PHYSICS ANALYSIS AT THE CMS EXPERIMENT

SUPERVISOR:
Mirena Paneva

SUPERSTUDENTS:
Simona Todorova
Kaloyan Botev

BASIC EQUATIONS IN CERN:

HIGGS BOSON

CERN'S PHYSICS

FITTING

HIGGS BOSON ≠ X BOSON

PHYSICS IN CERN ≈ COMPUTING

UNFITTED HISTOGRAM ≠ GOOD HISTOGRAM

THE PROJECT

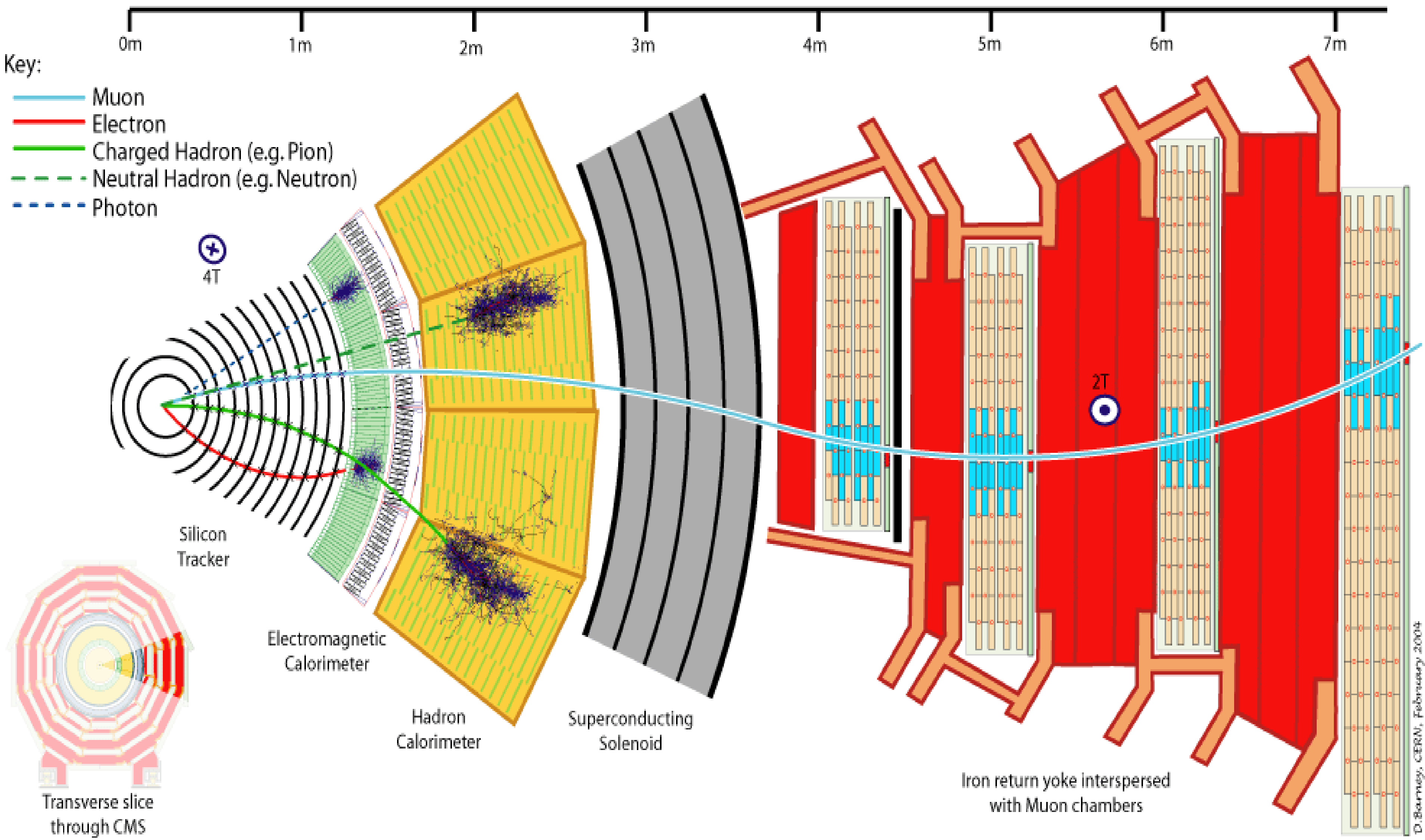
WORK:

THE BASICS OF PHYSICS ANALYSIS AT THE CMS EXPERIMENT USING SOME OF THE SOFTWARE TOOLS TO VISUALIZE AND ANALYZE DATA

TASK:

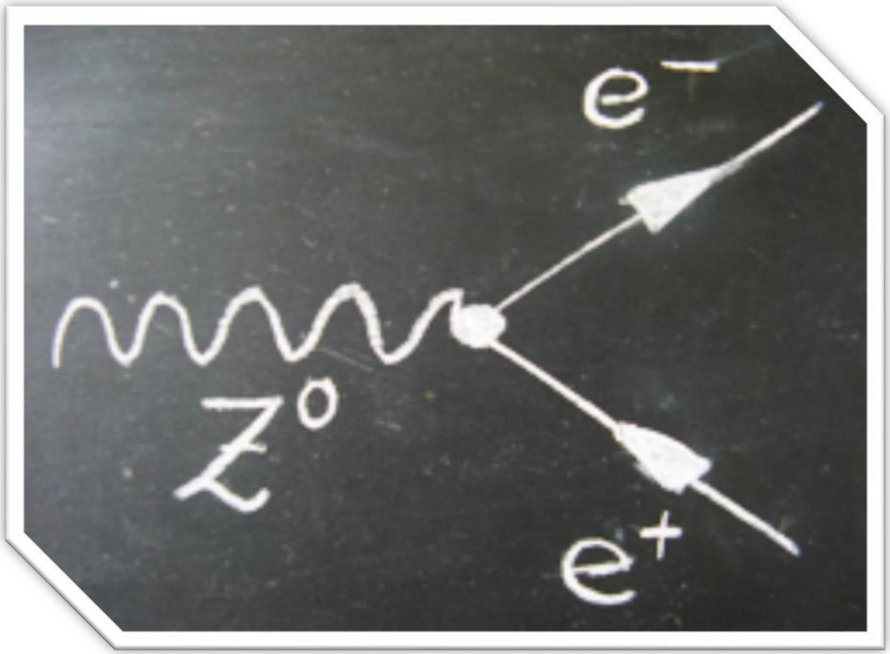
RECONSTRUCTION OF THE **Z BOSON**

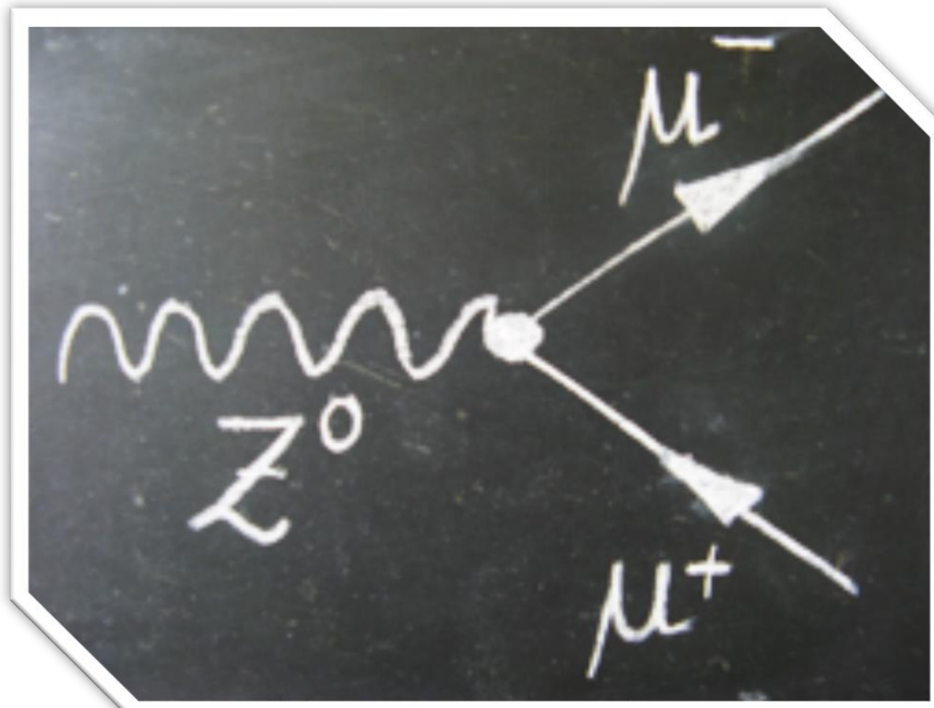
WHAT IS CMS ABOUT?



HOW TO RECONSTRUCT THE Z BOSON?

WHAT IS Z BOSON?

