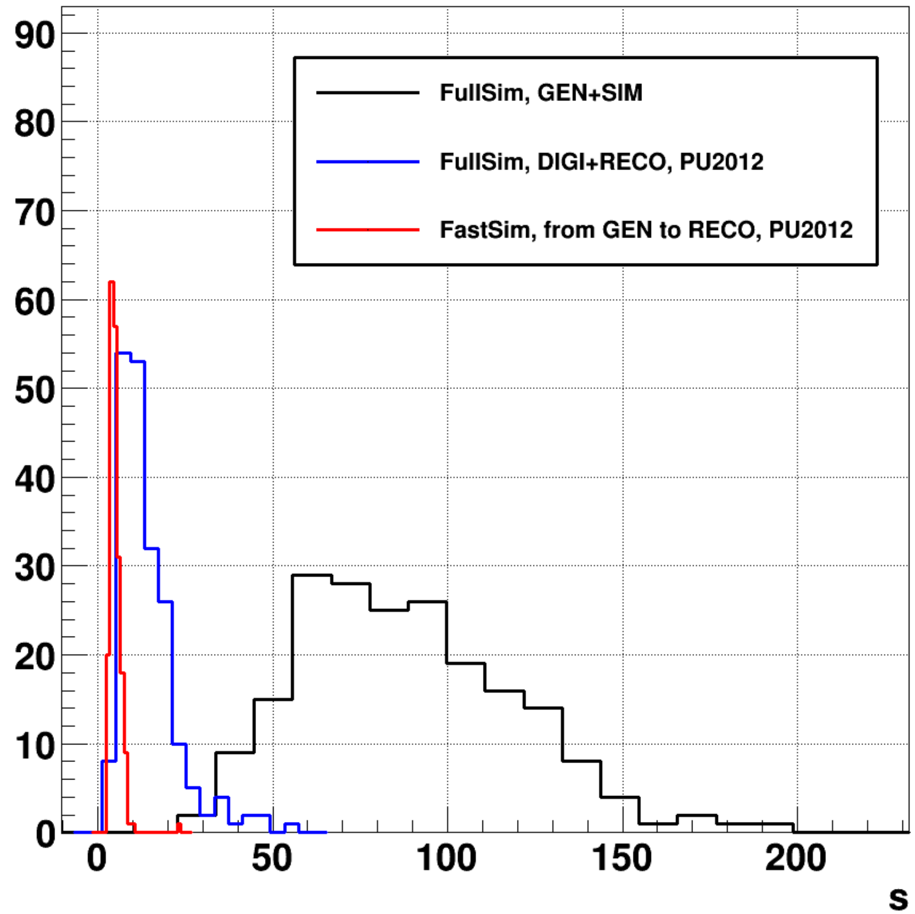
The CMS Fast Simulation

A.Giammanco (UCL Louvain & FNRS & NICPB)
representing the CMS Fast Simulation Group



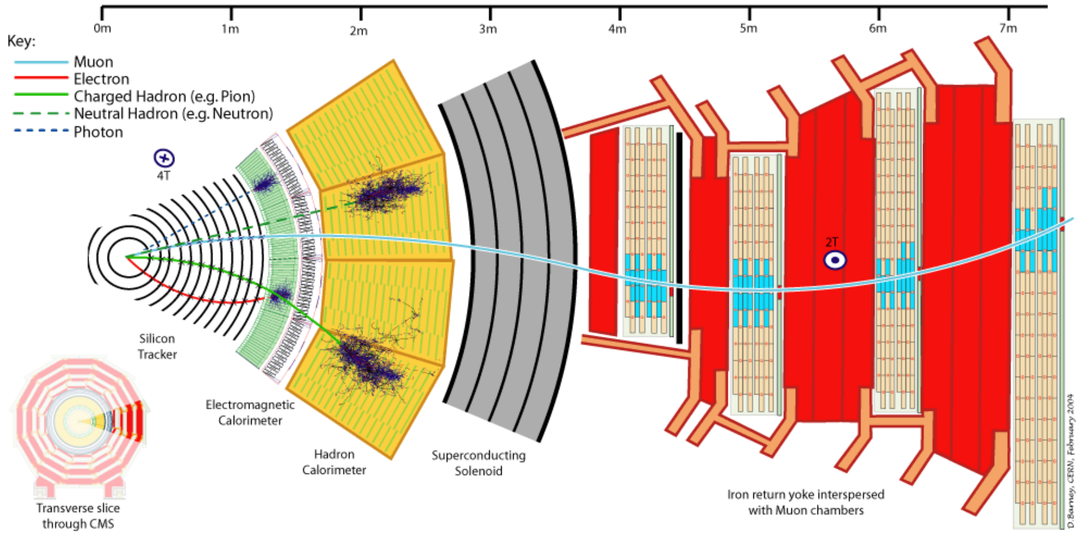
CHEP 2013, Amsterdam (Netherlands)

