UCL
Université catholique de Louvain

Delphes

S. Ovyn
X. Rouby

Delphes

A framework for fast simulation of a generic collider experiment

Séverine Ovyn, Xavier Rouby

Université catholique de Louvain, Belgium

Center for Particle Physics and Phenomenology (CP3)

GDR Terascale@Heidelberg, 16/10/2009

References

Website :

http://www.fynu.ucl.ac.be/delphes.html

News / Download / User manual / FAQ

Current version: Delphes V 1.8

Paper + User manual :

arXiv:0903.2225[hep-ph]

DELPHES, a framework for fast simulation of a generic collider experiment

S. Ovyn, X. Rouby and V. Lemaître
Center for Particle Physics and Phenomenology (CP3)
Université catholique de Louvain
B-1348 Louvain-la-Neuve, Belgium

Contacts :

severine.ovyn@uclouvain.be xavier.rouby@cern.ch

S. Ovyn
X. Rouby

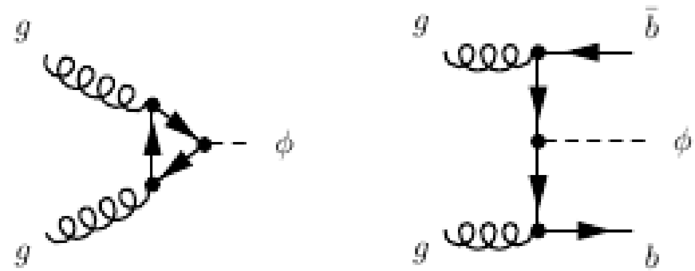
Motivation: from theory to detectors...

From theory to detectors...

S. Ovyn
X. Rouby

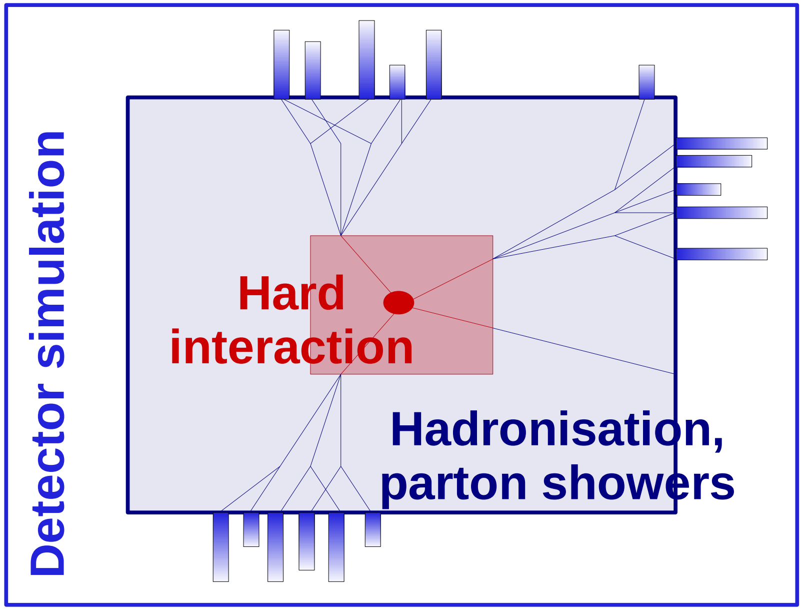
Knowing if theoretical predictions will be visible and measurable in a high energy experiment is complex and requires several steps:

1° Development of a new model



2° Implementation and generation of hard interaction

- MadGraph/MadEvent (MG/ME)
- CalcHep



3° Simulation of hadronisation and parton showers

- Pythia
- Herwig

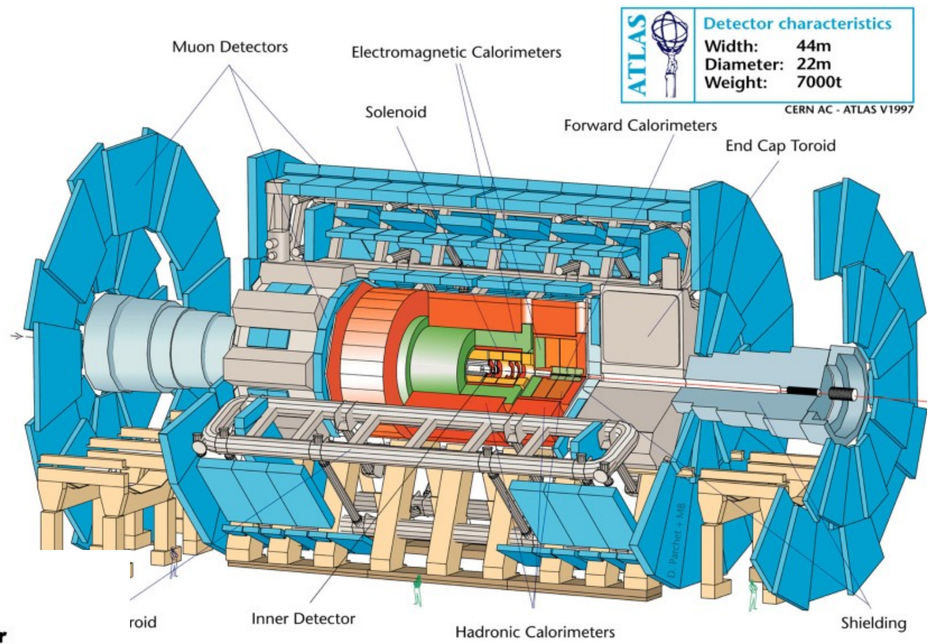
Complexity of HE detectors...

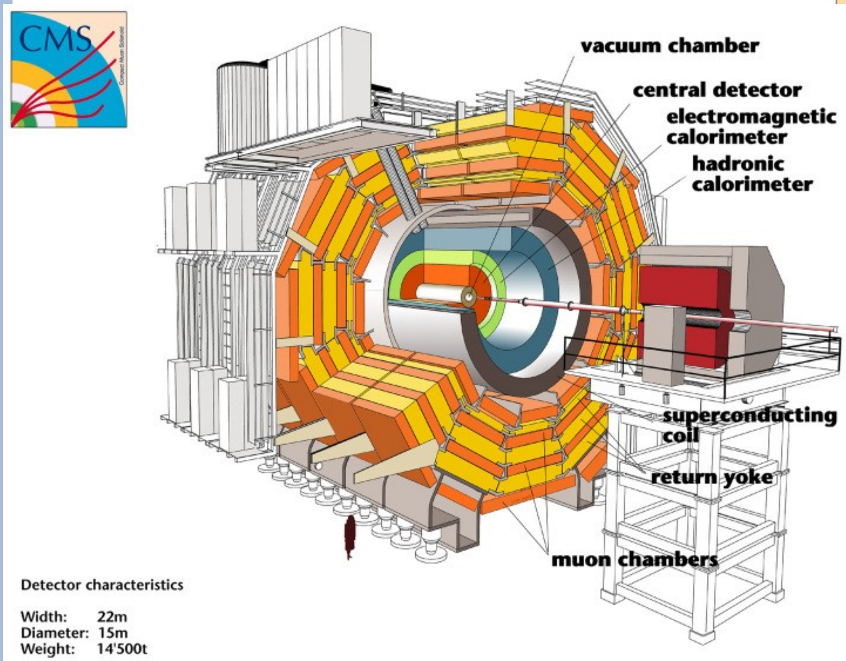
S. Ovyn
X. Rouby

4° Simulation of the response of a high energy experiment

ATLAS ➡

`Athena, ATLFAST I & II`





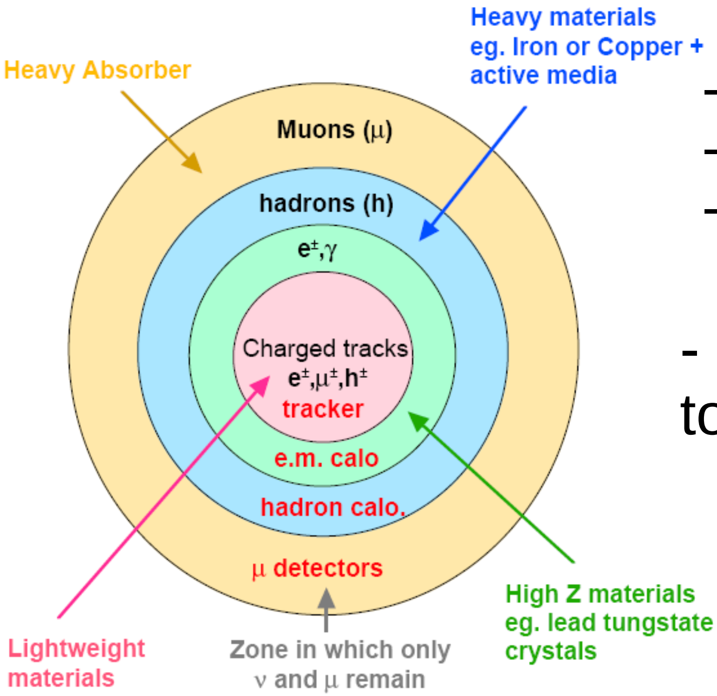
⬅ **CMS**

`CMSSW, FastSim`

Complexity of HE detectors...

S. Ovyn
X. Rouby

General structure



- Complexity of the related subdetectors
  - tracker
  - electromagnetic and hadronic calorimeters
  - muon chambers

- Requires the use of complex softwares to simulate
  - detailed energy deposition from ionization, showering
  - secondary interactions
  - detector inefficiencies
  - multiple scattering
  - ...

