Fast-Simulation and Digitization in CMS

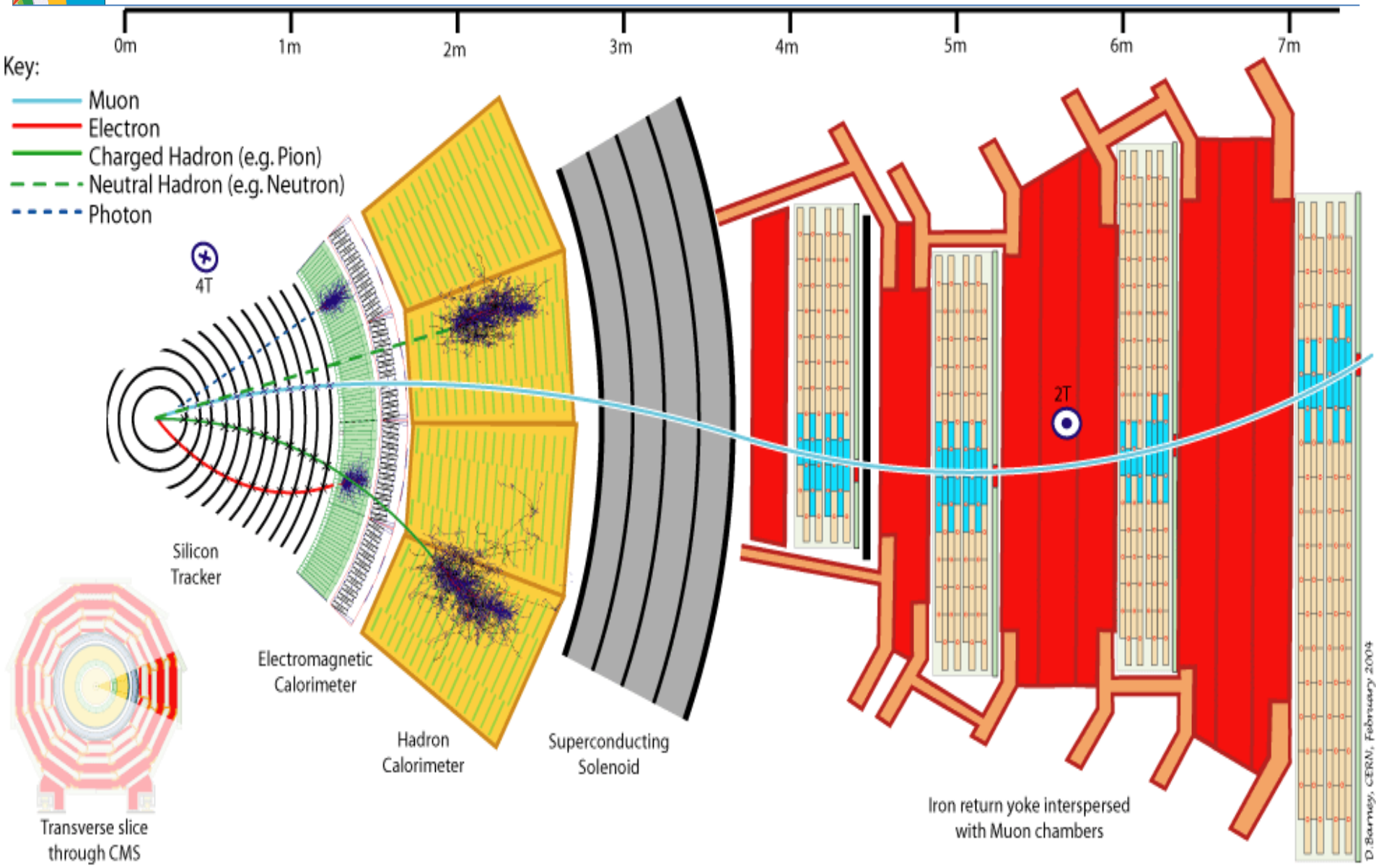
Federica Primavera
University of Bologna & INFN (IT)
on behalf of the CMS collaboration

Fast Detector Simulation in High Energy Physics
15-17 Jan 2013, DESY-Zeuthen

Outline

- Introduction to CMS
- Fast Simulation:
  - Overview
  - Why do we need digitization?
  - Muon system
  - Calorimeters
  - Plots of the performances
- Summary

CMS detector



Fast-Simulation in CMS

The Fast Simulation of CMS is an object-oriented subsystem of the general CMS C++ based software.

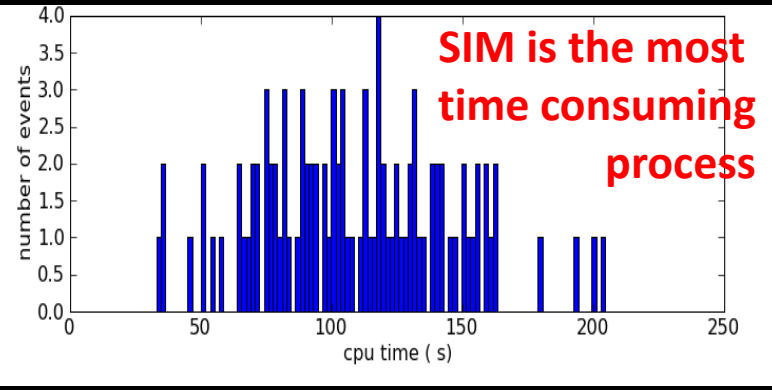
The Fast Simulation produces the same data format as the Geant4-based Full-Simulation.

Goal:
event production rates are of the order of 100 times faster than the corresponding Full Simulation one with nonetheless comparable accuracy for most of the physics objects typically considered in the analyses.

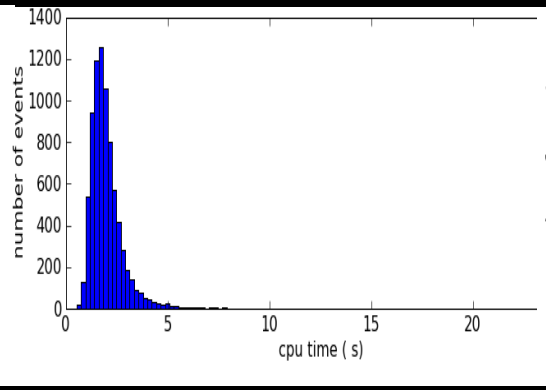
CPU time: FullSim vs FastSim

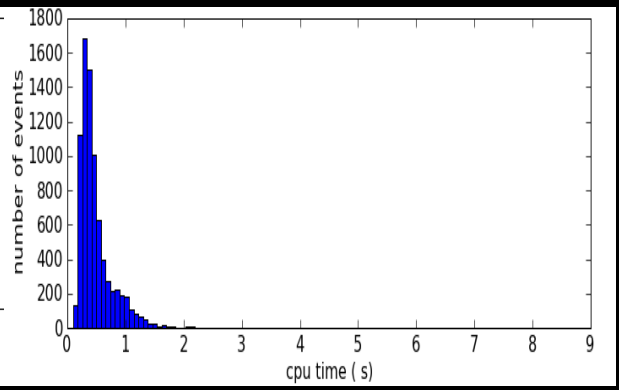
TTbar events **without Pile-Up** processed by – **slc5_amd64_gcc472 – 8 cores**