What is the CMS FastSim

- OO subsystem of the CMS C++ based software (CMSSW)
- Alternative and complementary to GEANT4-based approach (“FullSim”)
  - FastSim is regularly validated/tuned by comparing to FullSim
- Differently from PGS or Delphes, which are fully parametric simulations, we don't smear gen-level to analysis-level
- Differently from GEANT4 it is not an ab-initio simulation, we parametrize material effects according to their known distributions
- We do a realistic simulation of low-level objects (hits, clusters)
- On these we apply the same high-level modules (lepton reco, particle flow, jet finding, b/τ-tagging, isolation ...) as in FullSim and data
- The only case where reconstruction is customized is tracking
  - FastTracking: seeding cheats by using MC truth
  - Although the option exists to run real tracking on the hits

Interactions

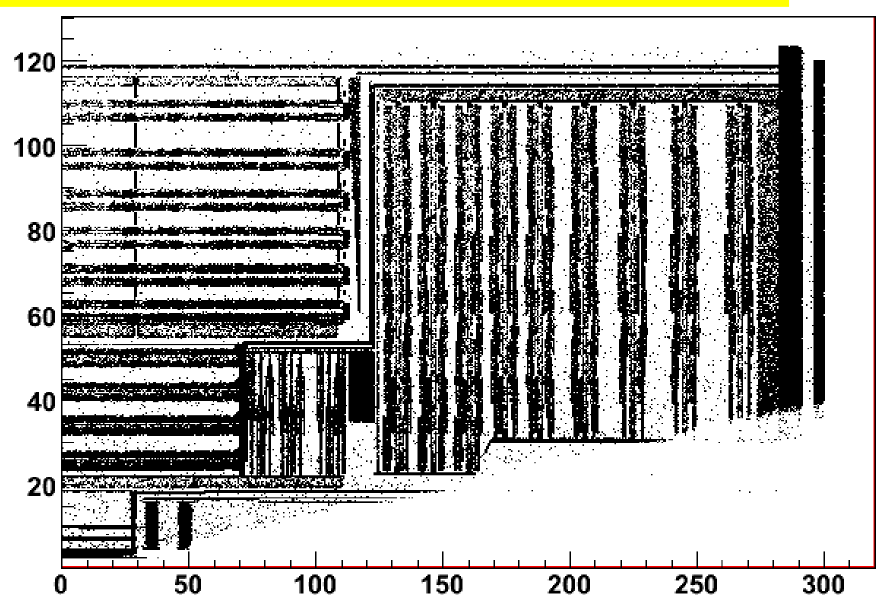
- All material effects are simulated when crossing a layer (point-like approach, as opposed to cumulative effects in the bulk)
- The interactions considered in the fast simulation:
  - Electron Bremsstrahlung, γ conversion (inner tracker)
  - Energy loss by ionization; multiple scattering (inner tracker, muon chambers)
  - Nuclear interactions (inner tracker)
  - Electron, photon and hadron showering (calorimeters)
- δ-rays are ignored (effects absorbed in energy loss, or parametrized)