Such a simulation is very complex and a large CPU per event

Phenomenological studies may require only fast but realistic estimates of detector response → Delphes

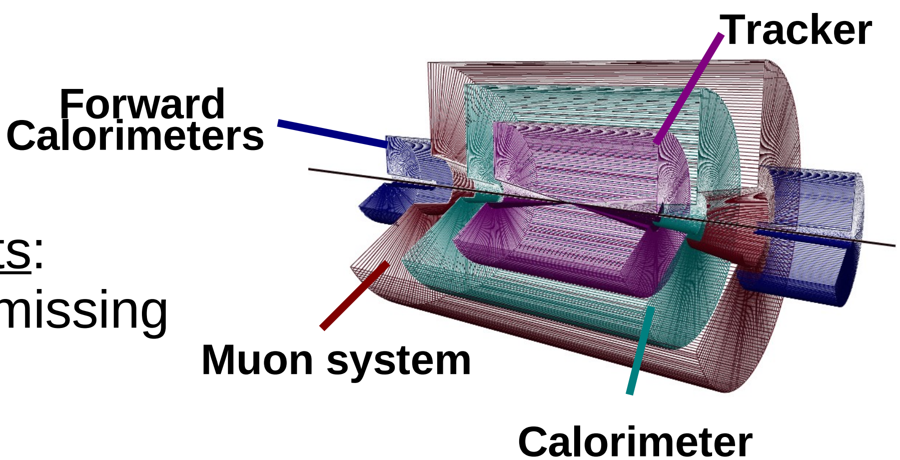
Fast simulation utilitiy

S. Ovyn
X. Rouby

Delphes provides:

- Realistic simulation taking into account subdetector extensions, types, segmentations and resolutions

  - A tracker in a solenoidal magnetic field
  - Calorimeters with electromagnetic and hadronic sections
  - Muon system



- Reconstruction of physics objects: leptons, jets, b-jets, tau-jets, and missing transverse energy

- Trigger emulation

- An event display

Delphes allows easy connection between theoretical and experimental *(distant)* worlds

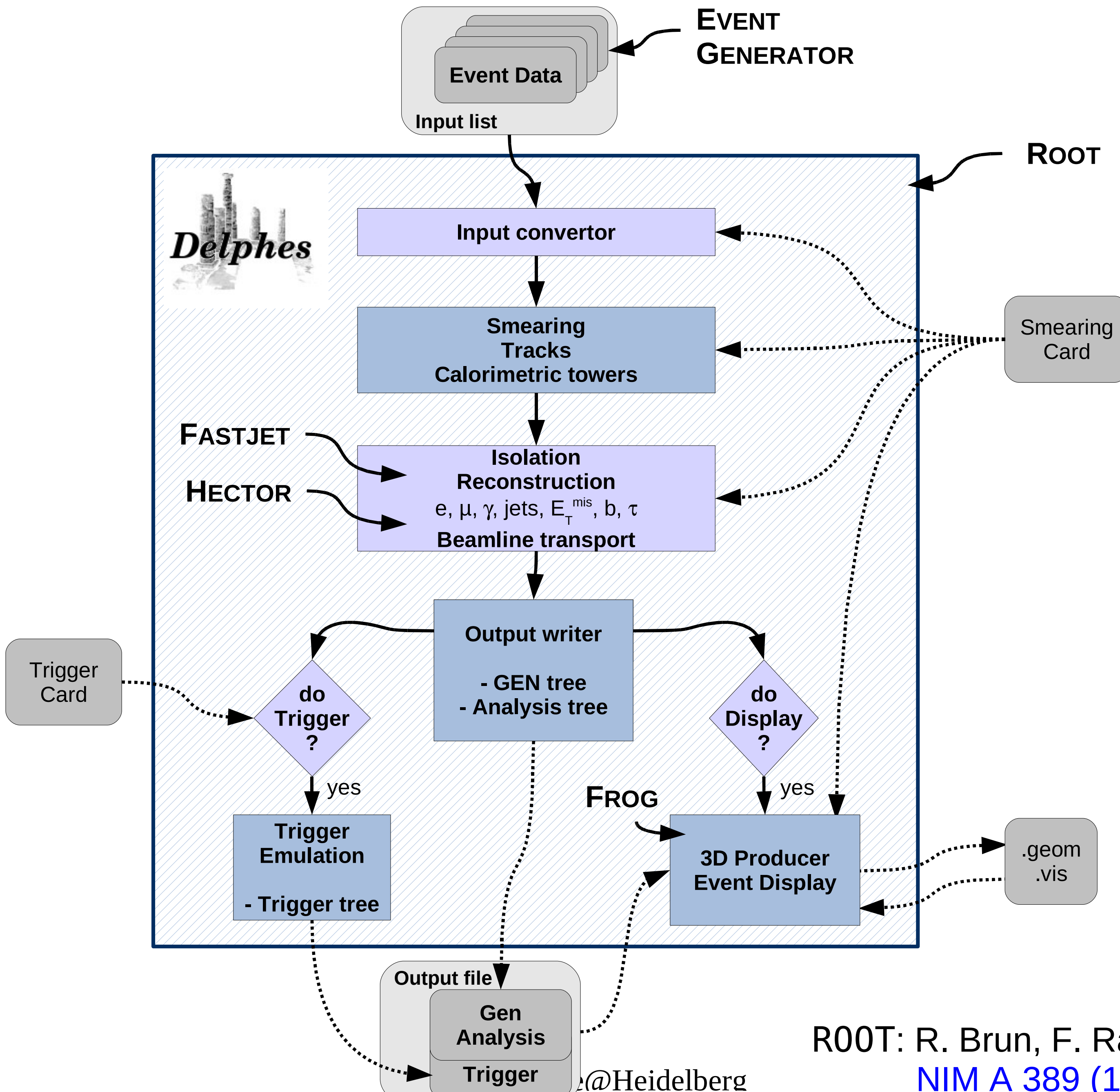
The code is also independent from any collaboration

S. Ovyn
X. Rouby

C++ implementation of the simulation

C++ / ROOT implementation

S. Ovyn
X. Rouby

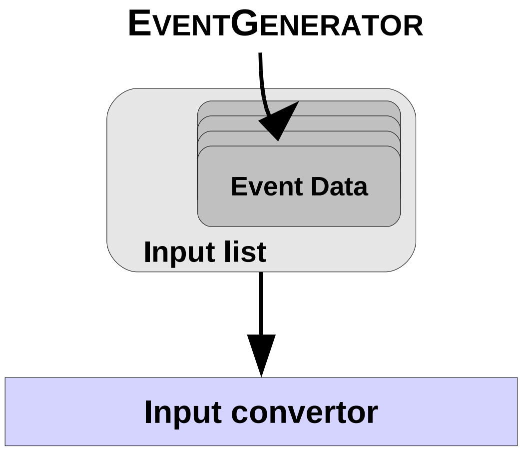


ROOT: R. Brun, F. Rademakers, NIM A 389 (1997) 81-86.

Delphes flow

S. Ovyn
X. Rouby

Interface:



First step of ***Delphes***: conversion of input events into a ROOT tree readable by the rest of the code

➡ Result of the conversion stored in a `GEN tree`

➡ Allow easy checks between various generators

```
./Delphes inputlist.list OutputRootFileName.root data/DetectorCard.dat
data/TriggerCard.dat
```

- Input events : ***Delphes*** is interfaced to standard file formats

- `StdHEP`
- `ROOT` files obtained with `h2root` (hbook)
- Les Houches Event Format
- `HepMC`

➡ Compatible with - MG/ME, Pythia, Herwig,...

Delphes flow

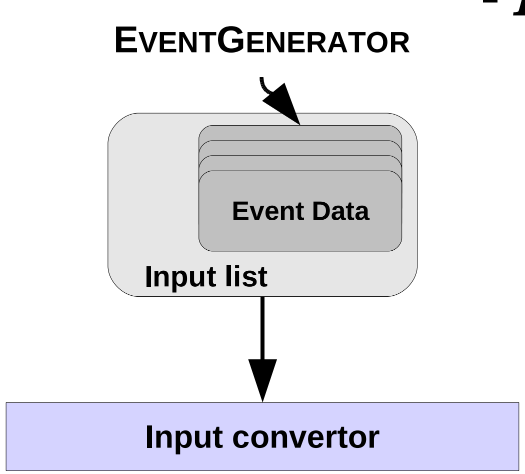
S. Ovyn
X. Rouby

Interface:

The input file list contains the location of the input files with one file per line!

```
/nfs/cms/mass10/o/ovyn/Analysis2/Wt_2l/apatWt_an_1.root
/nfs/cms/mass10/o/ovyn/Analysis2/Wt_2l/apatWt_an_2.root
/nfs/cms/mass10/o/ovyn/Analysis2/Wt_2l/apatWt_an_3.root
```

➡ Automatic detection of the file extension (.lhe, .hep, .root, .hepmc)



- ***Delphes*** is driven by two input cards defining

(a) detector parametrisation
(b) trigger definitions
(c) parameters on physics objects (cuts,...)

➡ Default detector cards and trigger tables available for ATLAS & CMS experiments

```
./Delphes inputlist.list   OutputRootFileName.root
                 data/DetectorCard.dat   data/TriggerCard.dat
```

Delphes flow

S. Ovyn
X. Rouby

Output file format: ***Delphes*** yields realistic observables for all reconstructed high level objects in two formats:

1°) Analysis tree in ROOT files, using ExRootAnalysis, P. Demin

- GEN tree (Monte Carlo level information)
- Analysis tree (detector level information)
- Trigger tree (trigger acceptance)

2°) LHCO http://v1.jthaler.net/olympicswiki/doku.php

Column format

#	typ	eta	phi	pt	jmass	ntrk	btag	had/em	dummy	dummy

Typ: 0 = photon , 1 = electron , 2 = muon , 3 = tau-jet , 4 = jet , 6 = MET

Ntrk: number of tracks associated with the object. For of a lepton, this number is multiplied by the charge of the lepton.

had/em: ratio of the hadronic versus electromagnetic energy deposited in the calorimeter cells associated with the object; it is typically > 1 for a jet and << 1 for an electron or γ.

Trees in the output ROOT file

Delphes.root

- GEN

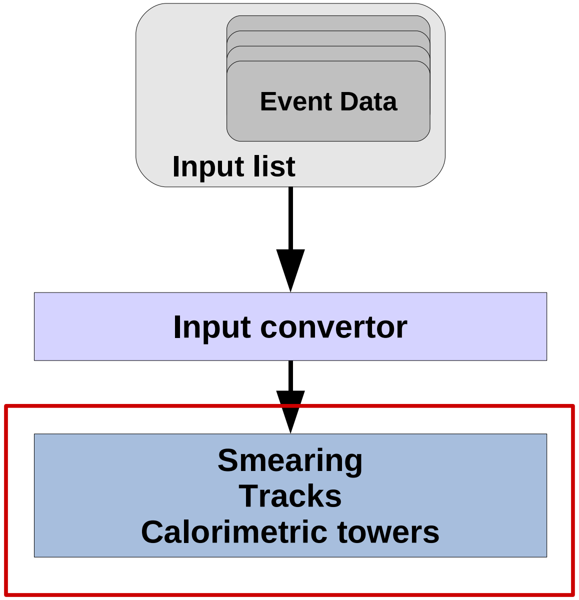
  Monte Carlo level information

  - Event
    - Event.Number
  - Particle
    - Particle.PT
    - Particle.E
    - Particle.Px
    - Particle.D1
    - Particle.D2
    - Particle.M1
    - Particle.M2
    - Particle.Status
    - Particle.PID