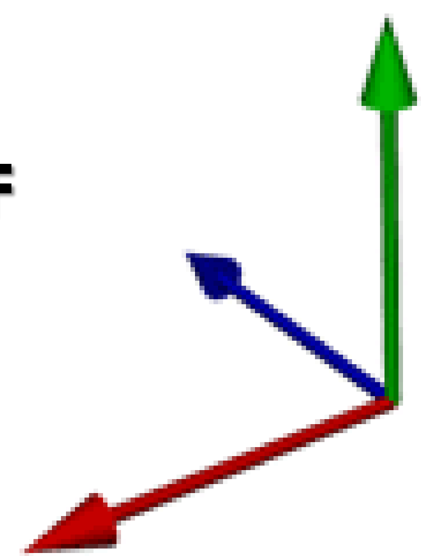




HOW TO RECONSTRUCT THE Z BOSON?

WHAT IS INVARIANT MASS?

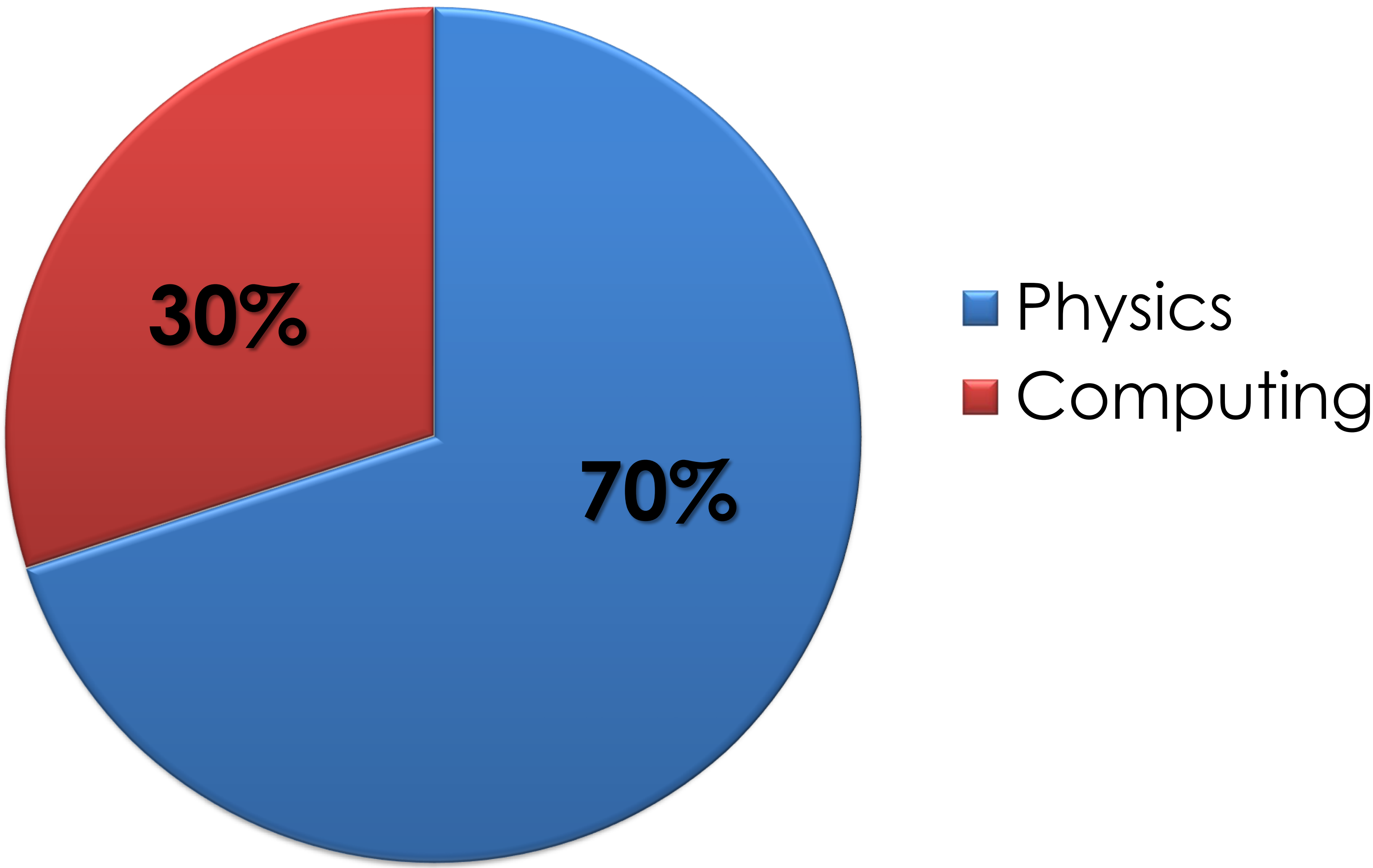
CHARACTERISTIC OF THE SYSTEM'S TOTAL **ENERGY** AND **MOMENTUM** THAT IS THE SAME **IN ALL FRAMES OF REFERENCE**



WHAT DO YOU NEED?

- LINUX
- LOGICAL THINKING
- PINCH OF PYTHON SKILLS
- 100 C++ PROGRAMING SKILLS
- CMSSW ACCESS (NEARLY IMPOSSIBLE)

AS IN CERN WE LOVE THE DIAGRAMS..



THEORETICAL EXPLANATION

SPECIAL RELATIVITY?

ENERGY-MOMENTUM RELATION:

$$E^2 = (pc)^2 + (m_o c^2)^2$$

E - ENRGY
P - MOMENTUM
Mo - INVARIANT MASS

THEORETICAL EXPLANATION

SPECIAL RELATIVITY?

$$m^2 = \frac{E^2}{c^4} - \frac{p^2}{c^2}$$

Working in natural system : c= h=1

$$m_o = \sqrt{(E' + E'')^2 - (p' + p'')^2}$$

E '– ENRGY OF FIRST PARTICLE

E"- ENERGY OF SECOND PARTICLE

P'- VECTOR OF THE MOMENTUM OF FIRST PARTICLE

P"- VECTOR OF THE MOMENTUM OF SECOND PARTICLE

THE IMPORTANCE OF COMPUTING

"THE OLD-FASHIONED WAY"

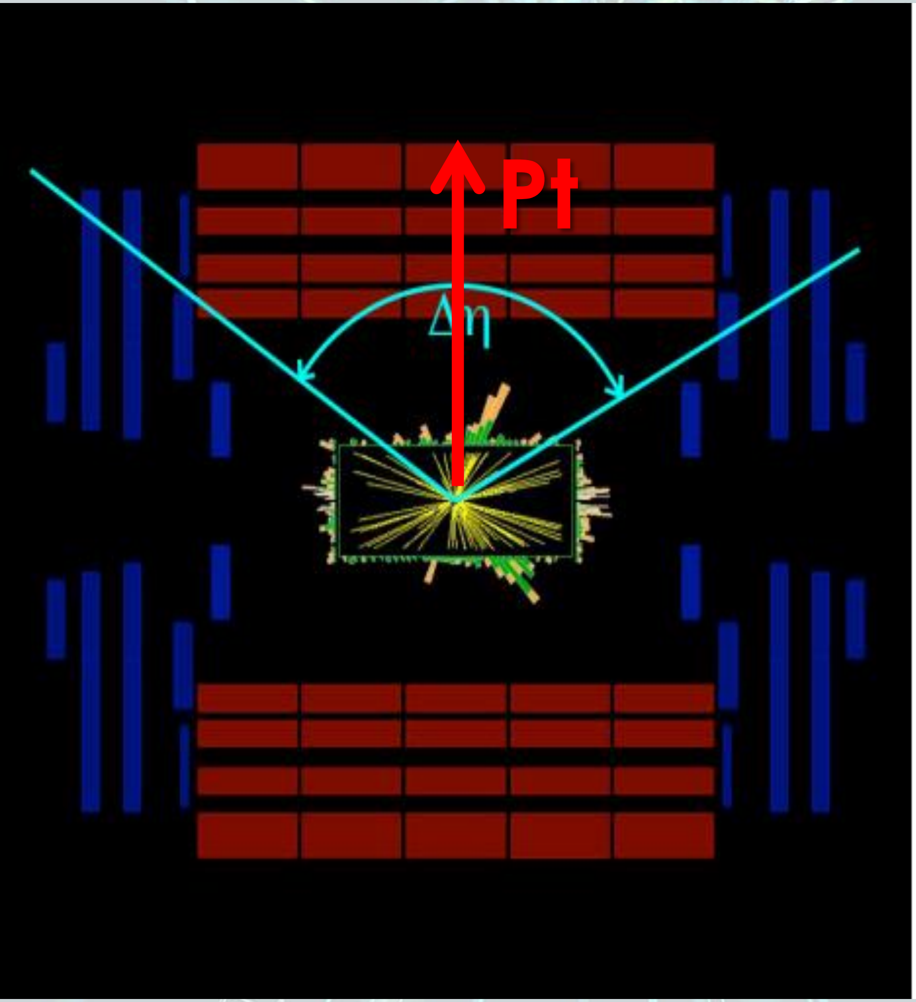
```
for (uint i = 0; i < muons->size(); i++)
 { pat::Muon mu1 = (*muons)[i];
  for (uint j = i + 1; j < muons->size(); j++)
 { pat::Muon mu2 = (*muons)[j];
   if (!(mu1.charge() + mu2.charge() == 0)) continue;
 double energy1 = sqrt (mu1.px()*mu1.px() +  mu1.py()*mu1.py() + mu1.pz()*mu1.pz() + MuMass*MuMass);
 double energy2 = sqrt (mu2.px()*mu2.px() +  mu2.py()*mu2.py() + mu2.pz()*mu2.pz() + MuMass*MuMass);
 double mx = (mu1.px() + mu2.px())*(mu1.px() + mu2.px());
 double my = (mu1.py() + mu2.py())*(mu1.py() + mu2.py());
 double mz = (mu1.pz() + mu2.pz())*(mu1.pz() + mu2.pz());
 double InvM = sqrt (((energy1 + energy2)*(energy1+energy2)) - (mx+my+mz));
```

