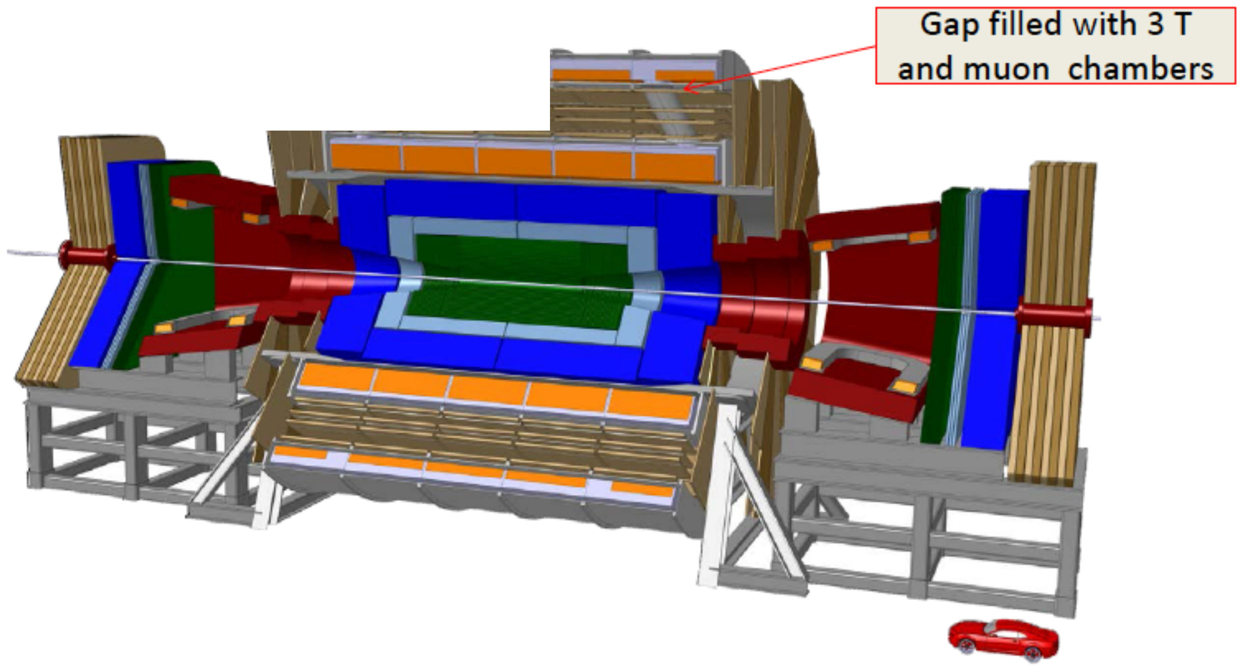
FCC-hh

Towards a DELPHES description for FCC

Heather Gray, Filip Moortgat (CERN)



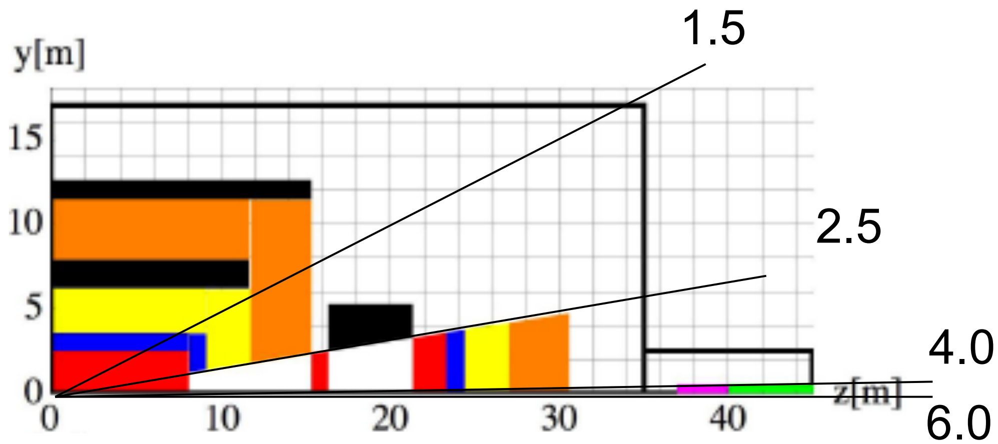
Introduction

- DELPHES seems the most appropriate simulation package for generic FCC-hh physics studies at the moment.
- FCC software framework will include DELPHES interface, plus other detector simulation options (see talk by Benedikt later)
- During the last meeting we made a proposal for a DELPHES implementation of an generic FCC detector
- Received useful comments from Daniel, Paolo, Andre, Sergei, Ana and Mike.
- Most of the suggestions have been incorporated in the revised card shown today.