S. Ovyn
X. Rouby

S. Ovyn
X. Rouby

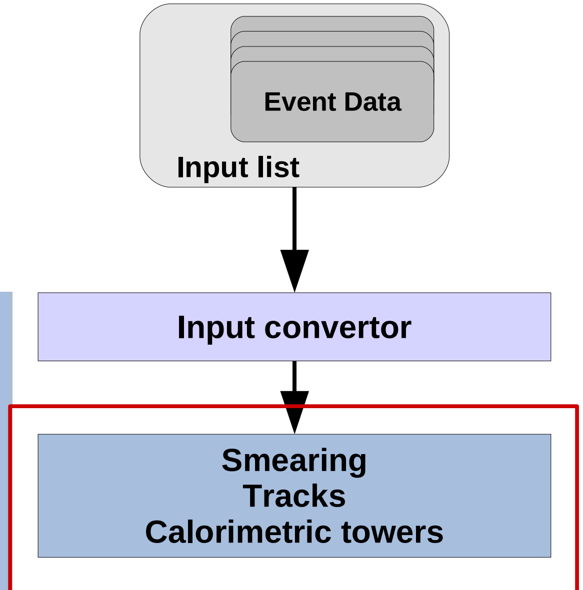
Low objects, detector level information



Delphes flow

S. Ovyn
X. Rouby

Motivations
Simulation
Interface
Tower-tracks
photon-e/μ
jets
tau-jets-MET
Forward det.
Validation
BUT also...
Tutorial
Conclusion



Smearing: Response of each subdetector parametrised as a function of the energy:

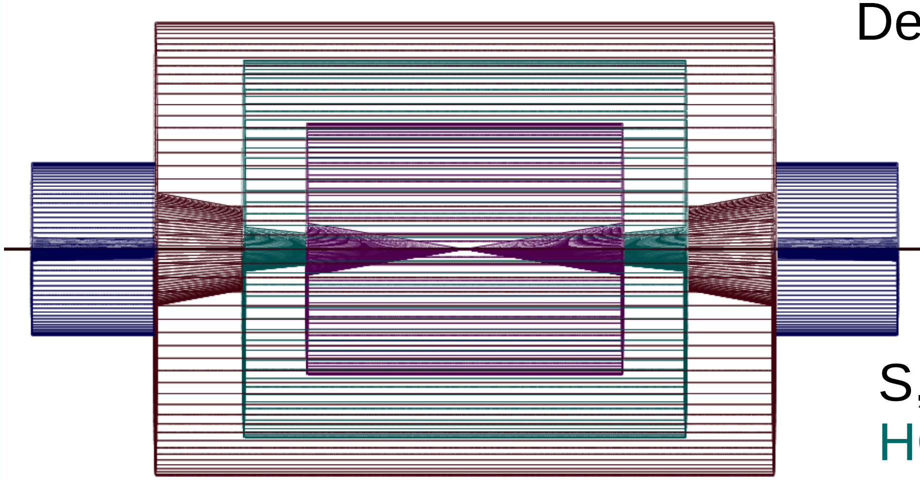
$$\frac{\sigma}{E} = \frac{S}{\sqrt{E}} \oplus \frac{N}{E} \oplus C$$

With different response to
- electromagnetic objects
- hadrons

Muons: smearing on the p_T

Parameters controllable using the input datacard

Schematic view of the *Delphes* detector



Detector extension in pseudorapidity
- tracker coverage
- central calorimeter coverage
- forward calorimeter coverage
- muon chambers coverage

S, N and C term of the ECAL, HCAL, FCAL

Delphes flow

Delphes

S. Ovyn
X. Rouby

Low level objects : Tracks

For all charged particles in the tracking coverage, considering « energy flow »

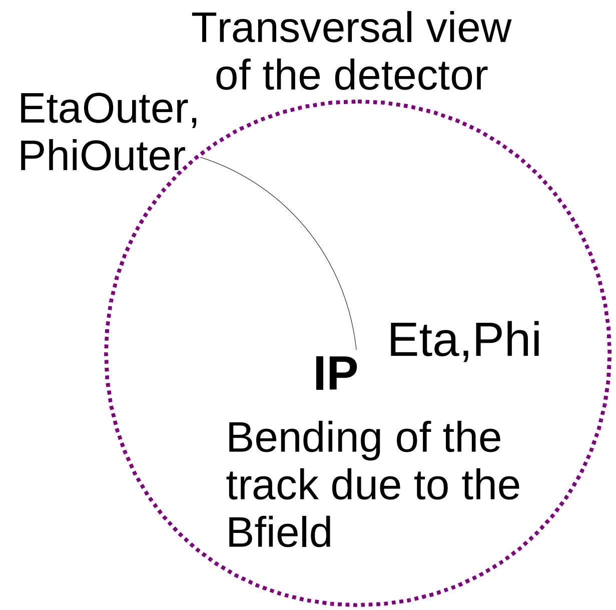
The tracker is embedded in a magnetic B field

- Position of charged particles is modified
- The values of the length and radius of the tracker are important parameters

Impact of the η modification important when the p_T of the particle is too small to reach the central calorimeters

The inner and outer value of the tracks are stored in the `Tracks` branch of the ***Delphes*** `ROOT` file

→ Eta, Phi, EtaOuter, PhiOuter



!! The particle energies are smeared according to the resolution of the calorimeter subdetector they reach !!

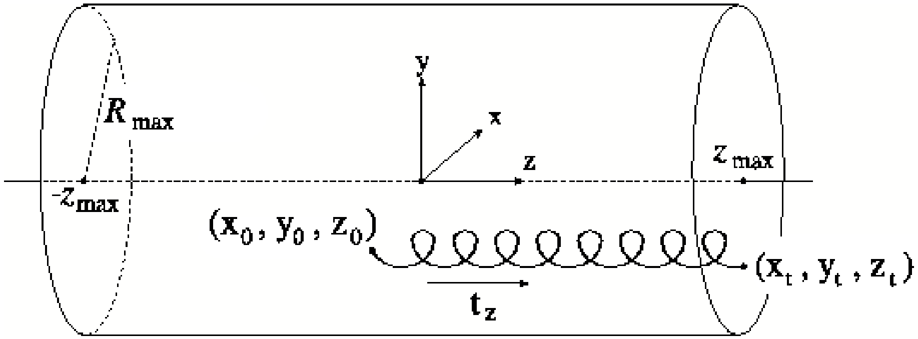
Delphes flow

S. Ovyn
X. Rouby

Simulation of the central solenoidal magnetic field

Bx = By = 0 ➡ Exact calculation of the transport of a charged particle

The magnetic field is supposed to be
- homogeneous
- constant inside a cylinder of length $2 \times z_{max}$ and of radius R_{max}.



To make the code faster, the time of flight needed to exit the cylinder is computed

$$\Rightarrow \quad t_{max} = min(t_T, t_z) \quad \begin{cases} t_z \text{ such that } |z(t_z)| = z_{max} \\ t_T \text{ such that } R(t_T) = R_{max} \end{cases}$$

Bx ≠ 0 By ≠ 0 ➡ iterative method, step by step until the particle exits the tracker region (slower method)

Limitation: magnetic field outside the tracker (e.g. in muon system) not simulated with ***Delphes***.

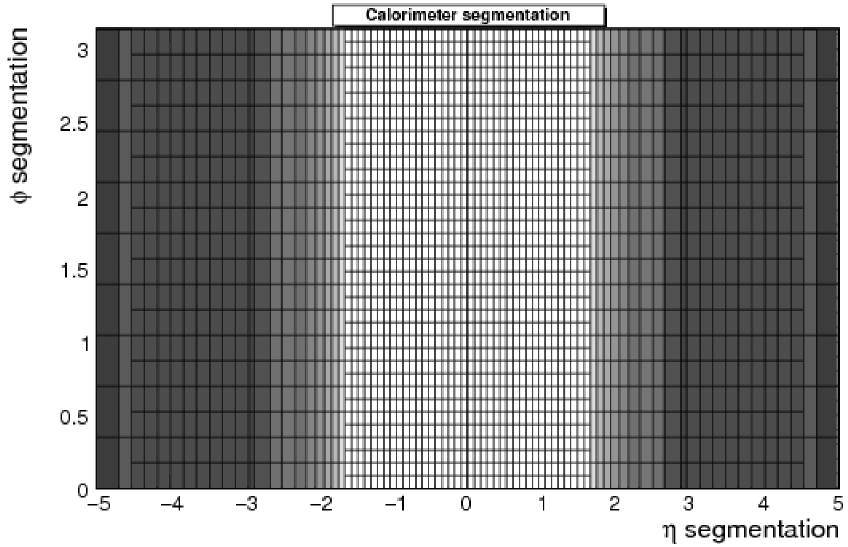
Delphes flow

Delphes

S. Ovyn
X. Rouby

Calorimetric cells

Segmentation in eta/phi



Need to enter in the datacard the number of cells in pseudorapidity as well as the edges of the towers in eta/phi

Assumptions:
- the detector is supposed to be symmetric in η!
- all cells have an identical ϕ-size for a given η value
- identical segmentation for ECAL and HCAL

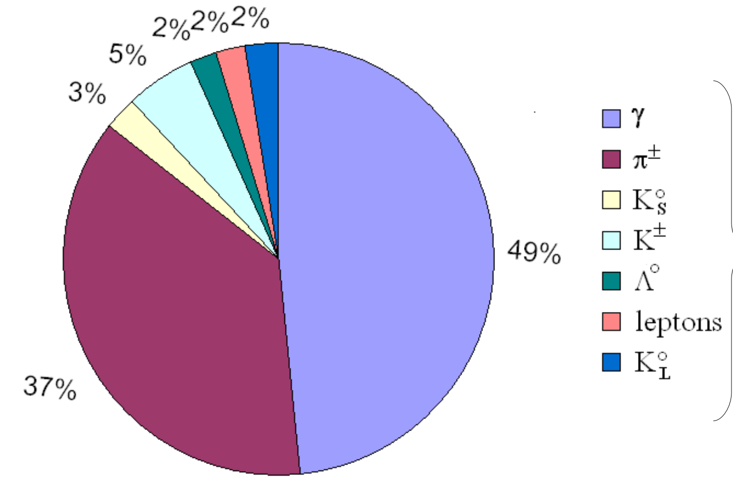
e.g.