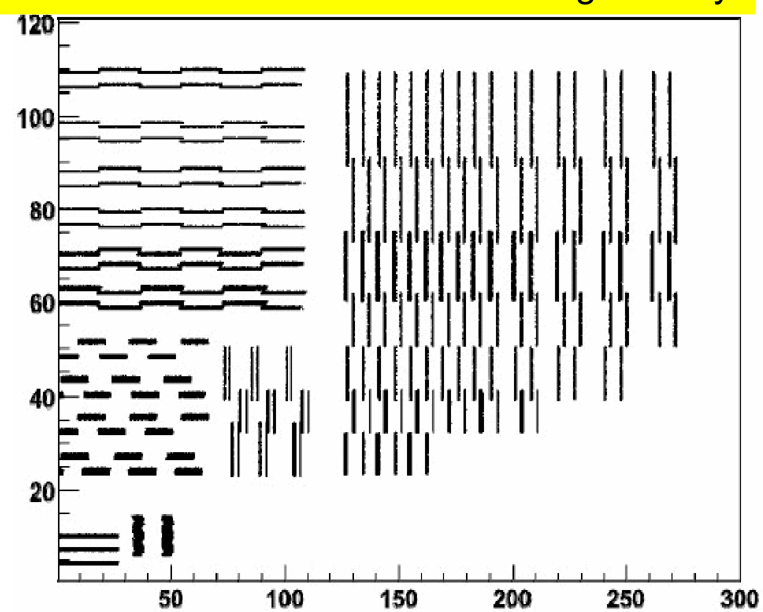
Tracker geometry and material

- Great gain in time by simplified geometry
- Interaction geometry: connected cylindrical volumes, navigated from a layer to the next; material is mapped onto layers
- Exact description of sensitive elements by using reconstruction geometry
- Direct propagation between volume boundaries, but taking into account detailed magnetic field map
- Dead modules are considered for the pixels (from conditions database)

CMS-tracker “tomography” with FullSim geometry

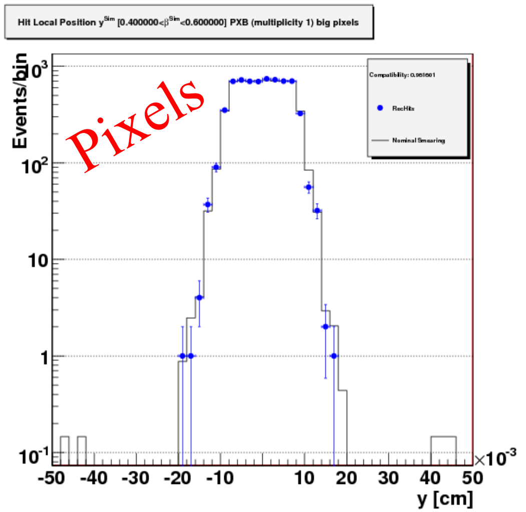


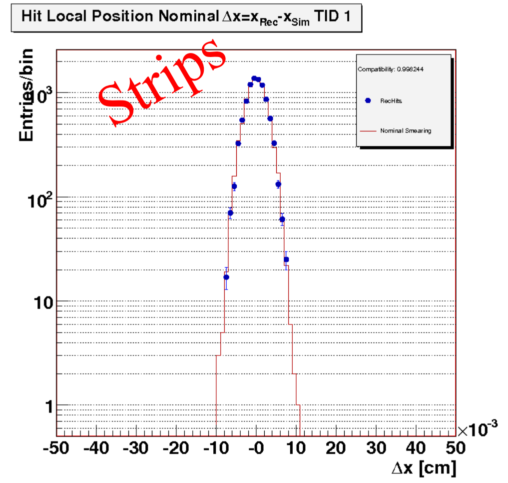
...and with reconstruction/FastSim geometry



