
    pythia.readFile("main03.cmnd",0);
    pythia.readString("Random::setSeed = on");
    pythia.readString("Random::seed = 0");
    pythia.readString("Beams:eCM = 7000.");
    pythia.readString("HardQCD:all = on");
    pythia.readString("PhaseSpace:pTHatMin = 20.");
    pythia.readString("WeakBosonAndParton:qqbar2gmZg = on");
    pythia.readString("WeakBosonAndParton:qg2gmZq = on");
    pythia.readString("23:onMode = off");
    pythia.readString("23:onIfAny = 13");
    pythia.init();
    Hist mult("charged multiplicity", 100, -0.5, 799.5);
```

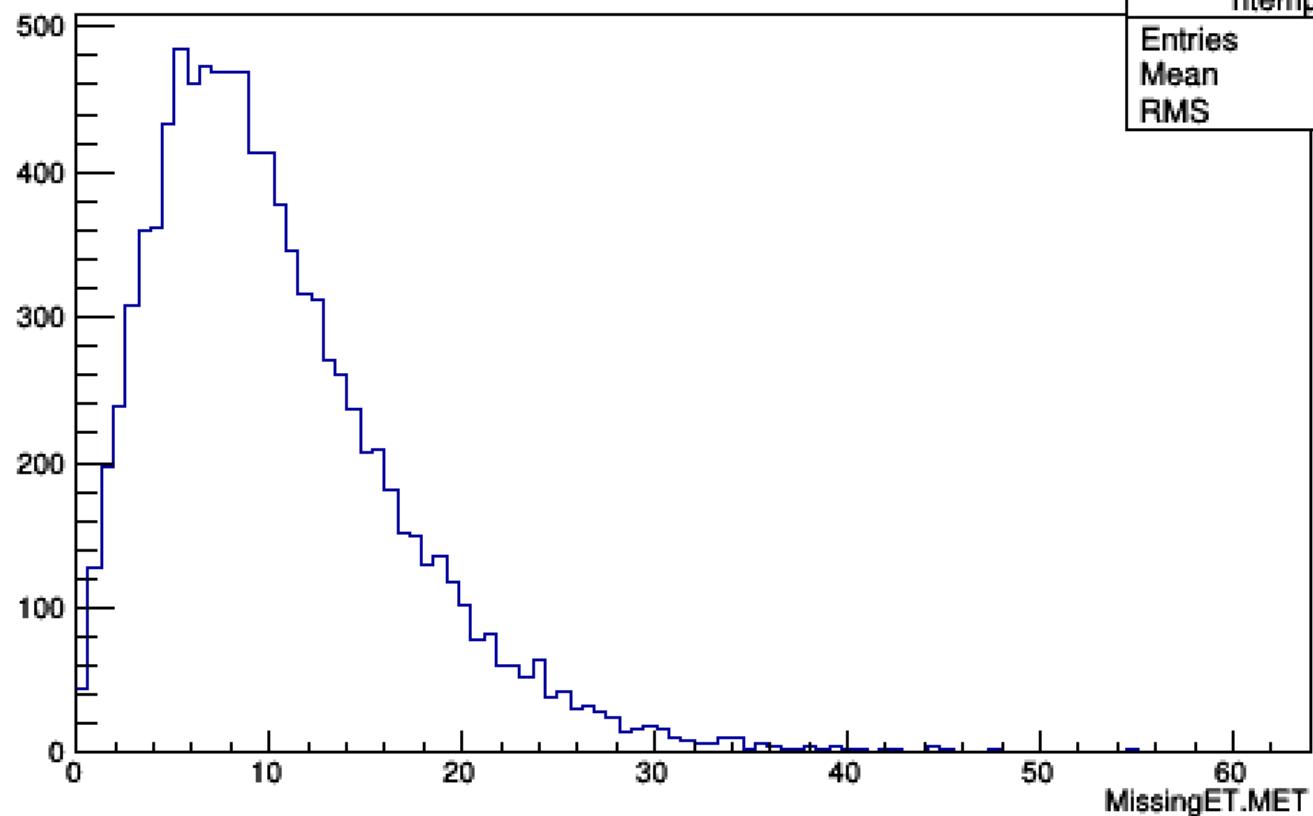
Jet.PT

htemp	
Entries	11090
Mean	32.45
RMS	11.06



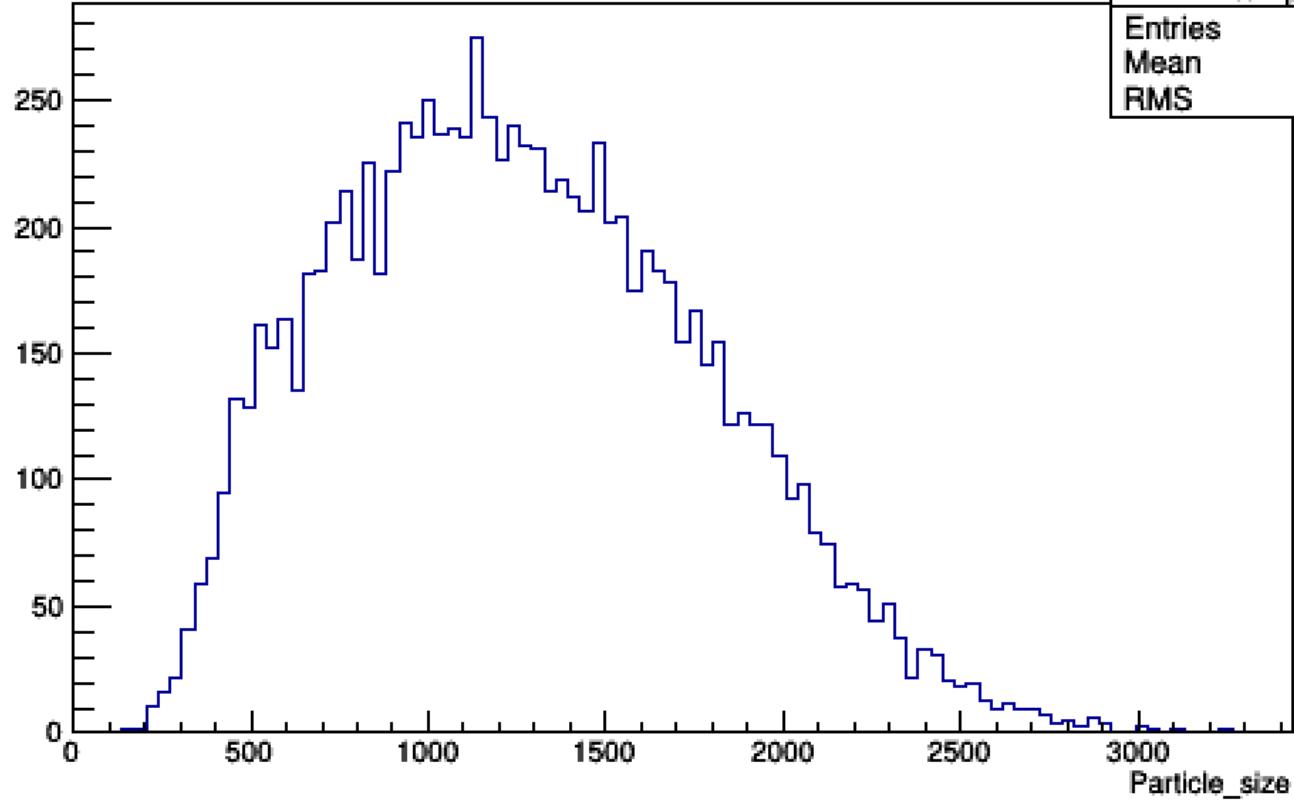
MissingET.MET

htemp	
Entries	10000
Mean	10.46
RMS	6.621



Particle_size

htemp	
Entries	10000
Mean	1267
RMS	517.9