Detector geometry

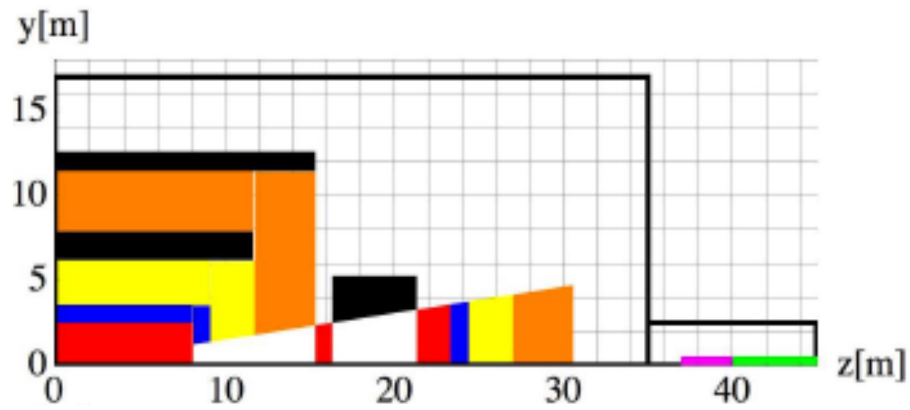
4 regions in eta:

- Barrel: 0 < eta < 1.5
- Endcap: 1.5 < eta < 2.5
- Forward: 2.5 < eta < 4.0
- Very forward: 4.0 < eta < 6.0



Magnet choice

First choice: pick Twin Solenoid option as baseline



Twin Solenoid
+ Dipole ?

Delphes card:

```
# radius of the magnetic field coverage, in m
set Radius 6.0

# half-length of the magnetic field coverage, in m
set HalfLength 11.5

# magnetic field
set Bz 6.0
```

Note: dipole field not yet implemented in Delphes. Will follow later.
If this feature is critical for you, you probably need a more detailed simulation anyway.

Charged hadron efficiency

Reference: http://indico.cern.ch/event/340703/session/80/contribution/167/material/slides/1.pdf

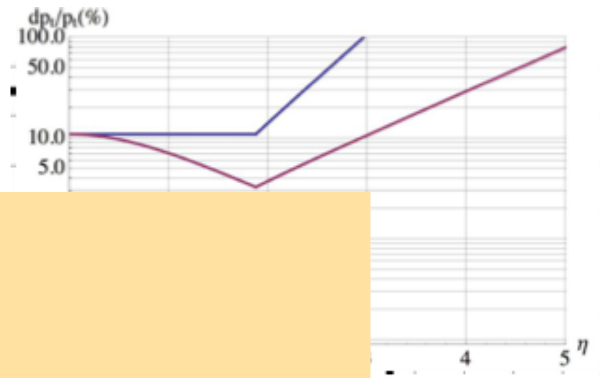
```
set EfficiencyFormula {                                (pt <= 0.5)   * (0.00) + \

                     (abs(eta) <= 1.5) * (pt > 0.5   && pt <= 1.0)   * (0.90) + \

                                                                       5) + \

}
```

See Zbynek's talk

- Eta coverage to 4 or 6?
- Minimal pT cut: 0.5 GeV
- Material distribution will drive inefficiency

Charged hadron resolution

```
set ResolutionFormula {             (abs(eta) <= 1.5) * (pt > 0.5   && pt <= 1.0)   * (0.02) + \

                          (abs(eta) <= 1.5) * (pt > 1.0   && pt <= 1.0e1) * (0.01) + \

                          (abs(eta) <= 1.5) * (pt > 1.0e1 && pt <= 2.0e2) * (0.03) + \


                 (abs(eta) >

                 (abs(eta) >

                 (abs(eta) >

                 (abs(eta) >
```



See Zbynek's talk

η: 0-1.5; 1.5-2.5; 2.5-4; 4-6

p_T: 0.5, 1, 10, 20, > 20

$$\frac{\sigma(p_T)}{p_T} = \frac{\sigma_x \cdot p_T}{0.3BL^2}\sqrt{\frac{720}{(N+4)}}$$

