

# Short exercise - part 1

S. Liechti, S. Leontsinis  
University of Zurich



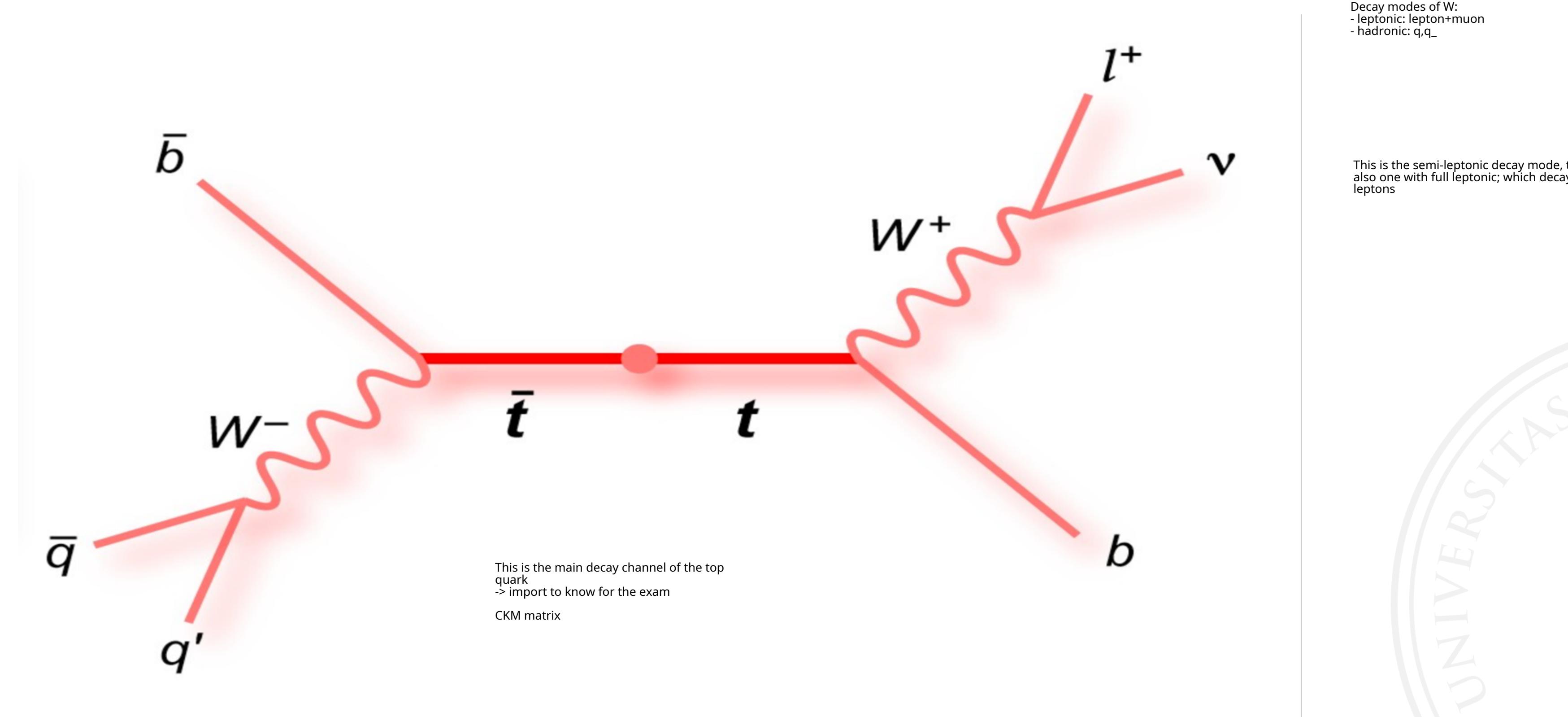
PHY451 Particle Physics I  
18<sup>th</sup> April 2023



# Short exercise - part 1

## Introduction

- The short exercise consists in measuring the production cross section of top quark pairs performing a physics analysis of 50 pb-1 of data collected by the CMS experiment at the proton collider LHC
- The top quark mean lifetime is about  $5 \times 10^{-25}$ s
  - too short to enable direct top quark reconstruction
  - the alternative is to measure final state decay products of top quark from energy deposits in the detector and indirectly reconstruct the top quark



## Short exercise - part 1

### Hunt for tt production

- For this exercise only semileptonic decay of top-quark pair will be considered (in the muon channel)
  - expected signal final state: 4 jets out of which 2 from b quarks, 1 lepton and missing momentum from non-reconstructed neutrino
- The hunt for  $t\bar{t}$  production (signal) consists in correctly identifying the final state products to distinguish the reconstructed final state from other processes which have similar signature (called background processes)
  - examples: QCD jet production, electroweak W or Z boson production in association to jets
  - moreover, additional objects such as jets from radiation or from overlapping collisions can be produced complicating the top reconstruction
- Physics analysis of experimental data workflow
  1. Study of **kinematical properties** distinguishing signal events against background using Monte Carlo simulation (or data driven)
  2. Define a **selection strategy** based on these studies to suppress background and select signal-like data
  3. **Apply this selection** on experimental data
  4. Measure! In this case, **estimate  $t\bar{t}$  production cross section**