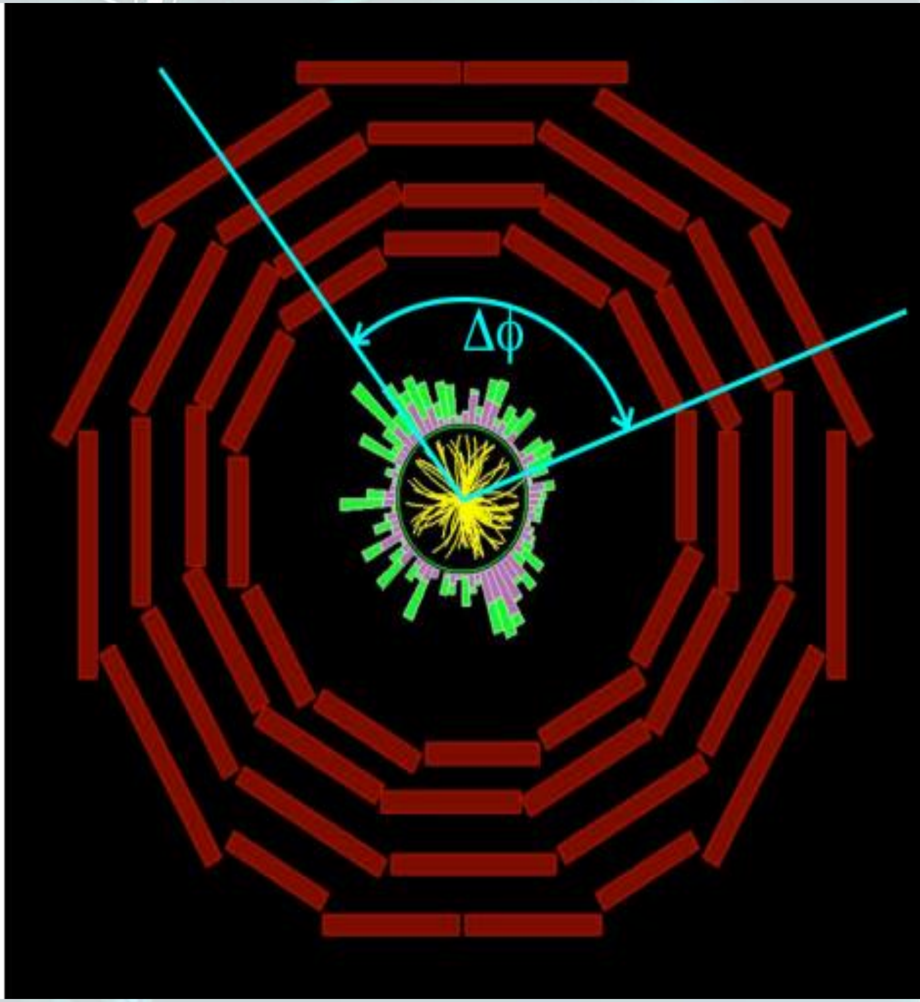
TLORENTZVECTOR- THE LOVED ONE

```
for (uint i = 0; i < muons->size(); i++)
 { pat::Muon mu1 = (*muons)[i];
  for (uint j = i + 1; j < muons->size(); j++)
 { pat::Muon mu2 = (*muons)[j];
  if (!(mu1.charge() + mu2.charge() == 0)) continue;
 double energy1 = sqrt (mu1.px()*mu1.px() +  mu1.py()*mu1.py() + mu1.pz()*mu1.pz() + MuMass*MuMass);
 double energy2 = sqrt (mu2.px()*mu2.px() +  mu2.py()*mu2.py() + mu2.pz()*mu2.pz() + MuMass*MuMass);
 double mx = (mu1.px() + mu2.px())*(mu1.px() + mu2.px());
 double my = (mu1.py() + mu2.py())*(mu1.py() + mu2.py());
 double mz = (mu1.pz() + mu2.pz())*(mu1.pz() + mu2.pz());
 double InvM = sqrt (((energy1 + energy2)*(energy1+energy2)) - (mx+my+mz));
```

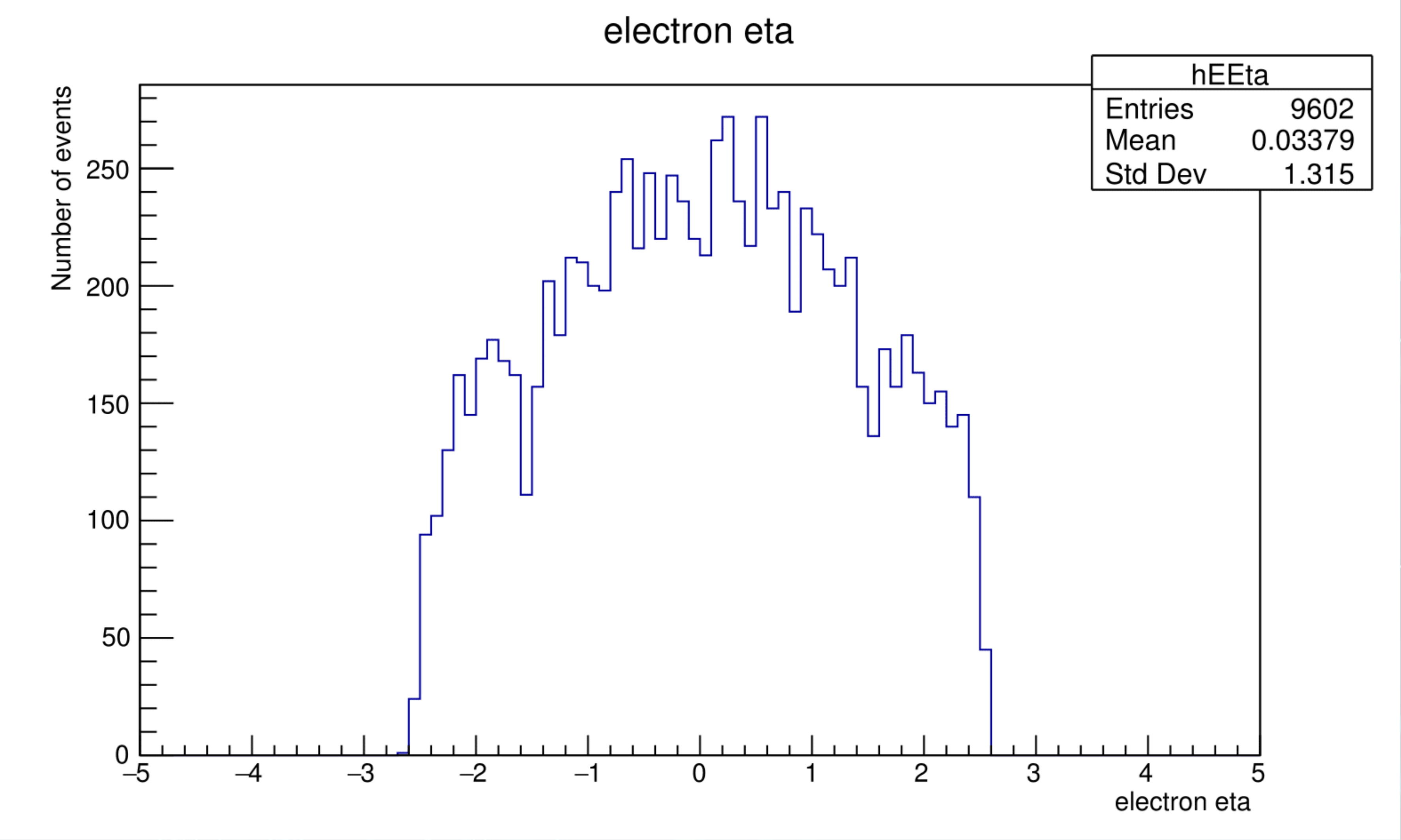
```
for (uint i = 0; i < muons->size(); i++)
 { pat::Muon mu1 = (*muons)[i];
  for (uint j = i + 1; j < muons->size(); j++)
 { pat::Muon mu2 = (*muons)[j];
  if (!(mu1.charge() + mu2.charge() == 0)) continue;
 if ( mu1.pt()< 3 || mu2.pt()<3 ) continue;
  TLorentzVector lv1;
  TLorentzVector lv2;
 double energy1 = sqrt (mu1.px()*mu1.px() +  mu1.py()*mu1.py() + mu1.pz()*mu1.pz() + MuMass*MuMass);
 double energy2 = sqrt (mu2.px()*mu2.px() +  mu2.py()*mu2.py() + mu2.pz()*mu2.pz() + MuMass*MuMass);
 lv1.SetPxPyPzE(mu1.px(), mu1.py(), mu1.pz(), energy1);
 lv2.SetPxPyPzE(mu2.px(), mu2.py(), mu2.pz(), energy2);
```