GEN, SIM,DIGI,L1,DIGI2RAW | RAW2DIGI,RECO | HLT

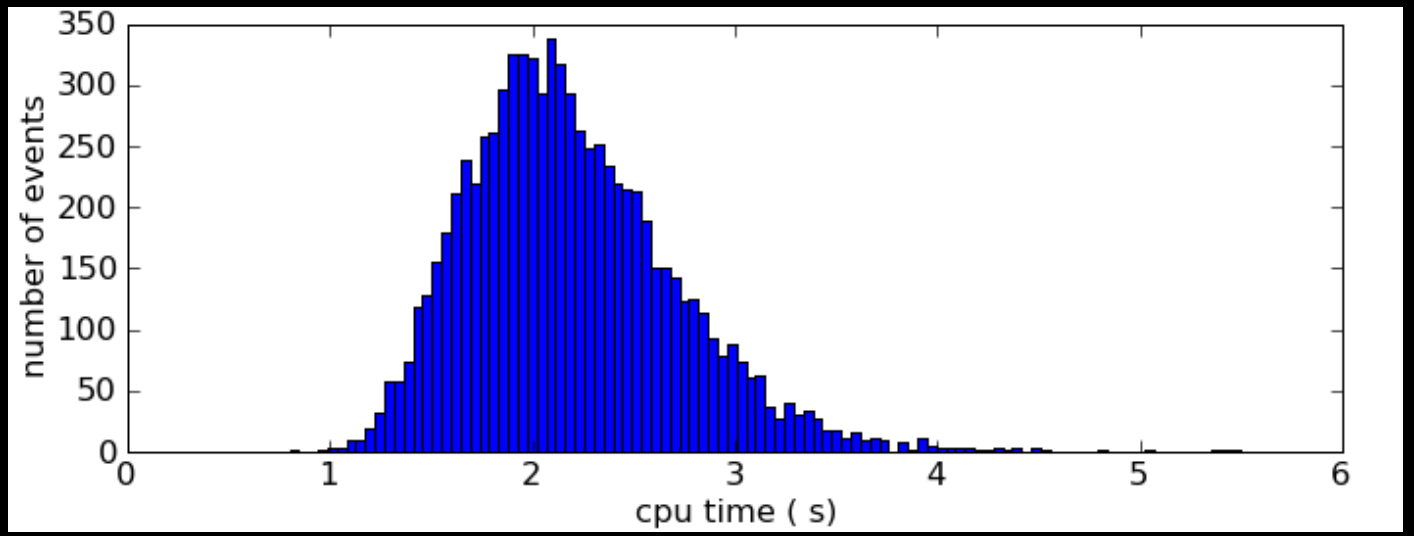


SIM is the most time consuming process



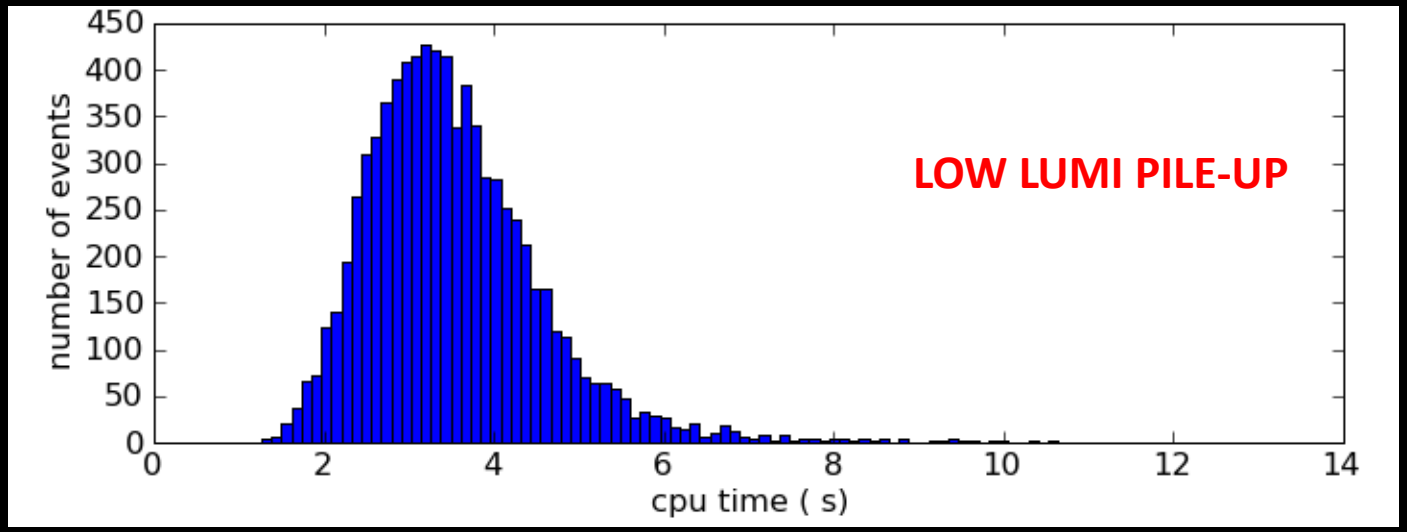


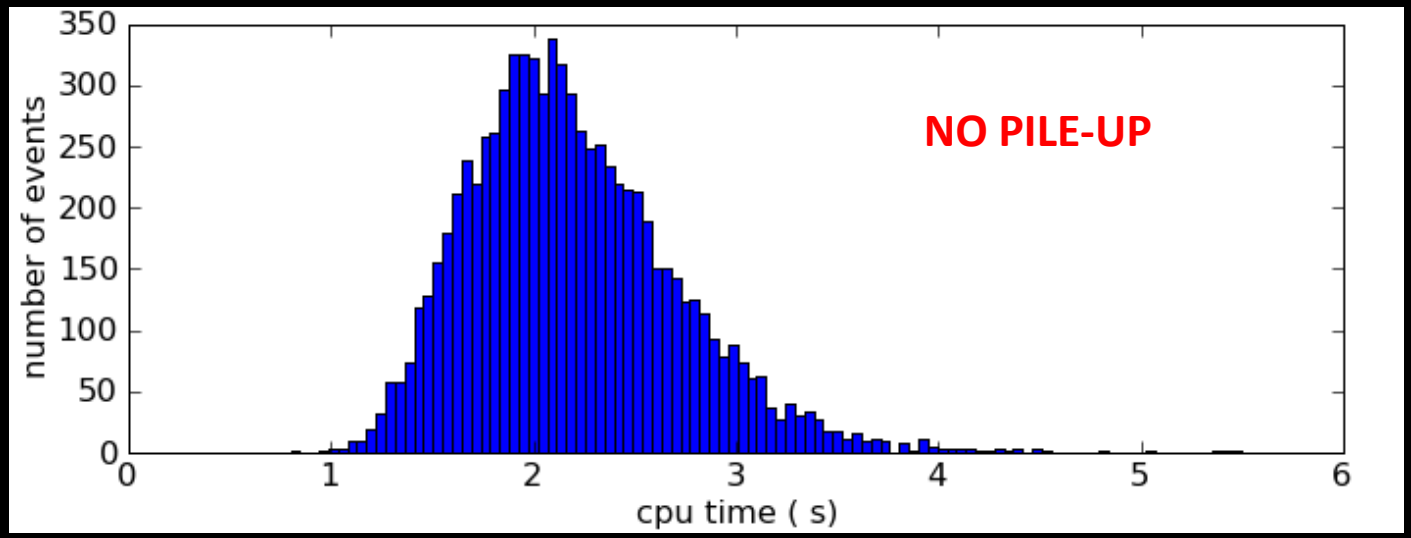
GEN, FASTSIM, HLT



CPU time: FastNoPU vs FastLowPU

TTbar events processed by – **slc5_amd64_gcc472 – 8 cores** GEN, FASTSIM,HLT





Fast-Simulation in CMS

The Fast Simulation of CMS is an object-oriented subsystem of the general CMS C++ based software.

The Fast Simulation produces the same data format as the Geant4-based Full-Simulation.

Goal:
event production rates are of the order of 100 times faster than the corresponding Full Simulation one with nonetheless comparable accuracy for most of the physics objects typically considered in the analyses.

How?

SIMULATION:
dedicated algorithm and simplified tracker geometry

RECONSTRUCTION (more time consuming):
is the same as the full simulation, but **may resort to some parameterizations** to be faster as in case of calorimeter response.

Fast Simulation workflow

GENERATED PARTICLES → SIMULATED PARTICLES

PYTHIA

PILE-UP PARTICLES:

- **in time** – overlapped collisions at the same bunch crossing;
- **out of time** – overlapped collisions at different bunch crossing.

Pile-Up mixing

GENERATED PARTICLES → SIMULATED PARTICLES

PYTHIA

PILE-UP PARTICLES:

- **in time** – overlapped collisions at the same bunch crossing;
- **out of time** – overlapped collisions at different bunch crossing.

- **in time** – the mix is done at particle level, and the vertices from the different events are taken into account only during the reconstruction step

- **out of time (OOT)** – the mix has to be done at event generator level, and particles have to contain the information about the production bunch crossing.

Fast Simulation workflow

GENERATED PARTICLES → SIMULATED PARTICLES

PYTHIA

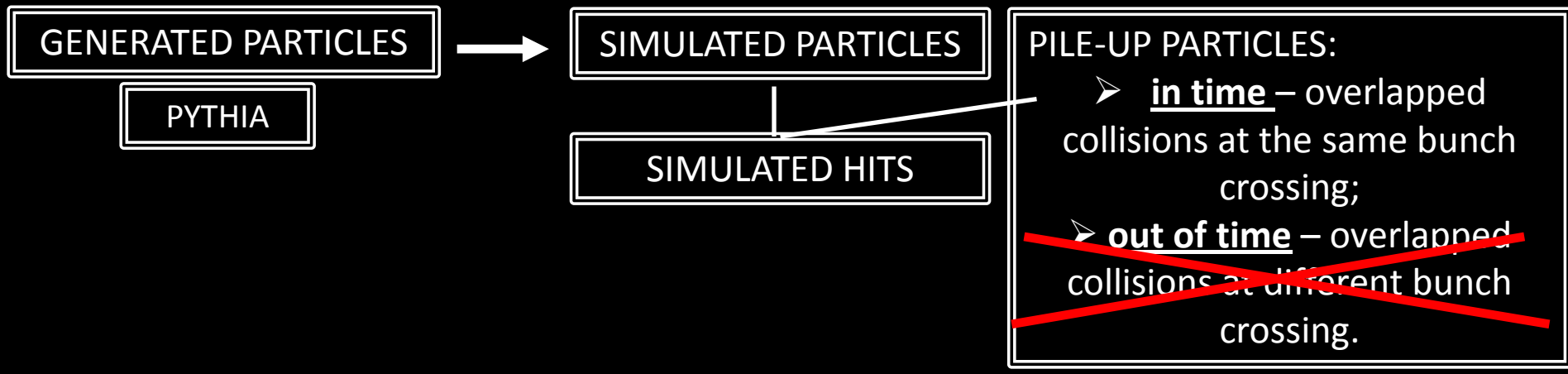
PILE-UP PARTICLES:

- **in time** – overlapped collisions at the same bunch crossing;
- ~~**out of time** – overlapped collisions at different bunch crossing.~~

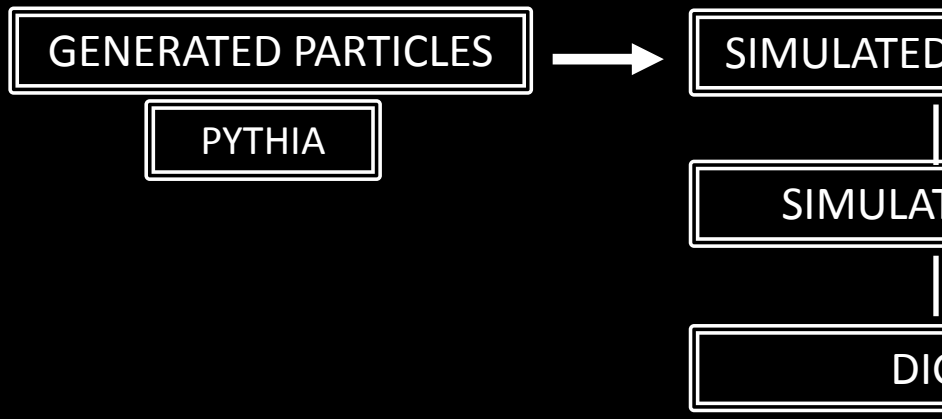
NOT YET IN FAST SIMULATION

WORK IN PROGRESS AIMING AT THE NEXT MAJOR RELEASE

Fast Simulation workflow



Fast Simulation workflow



PILE-UP PARTICLES:

- **in time** – overlapped collisions at the same bunch crossing;
- ~~**out of time** – overlapped collisions at different bunch crossing.~~

Digitization

The digitization corresponds to the process of simulating the electronics signal read out by the detector. The software is subdivided into 3 domains:
SimTracker, **SimCalorimetry** and **SimMuon.**