# Short exercise - part 1

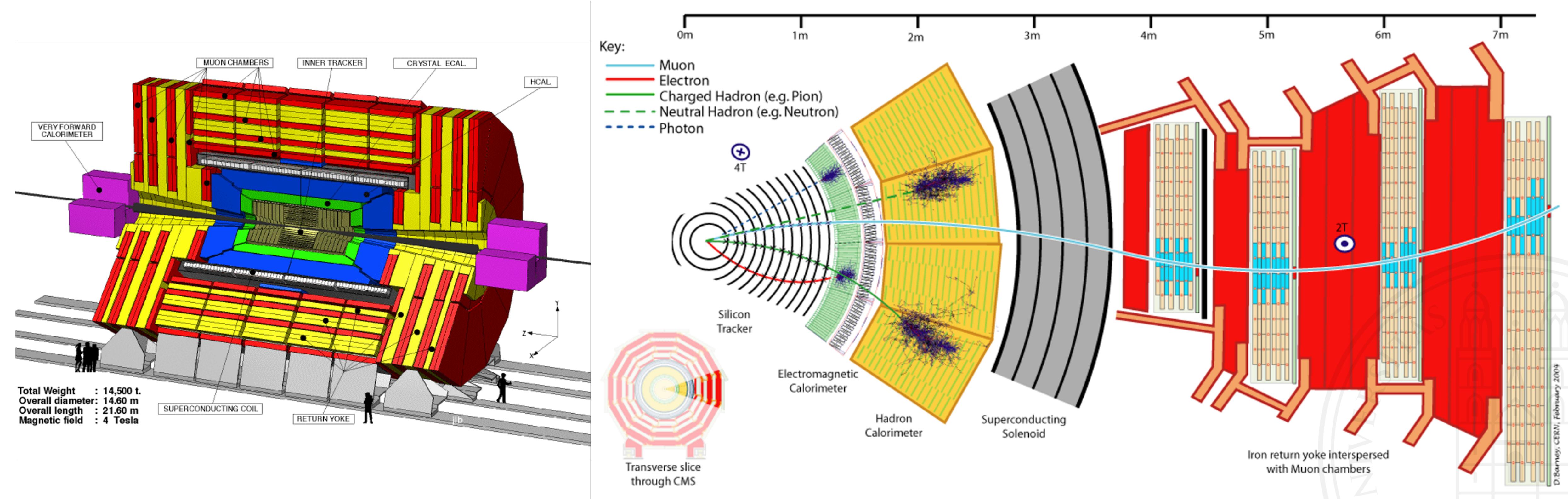
## The CMS experiment

- The Compact Muon Solenoid, CMS

- total weight: 14.5 t
- overall diameter: 14.8 m
- overall length: 21.6 m
- internal magnetic field: 3.8 T
- external magnetic field: 1.8 T

- CMS main features

- superconducting solenoidal magnet
- High-quality tracker system:
- high-performance electromagnetic calorimeter
- superconducting coil
- return Yoke
- highly hermetic hadron calorimeter



# Short exercise - part 1

## The LHC

- The Large Hadron Collider is built at CERN (Geneva), on the Swiss-French border
  - accelerate bunch of protons through a circumference of 27 km up to 13 TeV c.m.e
  - this exercise makes use of 50 pb-1 of p–p collision data delivered by LHC at a  $\sqrt{s} = 7 \text{ TeV}$



# Short exercise - part 1

## Files

- Data and MC simulation samples together with a basic analysis framework are available at the following path:
  - `/home/hep/stereo/PHY451/exercise`
- The data samples for this exercise are stored in root trees, you can find them in the folder files
  - file names indicate which kind of simulated data are stored in the file (`ttbar.root`, `qcd.root`, etc.) or whether it contains real data (`data.root`)
  - `data.root` is a sample of real data collected by the CMS experiment in 2011
    - corresponding to  $50 \text{ pb}^{-1}$  of proton-proton collisions at  $\sqrt{s} = 7 \text{ TeV}$
  - given the very large collision rate, not all produced data are stored
    - a selection is performed by the trigger system to store only “interesting” data
    - only events containing a reconstructed, isolated muon are saved
  - data are stored in files as ROOT trees
    - each tree contains a selection of variables useful for the analysis
    - a list of all variables stored in the trees follows
- The analysis framework will serve as base for your own analysis

Calculate trigger efficiency:  
(selection of events with applied trigger efficiency)/(selection of events you want)

If we can calculate it theoretically:

- it is acceptance

if not, it is usually efficiency

$$N = L * \rho$$

with  $\rho$  pure mathematical in barn  
 $L$  in  $b^{-1}$

Does not hold if we are limited by efficiency (e.g. in the tracker) and acceptance (i.e. angle not covered by detector):

$$N = L * \rho * a * e$$

Here we have the Branching ratios:  
 $B(pp \rightarrow tt)$

Example: The muonpT is a distribution, and we need to reject e.g. low-pt muons (cannot see this). This lowers the cross-section (-> fiducial cross-section), and we need to increase the acceptance. We also need to fold in the efficiency (e.g. if the detector sees two particles as one).

# Short exercise - part 1

## Structure of a ntuple

Event	Variables	Content (size of each vector indicates the number of candidates per event)
1	$p_x$ $p_y$ $p_z$	-138.0434 126.42235 30.451496 -19.39185 -40.10309 32.827720 31.058544 -23.21485 9.2168264 48.834972 -74.05298 37.526317
2	$p_x$ $p_y$ $p_z$	1.4830336 35.448909 159.64926
4	$p_x$ $p_y$ $p_z$	53.272769 -46.61022 -34.74116 18.753765 -363.9992 49.854187
....	$p_x$ $p_y$ $p_z$	....

This indicates e.g. 2 muons; if we have each two values of every coordinate of momentum