Calorimeters

Simple calorimeter (no longitudinal segmentation at the moment):

ECAL: granularity : 0.0125 x 0.0125 for eta<2.5,
0.025 x 0.025 for eta<4.0,
0.05 x 0.05 for eta<6.0

```
# set ECalResolutionFormula {resolution formula as a function of eta and energy}
  set ResolutionFormula {       (abs(eta) <= 4.0) * sqrt(energy^2*0.01^2 + energy*0.10^2) + \
                         (abs(eta) > 4.0 && abs(eta) <= 6.0) * sqrt(energy^2*0.01^2  + energy*0.10^2)}
```

i.e. stochastic term: 10%, constant term: 1%

HCAL: granularity : 0.05 x 0.05 for eta<2.5,
0.1 x 0.1 for eta<4.0,
0.2 x 0.2 for eta<6.0

```
# set HCalResolutionFormula {resolution formula as a function of eta and energy}
  set ResolutionFormula {        (abs(eta) <= 4.0) * sqrt(energy^2*0.03^2 + energy*0.50^2) + \
                         (abs(eta) > 4.0 && abs(eta) <= 6.0) * sqrt(energy^2*0.05^2 + energy*1.00^2)}
```

i.e. stochastic term: 50% (100%), constant term: 3% (5%)

Electrons

```
# tracking efficiency formula for electrons
set EfficiencyFormula { (pt <= 10) * (0.00) + \
(abs(eta) <= 1.5) * (pt > 10 && pt <= 50) * (0.80) + \
(abs(eta) <= 1.5) * (pt > 50) * (0.90) + \
(abs(eta) > 1.5 && abs(eta) <= 4) * (pt > 10 && pt <= 50) * (0.80) + \
(abs(eta) > 1.5 && abs(eta) <= 4) * (pt > 50) * (0.90) + \
(abs(eta) > 4 && abs(eta) <= 6) * (pt > 10 && pt <= 50) * (0.70) + \
(abs(eta) > 4 && abs(eta) <= 6) * (pt > 50) * (0.80) + \
```

Identification efficiency included

Track-based isolation
PTMin 0.5

```
# resolution formula for electrons
set ResolutionFormula {
(abs(eta) <= 1.5) * sqrt(energy^2*0.01^2 + energy*0.10^2) + \
(abs(eta) > 1.5 && abs(eta) <= 2.5) * sqrt(energy^2*0.01^2 + energy*0.10^2) + \
(abs(eta) > 2.5 && abs(eta) <= 4.0) * sqrt(energy^2*0.01^2 + energy*0.10^2) + \
(abs(eta) > 4.0 && abs(eta) <= 6.0) *  sqrt(energy^2*0.01^2  + energy*0.10^2)}
```

= ECAL resolution

Muons

```
set EfficiencyFormula {          (pt <= 5)   * (0.00) + \
  (abs(eta) <= 1.5) * (pt > 5 )   * (0.99) + \
  (abs(eta) > 1.5 && abs(eta) <= 4.0) * (pt > 5)          * (0.99) + \
  (abs(eta) > 4.0)                              * (0.00)}
```

99% efficiency for
Pt > 5 GeV and eta < 4.0

```
set ResolutionFormula {
  (abs(eta) <= 0.5) * (pt > 5.0   && pt <= 1.0e2) * (0.015) + \
  (abs(eta) <= 0.5) * (pt > 1.0e2 && pt <= 2.0e2) * (0.03) + \
  (abs(eta) <= 0.5) * (pt > 2.0e2)              * (0.05 + pt*1.e-4) + \
  (abs(eta) > 0.5 && abs(eta) <= 1.5) * (pt > 5.0   && pt <= 1.0e2) * (0.02) + \
  (abs(eta) > 0.5 && abs(eta) <= 1.5) * (pt > 1.0e2 && pt <= 2.0e2) * (0.04) + \
  (abs(eta) > 0.5 && abs(eta) <= 1.5) * (pt > 2.0e2)              * (0.05 + pt*1.e-4) + \
  (abs(eta) > 1.5 && abs(eta) <= 4.0) * (pt > 5.0   && pt <= 1.0e2) * (0.035) + \
  (abs(eta) > 1.5 && abs(eta) <= 4.0) * (pt > 1.0e2 && pt <= 2.0e2) * (0.05) + \
  (abs(eta) > 1.5 && abs(eta) <= 4.0) * (pt > 2.0e2)              * (0.05 + pt*1.e-4)}
}
```