For each one is simulated:

the **Analog to Digital Converter (ADC)** module

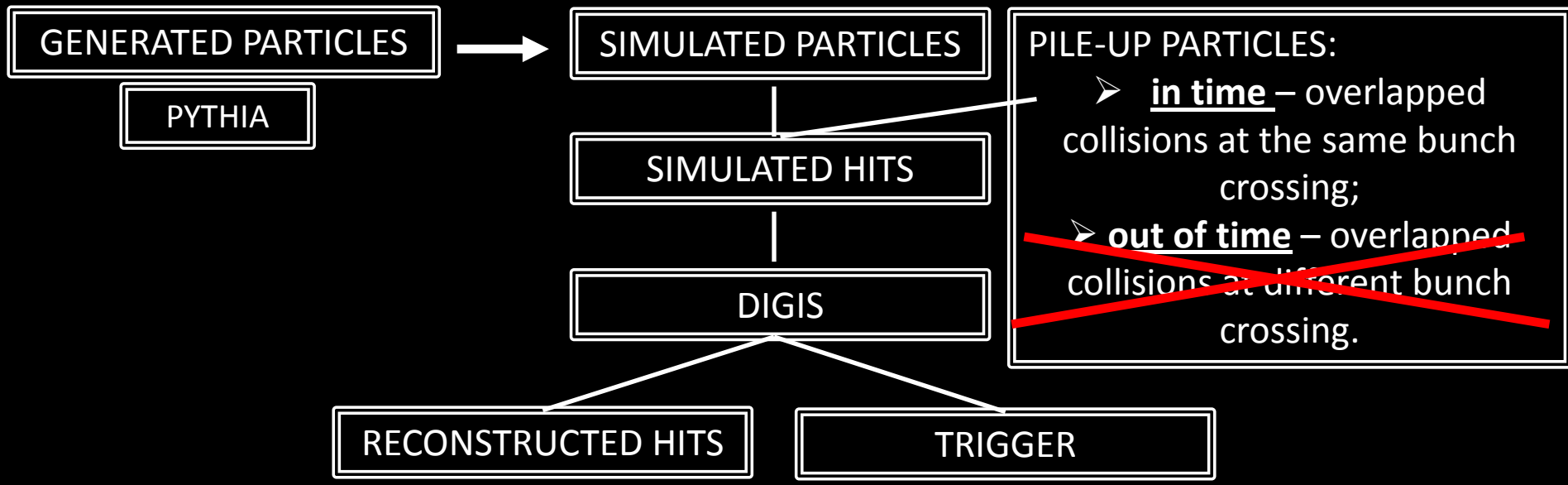
- Subdetectors response proportional to the particle energy
- Quantized signal
- Threshold

the **noise** from the electronic devices, other effects as **saturation** and **cross-talk** and the **Zero Suppression.** → Channels giving signal comparable with noise are ignored

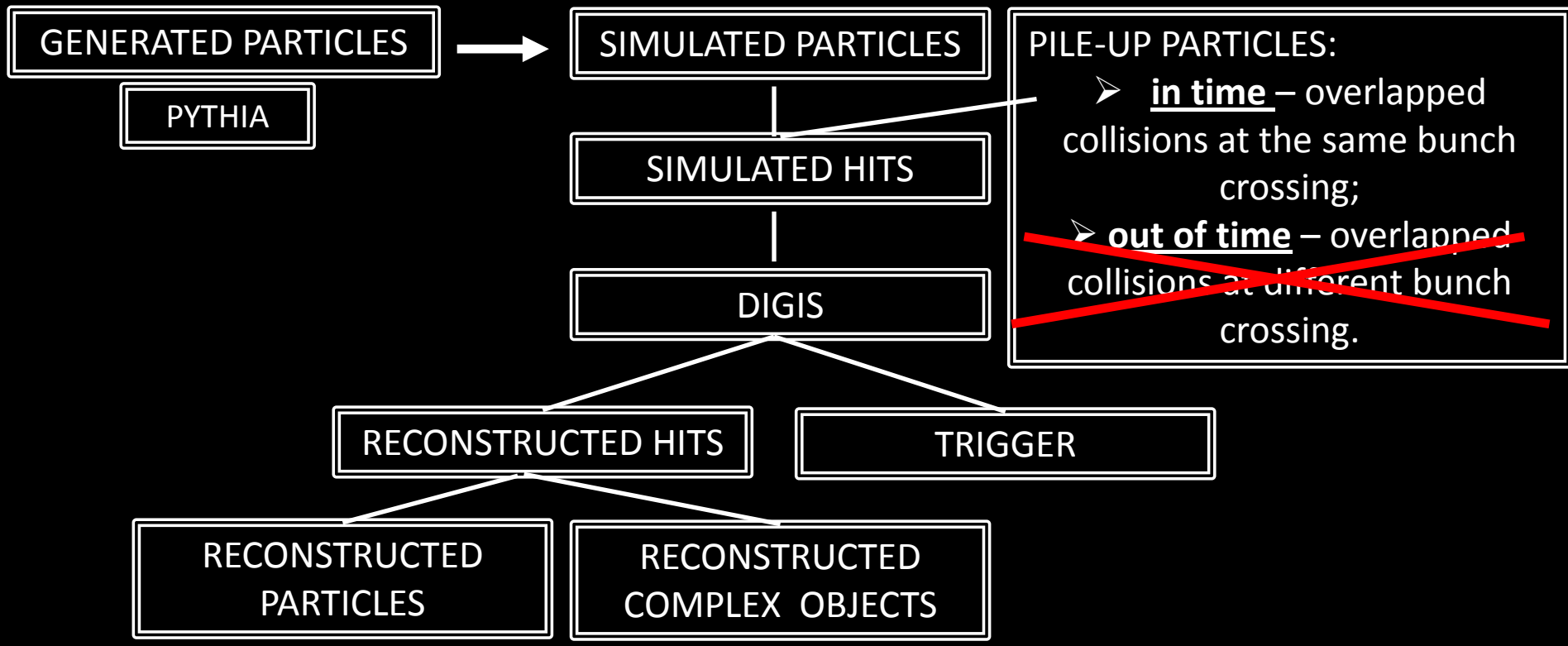
Why do we use the digitizers?

To obtain the same data format to give as input to the local reconstruction modules, and to simulate the electronic effects (especially noise) in the same realistic way as FullSim. [see the EM isolation in the next plots]

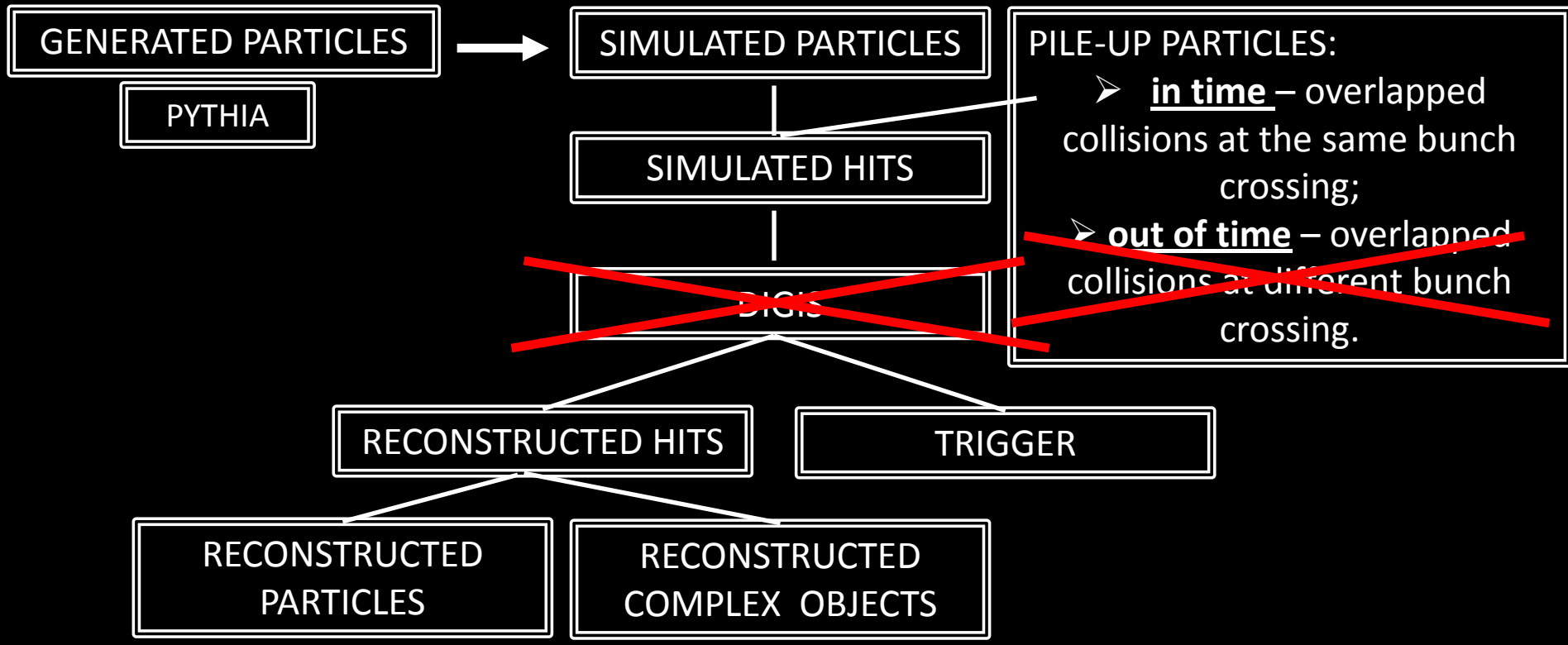
Fast Simulation workflow



Fast Simulation workflow



Fast Simulation workflow



SKIPPED IN THE FAST SIMULATION (apart from muons)– Why?

- the collection of digis is not necessary for the analyses
- one less iteration for each subdetector

OOT Pile-Up

GENERATED PARTICLES → SIMULATED PARTICLES

PYTHIA

PILE-UP PARTICLES:

- **in time** – overlapped collisions at the same bunch crossing;
- **out of time** – overlapped collisions at different bunch crossing.

- **in time** – the mix is done at particle level, and the vertices from the different events are taking into account only during the reconstruction step

- **out of time** (OOT)– ***the mix has to be done at event generator level*, and particles have to contain the information about the bunch crossing of production.**

Mixing Module

Digitizer

…how do they work?

Mixing Module

- It was designed for FullSim;
- it is able to mix several kind of object as “SimHit”, “RecHit”, “Events”, etc…
- now it works also in FastSim;
- it is used to mix particles of main event to particles of Minbias events at the “generator level”

Mixing Module

- It was designed for FullSim;
- it is able to mix several kind of object as “SimHit”, “RecHit”, “Events”, etc…
- now it works also in FastSim;
- it is used to mix particles of main event to particles of Minbias events at the “generator level”

Mixing Module

- It was designed for FullSim;
- it is able to mix several kind of object as “SimHit”, “RecHit”, “Events”, etc…
- now it works also in FastSim;
- it is used to mix particles of main event to particles of Minbias events at the “generator level”, to obtain the information about the time and the position of production vertex.

Which time?

Mixing Module

- It was designed for FullSim;
- it is able to mix several kind of object as “SimHit”, “RecHit”, “Events”, etc…
- now it works also in FastSim;
- it is used to mix particles of main event to particles of Minbias events at the “generator level”, to obtain the information about the time and the position of production vertex.