# Short exercise - part 1

## Tree structure - jets

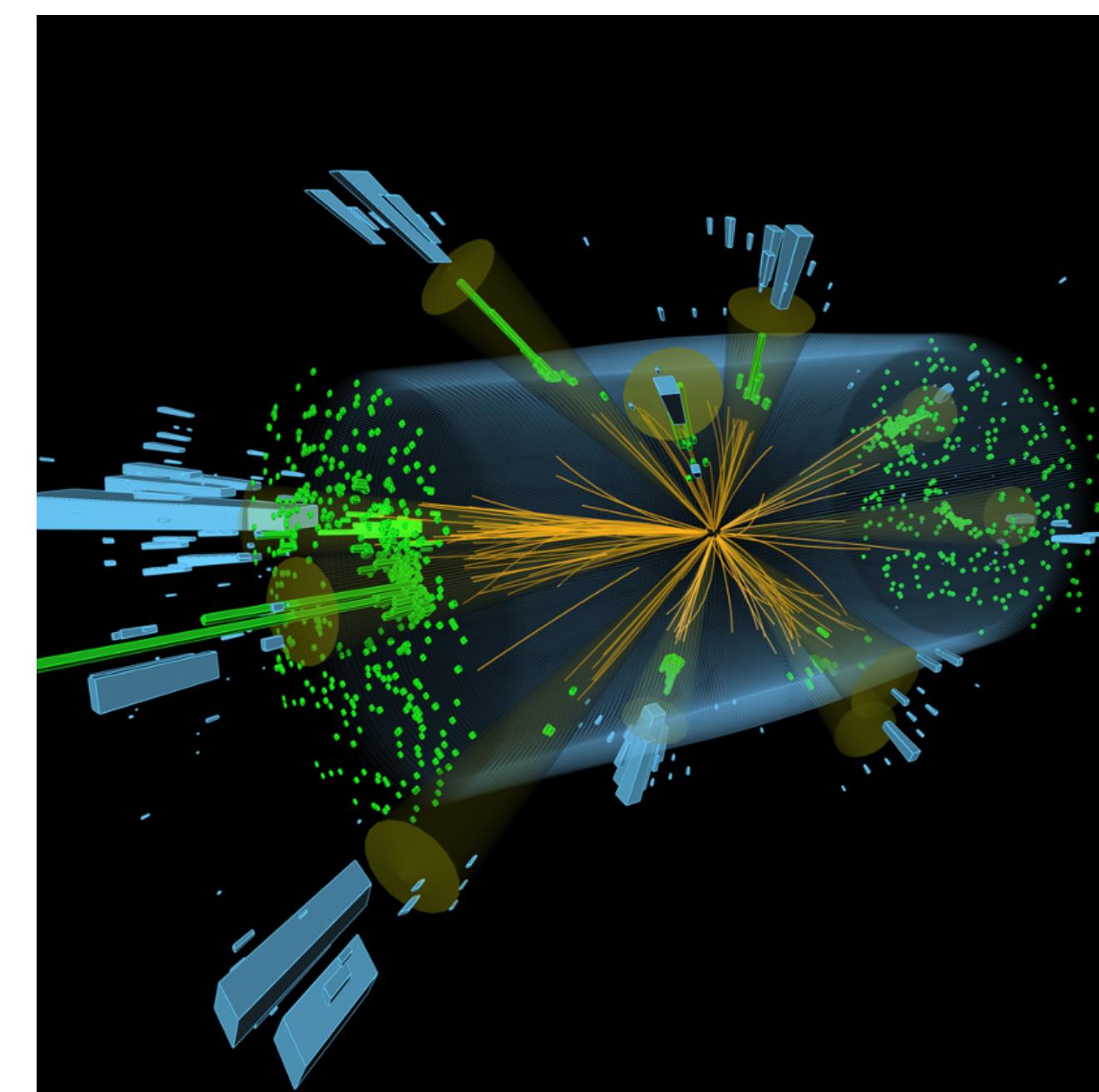