TOWER_number 40
TOWER_eta_edges 0. 0.087 0.174 0.261 0.348 0.435 0.522 0.609 0.696 0.783 0.870 0.957 1.044 1.131 1.218 1.305 1.392 1.479 1.566 1.653 1.740 1.830 1.930 2.043 2.172 2.322 2.500 2.650 2.868 2.950 3.125 3.300 3.475 3.650 3.825 4.000 4.175 4.350 4.525 4.700 5.000
TOWER_dphi 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 20 20

Delphes flow

S. Ovyn
X. Rouby

- Charged and neutral final-state hadrons interact with the ECAL, HCAL and FCAL



$$E_{HCAL} = E \times F$$

$$E_{ECAL} = E \times (1 - F)$$

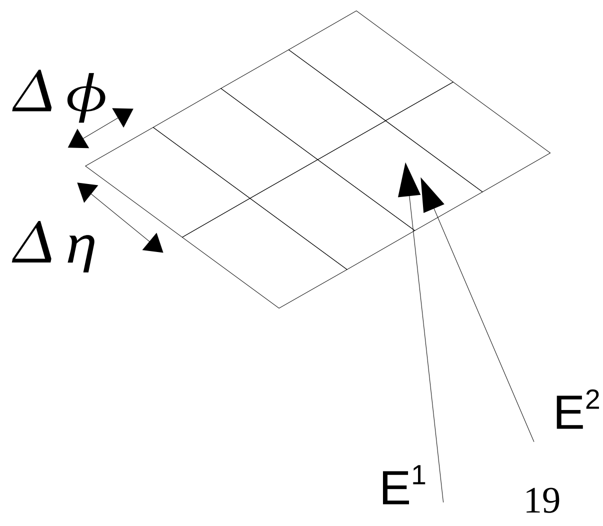
Smearing of particles performed using the expected fraction of the energy, determined according to their decay products, that would be deposited into the ECAL (E_{ECAL}) and into the HCAL (E_{HCAL})

- Summing energy of multiple impacts in identical towers

$$E^{tower}_{ECAL} = E^1_{ECAL} + E^2_{ECAL} \quad \text{and} \quad E^{tower}_{HCAL} = E^1_{HCAL} + E^2_{HCAL}$$

➡ Smearing of the corresponding energies

$$E^{tower} = E^{tower}_{SHCAL} + E^{tower}_{SECAL}$$



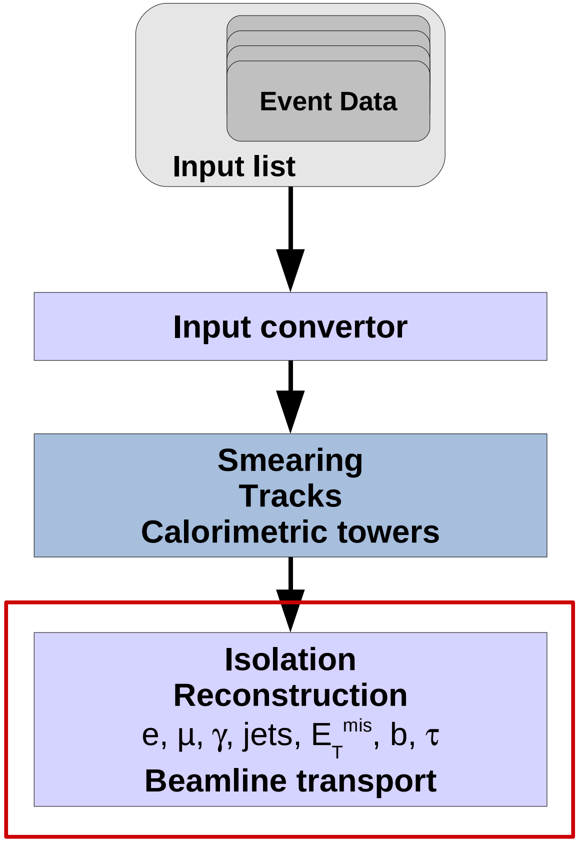
Trees in the output ROOT file

S. Ovyn
X. Rouby

Delphes.root

- **GEN** — Monte Carlo level information
  - Event
  - Particle
- **Analysis** — Analysis level information
  - CaloTower
    - CaloTower.E
    - CaloTower.ET
    - CaloTower.E_em
    - CaloTower.E_had
    - CaloTower.Eta
    - CaloTower.Phi
  - Tracks
    - Tracks.Charge
    - Tracks.E
    - Tracks.PT
    - Tracks.Phi
    - Tracks.Eta
    - Tracks.PhiOuter
    - Tracks.EtaOuter
    - Tracks.Px
    - Tracks.Py
    - Tracks.Pz

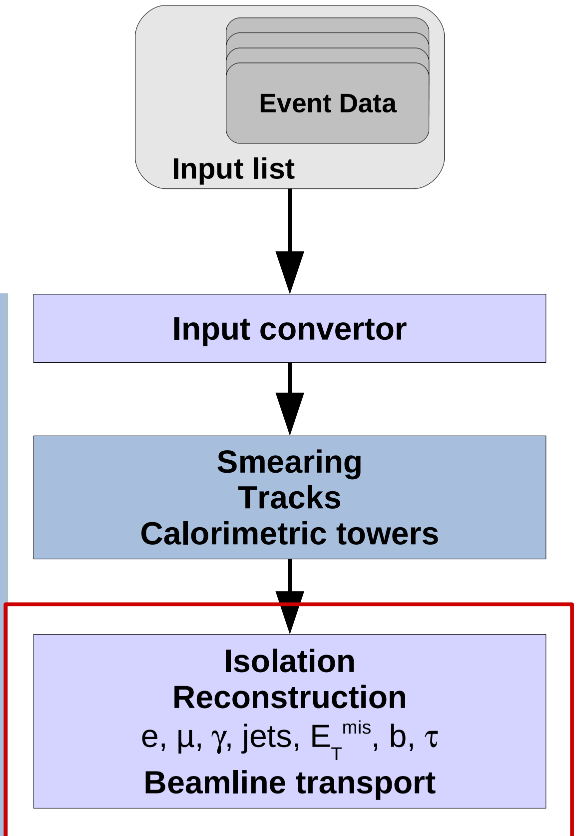
S. Ovyn
X. Rouby



High level objects, final reconstructed information

Delphes flow

S. Ovyn
X. Rouby



Photons :

- identified with true PID
- reconstructed if they fall into the tracker coverage
- measured eta/phi from the corresponding calorimeter cell

Electrons and muons :

- identified with true PID
- reconstructed if they fall into the tracker coverage
- muons do not leave energy in the calorimeters

Limitation: no fakes, no punch-through, no clustering

Lepton isolation:

- Isolation of charged particles using tracking information

↳ No other charge particles with p_T > 2 GeV/c within a cone

$$\Delta R = \sqrt{\Delta \eta^2 + \Delta \phi^2} < 0.5$$

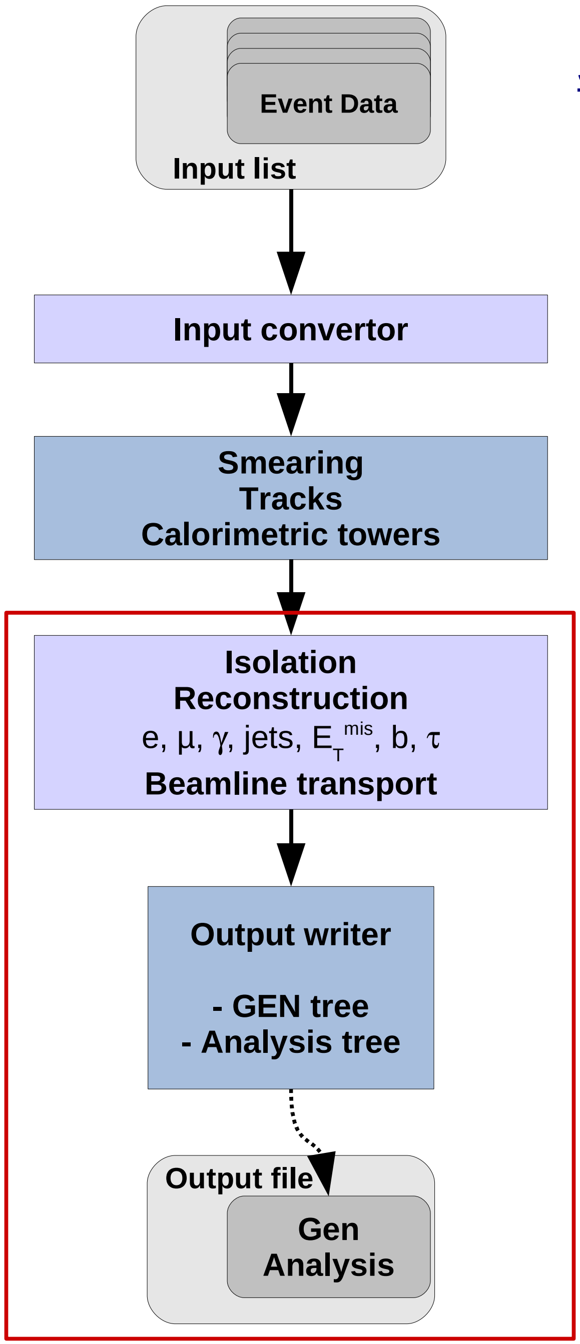
- Additional information available for calorimetric isolation

UCL
Université catholique de Louvain

Delphes

Delphes flow

S. Ovyn
X. Rouby



Jets :

- Treatement of particles which hadronise using jet reconstruction

- Uses reconstruction algorithms implemented in FastJet

  - CDF jet algorithm (cone)
  - CDF Midpoint algorithm
  - SIS Cone jets
  - Longitudinally invariant k_t jets
  - Cambridge / Aachen jets
  - Anti k_t jets

Jet algorithms differ
- in their sensitivity to soft particles or collinear splittings
- their computing speed performances.

`FastJet`: M. Cacciari, G.P. Salam, Phys. Lett. B 641 (2006) 57.

Delphes flow

S. Ovyn
X. Rouby

Choice of the jet algorithm, jet parameters in the detector datacard

In addition to the standard E, Px, Py, Pz, Eta, Phi variables, the jet collections also contains

- The number of tracks associated to the jet
- The hadronic and electromagnetic contents of the jet
- A b-flag indicating if the jet has been b-tagged

Assumptions for the *b*-tagging

- identical in the entire tracker coverage
- independent of the p_T of the jet
- efficiency controllable in the datacard (default= 40%)
- mis-identification of *c* (10%) and light jets (1%)

Final remark

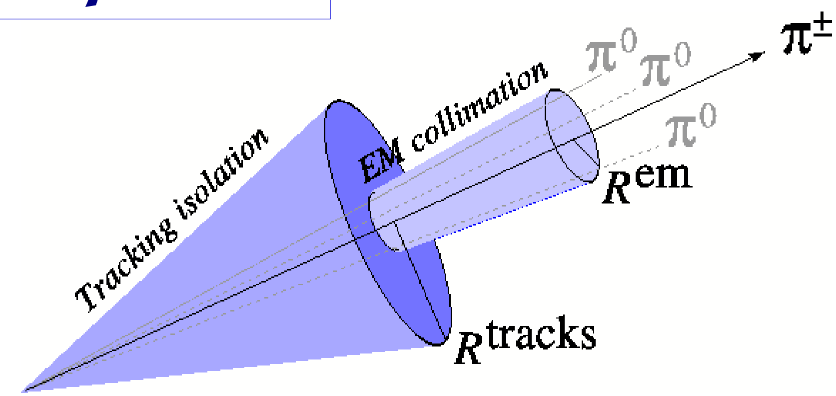
The user can choose if a perfect energy recontruction is applied in the tracker coverage (perfect energy flow).
If not, jets are taking as input the calorimetric cells.