Which time?

time of production in terms of bunch crossing

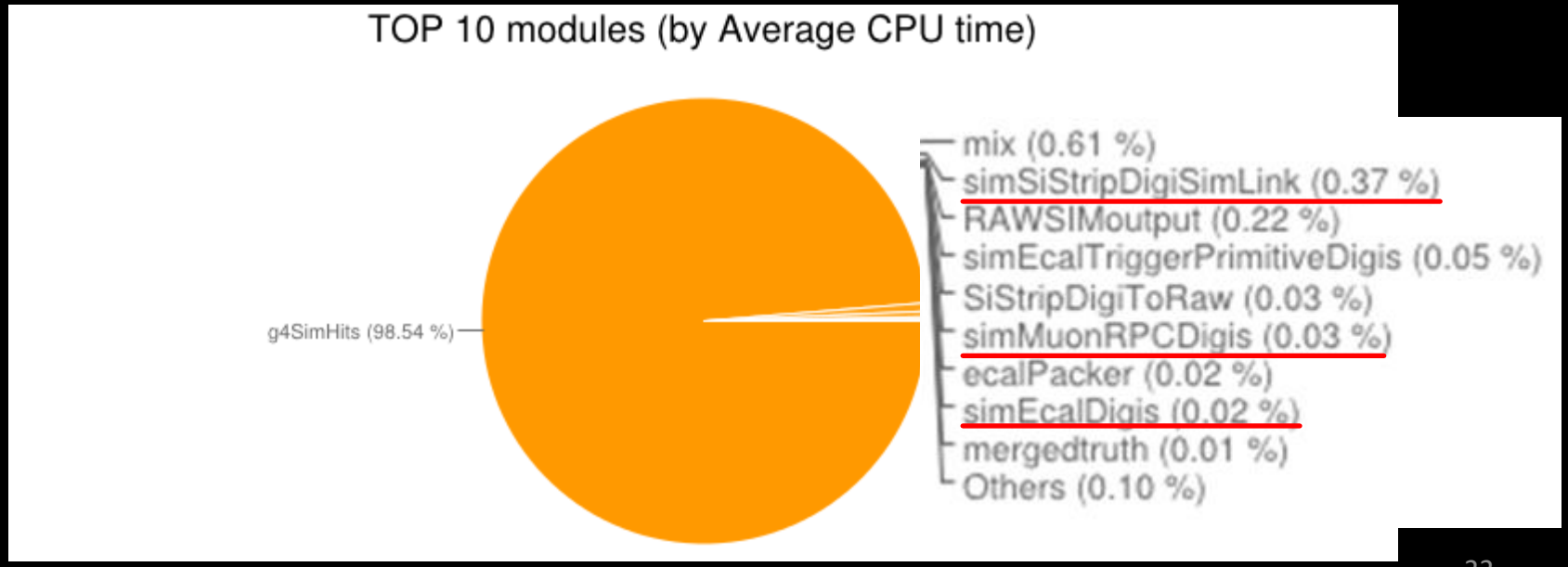
we also need of the “time of flight”:

- **it’s the time respect to the production vertex;**
- **it’s translated as latency time in the subdectors;**
- **it affects the subdetectors used to trigger (calorimeters and muon chambers for L1 trigger);**
- **it’s taken into account by digitizers;**

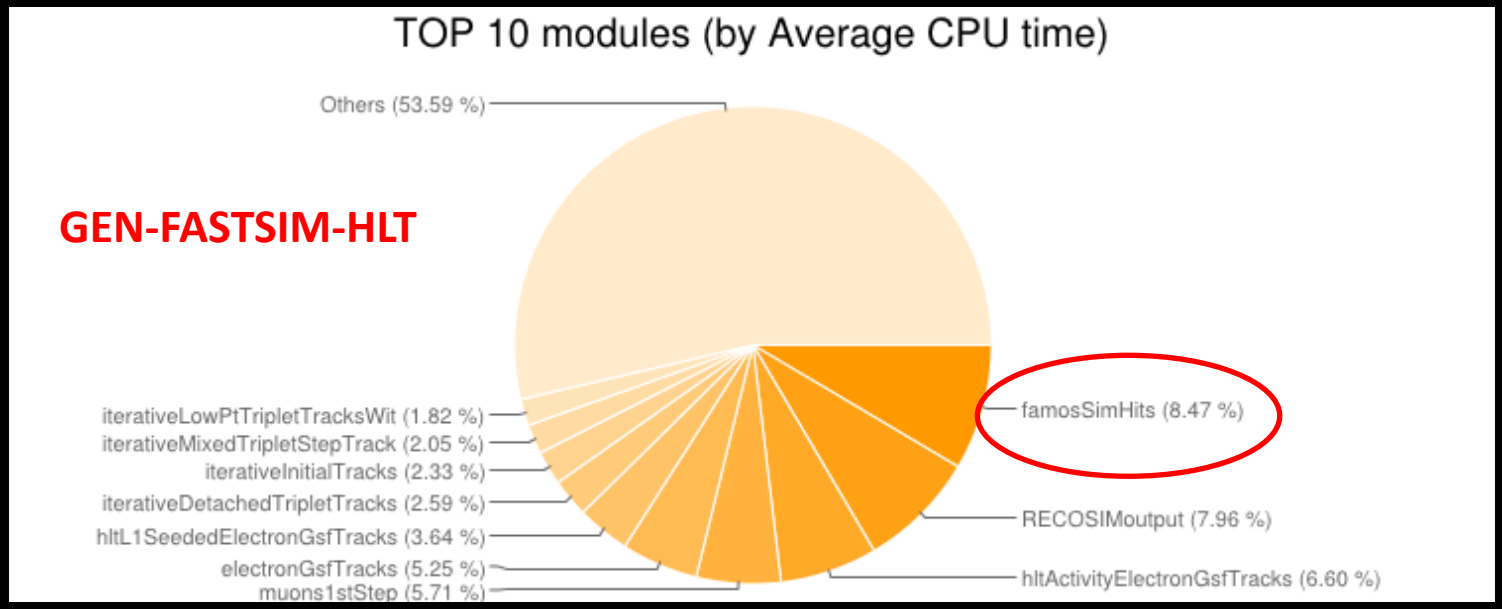
Digitizers

The digitizers are taken from the Full Simulation for the:

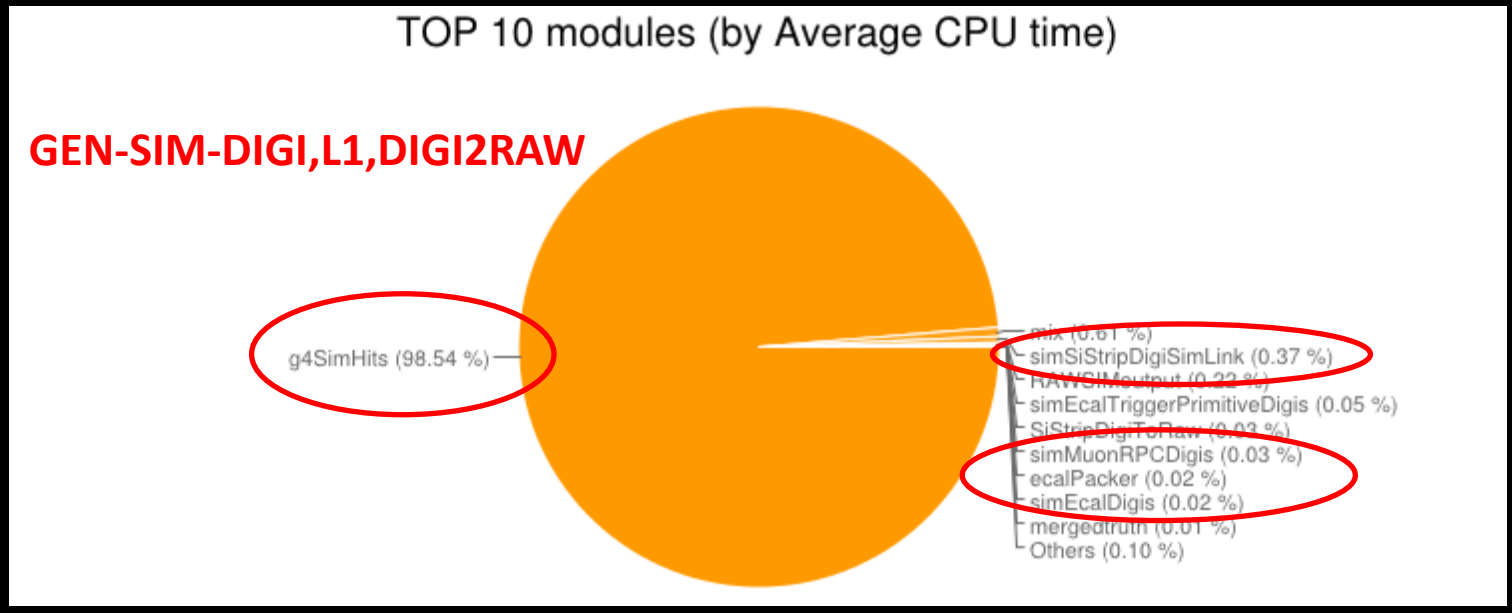
- **muon system;**
- **electromagnetic calorimeter;**
- **hadronic calorimeter ;**
- **the tracker module is skipped for the moment, because is the most time consuming.**



Timing



In FullSim the timing for digis is dominated by the tracker.



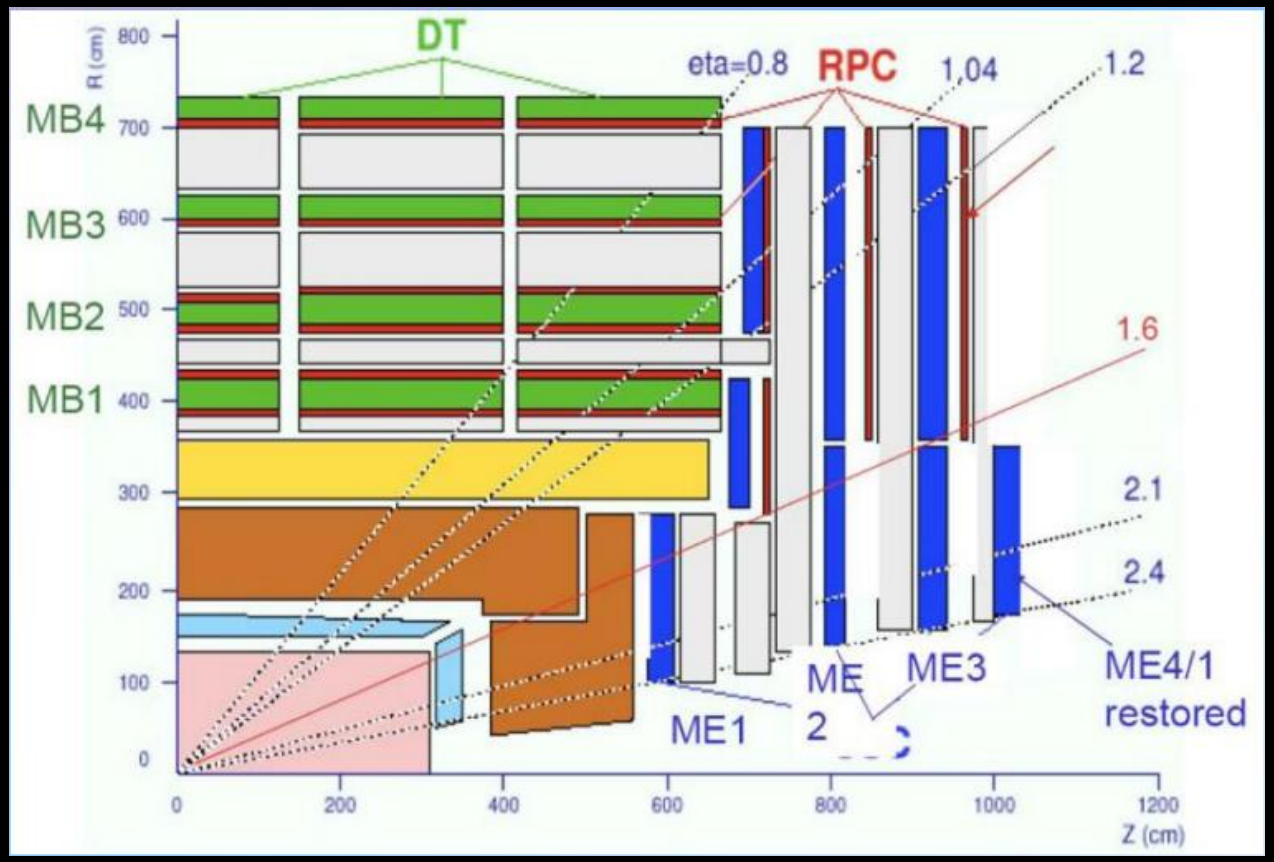
The digitization for calorimeters and muon chambers is not so time consuming.