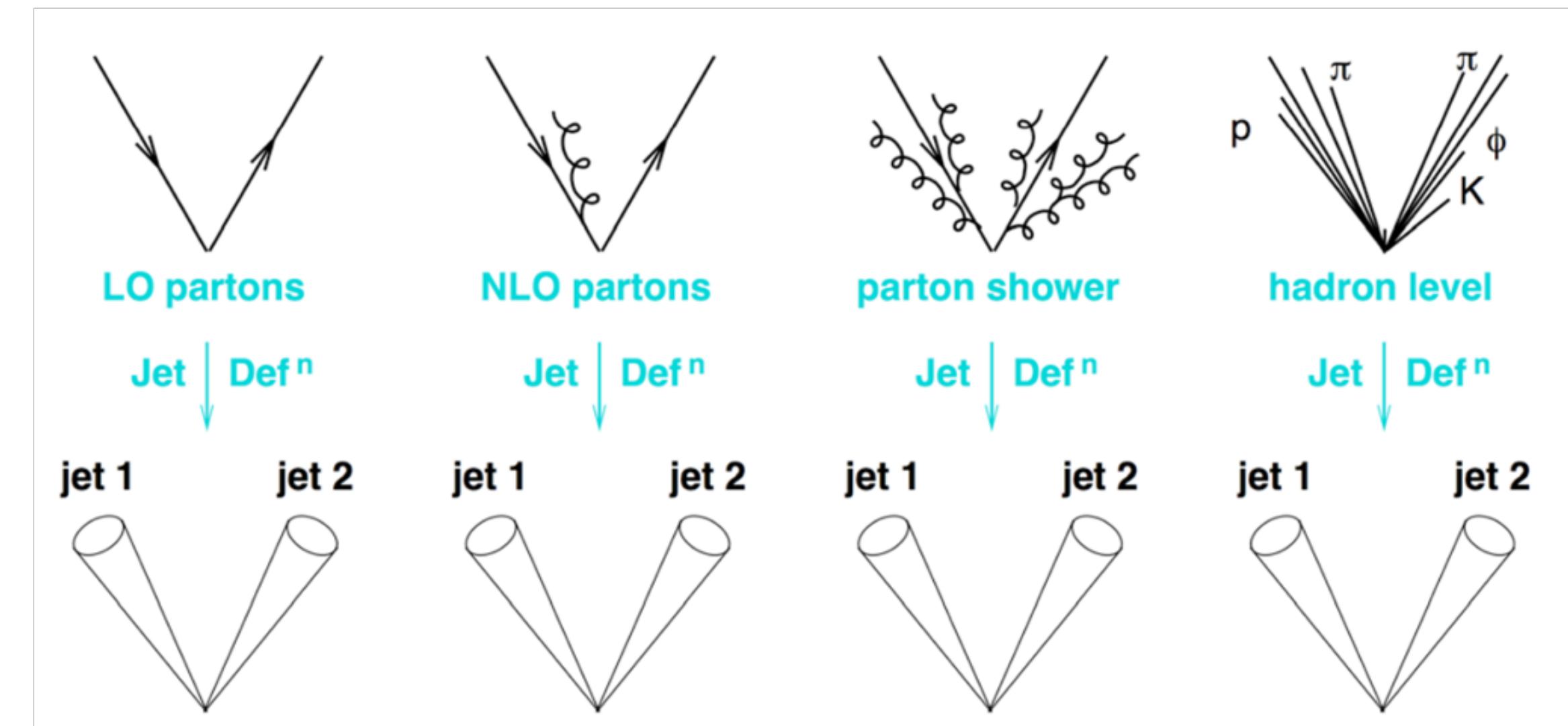
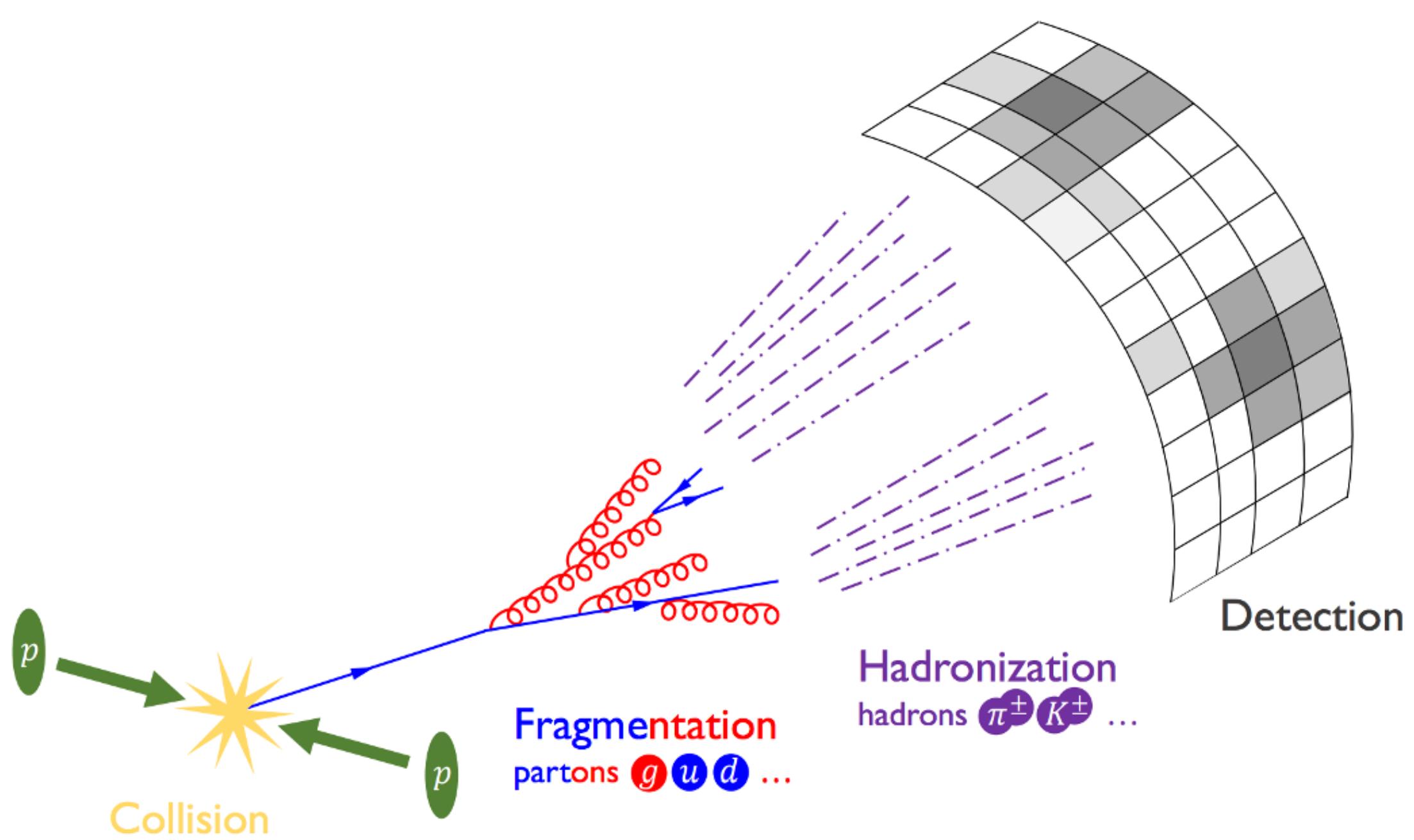
Usually there are jet-related questions on the exam