Photons

```
set EfficiencyFormula {      (pt <= 10.0) * (0.00) + \
(abs(eta) <= 1.5) * (pt > 10.0)  * (0.95) + \
(abs(eta) > 1.5 && abs(eta) <= 4.0) * (pt > 10.0)  * (0.90) + \
(abs(eta) > 4.0 && abs(eta) <= 6.0) * (pt > 10.0)  * (0.80) + \
(abs(eta) > 6.0) * (0.00)   }
```

Resolution & isolation : same as electrons.

Jet & MET

Then:

jets:
- default Anti-Kt (FastJet) JetAlgorithm 7 (includes substructure variables)
- using Eflow objects as input
- default cone 0.4
- default PT>30 GeV

MET = negative vector sum of Eflow objects

Note: no PU at the moment. PU configurations can be foreseen, but requires a separate card. Coming.

B-tagging

- Suggest using parametrised efficiency and fake rate
- set DeltaR 0.4
- set BPTMin 20.0
- set BEtaMax 4.0
- add EfficiencyFormula {1,2,3} {0.001}
- add EfficiencyFormula {4} {0.05}
- add EfficiencyFormula {5} {0.70}

Efficiency for b's: 70%, charm: 5%, light quarks: 0.1%

Added a separate tune for charm

add EfficiencyFormula {1,2,3} {0.01}

add EfficiencyFormula {5} {0.10}

add EfficiencyFormula {4} {0.25}

Efficiency for charm: 25%, b's: 10%, light quarks: 1%

Tau-tagging

Parametrized description (40% efficiency, 0.1% fake rate, 0.5% electron fakes)

```
module TauTagging TauTagging {
  set ParticleInputArray Delphes/allParticles
  set PartonInputArray Delphes/partons
  set JetInputArray JetEnergyScale/jets

  set DeltaR 0.4

  set TauPTMin 10.0

  set TauEtaMax 4.0

  # add EfficiencyFormula {abs(PDG code)} {efficiency formula as a function of eta and pt}

  # default efficiency formula (misidentification rate)
  add EfficiencyFormula {0} {0.001}
  add EfficiencyFormula {11} {0.005}
  # efficiency formula for tau-jets
  add EfficiencyFormula {15} {0.4}
}
```

No muon fake rate at the moment

BACKUP

FCC event database

Clement Helsens has set up a database that collects FCC samples:

https://test-fcc.web.cern.ch/test-FCC/FCCevents.php

It contains two types of samples:

- LHE files
- generated events in FCC event data format

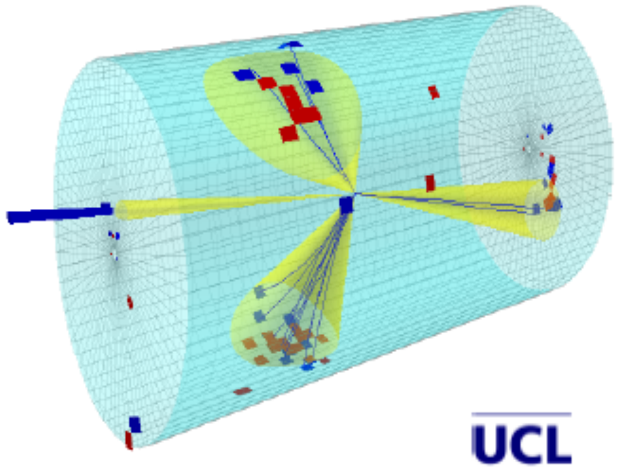
	name	nevents	nfiles	outputdir	mainprocess	finalstates
1	B-4p-0-1	91000	10	/eos/fcc/hh/generation/snowmass/FCCEvents/v0_0/B_4p/B-4p-0-1_100TEV/	vector boson + jets	V+0J
2	BB-4p-0-300	55000	10	/eos/fcc/hh/generation/snowmass/FCCEvents/v0_0/BB_4p/BB-4p-0-300_100TEV/	divector boson + jets	V+nJ
3	BB-4p-1400-2900	55000	10	/eos/fcc/hh/generation/snowmass/FCCEvents/v0_0/BB_4p/BB-4p-1400-2900_100TEV/	divector boson + jets	V+nJ
4	BB-4p-2900-5300	46177	10	/eos/fcc/hh/generation/snowmass/FCCEvents/v0_0/BB_4p/BB-4p-2900-5300_100TEV/	divector boson + jets	V+nJ
5	BB-4p-300-1400	55000	10	/eos/fcc/hh/generation/snowmass/FCCEvents/v0_0/BB_4p/BB-4p-300-1400_100TEV/	divector boson + jets	V+nJ
6	BB-4p-5300-8800	31498	10	/eos/fcc/hh/generation/snowmass/FCCEvents/v0_0/BB_4p/BB-4p-5300-8800_100TEV/	divector boson + jets	V+nJ
7	BB-4p-8800-100000	9392	10	/eos/fcc/hh/generation/snowmass/FCCEvents/v0_0/BB_4p/BB-4p-8800-100000_100TEV/	divector boson + jets	V+nJ
8	BBB-4p-0-1200	55000	10	/eos/fcc/hh/generation/snowmass/FCCEvents/v0_0/BBB_4p/BBB-4p-0-1200_100TEV/	tri-vector + jets, Higgs associated + jets	(VVV+nJ),(