Muon digitizer

The digitizers for the muon chambers were implemented since the beginning in Fast Simulation because: - is not so time consuming;

- was convenient to have a uniform code and tuning with FullSim;
- in particular was convenient for the trigger emulator.

the good agreement with FullSim and data is shown in A.Giammanco's talk



BARREL:
Drift-Tubes (DT),
Resistive-Plate-Chambers (RPC);

ENDCAP:
CSC and RPC.

DT, CSC, and RPC are composed by several chambers in turn containing strips or wires.

Calorimeter digitizers: Ecal

The digitizers for the calorimeters have just been implemented in FastSim (for the moment it can switch on separately Ecal and Hcal).

The process is similar in both calorimeters so an unified framework is used. The digis sequence of the whole Calorimeter system is divided in **ecalDigiSequence** and **hcalDigiSequence**;

Ecal digitizer:

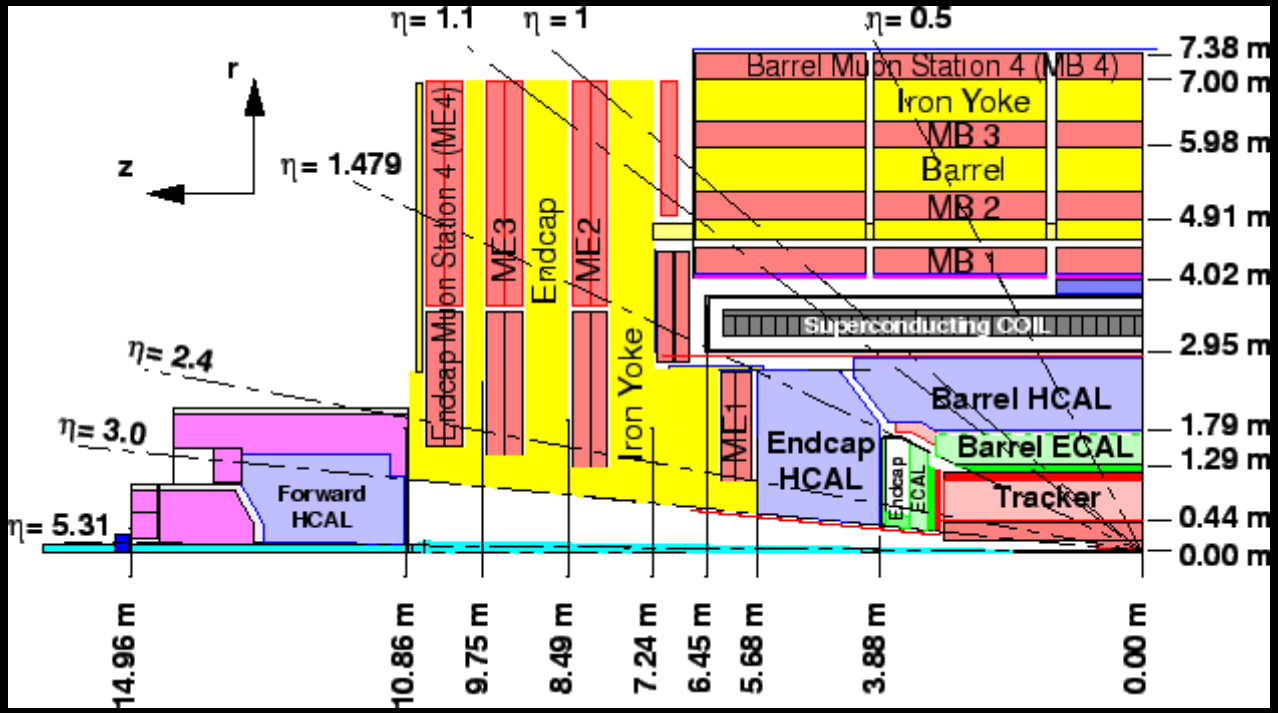
1. **conversion of the SimHit energy deposition in the channels into the electronics signal;**
2. **simulation of the noise, in EcalBarrel, EcalEndcap, and ES (Preshower);**
3. **modeling of the trigger primitives;**
4. **application of the zero suppression;**

Each module comes with its set of configurable parameters.

Calorimeter digitizers: Hcal

… for Hcal is a bit more complicated:

An HCAL Digi represents the signal in one readout channel, and consists of ten coded integers (time slice), each representing charge deposited in a 25 ns time bin.
The SimHits are the total signal in a given readout channel in a nanosecond.



Different electronic read out gives different units for these SimHits:
GeV for HB, HE, and HO, and in units of photoelectrons for HF.

Calorimeter digitizers: Hcal

Hcal digitizer:

1. conversion of the SimHits amplitudes into photoelectrons:
   - taking into account the gains;
   - modelling the channel response as well as a ring-dependent sampling factor.

2. The photoelectrons are then subject to Poisson statistics, and have their timing adjusted to correct for time of flight.

3. Conversion of each group of photoelectrons into an electronics pulse (units of fc);

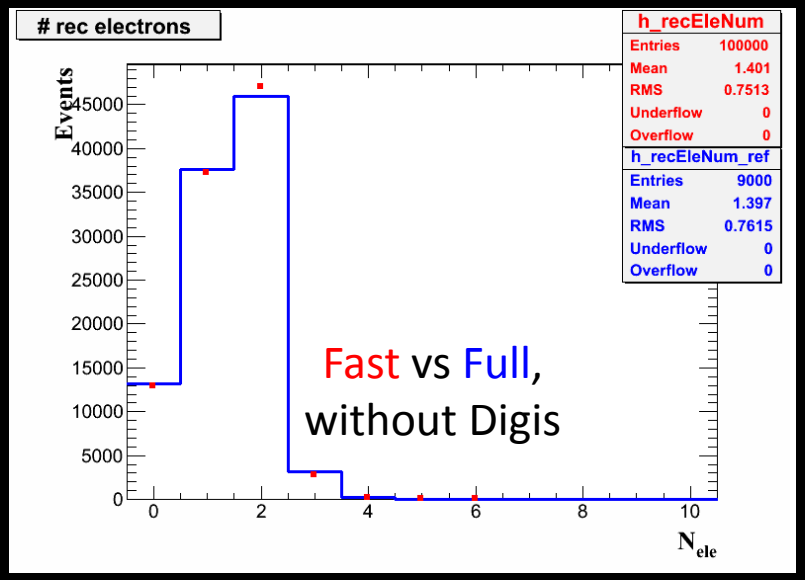
4. Simulation of the noise.

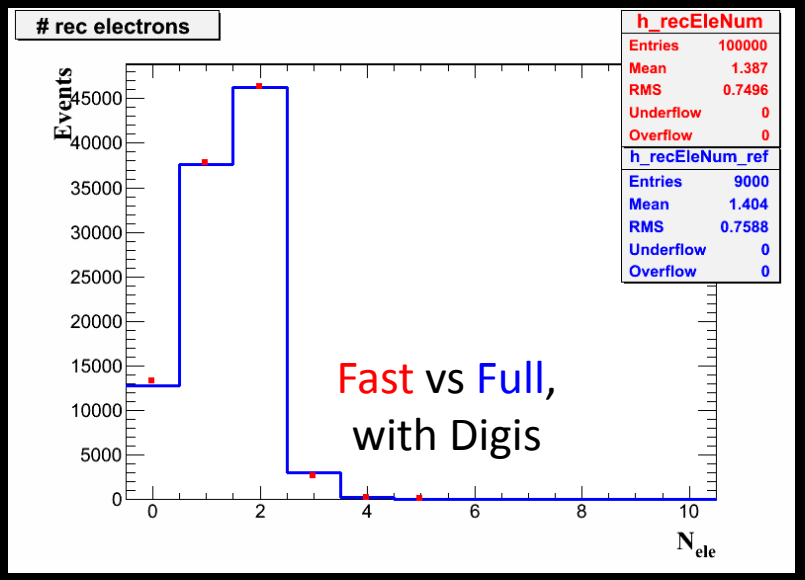
5. Trigger primitives.

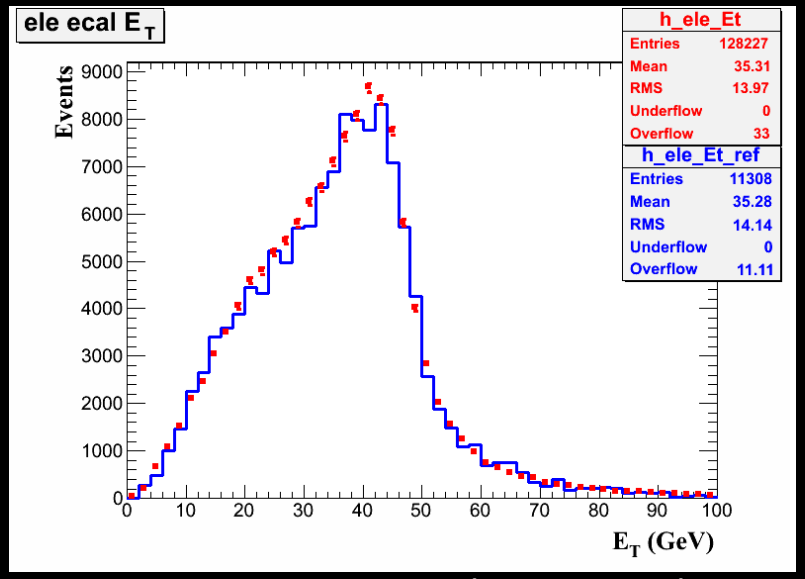
6. Zero suppression.