HISTOGRAMS AND PARAMETERS

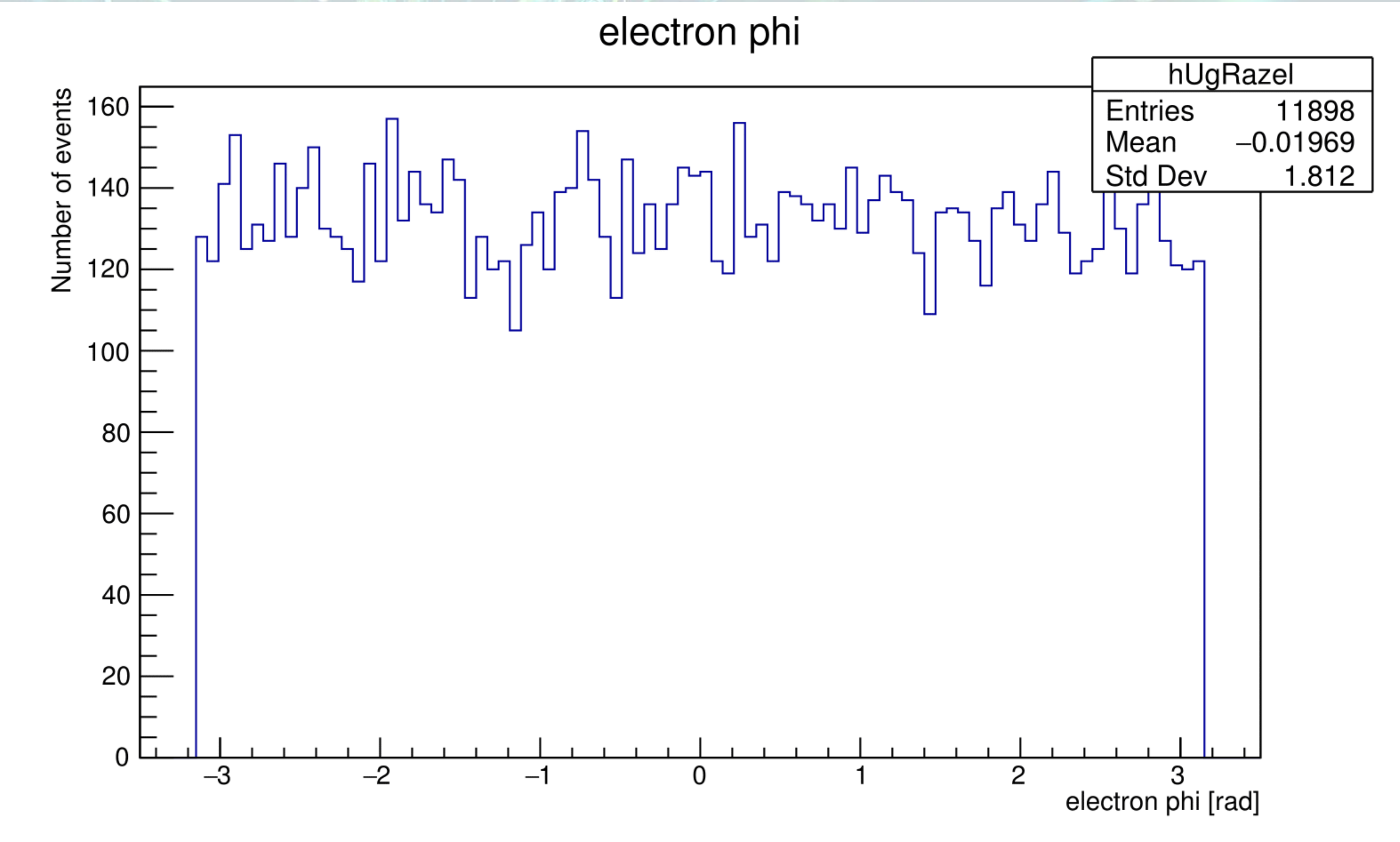




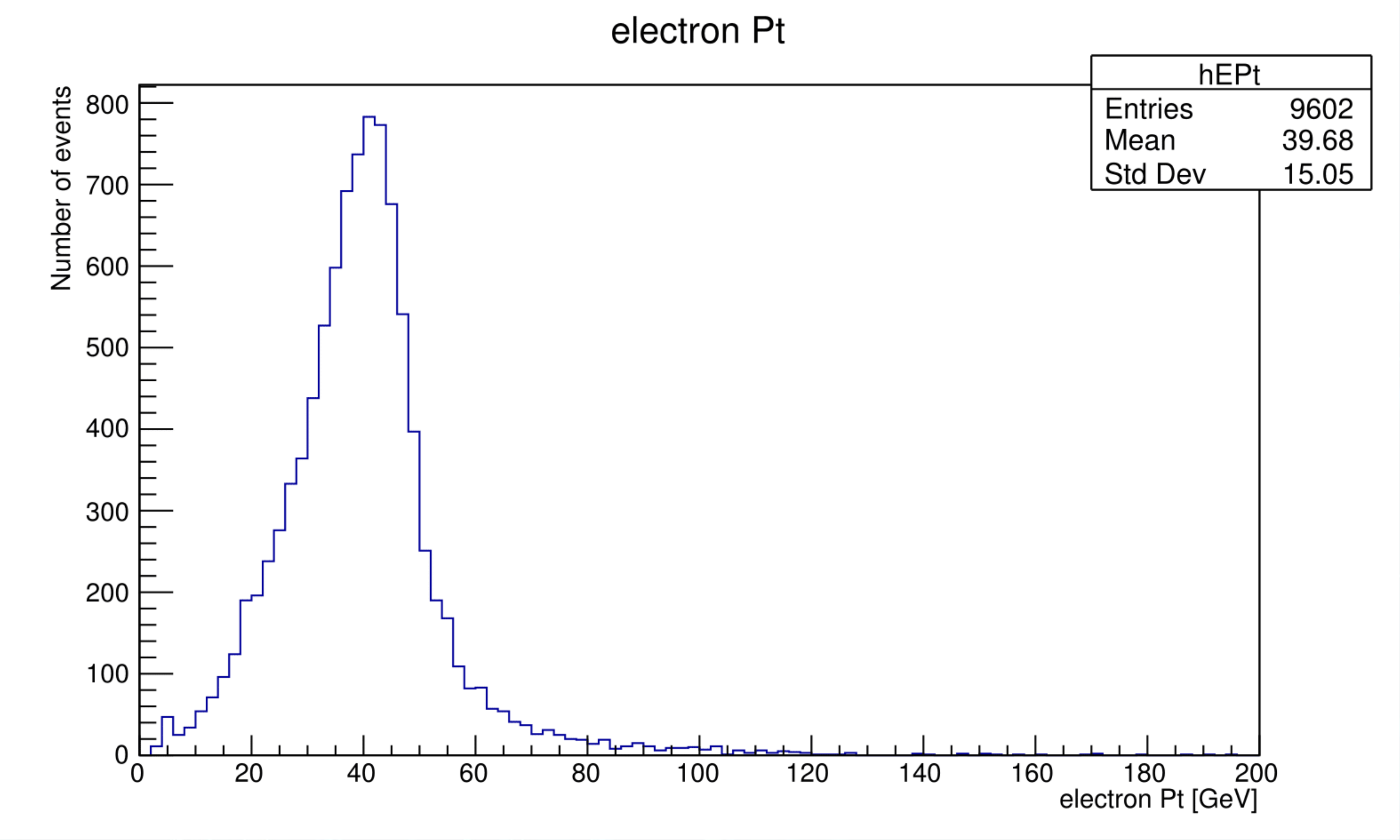
HISTOGRAMS AND PARAMETERS



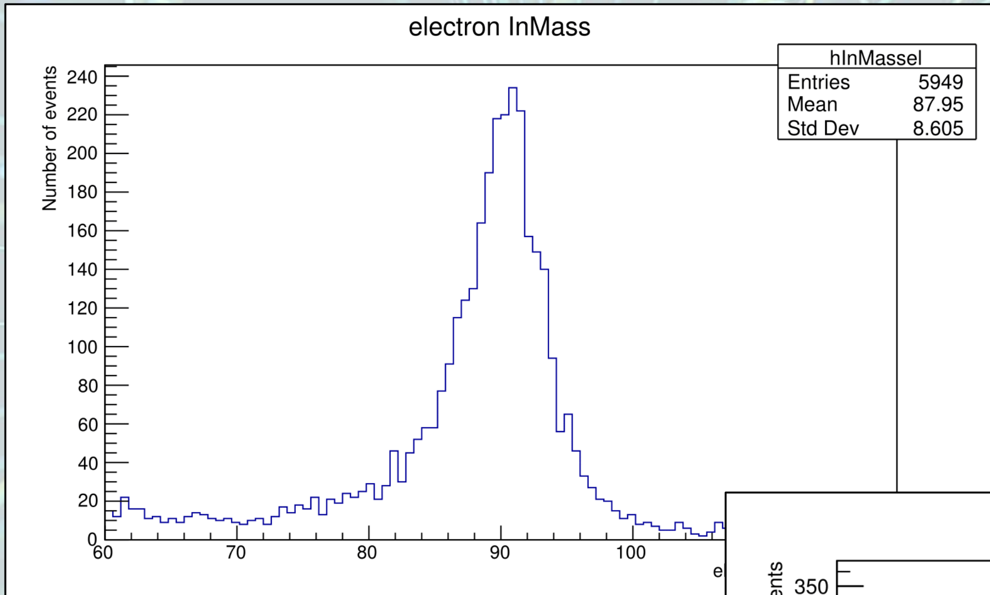
HISTOGRAMS AND PARAMETERS

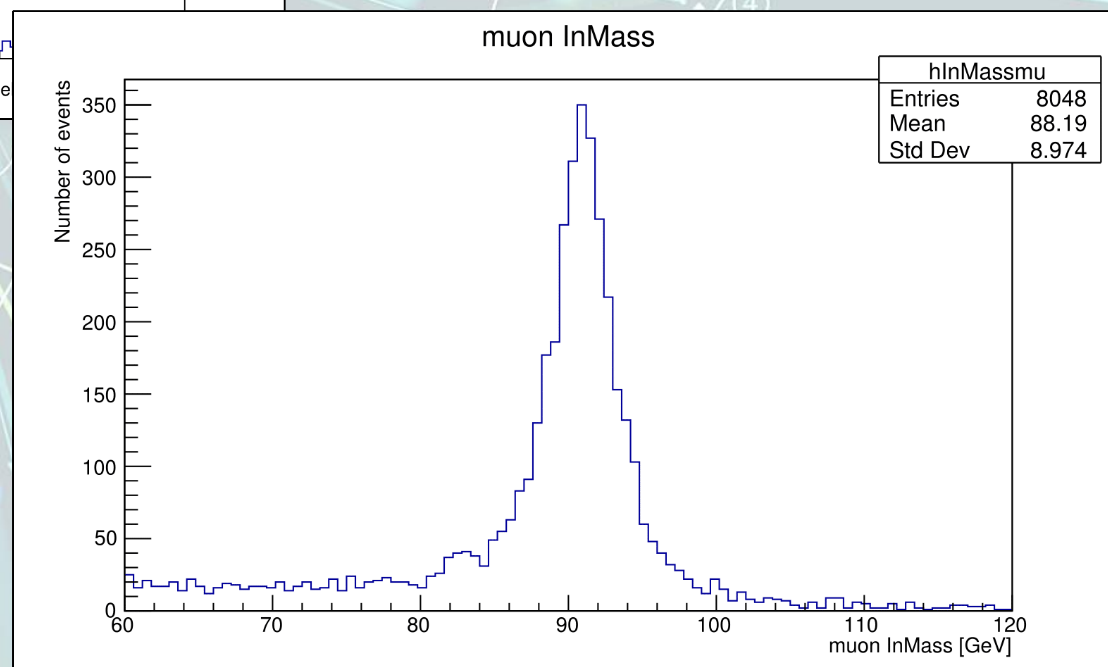


HISTOGRAMS AND PARAMETERS



HISTOGRAMS AND PARAMETERS



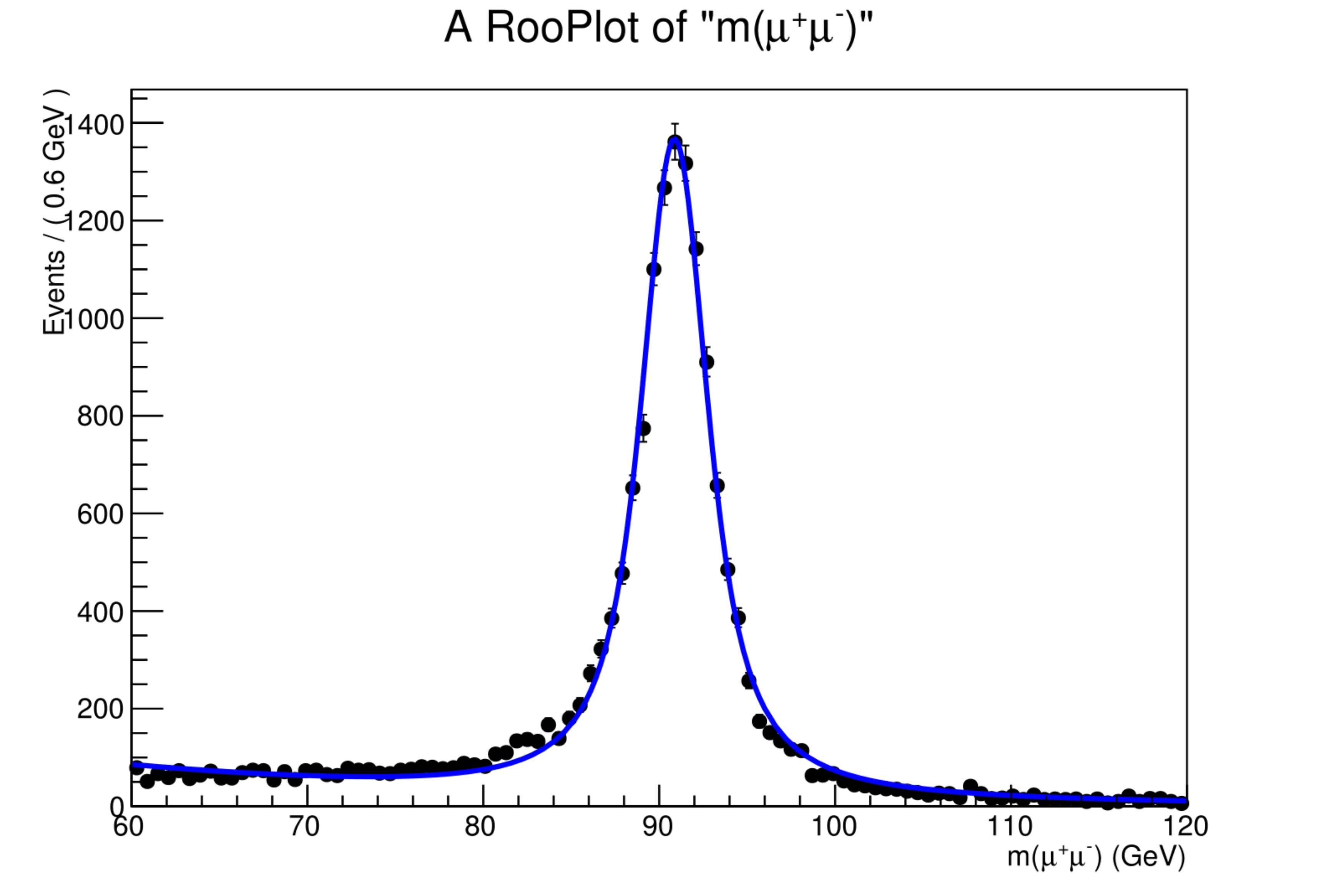


FITTING HISTOGRAMS

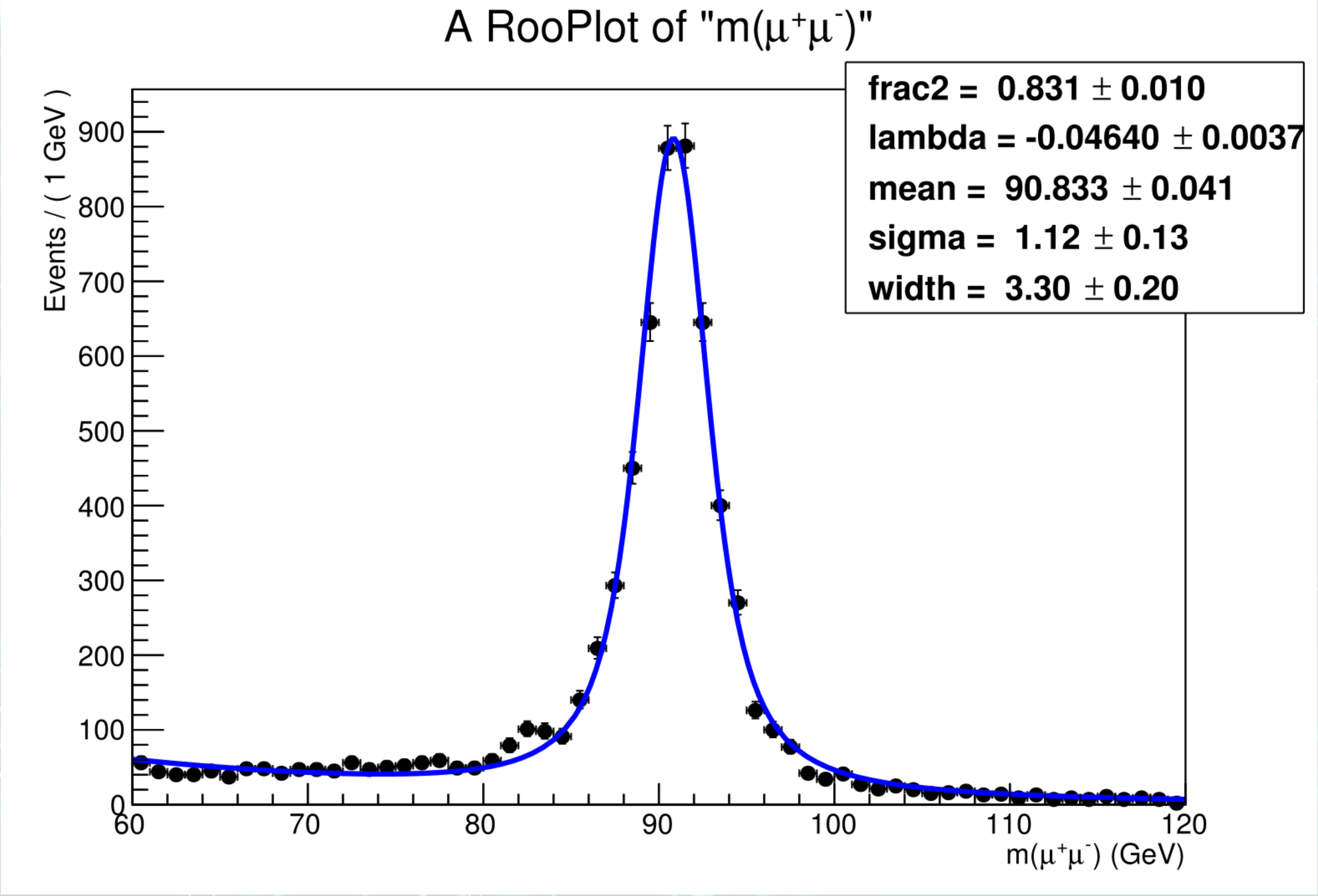
WHY BOTH GAUSS AND BREIT-WIGNER?

```
// Mean of the J/psi mass peak
RooRealVar mean("mean","mean", 91, hmin, hmax);

// Construct Gaussian1 PDF for signal
RooRealVar sigma1("sigma1","sigma1", 1, 0.01, 7);
RooGaussian gauss1("gauss1","gauss1", x, mean, sigma1);

// Construct Gaussian2 PDF for signal
RooRealVar sigma2("sigma2","sigma2", 0.02, 0.001, 7);
RooGaussian gauss2("gauss2","gaus2s", x, mean, sigma2);

// Construct a double Gaussian function to fit the signal component
RooRealVar frac1("frac1","fraction1", 0.8, 0, 1); // fraction of gauss1 to gauss2 componenct
RooAddPdf signalModel("signal model","gauss1+gauss2",RooArgList(gauss1,gauss2),RooArgList(frac1));

// Construct a Voigtian to fir the signal component
RooRealVar width("width","width", 0.2, 0.001,7);
RooRealVar sigma("sigma","sigma", 0.02, 0.001, 7);
RooVoigtian voigt("voigt","voigt",x,mean,width,sigma);

// Construct exponential PDF to fit the bkg component
RooRealVar lambda("lambda", "slope", -2, -5, 5);
RooExponential expo("expo", "exponential PDF", x, lambda);

// Construct signal + bkg PDF
RooRealVar frac2("frac2","fraction2", 0.8, 0, 1); // fraction of the signal to bkg component
```