Variable	Type	Content
NJet	Integer	Number of jets in the event
Jet_Px	vector of floats	x-component of jet momentum (only jets with $p_T > 30\text{GeV}$ are stored)
Jet_Py	vector of floats	y-component of jet momentum (only jets with $p_T > 30\text{GeV}$ are stored)
Jet_Pz	vector of floats	z-component of jet momentum (only jets with $p_T > 30\text{GeV}$ are stored)
Jet_E	vector of floats	Energy of the jet (only jets with $p_T > 30\text{GeV}$ are stored)
Jet_btag	vector of floats	b tag discriminator value
Jet_ID	vector of bools	Jet quality identifier - Used to reject jets from detector noise. A good jet has true value for this variable

# Short exercise - part 1

## Short introduction to jets

- From the theory point of view
  - jets of partons (gluons and quarks)
- From the experimental point of view
  - Quarks and gluons hadronize quickly and we detect sprays of hadronic particles in our detectors
  - observables are protons, neutrons, pions, photons, electrons, muons



Experimentally, jets are just cones of energy.

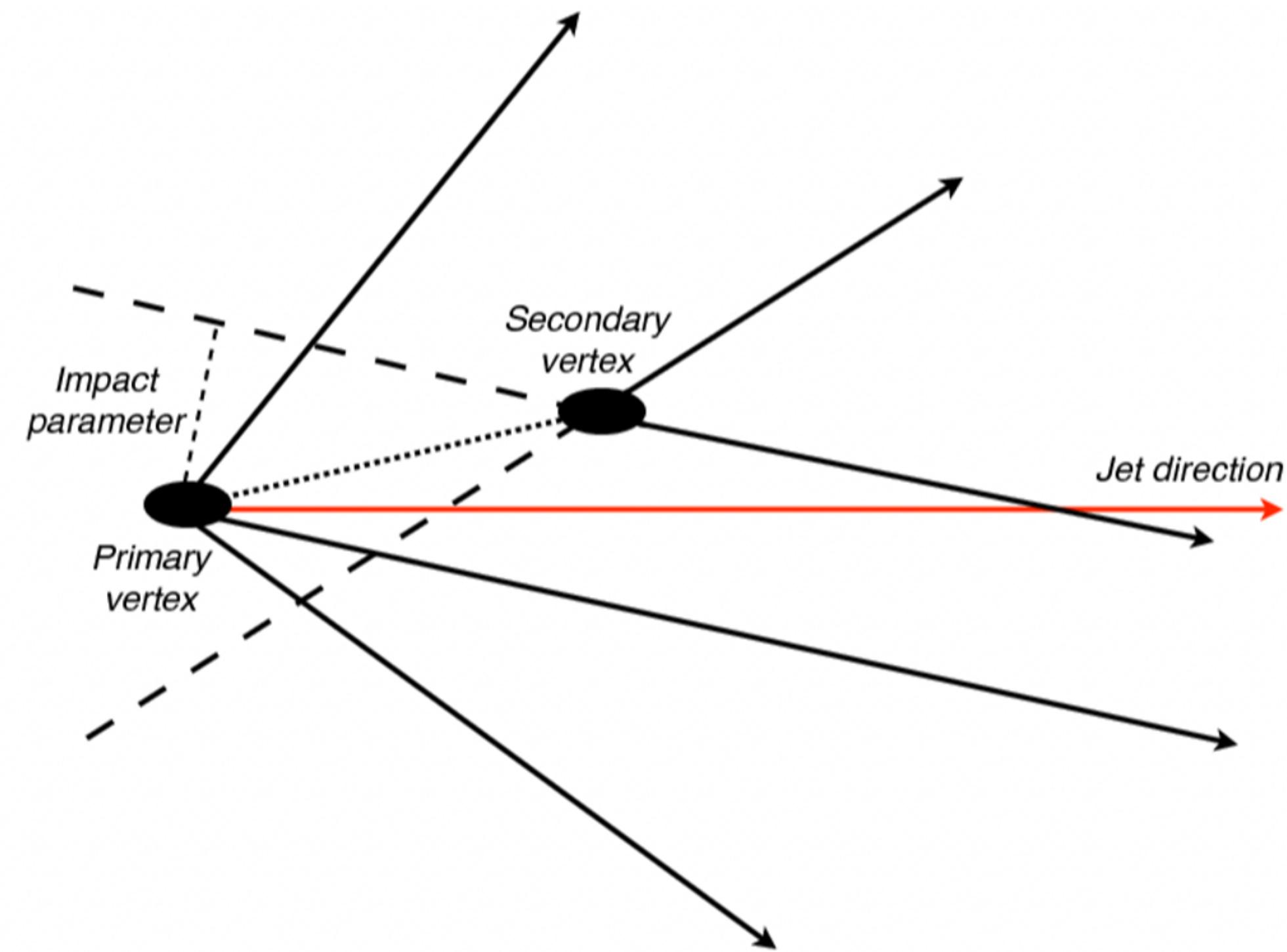
Light jets and heavy jets (b quarks, long lifetime).