```
#include "TCanvas.h"
#include "TStyle.h"
#include "TH1.h"
#include "TGaxis.h"
#include "TRandom.h"
#include "TTree.h"
#include "TClonesArray.h"
#include " TBranch.h"

void jet()

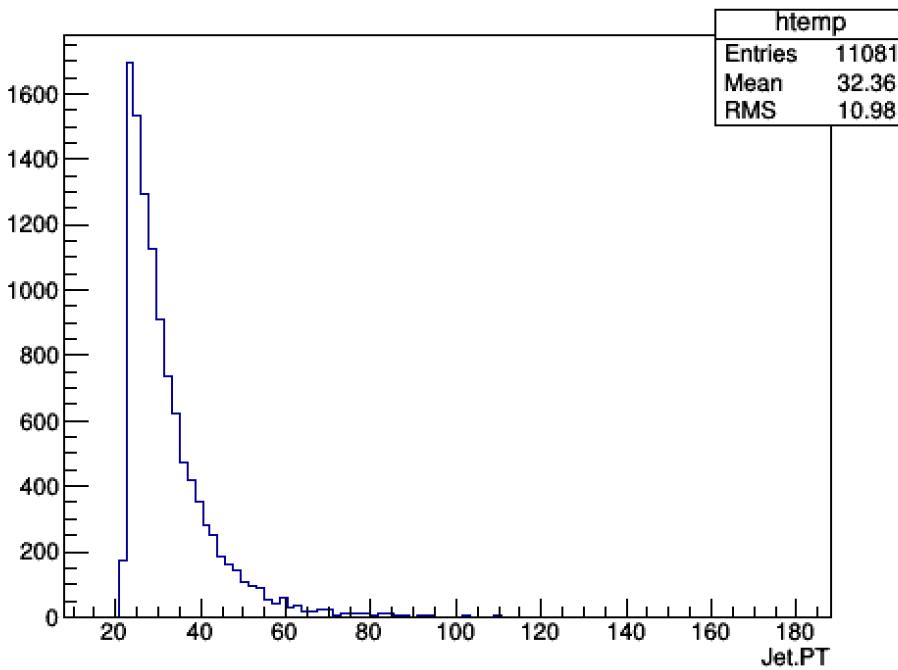
{
    TFile* f1 = new TFile("/home/cakir/Programs/Delphes-3.4.0/leptonProdout2.root");
    TFile* f2 = new TFile("/home/cakir/Programs/Delphes-3.4.0/leptonProdout1.root");
    TTree* Delphes1;
    TTree* Delphes2;
    f1->GetObject("Delphes",Delphes1);
    f2->GetObject("Delphes",Delphes2);