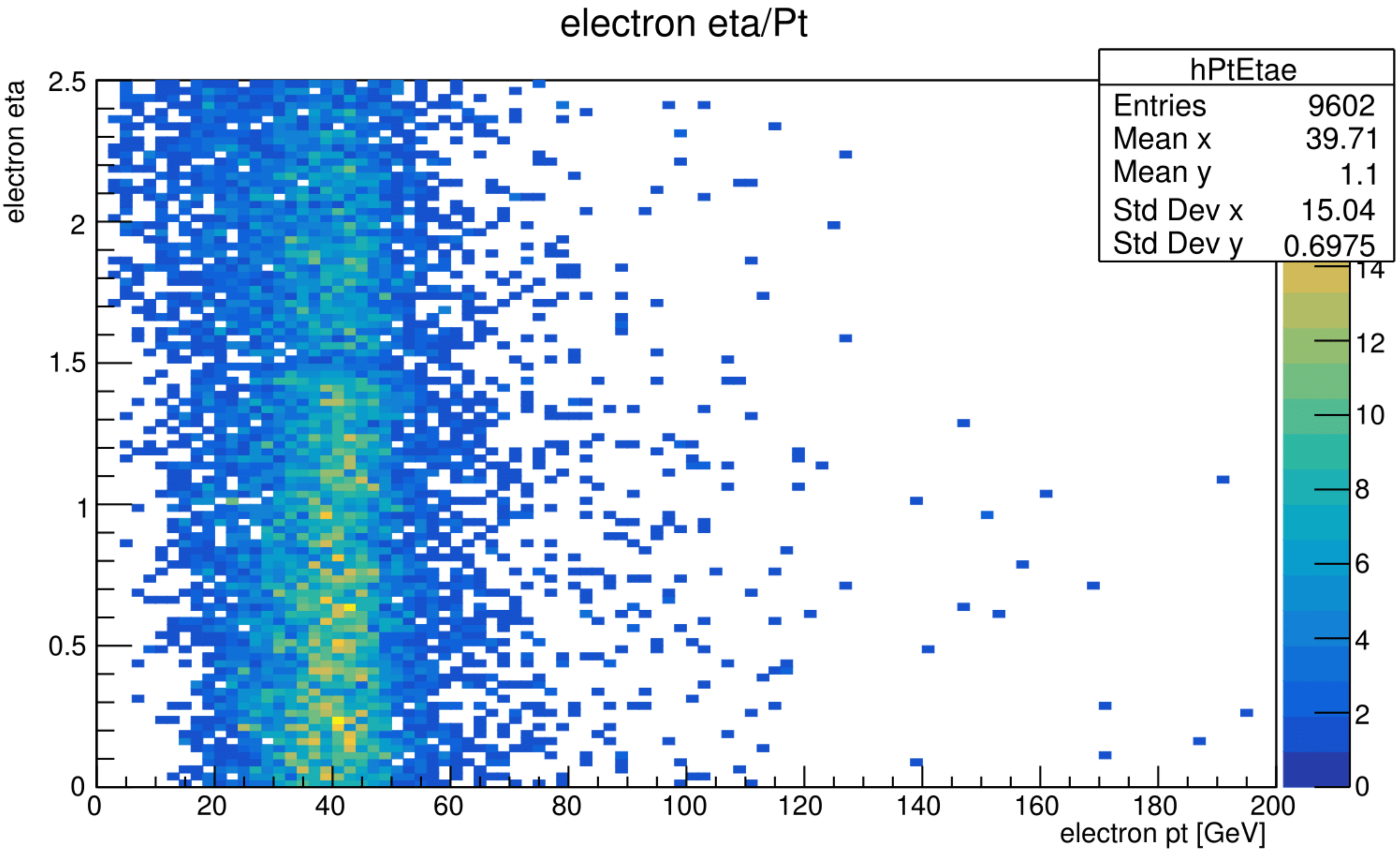
FITTING HISTOGRAMS



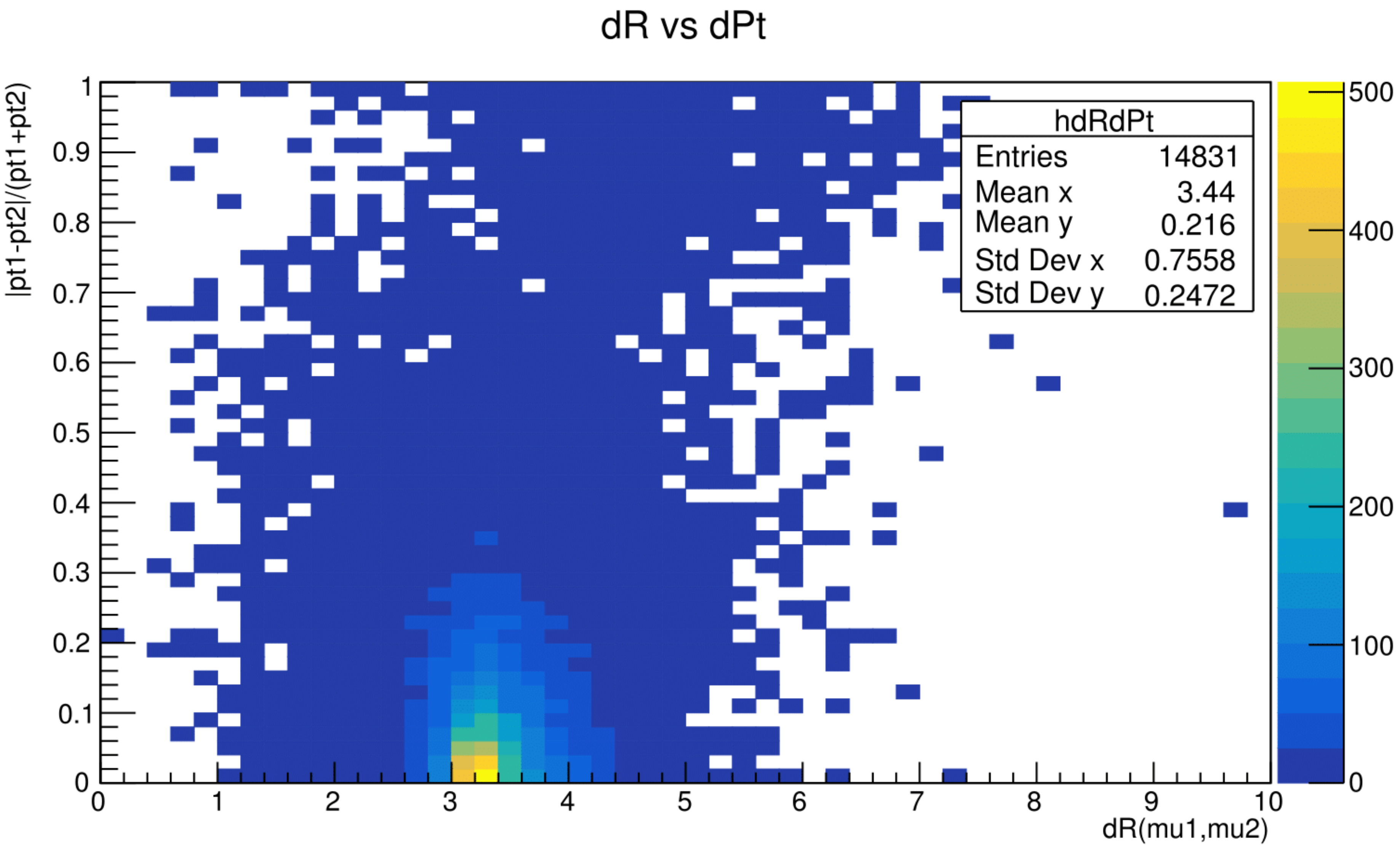
FITTING HISTOGRAMS



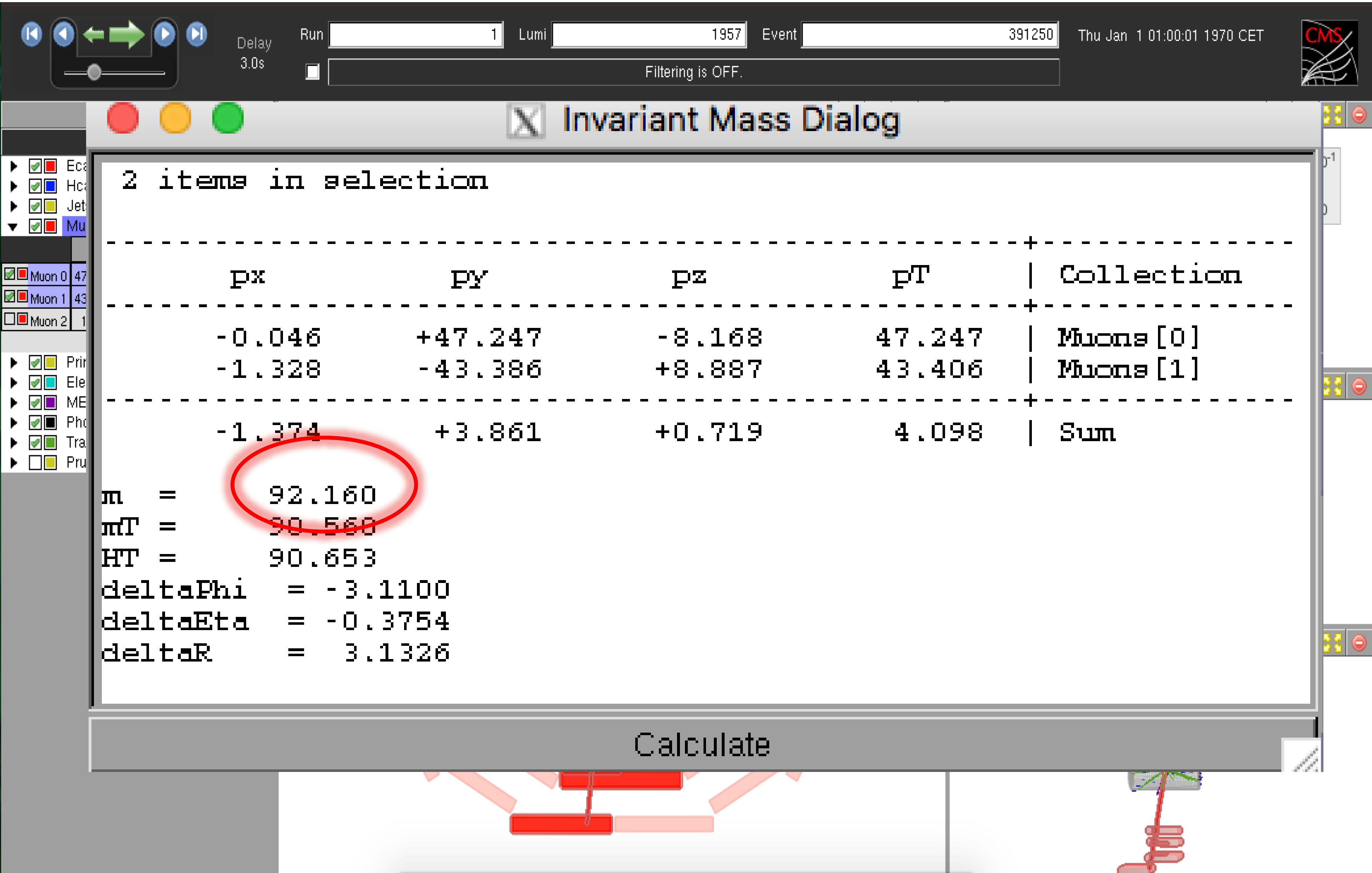