Delphes flow

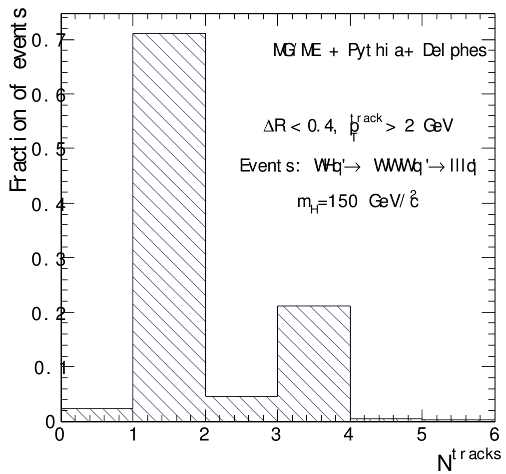
S. Ovyn
X. Rouby

Tau-jets reconstruction:

Selected from the jet collection



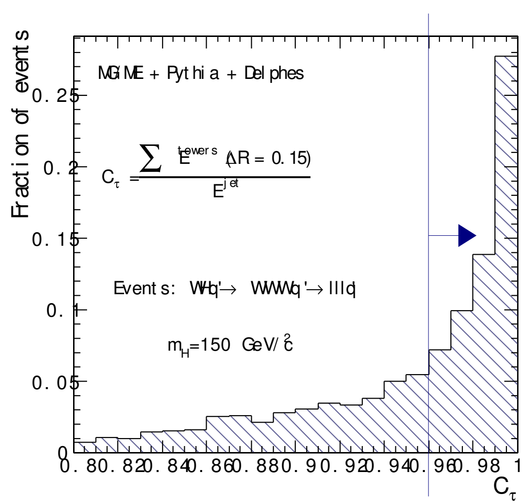
1) Requirement of tracking isolation



Number of tracks associated to a particle with p_T>2 GeV/c is one and only one in a cone of radius R_{tracks}

- 3-prong τ dropped.
- Cone should be entirely incorporated into the tracker

2) Use of the narrowness of the tau-jet



C_τ = sum of the energy of towers in a small cone of radius R_{em} around the jet axis, divided by the energy of the reconstructed jet.

C_τ expected to be large

Delphes flow

- Measurement of Missing Transverse Energy (MET) in an ideal detector

Momentum conservation imposes the transverse momentum of the observed final state (p_T^{obs}) to be equal to the vector sum of the invisible particles,

$$\vec{p}_T = \begin{pmatrix} p_x \\ p_y \end{pmatrix} \text{ and } \begin{cases} p_x^{miss} = -p_x^{obs} \\ p_y^{miss} = -p_y^{obs} \end{cases}$$

- MET reconstruction in ***Delphes***

Missing Transverse Energy (MET) calculation based on the calorimetric towers:

$$\vec{E}_T^{\,miss} = -\sum_i^{towers} \vec{E}_T(i)$$

Limitations:

- dead channels, misalignment, noisy towers, cracks of the detector that worsen directly the MET not taken into account
- based on the calorimetric towers only

➡ muons not used to reconstruct MET but can be added by hand

S. Ovyn
X. Rouby

UCL
Université catholique de Louvain

Delphes

Trees in the output ROOT file

S. Ovyn
X. Rouby

Delphes.root

- GEN — Monte Carlo level information
  - Event
  - Particle
- Analysis — Analysis level information
  - CaloTower
  - Tracks
  - Electron
    - Charge
    - IsolFlag
    - PhiCalo
    - EtaCalo
  - Muon
    - Charge
    - IsolFlag
    - PhiCalo
    - EtaCalo
  - Jet
    - NTracks
    - Btag
  - TauJet
    - NTracks
    - Charge
  - ETmis
    - ET
    - Phi
    - Px
    - Py
  - Photon

Common leaves to Electron, Muon, Jet, TauJet and Photon branches

- E
- PT
- Phi
- Eta
- Px
- Py
- Pz
- EHoverEE

S. Ovyn
X. Rouby

Very forward detector information

S. Ovyn
X. Rouby

Near-beam components

Most of recent experiments in HE physics have additional instrumentation along the beamline

➡ In addition to the central detector, ***Delphes*** includes

- Forward detectors to extend the eta coverage to higher values

e.g. : Zero Degree Calorimeters

- (very) forward near-beam detectors

Central detector coverage

CMS tracking : $0 < |\eta| < 2.5$

CMS calorimetry : $0 < |\eta| < 5$

Very forward extensions

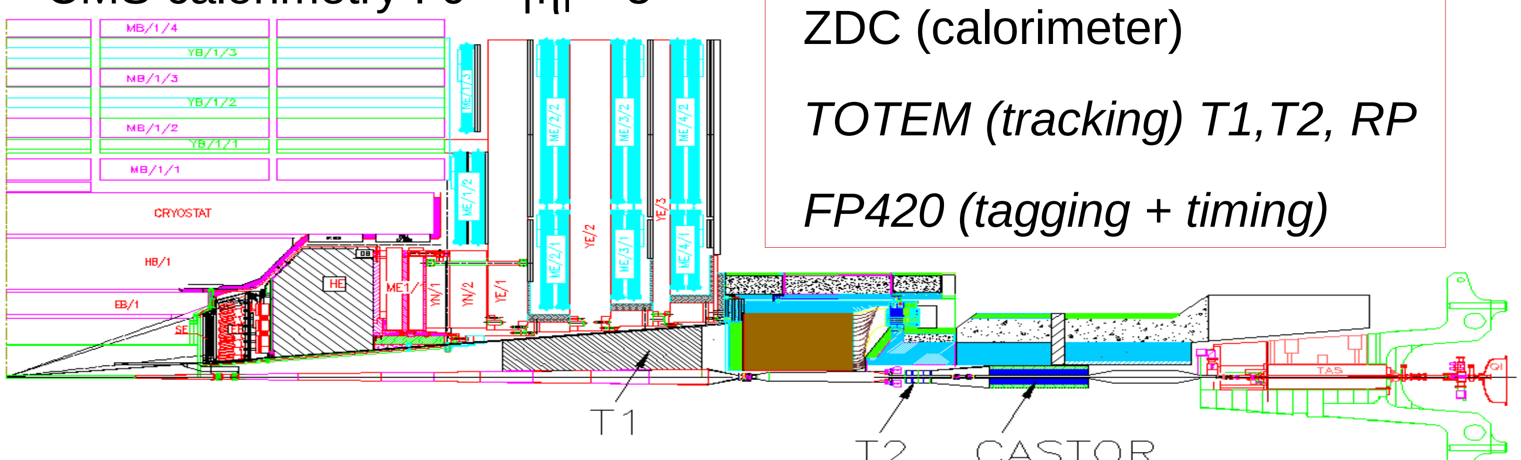
CASTOR (calorimeter)

Delphes

ZDC (calorimeter)

TOTEM (tracking) T1,T2, RP

FP420 (tagging + timing)



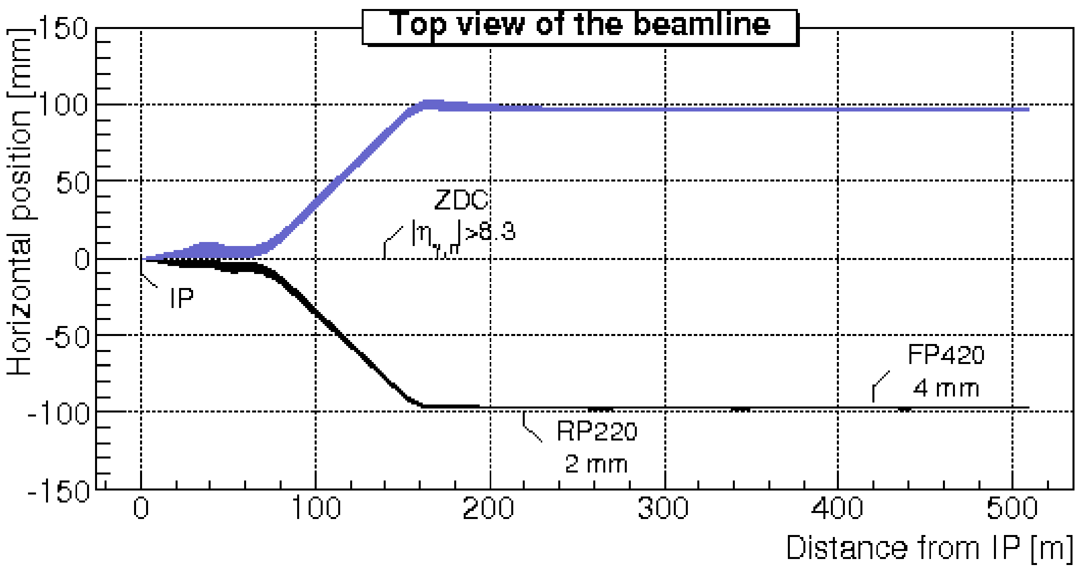
CMS & TOTEM Coll [CERN/LHCC 2006-039/G-124]

HECTOR implementation

S. Ovyn
X. Rouby

Delphes uses HECTOR to perform particle transport in beamlines

X. Rouby, J. de Favereau and K. Piotrzkowski, JINST 2(2007) P09005



Incoming beam

outgoing beam

<u>Input needed:</u>

- effective field strength / length
- magnetic position/aperture

→ data/LHCB1IR5_v6.500.tfs
data/LHCB2IR5_v6.500.tfs

Acceptance of the very forward and near-beam detectors are easily modified using the Detector card

HECTOR is not used for ZDC acceptance (direct extrapolation for neutrals).
Limitation: only neutrals in ZDC, HECTOR can currently not compute the charged particle contamination.