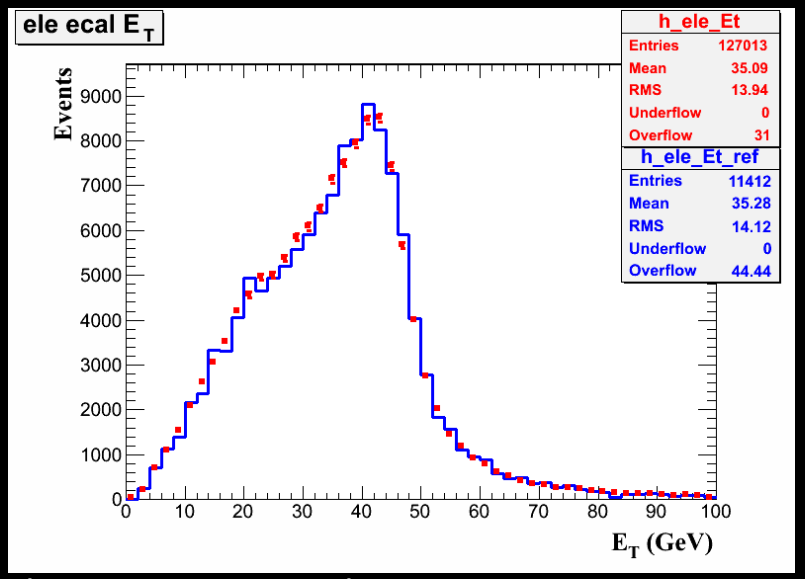
Ecal digitizers

Electrons in Z→ee events



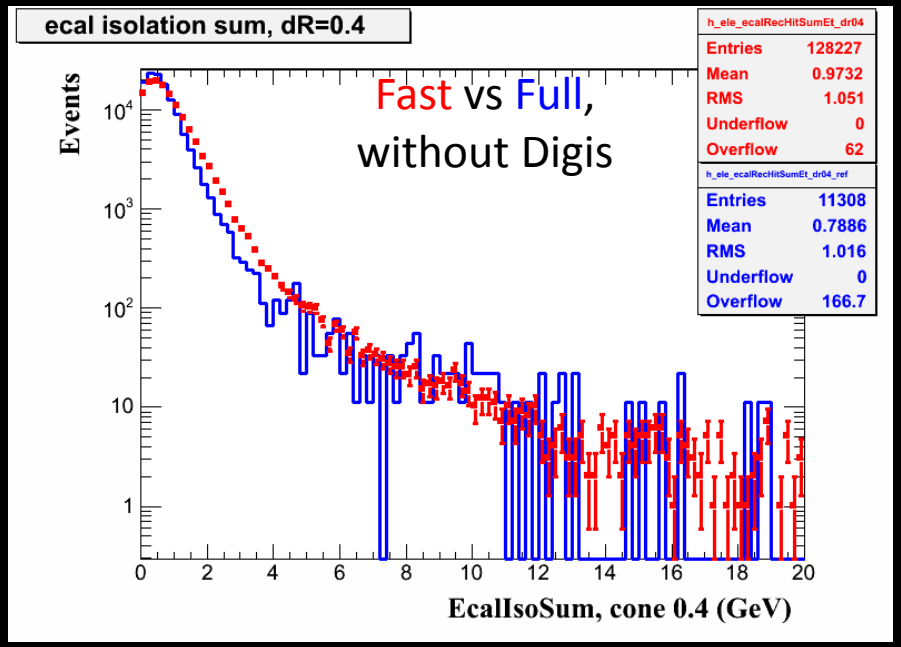


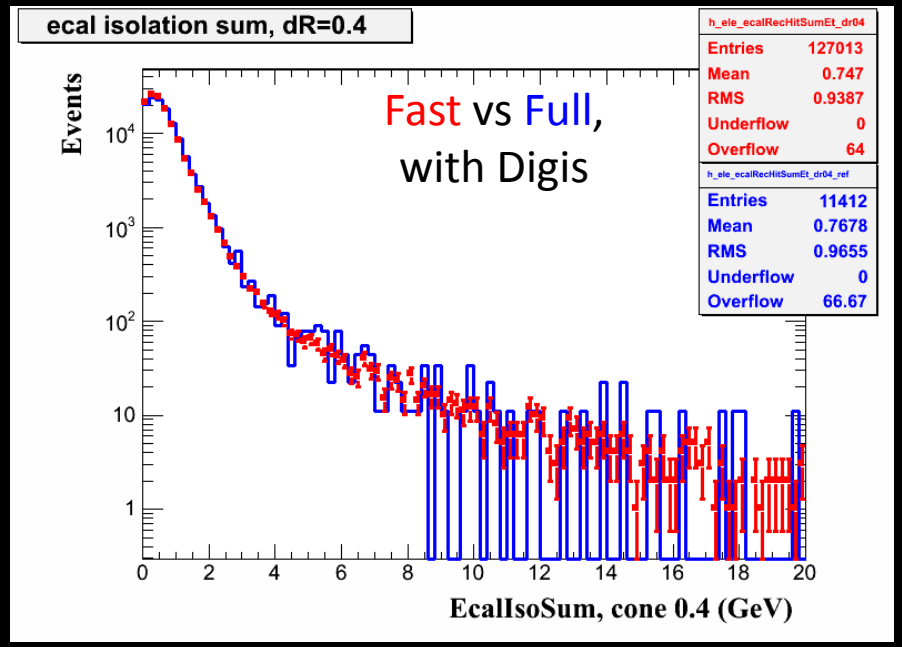




Ecal digitizer

Electrons in Z→ee events

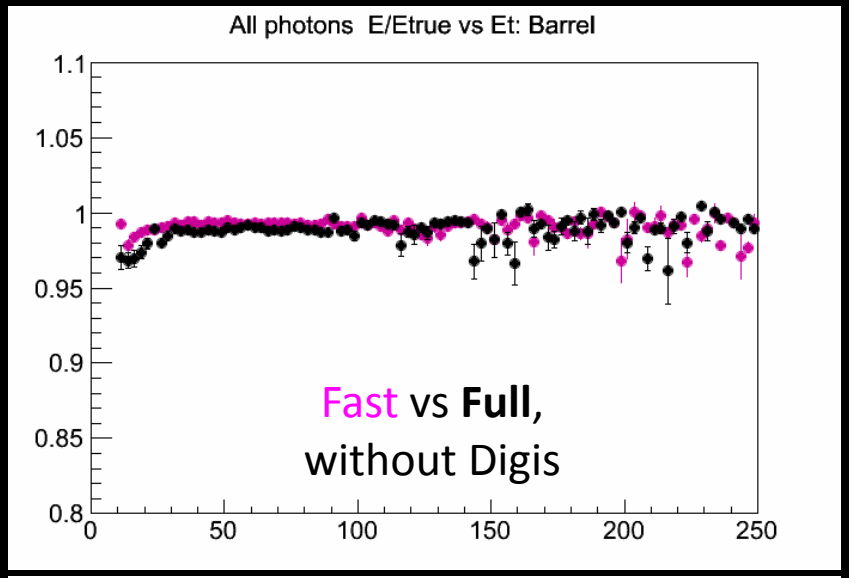


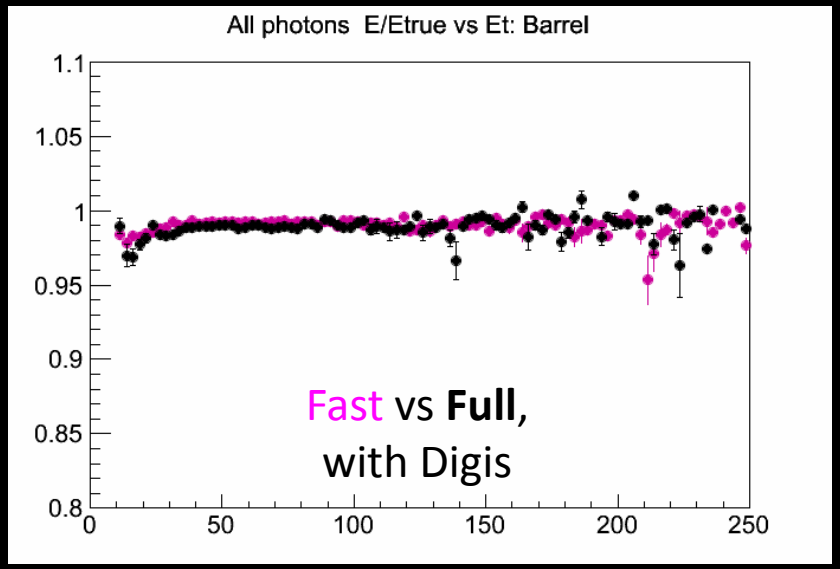


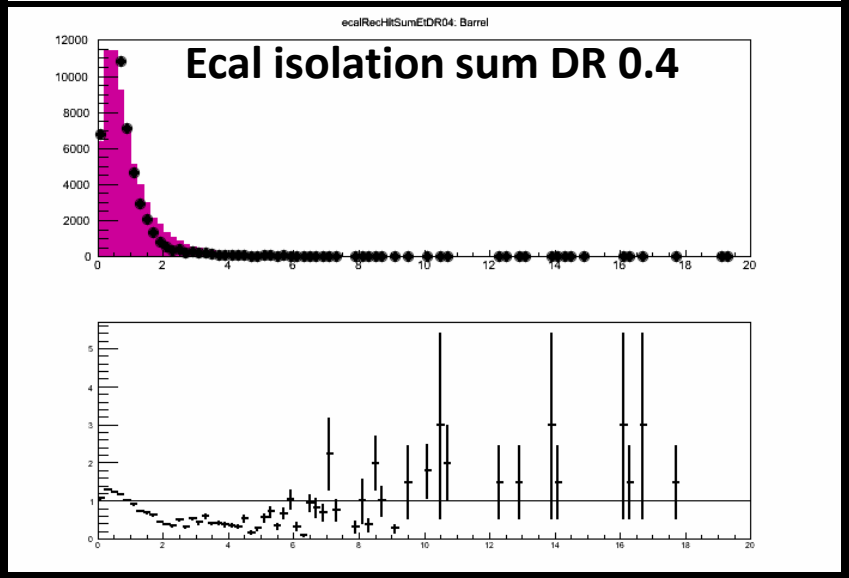
All Fast-Full comparisons stayed the same or improved.
Noise-related ones improved a lot