# Short exercise - part 1

## B-jet identification

- The long lifetime of B hadrons can be exploited to identify jets originating from b-quarks
  - the reconstruction of secondary vertex due to the decay of B mesons indicates the jet is a b-jet
- A b-tagging discriminator (Jet\_btag) is obtained from an algorithm that identifies secondary vertices
  - higher values of b-tagging discriminator indicates a higher probability that the jet comes from a b-quark
- A jet is tagged as b-jet if b-tag discriminant is greater than 2.0 ( $\text{Jet\_btag} > 2.0$ )

b-jets are slightly shifted. With good enough impact parameter resolution, we are able to distinguish them.



# Short exercise - part 1

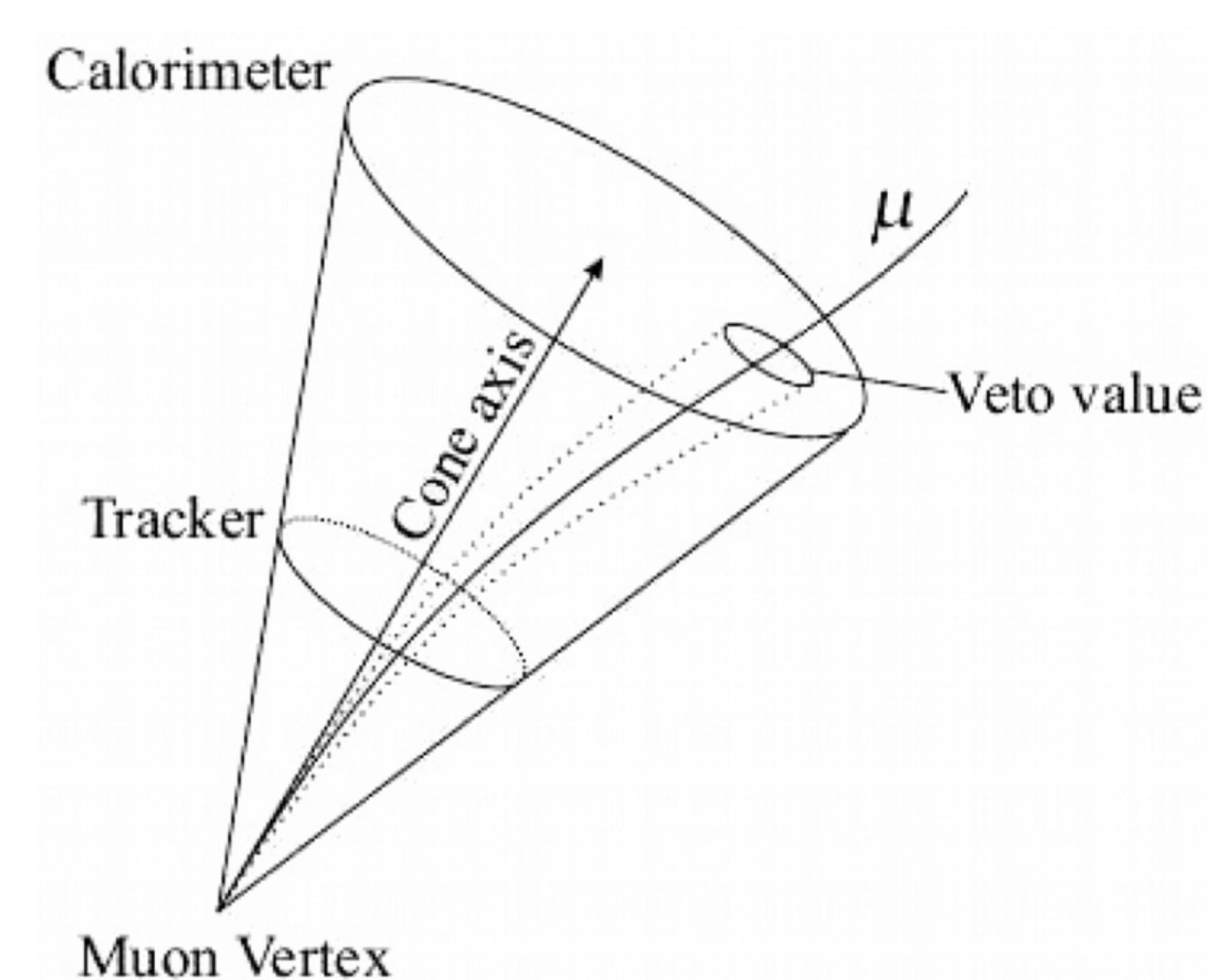
## Tree structure - muons

Variable	Type	Content
NMuon	Integer	Number of muons in the event
Muon_Px	vector of floats	x-component of muon momentum
Muon_Py	vector of floats	y-component of muon momentum
Muon_Pz	vector of floats	z-component of muon momentum
Muon_E	vector of floats	Muon energy
Muon_Charge	vector of integer	Muon charge <small>important to reconstruct whether it was W+ or W-</small>
Muon_Iso	vector of floats	Muon isolation value

# Short exercise - part 1

## Isolation

- Muons originating from Z/W boson decay manifest themselves in the detector as a clean trajectory traversing the inner tracker up to the muon detectors
  - muons from b quark decay are instead produced within a jet
- The isolation,  $I$  (Muon\_Iso), is defined as the sum of  $p_T$  of reconstructed particles within a cone of size  $\Delta R = \sqrt{(\Delta\eta^2 + \Delta\phi^2)}$  around the muon. It is used to discriminate muons originated from Z/W decays from those produced in jets, which are surrounded by a multitude of particles



We need to only look at isolated muons,  
because muons can also come from b-decays.