Trees in the output ROOT file

S. Ovyn
X. Rouby

Delphes.root

- **GEN** — Monte Carlo level information
  - Event
  - Particle
- **Analysis** — Analysis level information
  - CaloTower
  - Tracks
  - Electron
  - Muon
  - Jet
  - TauJet
  - ETmis
  - Photon
  - FP420hits
  - RP220hits
  - ZDChits
    - ZDChits.E
    - ZDChits.Eta
    - ZDChits.Phi
    - ZDChits.PT
    - ZDChits.Px
    - ZDChits.Py
    - ZDChits.Pz
    - ZDChits.T
    - ZDChits.pid
    - ZDChits.side

Common leaves to FP420hits and RP220hits branches

- E
- S
- T
- side
- Tx
- Ty
- X
- Y

S. Ovyn
X. Rouby

Validation

Validation: jet resolution

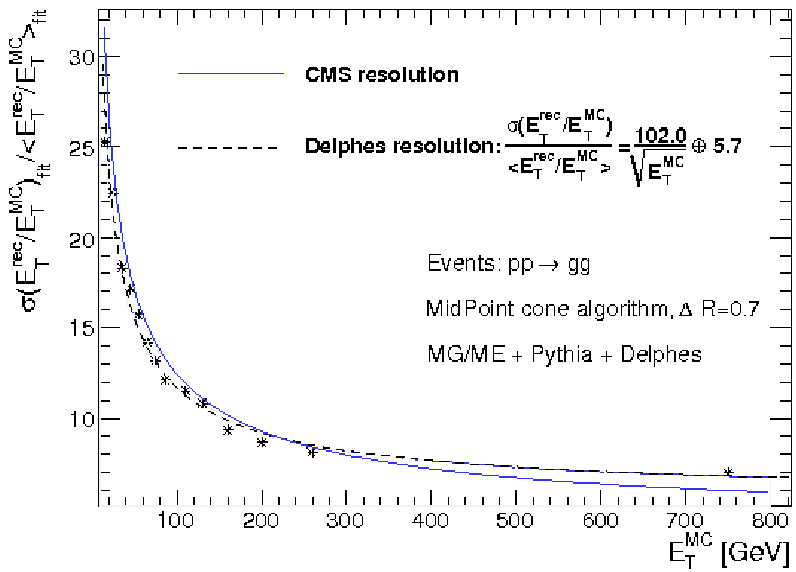
S. Ovyn
X. Rouby

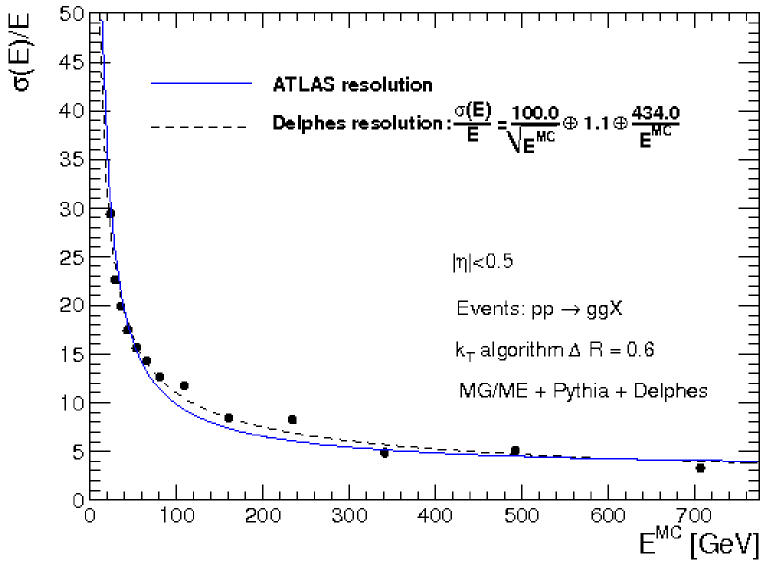
Validation procedures using CMS-like and ATLAS-like detectors

CMS resolution from: The CMS Collaboration, CERN/LHCC 2006-001.
ATLAS resolution from: The ATLAS Collaboration, CERN-OPEN 2008-020.

The majority of interesting processes contain jets in the final state.

➡ The jet resolution is therefore a crucial point





Used samples: pp→ggX, following the methods of the corresponding reference

An excellent agreement is obtained comparing values of ***Delphes*** with the expectations of CMS and ATLAS detectors

Validation: MET resolution

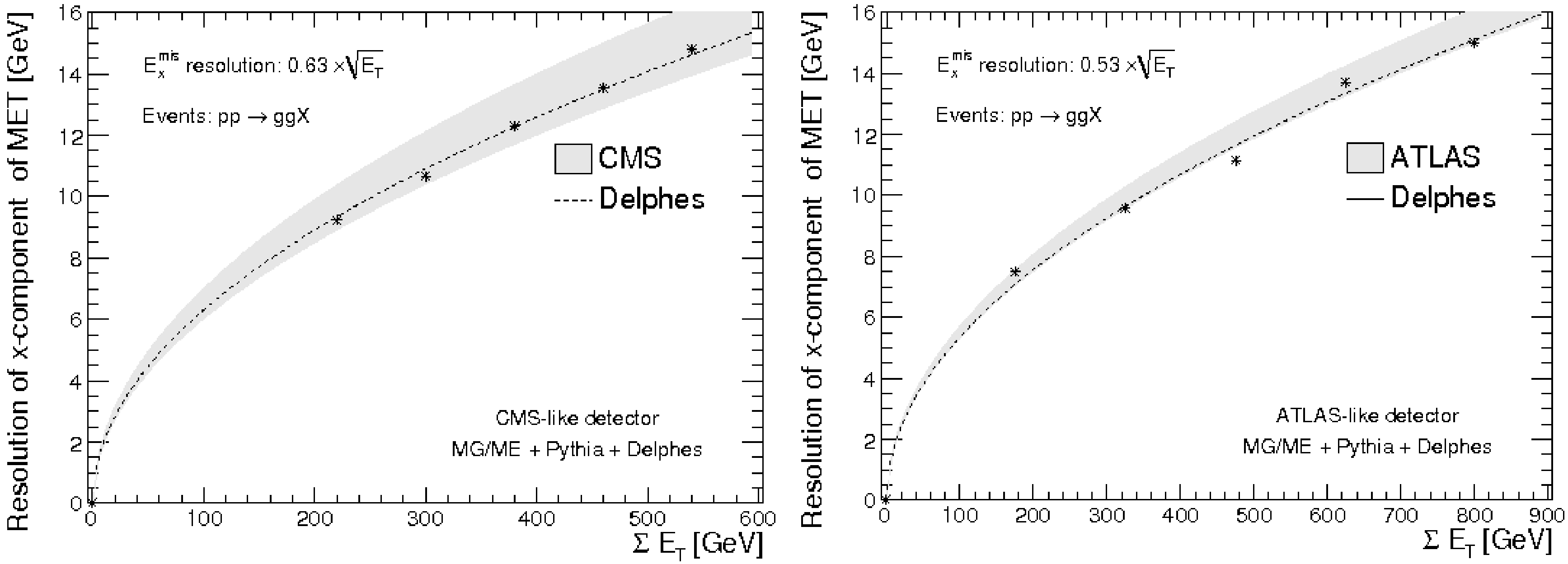
S. Ovyn
X. Rouby

Validation procedures using CMS-like and ATLAS-like detectors

CMS resolution from: The CMS Collaboration, CERN/LHCC 2006-001.
ATLAS resolution from: The ATLAS Collaboration, CERN-OPEN 2008-020.

HEP detectors designed to be as much hermetic as possible

➡ MET resolution is a crucial point



Used samples: pp→gg : muon contribution is negligible

An excellent agreement is obtained comparing values of ***Delphes*** with the expectations of CMS and ATLAS detectors

S. Ovyn
X. Rouby

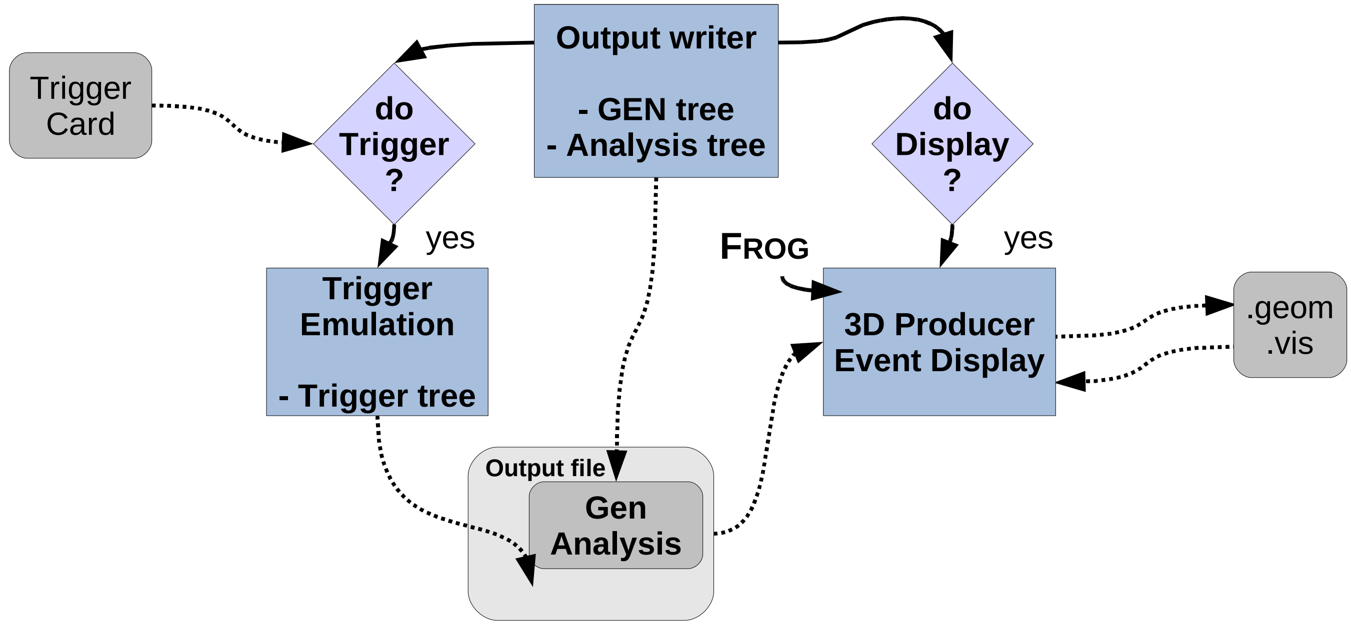
Additional features

UCL
Université catholique de Louvain

Delphes

Additional features

S. Ovyn
X. Rouby



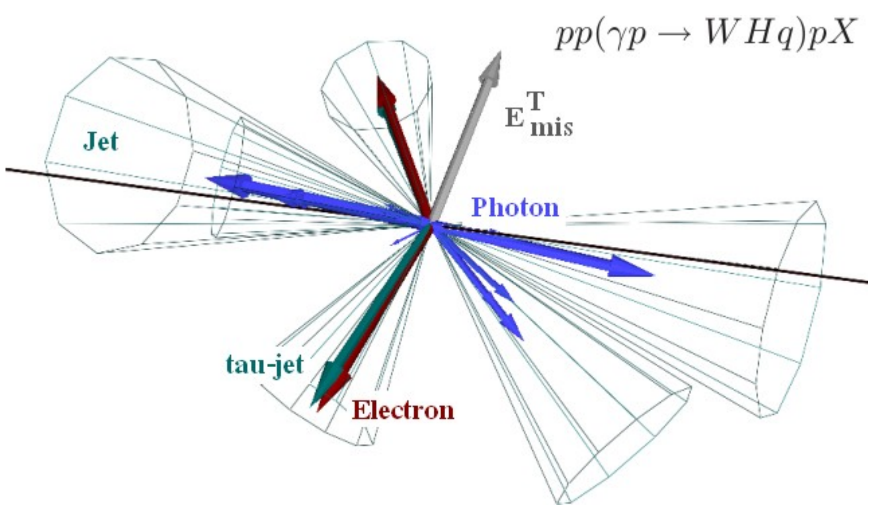
Trigger emulation

Application of user-defined trigger selection using the Trigger card

Result of the ***Delphes*** trigger selection is stored in the « Trigger tree » in the output root file