    TCanvas* myc = new TCanvas("c2", "my histogram", 500,800);
    myc->Divide(1,2);

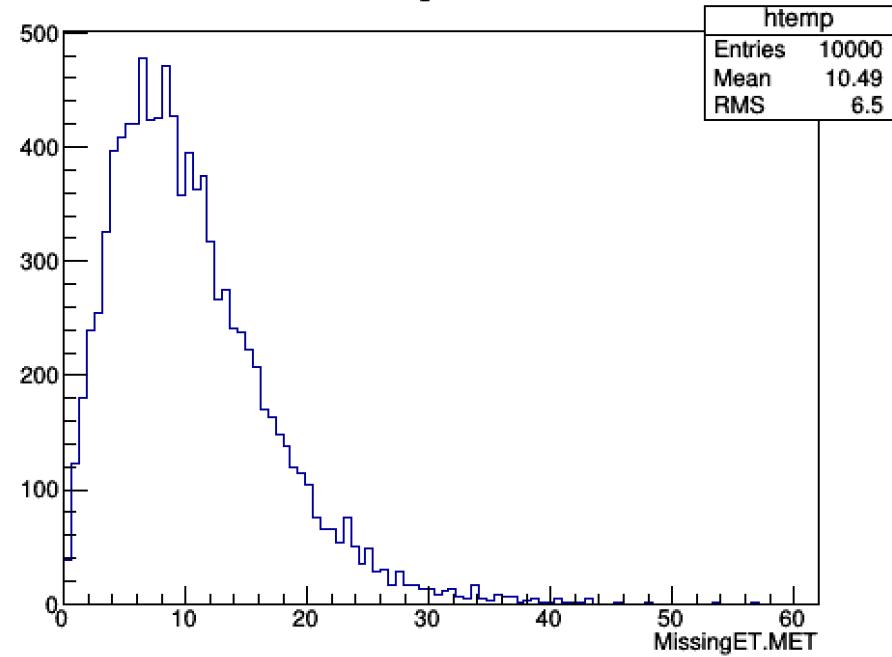
    myc->cd(1);
    Delphes1->Draw("Jet.PT");

    myc->cd(2);
    Delphes2->Draw("Jet.PT");
```

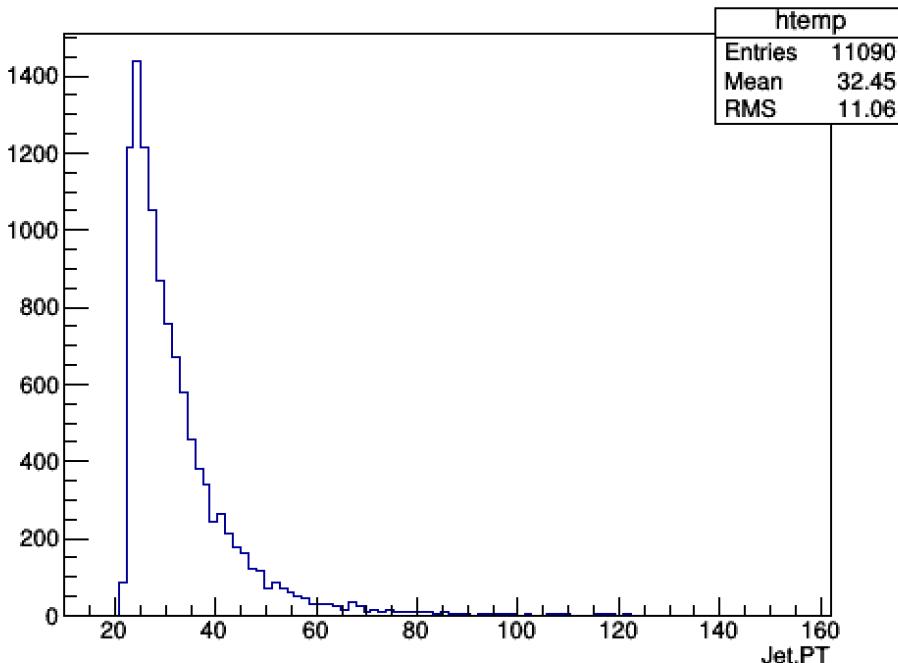
Jet.PT



MissingET.MET



Jet.PT



MissingET.MET