# Short exercise - part 1

## Tree structure - MET

Missing transverse energy (aka neutrino)  
Only x&y because we look at transverse (phi-angle)

Variable	Type	Content
Met_Px	vector of floats	x-component of the missing energy
Met_Py	vector of floats	y-component of the missing energy
EventWeight	vector of floats	Weight factor to be applied to simulated events to account for different sample cross sections
triggerIsoMu24	vector of bools	Trigger bit, it is true if the event is triggered, and false if it is not

single muon, isolated with  $\text{pt} > 24$

## Short exercise - part 1

### Analysis framework

- Develop your own analysis using the framework available
- The framework consists of different modules
  - MyAnalysis.py ← you'll have to modify this
  - example.py ← you'll have to modify this
  - Samples.py ← you probably **don't have to** modify this
  - Plotter.py ← you probably **don't have to** modify this

# Short exercise - part 1

## Analysis framework

### • MyAnalysis.py

- it's the **core of the analysis** implementation, here the selection strategy is implemented and histograms are booked and filled with selected events
- two main functions:
  - bookHistos: where histograms are defined
  - processEvent: where histograms are filled according to analyst criteria

```
def bookHistos(self):  
    h_nJet = ROOT.TH1F("NJet","#of jets", 6, -0.5, 6.5)  
    h_nJet.SetXTitle("%# of jets")  
    self.histograms["NJet"] = h_nJet  
  
....
```

```
def processEvent(self, entry):  
    tree = self.getTree()  
    tree.GetEntry(entry)  
    w = tree.EventWeight  
  
....  
    for m in xrange(tree.NMuon):  
        muon=ROOT.TLorentzVector(tree.Muon_Px[m],tree.Muon_Py[m], tree.Muon_Pz[m], tree.Muon_E[m])  
        self.histograms["Muon_Iso"].Fill(tree.Muon_Iso[m], w)
```

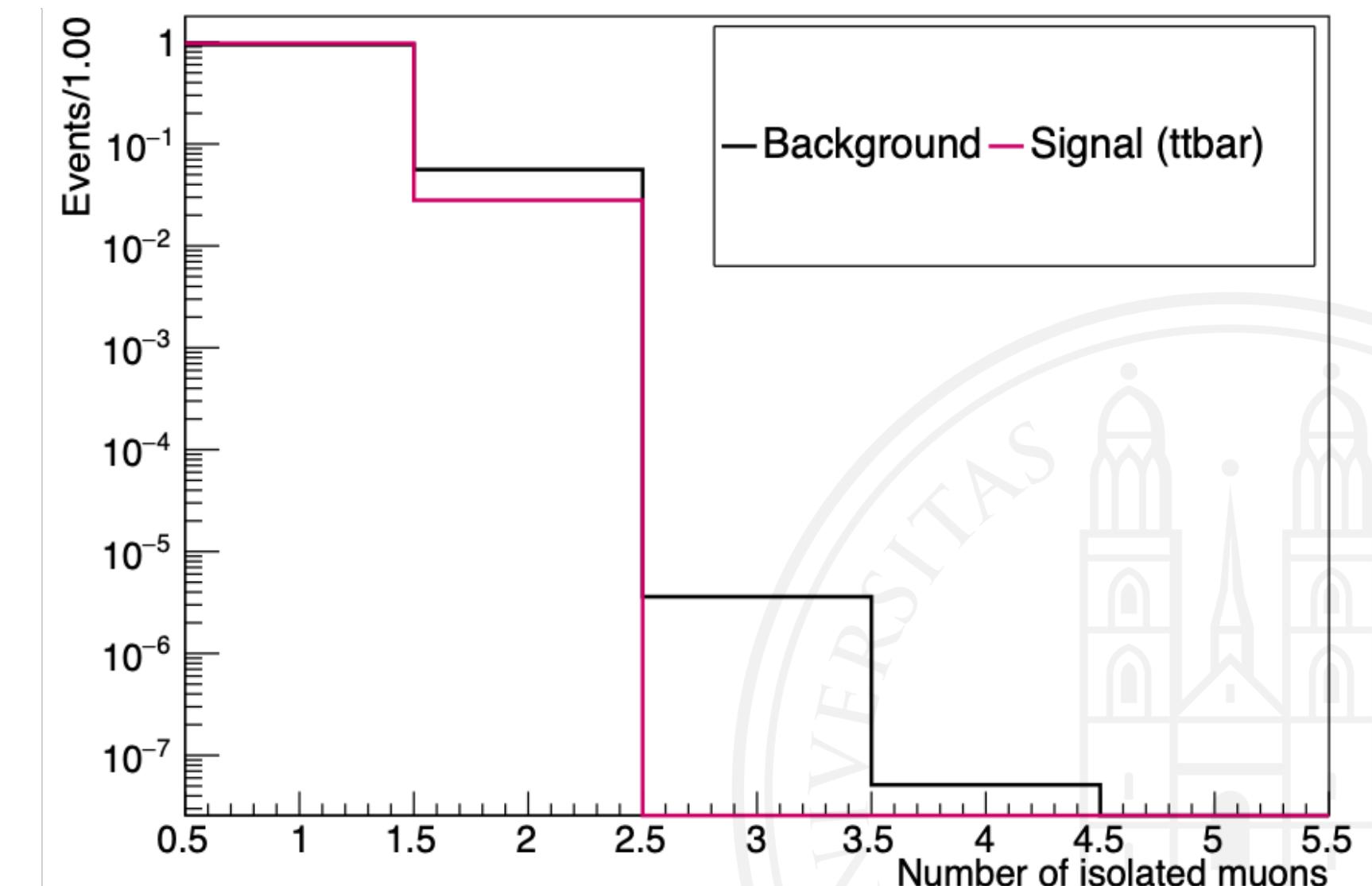
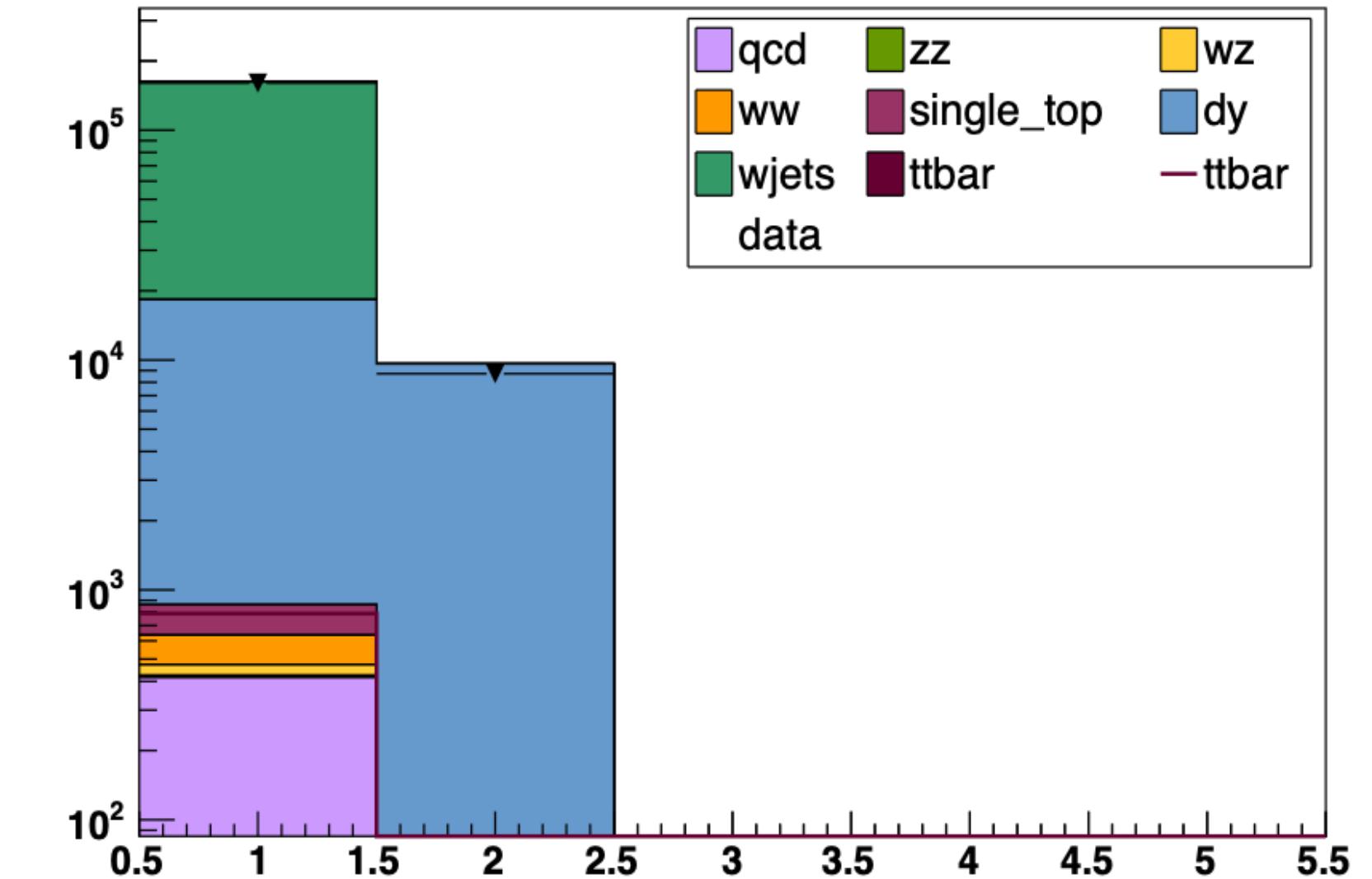
# Short exercise - part 1