3D Event Display

FROG interfaced to ***Delphes***



FROG: L. Quertenmont, V. Roberfroid, arXiv:0901.2718v1[hep-ex]

Summary and Outlook

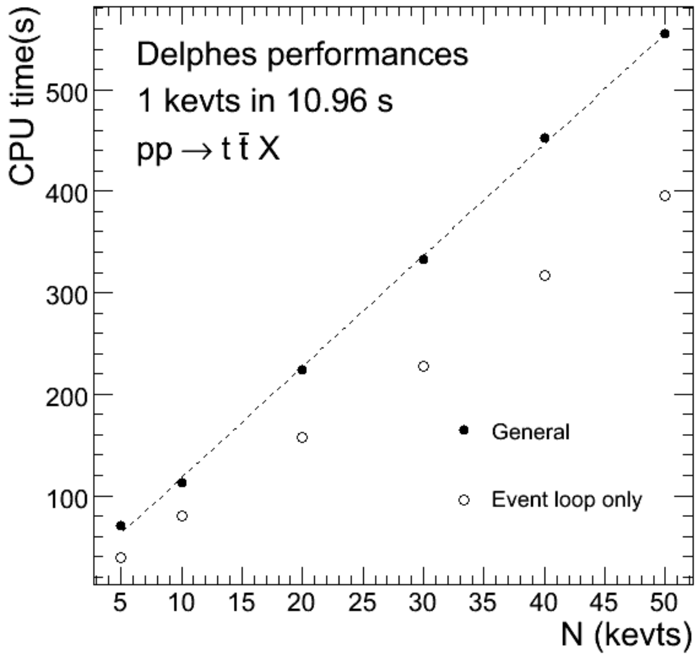
We present here a new framework for the fast simulation of a generic collider experiment

- Includes Trigger, forward near-beam detectors, 3D Event display

- Several input file types accepted by ***Delphes***
  - StdHEP
  - ROOT files
  - Les Houches Event Format
  - HepMC

- ***Delphes*** stores output information in
  - ASCII file of LHCO type
  - ROOT format

- ***Delphes*** performs a *fast* simulation:
  - 1 000 events, 10.96 s (regular laptop), 240 MB (physics dependent)



Can be used for fast evaluation of observability of new signals in phenomenology, as an illustration tool for tutorial sessions, ...

http://www.fynu.ucl.ac.be/delphes.html

S. Ovyn
X. Rouby

S. Ovyn
X. Rouby

Backup slides

Trigger emulation

- Trigger selection in a real experiment

New physics often characterised by low σ of new physics compared to values of Standard Model ones

➡ High statistics are required for data analyses ➡ high luminosity

BUT only a tiny fraction of the observed events can be stored for subsequent offline analyses,

↳ - large data rejection factor using dedicated algorithms
- Selection should be fast and very efficient

A trigger emulation is included in ***Delphes***, using a fully parametrisable trigger table

```
Inclusive electron >> ELEC1_PT:   '29'
di-electron        >> ELEC1_PT:   '17' && ELEC2_PT: '17'
```

- select events containing objects (i.e. jets, particles,met) with a p_T above some threshold.
- Logical combinations (AND) of several conditions are also possible.
- Default trigger tables available for ATLAS & CMS experiments

S. Ovyn
X. Rouby

Motivations
Simulation
BUT also...
Trigger
FROG
Convertors
Tutorial
Conclusion

3D Event Display

S. Ovyn
X. Rouby

FROG: L. Quertenmont, V. Roberfroid, arXiv:0901.2718v1[hep-ex]

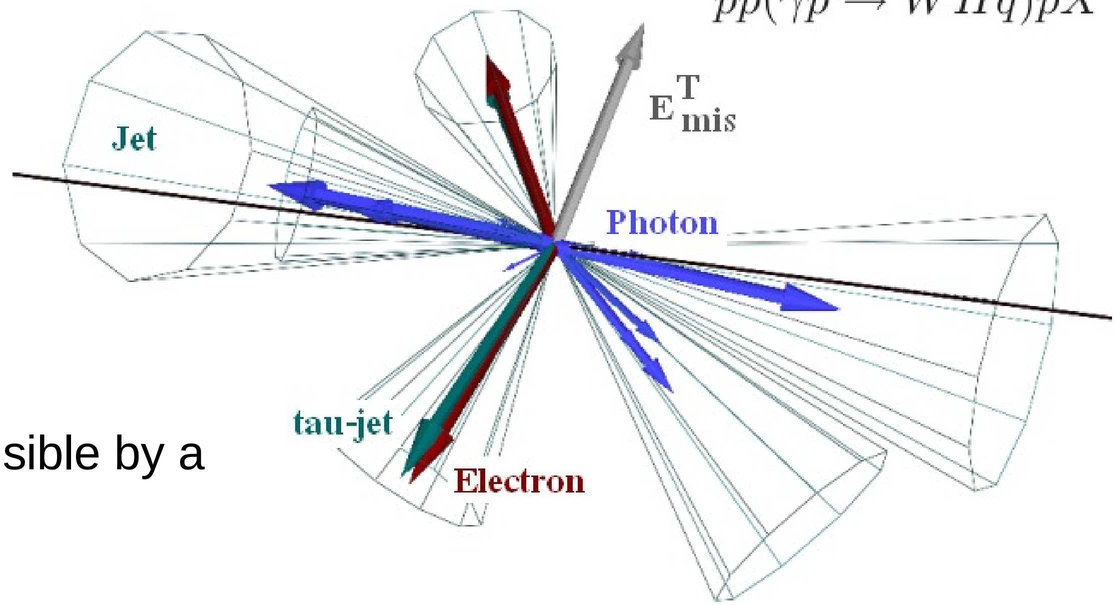
Visualisation is useful to convey information about the detector layout and the event topology in a simple way.

➡ The *Fast and Realistic OpenGL Displayer* FROG interfaced in ***Delphes***

$pp(\gamma p \rightarrow WHq)pX$

- Visibility of each objects ($e^{\pm}$, $\mu^{\pm}$, $\tau^{\pm}$, jets, MET) enhanced by a colour coding.

- Kinematics information is visible by a simple mouse action.



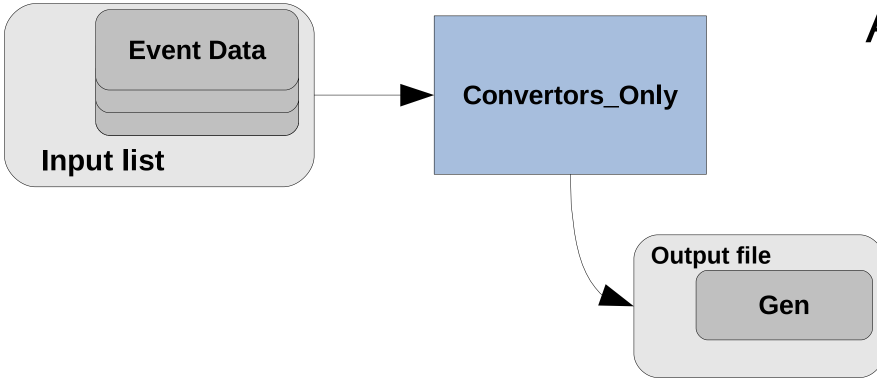
Utility of the event visualisation

- Deeper understanding of interesting physics processes

Additional features

S. Ovyn
X. Rouby

Convertor



All types of input files accepted by ***Delphes***

- StdHEP
- ROOT files
- Les Houches Event Format
- HepMC

Allow easy checks between various generators

GEN

Monte Carlo level information

- Event
  - Event.Number
- Particle
  - Particle.PT
  - Particle.E
  - Particle.Px
  - Particle.D1
  - Particle.D2
  - Particle.M1
  - Particle.M2
  - Particle.Status
  - Particle.PID

S. Ovyn
X. Rouby

Small tutorial...

Getting started : download

S. Ovyn
X. Rouby

Code download

from the website : « *download* » link

Delphes tar-ball is self-sufficient, it contains every dependencies needed for the physics.

or from a command line :

```
wget http://www.fynu.ucl.ac.be/users/s.ovyn/Delphes/files/Delphes_V_1.8.tar.gz
```

Requirements

A recent working ROOT version (http://root.cern.ch)

ROOT: R. Brun, F. Rademakers, NIM A 389 (1997) 81-86.

Delphes has been developped on ROOT > 5.18 on Linux with GNU gcc/g++ > 4.1.2, but any recent version should be fine.

Getting started : download

S. Ovyn
X. Rouby

Requirements

Checking ROOT installation

```
echo $ROOTSYS
```

If empty, check that the environment variables are defined.

In *bash* shell, check that these variables are in the .bashrc:

(e.g. assuming ROOT is in /usr/bin/root

```
export ROOTSYS=/usr/bin/root
export PATH=$PATH:$ROOTSYS/bin
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$ROOTSYS/lib
```

Test it :

```
root
```

If the FROG event display is to be run:

3D-OpenGL libraries are not included in the `tar.gz`, but required only if FROG is used. These libraries can be downloaded from here: http://curl.haxx.se/download.html

More on FROG requirements:

http://projects.hepforge.org/frog/index.php?page=Starting.php

Getting started : compile

S. Ovyn
X. Rouby

Motivations
Simulation
BUT also...
Tutorial
Conclusion

Untar – decompress the code sources

```
tar -xzf Delphes_V_1.8.tar.gz
```

Compile the sources

```
cd Delphes_V_1.8
./genMakefile.tcl > Makefile
make
>> Compiling tmp/Utilities/ExRootAnalysis/src/BlockClassesDict.cc
>> Compiling tmp/src/TreeClassesDict.cc
...
>> Building Analysis_Ex
Delphes has been compiled
Ready to run
```

Many lines are printed during the compilation.