2D HISTOGRAMS



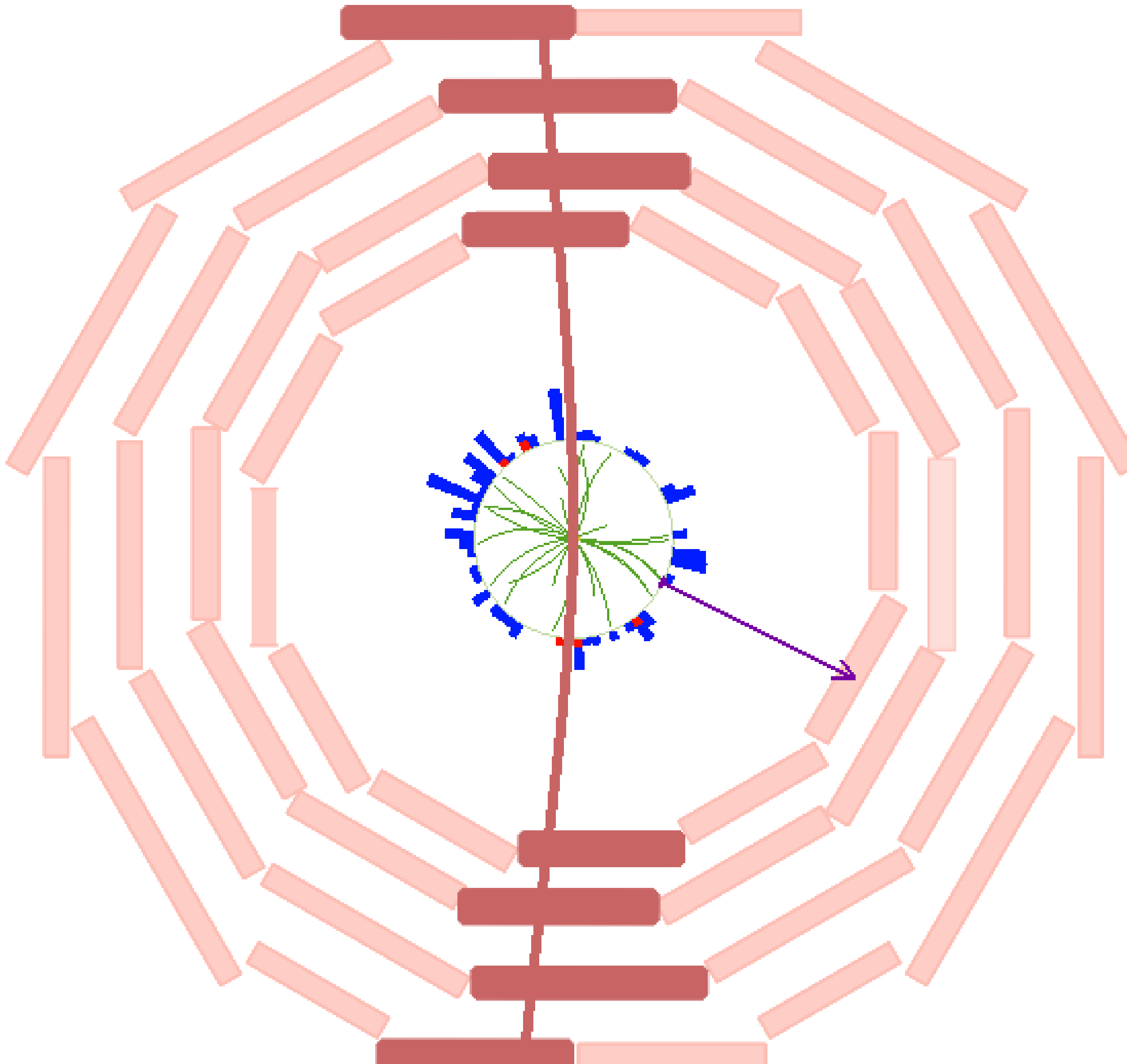
2D HISTOGRAMS



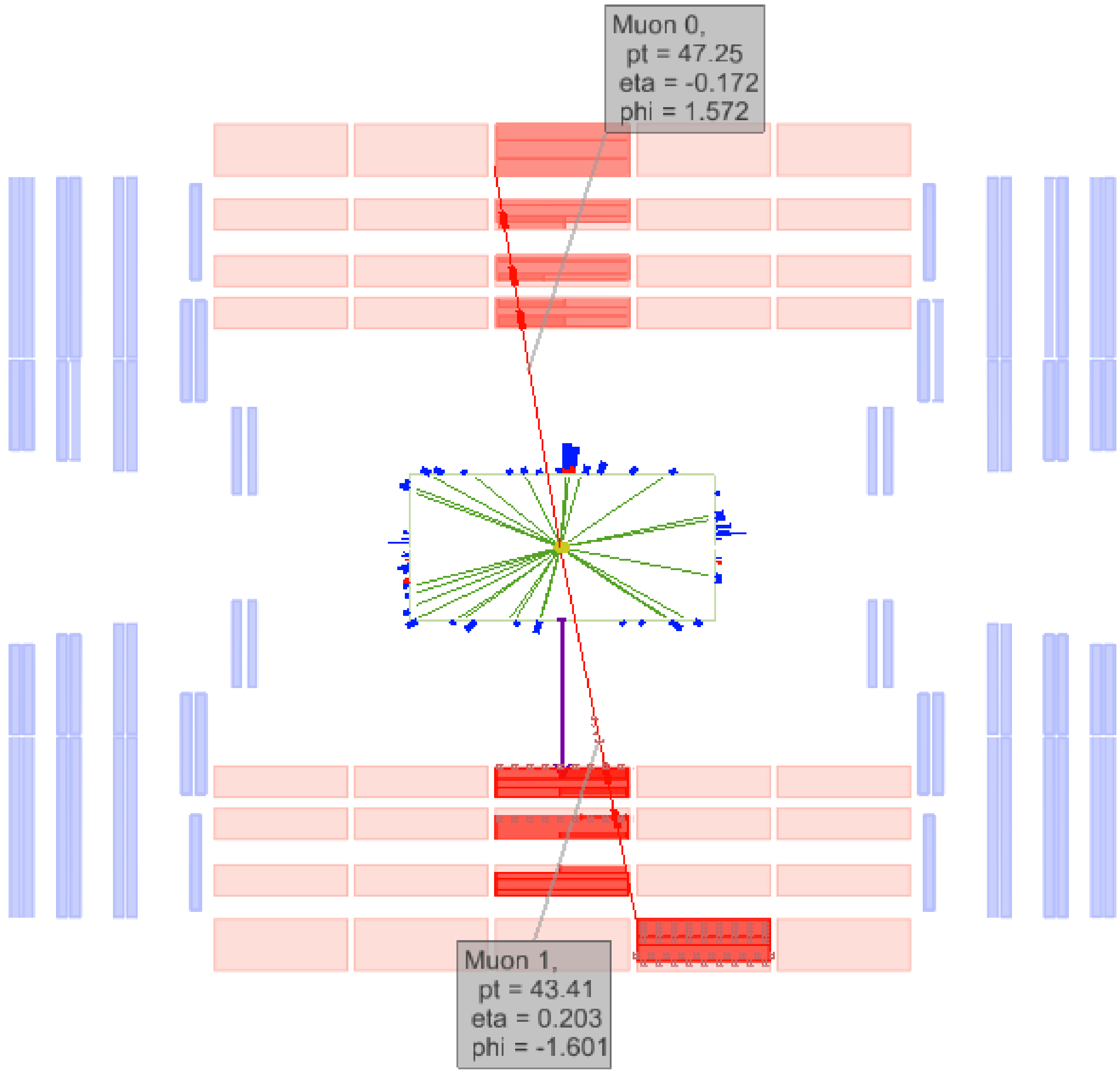