Please help in the validation and provide feedback!

DELPHES

- **Delphes** is a **modular framework** that simulates of the response of a multipurpose detector in a **parameterized** fashion

- **Includes**:
  - pile-up
  - charged particle **propagation** in magnetic field
  - electromagnetic and hadronic **calorimeters**
  - **muon** system



- **Provides**:
  - leptons (electrons and muons)
  - photons
  - jets and missing transverse energy (particle-flow)
  - taus and b's

- Website and manual: https://cp3.irmp.ucl.ac.be/projects/delphes

- Paper: JHEP 02 (2014) 057

M. Selvaggi,
A. Mertens,
P. Demin

Ingredients

What do we need for a DELPHES description of an FCC detector?

- basic geometry
  - radius of tracking system (before the magnet or calorimetry)
  - length of magnetic field coverage
  - magnetic filed intensity
  - tracking coverage in eta/theta
  - eta/phi granularity of hcal and ecal
- momentum resolution formula for charged tracks
- energy resolution for electrons and photons
- momentum resolution for muons
- impact parameters resolution (optional)
- identification and mis-identification efficiency for particles: muons, electron, pions, kaons, ...
- neutral hadron energy fraction lost in hcal and ecal (sum =1)
- energy resolution formula for jets (optional)
- b-tag efficiency (optional)

Twin Solenoid design

"System"	"ATLAS"	"CMS"	"Twin Sol0"
"DrTracker"	1.23	1.29	2.5
"DrCoil0"	0.05	0	0
"DrEcal"	1	0.48	1.1
"DrHcal"	1.97	1.23	2.6
"DrCoil"	0	0.312	1.73
"DrMuon"	6	4.2	3.57
"DrCoil2"	0	0	1.1
"LTracker"	2.8	2.9	8
"LEecal"	1	0.98	1.1
"LEhcal"	2.6	1.8	2.6
"LEmuon"	17	5.18	3.57
"LUtracker"	0	0	0
"LDipole"	0	0	0
"DrDipole"	0	0	0
"LDtracker"	0	0	0
"LFecal"	0	0	0
"LFhecal"	0	4	0
"LFmuon"	0	0	0
"EtaForward"	10	3	10
"Lstar"	23	23	40
"LTAS"	3	3	3
"Lcavern"	23.5	23.5	35
"Rcavern"	15	15	17
"Rtunnel"	2	2	2.5
"Rtriplet"	0.5	0.5	0.5
"zmax"	45	45	45
"ymax"	18	18	18
"eps"	0.1	0.1	0.1
"BBarrel"	2	3.8	6
"SigBarrel"	20*10^(-6)	30*10^(-6)	20*10^(-6)
"NBarrel"	15	15	15
"ForwardBFlag"	False	False	False
"ToroidFlag"	False	False	False

Jets/MET

Two options:

- A simple smearing of jet and MET (made up from genparticles)
- Or an implementation of a simple calorimeter
    - granularity
    - resolution
    - fraction of energy deposit per particle

Output: towers (full deposit) or EflowTowers (full deposit minus charged deposit)

First option: - easy to change the jet resolution
- but no way to study jet substructure, PU mitigation, …

Second option: - can also do substructure, PU mitigation, …
- but less direct to change resolution