In particular, the dependencies (like FastJet, mcfio, stdhep) lead to a few warning messages. This is normal and harmless.

Getting started : samples

S. Ovyn
X. Rouby

Input files from MC generator

Suggested samples for this introduction:

```
wget http://www.fynu.ucl.ac.be/users/s.ovyn/Delphes/files/tt_jj_small.hep.tar.gz
tar -xzf tt_jj_small.hep.tar.gz
mv samples/* .
```

These events are $\gamma p \to ttX$,

generated with MadGraph/MadEvent and

hadronised with Pythia

saved into StdHEP file format (*hep).

List of input files:

```
ls *list
TEST_small_tt.list
```

→

TEST_small_tt.list :

```
tt_jj_small.hep
```

text file containing one input data file per line

all data files must be of the same type

Getting started : run

Motivations
Simulation
BUT also...
Tutorial
Conclusion

S. Ovyn
X. Rouby

Running *Delphes* :

```
./Delphes
Usage: ./Delphes input_file output_file [detector_card] [trigger_card]
input_list - list of files in Ntpl, StdHep of LHEF format,
output_file - output file.
detector_card - Datacard containing resolution variables for the detector
  simulation (optional)
trigger_card - Datacard containing the trigger algorithms (optional)
```

Main things needed:

input_list : e.g. TEST_small_tt.list — List of input MC files

output_file : e.g. test.root — Output ROOT filename

Some options:

detector_card : e.g. data/DetectorCard_CMS.dat — Detector parameters

trigger_card : e.g. data/TriggerCard_CMS.dat — Trigger definitions

Try it:

```
./Delphes  TEST_small_tt.list  test.root
```

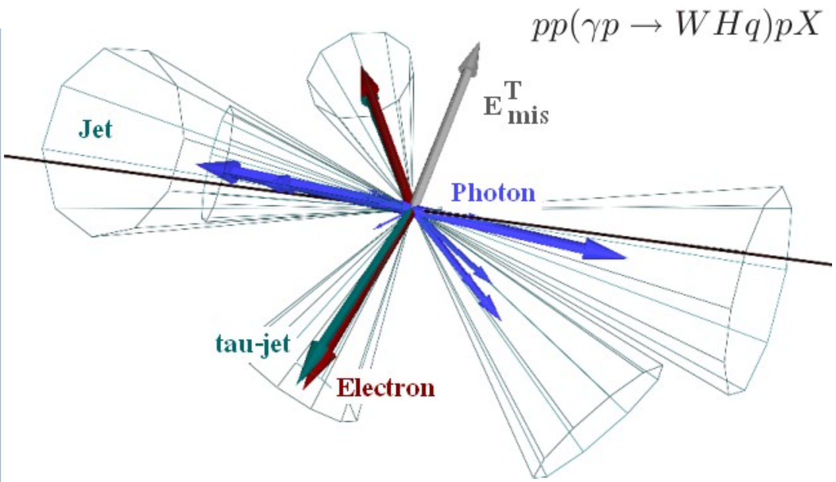
FROG

S. Ovyn
X. Rouby

Motivations
Simulation
BUT also...
Tutorial
Conclusion

3D Event Display

FROG interfaced to ***Delphes***



FROG: L. Quertenmont, V. Roberfroid,
arXiv:0901.2718v1[hep-ex]

To run FROG, some libraries are needed, as it uses the OpenGL free libraries, which are not in Delphes

```
OpenGL:
    xorg-x11-Mesa-libGL-6.8.2-1.EL.33.0.2
    xorg-x11-Mesa-libGLU-6.8.2-1.EL.33.0.2
GLUT:
    freeglut-2.2.0-14
    freeglut-devel-2.2.0-14
X11-devel:
    xorg-x11-devel-6.8.2-1.EL.33.0.2
CURL:
    libcurl
```

FROG runs on two files : *geom and *vis, which are created if the flag is ON

```
FLAG_frog          1   //1 to run the FROG event display
```

Then compile the source (NOT done automatically when Delphes is compiled)

```
cd Utilities/FROG
make
...
cd ../..
./Utilities/FROG/frog
```

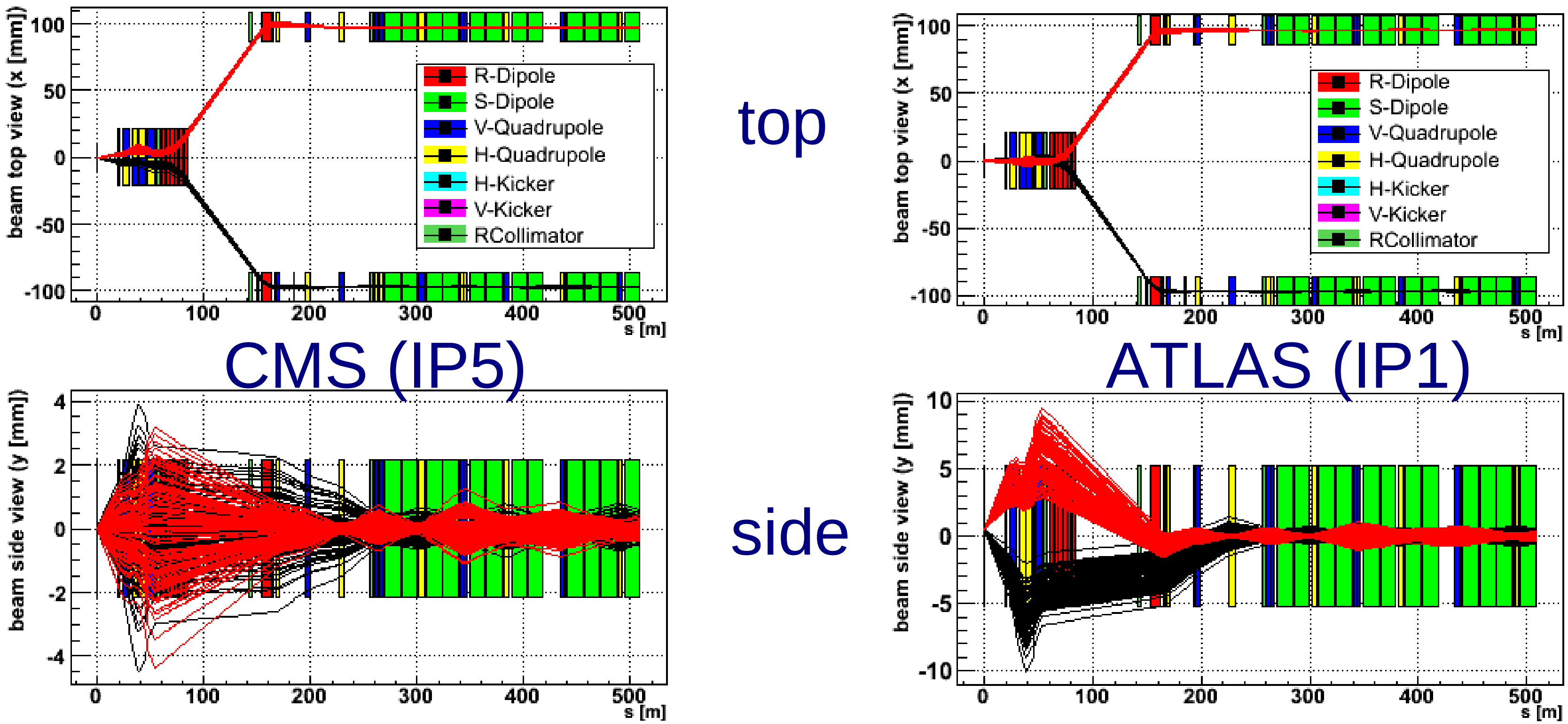
Compile it...

...Run it

HECTOR implementation

- ***Delphes*** uses HECTOR to perform particle transport in beamlines

X. Rouby, J. de Favereau and K. Piotrzkowski, JINST 2(2007) P09005



Horizontal crossing plane

Vertical crossing plane

Input needed:

- effective field strength / length
- magnetic position/aperture

→ data/LHCB1IR5_v6.500.tfs
data/LHCB2IR5_v6.500.tfs

Acceptance of the very forward and near-beam detectors are easily modified using the Detector card

S. Ovyn
X. Rouby

Validation: jet resolution

Validation procedures using CMS-like detector parameters

CMS resolution from: The CMS Collaboration, CERN/LHCC 2006-001.

S. Ovyn
X. Rouby

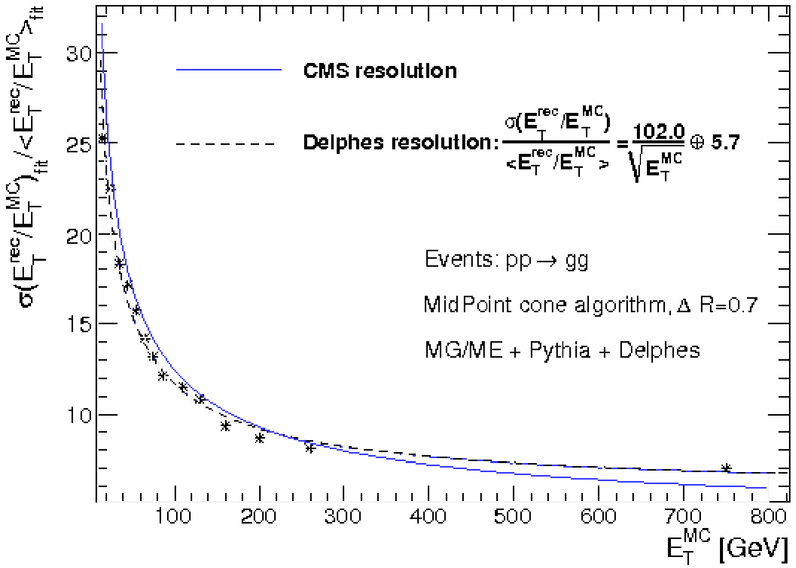
The majority of interesting processes contain jets in the final state.

➡ The jet resolution is therefore a crucial point

Sample used: pp→gg

- Arranged in 14 bins of gluon p_T .
- In each p_T bin, Delphes jets are matched to the closest GEN jet using

$$\Delta R = \sqrt{(\eta^{rec} - \eta^{MC})^2 + (\phi^{rec} - \phi^{MC})^2} < 0.25$$



- E_T^{rec}/E_T^{MC} histograms fitted with a Gaussian distribution in the interval ±2 rms centred around the mean value. The resolution in each p_T bin is obtained by

$$\frac{\sigma\left(\frac{E_T^{rec}}{E_T^{MC}}\right)_{fit}}{\left\langle \frac{E_T^{rec}}{E_T^{MC}} \right\rangle_{fit}} (\hat{p}_T(i))$$

An excellent agreement is obtained comparing values of ***Delphes*** with the expectations of the general purpose CMS detector