DIGITAL VISUALISATION



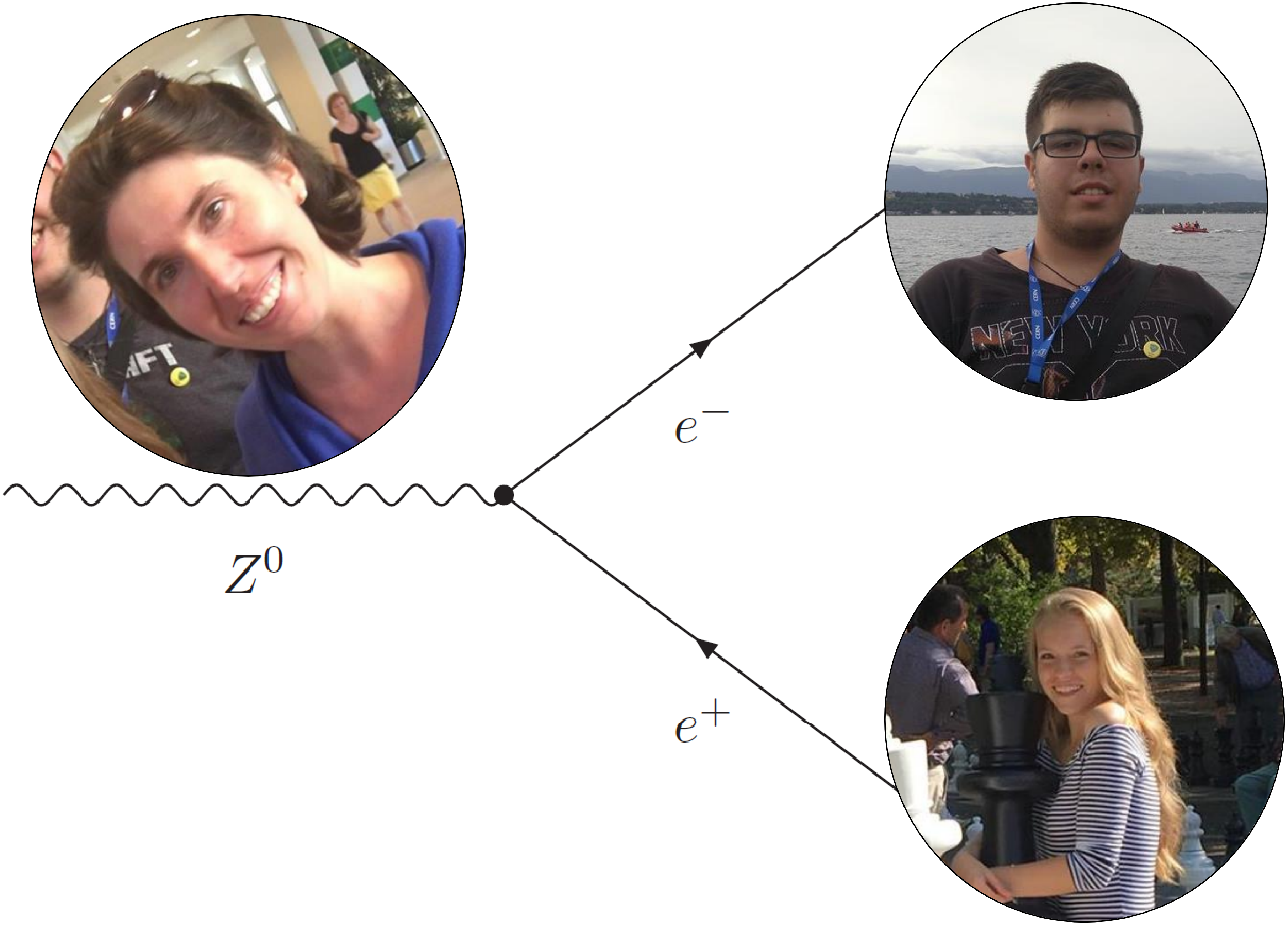
DIGITAL VISUALISATION



DIGITAL VISUALISATION



WHAT IS THE Z BOSON DECAY?




THANK YOU!