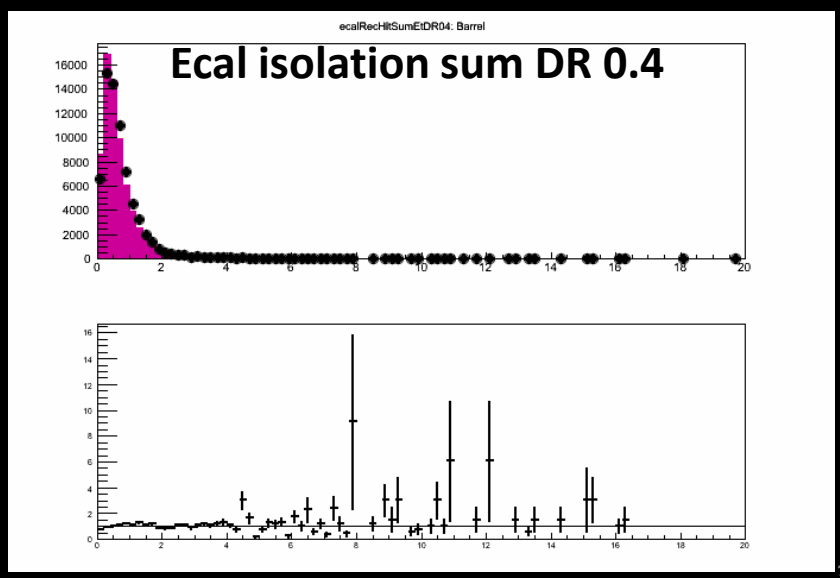
Ecal digitizers

Photons in H→γγ events





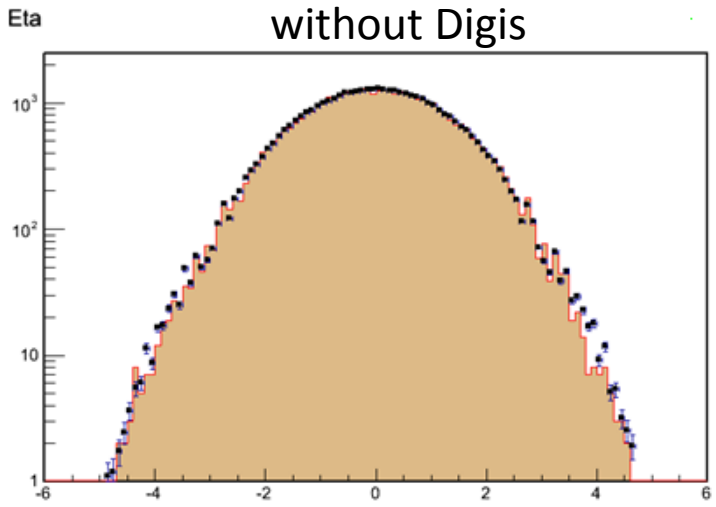




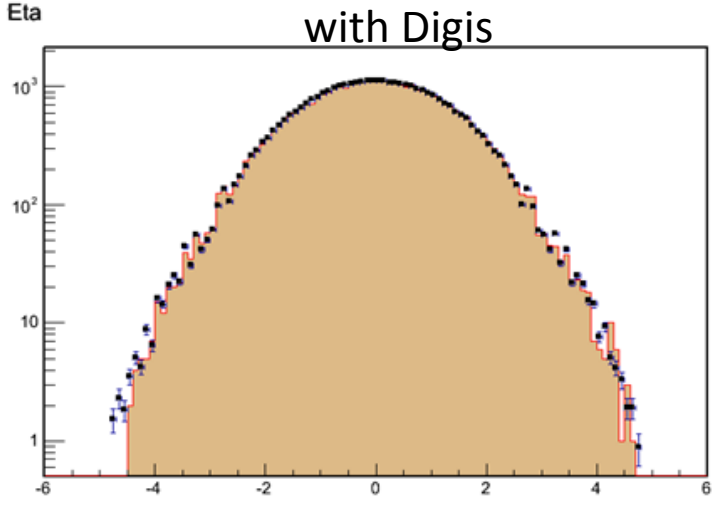
Ecal digitizers

Jets in ttbar events

Fast (solid) vs Full (dots), without Digis

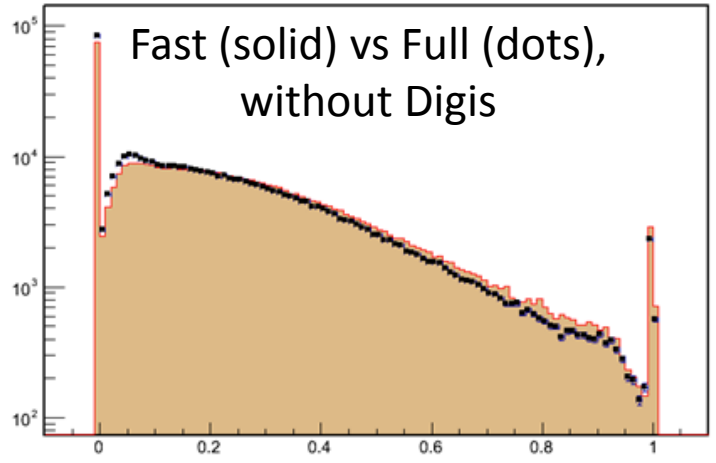


Fast (solid) vs Full (dots), with Digis



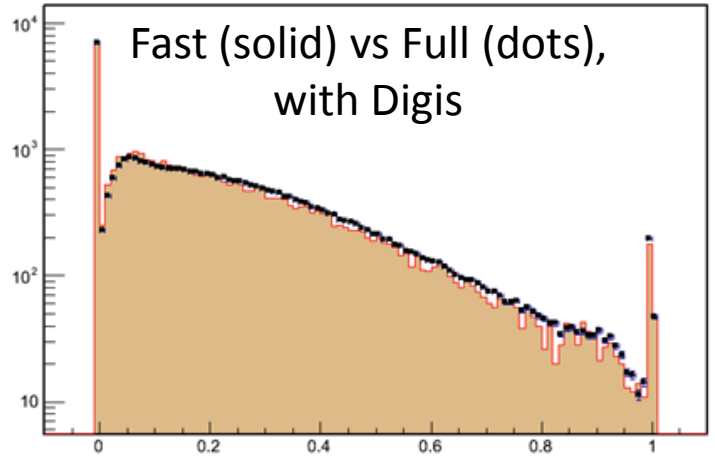
EM energy fraction in the Jet

Fast (solid) vs Full (dots), without Digis



EM energy fraction in the Jet

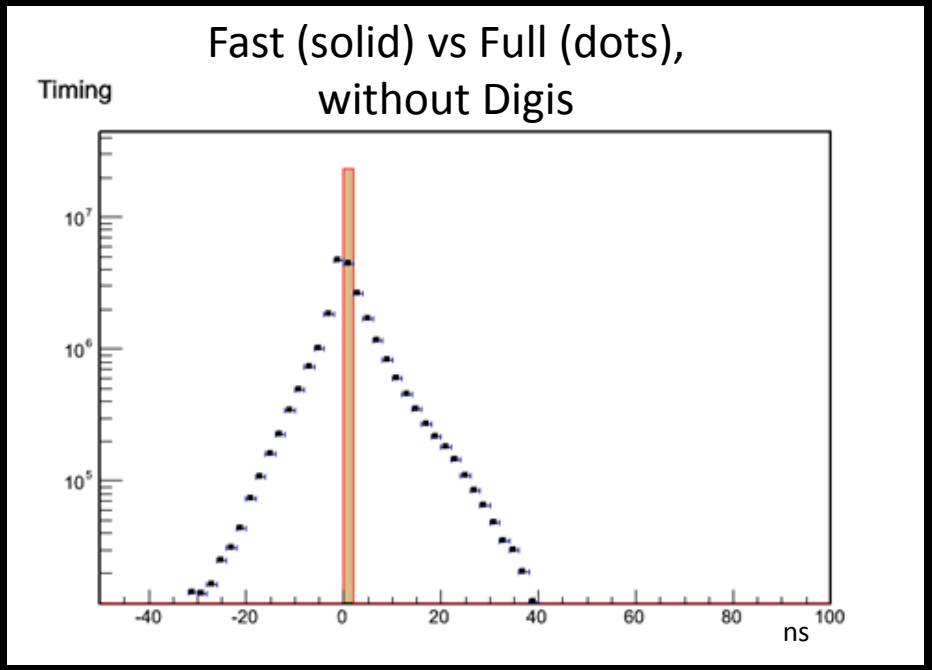
Fast (solid) vs Full (dots), with Digis



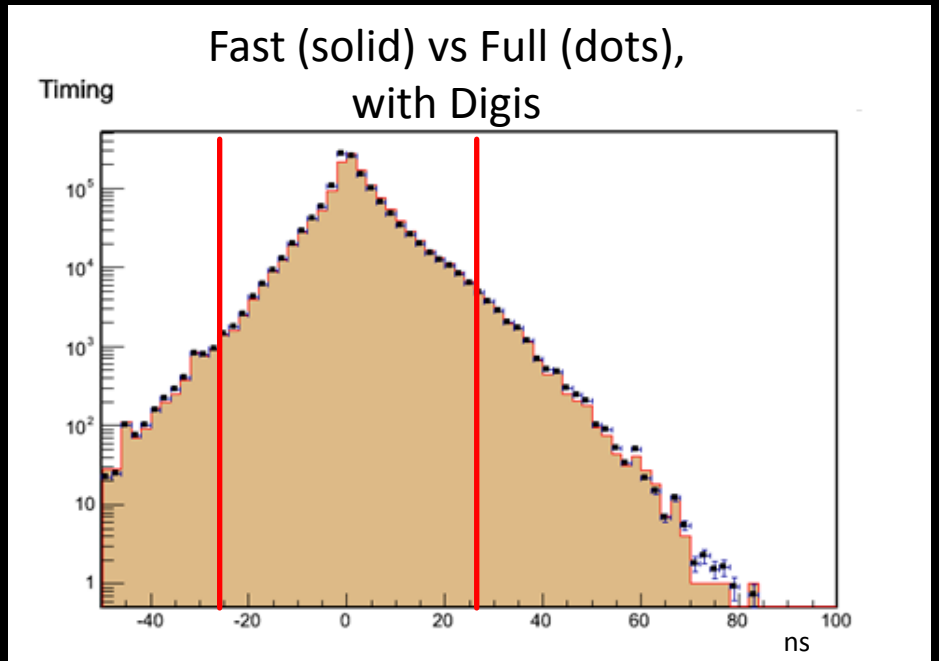
Ecal digitizers

EM hits in ttbar events

Fast (solid) vs Full (dots), without Digis



Fast (solid) vs Full (dots), with Digis



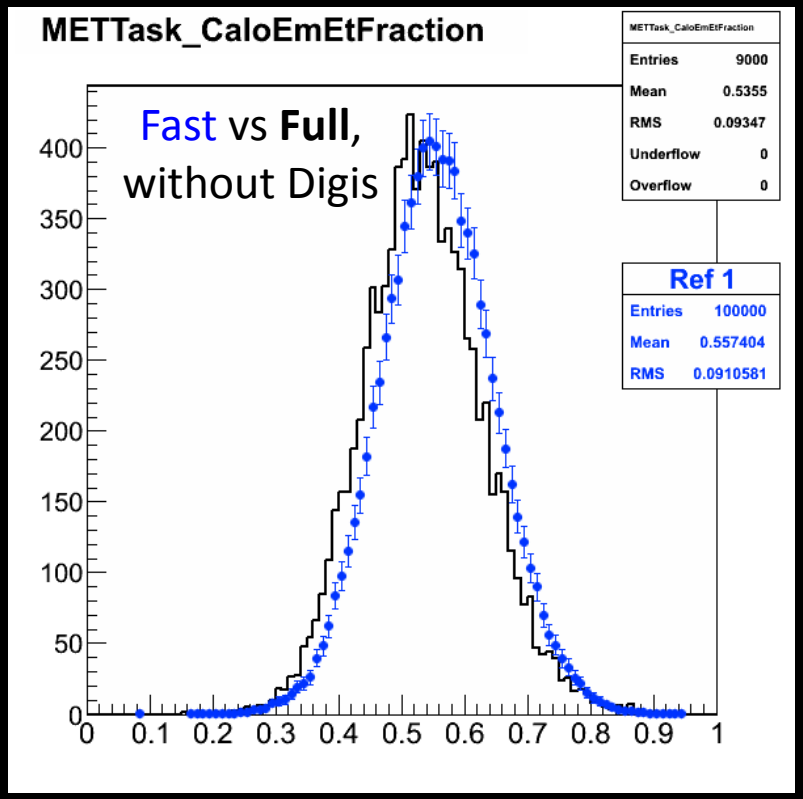
The **time** (related to the TOF) is a new feature in FastSim RecHit.
It is approximate (length of a straight line from 0,0,0 to the cell entrance, over c).

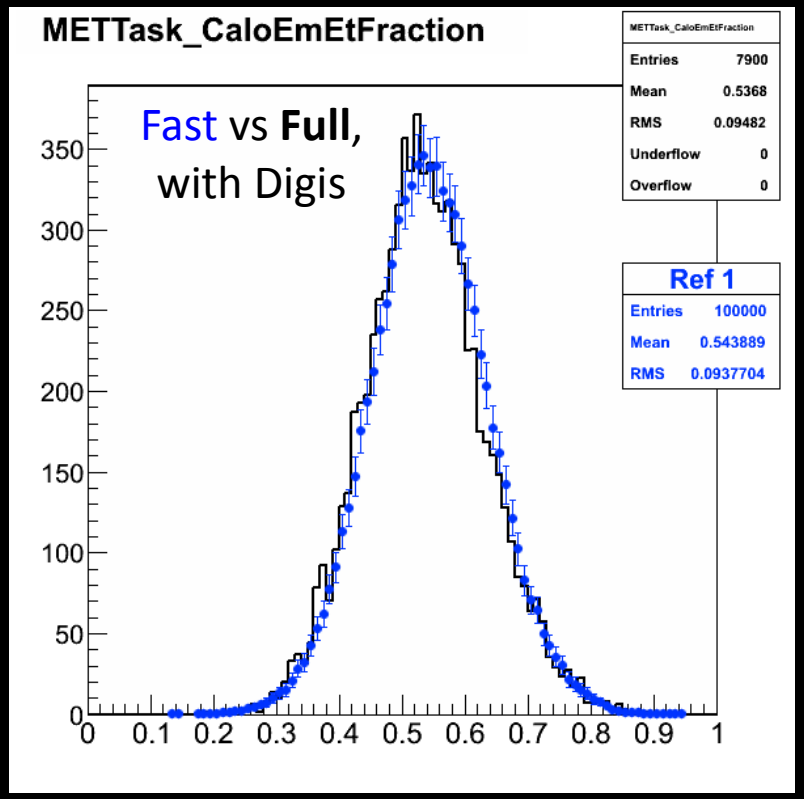
Important in view of the future implementation of OOT PU

What matter are the tails <-25 ns and > 25 ns, dictated by the digitizer

Ecal digitizers

MET in ttbar events

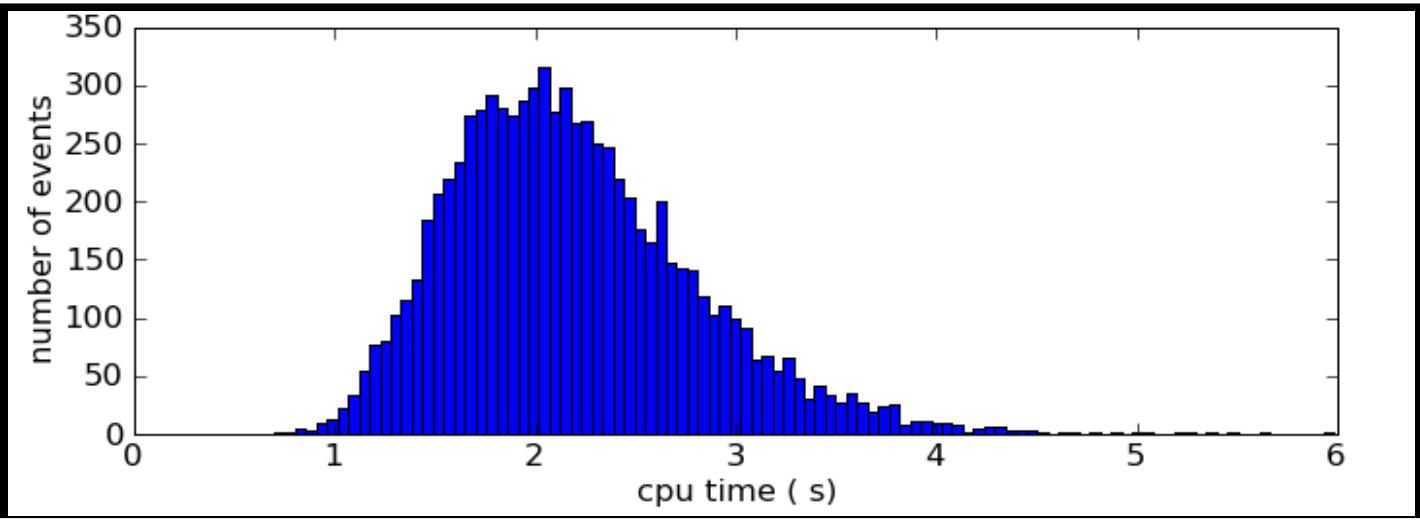




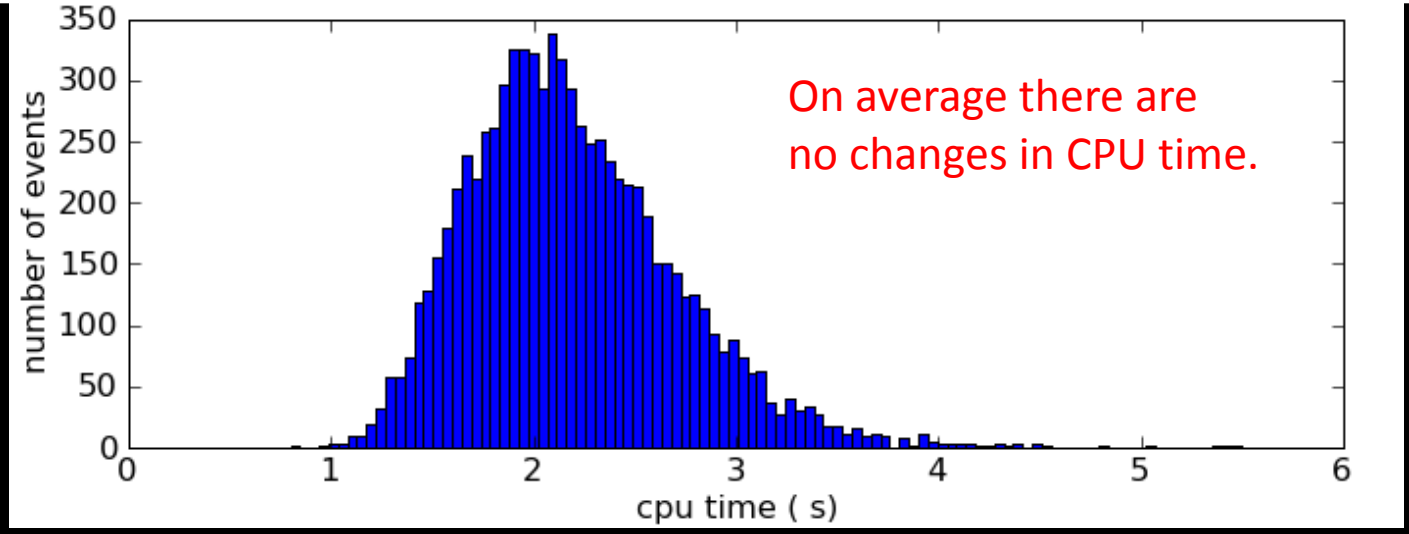
The comparisons for the all kinds of MET in ttbar events show a new good agreement with FullSim.

CPU time: FastNoDigis vs FastDigis

GEN-FASTSIM-HLT : without Ecal Digis



with Ecal Digis



Summary

- The use of Fast Simulation is important in particular for planning the forthcoming analyses at higher Pile-Up and luminosity of LHC;

- The use of the Full Digitizer, initially just for the muon system, is now implemented for the calorimeters:
  - already switched on by default for Ecal;
  - technically ready for Hcal, to be switched on when signal response will have been retuned;
  - In Ecal nice improvement of the performances, thanks to a more realistic simulation of the electronic effects as noise ;
  - The overall timing is not visibly affected.

Summary

- The use of Fast Simulation is important in particular for planning the forthcoming analyses at higher Pile-Up and luminosity of LHC;

- The use of the Full Digitizer, initially just for the muon system, is now implemented for the calorimeters:
  - already switched on by default for Ecal;
  - technically ready for Hcal, to be switched on when signal response will have been retuned;
  - In Ecal nice improvement of the performances, thanks to a more realistic simulation of the electronic effects as noise ;
  - The overall timing is not visibly affected.

Dank! ☺

Back up

Muon digitizer: DT

The DT digitizer simulates the signal collection in a single chamber:

- gas ionization;
- electrons propagation in the electric field geometry;
- drift velocity of electrons (400 ns);
- dead time;
- propagation of the signal along the anodic wire to the front end modules;