## Analysis framework

- **example.py**

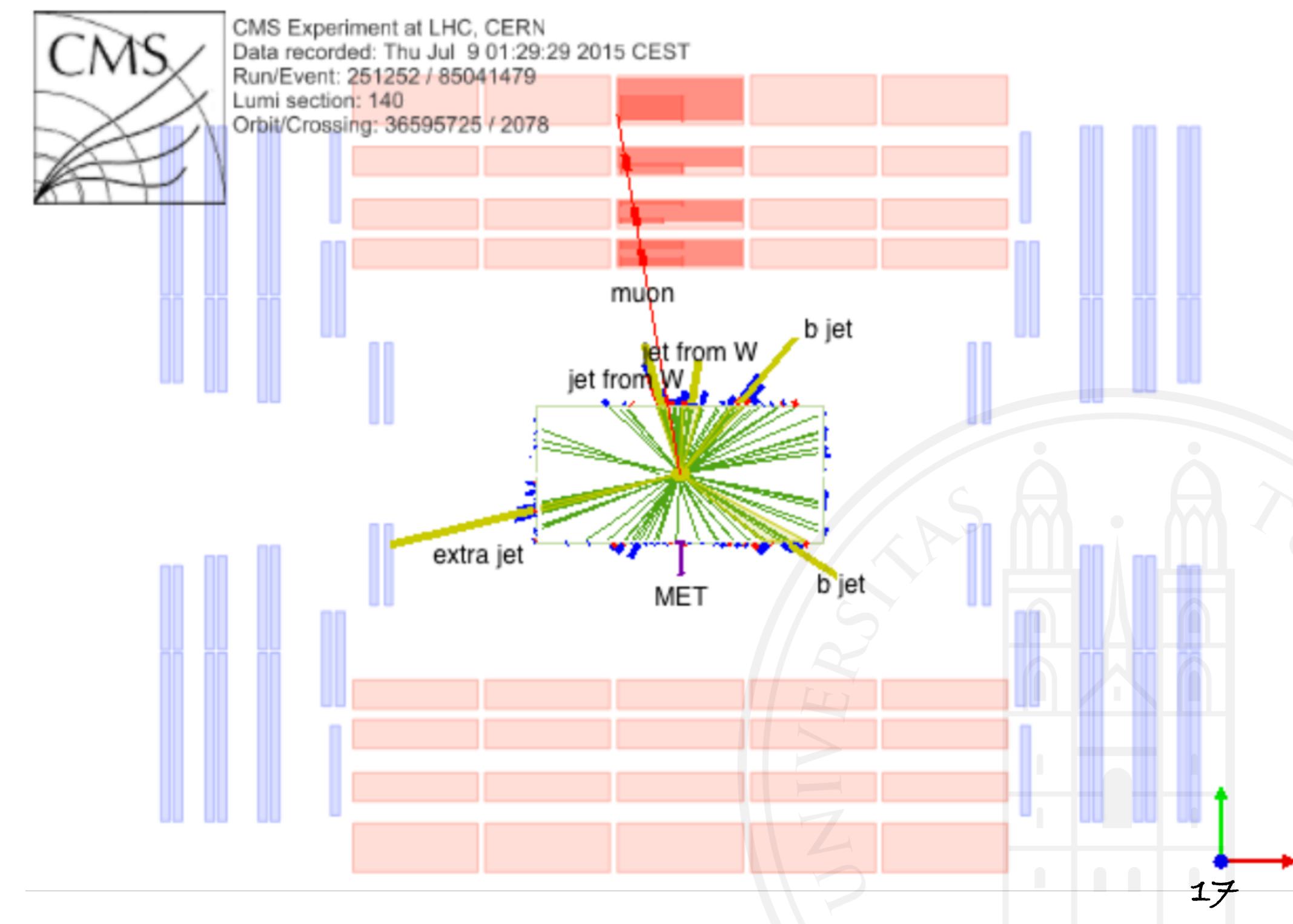
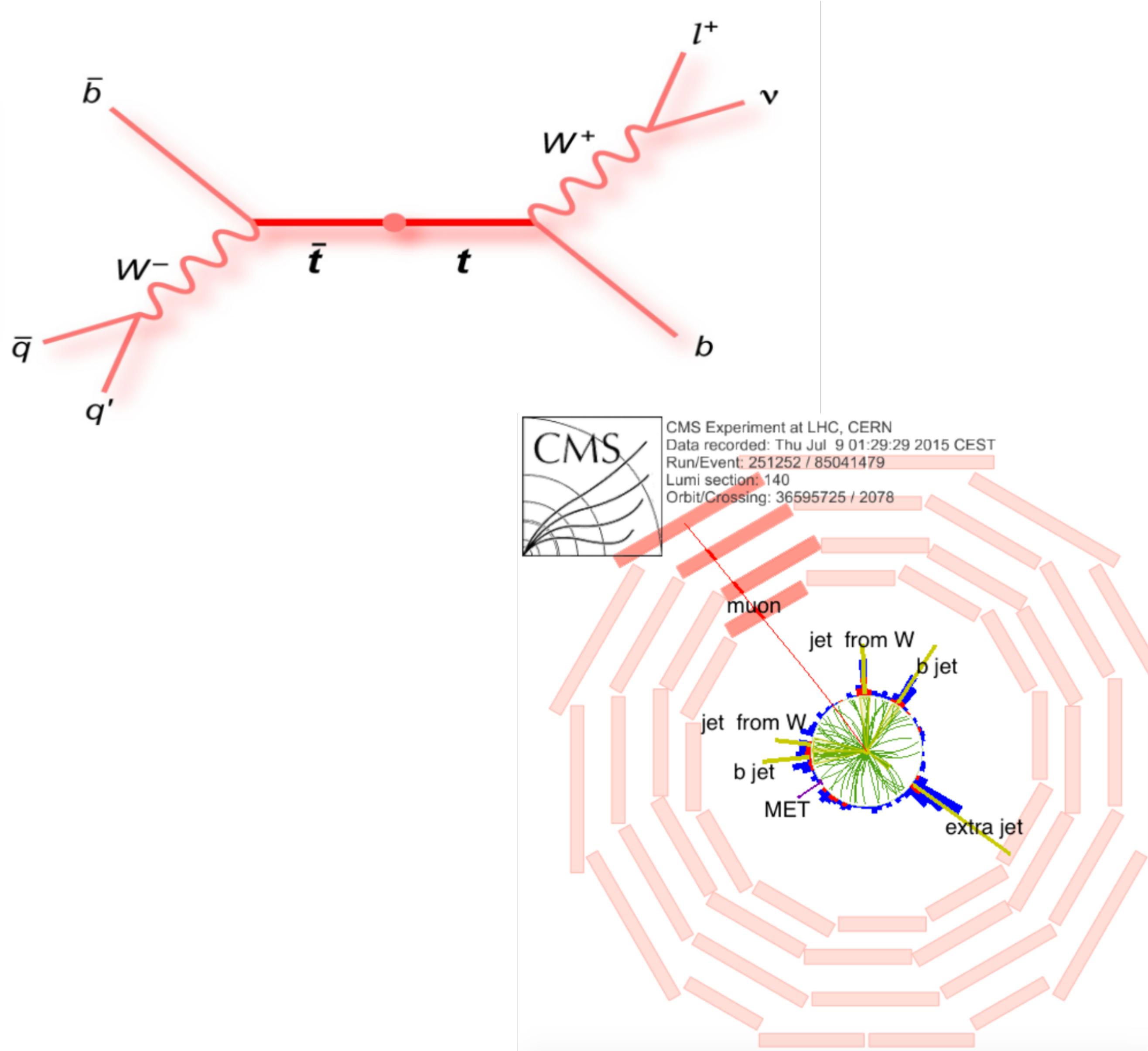
- it's the **main python file**. It calls the other modules and is used to define the actions, which samples should be processed, and which plots should be created.
- it returns .pdf files with plots for all indicated variables
- two kind of plots are produced:
  - a plot representing the sum of background plus signal and comparison to data
  - signal vs background distribution normalized to unity

```
for v in vars:  
    print "Variable: ", v  
    ### plotShapes (variable, samples,logScale )  
    plotShapes(v, samples, True)  
    ### plotVar(variable, samples, isData, logScale )  
    plotVar(v, samples, True, True)
```



# Short exercise - part 1

## Introduction



## Short exercise - part 1

### Objectives - part 1

- 1. Open the files data.root and ttbar.root and inspect them

- look interactively at the distributions of variables:

- NJet, NMuon, Jet\_Px/Py/Pz, Muon\_Px/Py/Pz, Muon\_Iso, Met\_px/py

- which is the most probable value for the number of Jets (NJet) and for the number of Muons (NMuon) in data? And in simulated t<sup>-</sup>t events?

Ans: most probable NJet: 0  
most probable NMuon: 0

triggerIsoMu24: 1

- pay attention to the variable triggerIsoMu24

- which values can you observe for data? And for MC?

- 2. Using the analysis framework select events with at least one isolated muon:

- Relative isolation (Muon\_Iso/ Muon\_PT) < 0.05

- Muon\_PT > 25 GeV

- 3. Compare the shapes of several distributions for the signal (ttbar) and the background (e.g. number of jets, number of tagged jets)

- which ones provide best discrimination between signal and background?

- 4. Find the most discriminating variables and choose a cut to apply in order to reject the background.

Optimise the selection choosing the cut that gives the highest S/√(S+B)

Ans for simulation:  
most probable Njet: 4  
most probable NMuon: 0  
triggerIsoMu24: 0

## Short exercise - part 1

## Objectives - part 2

- 5. Produce the histograms of these variables comparing MC simulation and data
  - use the function `plotVar(v, samples, True, True)` in `example.py`
- 6. Apply this selection to MC and data
  - what is the highest  $S/\sqrt{(S+B)}$  you can get (based on simulation)

5 already done

6) Add variables that help me: Missing  
transverse energy,  
mass of light jets, etc

backup