From SimHits to RecHits

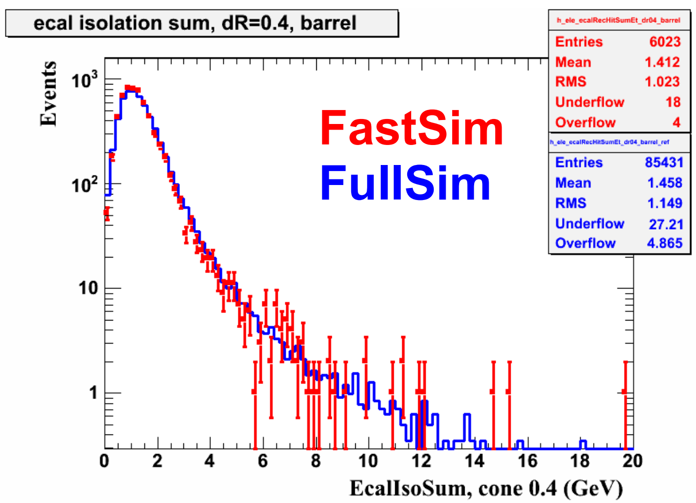
- Inner tracker: SimHits are directly converted into RecHits after a position smearing
  - Pixels: resolution distributions extracted from FullSim (might be from data) as functions of cluster multiplicity and track incidence angle
  - Strips: layer-dependent Gaussians
- Muon chambers:
  - SimHits are fed into the digitizer modules (same as in FullSim), and everything downstream is from standard reconstruction, including pattern recognition
- Calorimeters:
  - Until CMSSW 5.3 (Run-I legacy release) SimHits were directly converted into RecHits after a Gaussian energy smearing tuned on FullSim
  - Since CMSSW 6.2 (Run-II release), fully interfaced to FullSim digitizers; see next slide



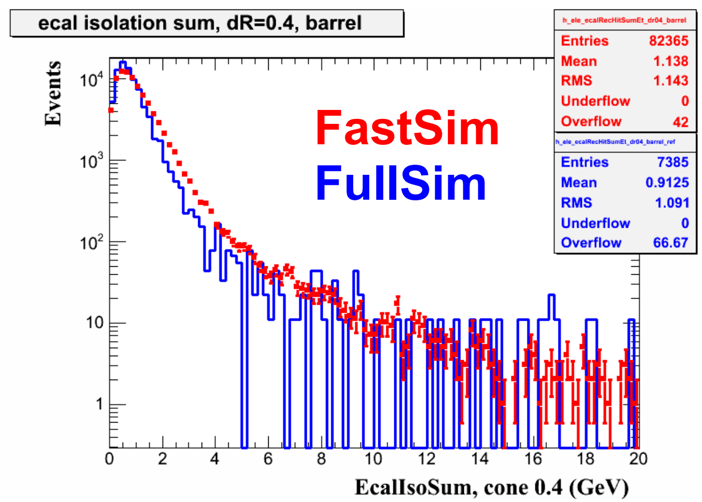


From smearing modules to digitizers

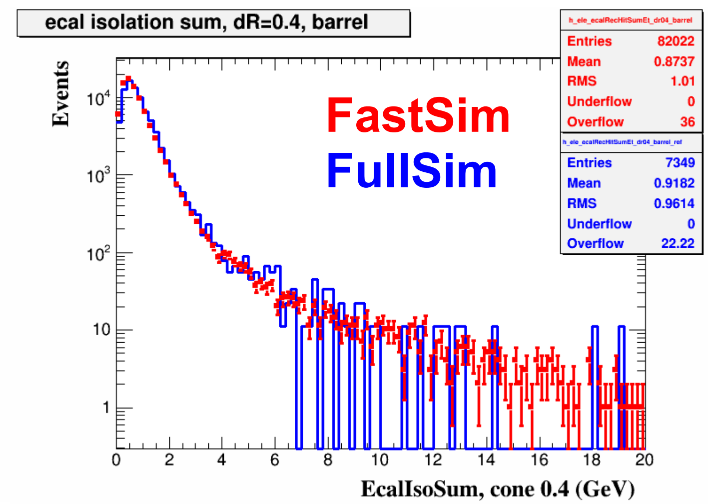
- No large penalty in cpu time, even with large pileup
- Improvements in some physics observables
- Most importantly, less maintenance burden, as electronic effects (like noise) don't need to be retuned to keep up with changes elsewhere:



CMSSW 4.4 (2011 release): ECAL noise applied in a FastSim module tuned on FullSim; good agreement in noise-sensitive variables (this example: isolation)



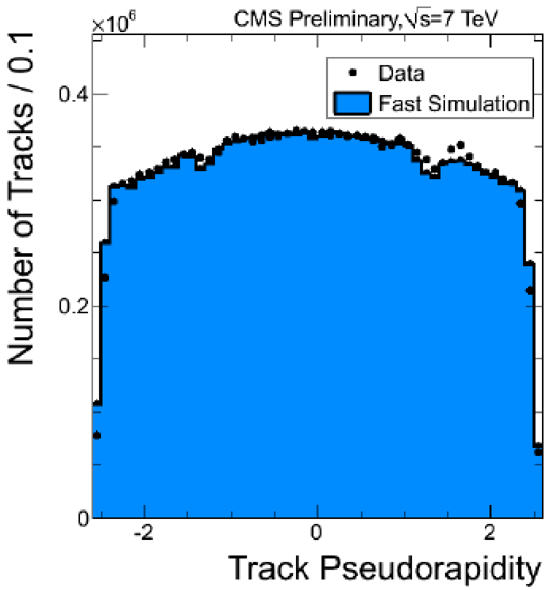
CMSSW 5.3 (2012 release): Data and FullSim's change in Selective Readout and Zero Suppression; FastSim-specific noise model became inadequate

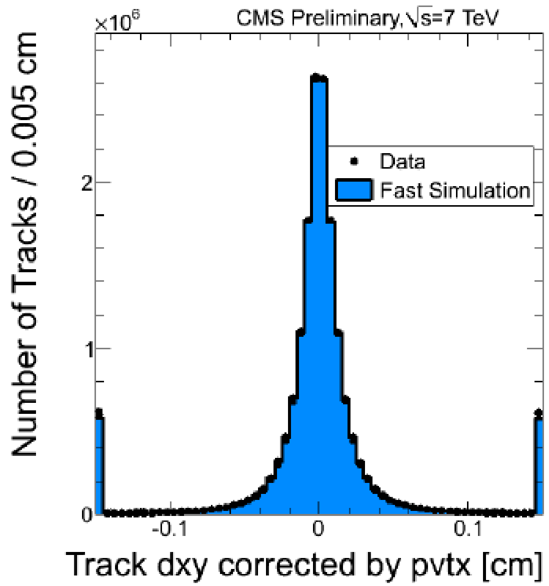


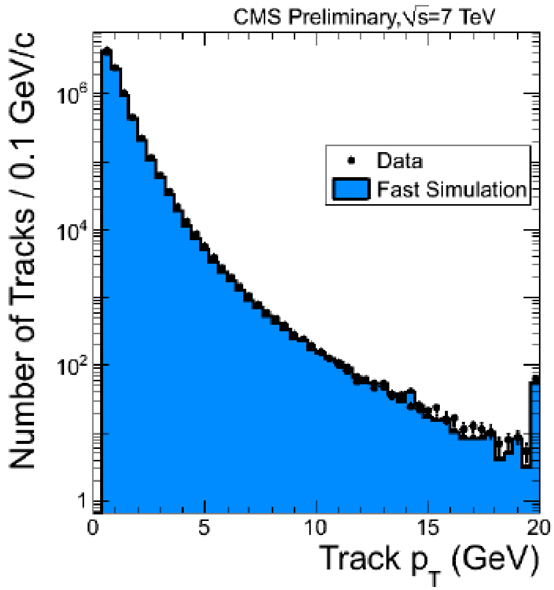
CMSSW 6.2 (2013 release): after integration with the FullSim digitizers, no need anymore for a FastSim-specific noise model

Tracking

- We save reconstruction time with **Fast Tracking**
  - FT emulates seeding efficiency (based on the hits of the MC-truth charged particle), performs fit, rejects outlier hits
  - Final track selection in FT uses same modules as real tracking
  - No fake tracks in FT (<1% of high-quality tracks in Run-I)
  - Excellent agreement with data after basic quality cuts
- Possibility to use standard tracking is available







b-tagging

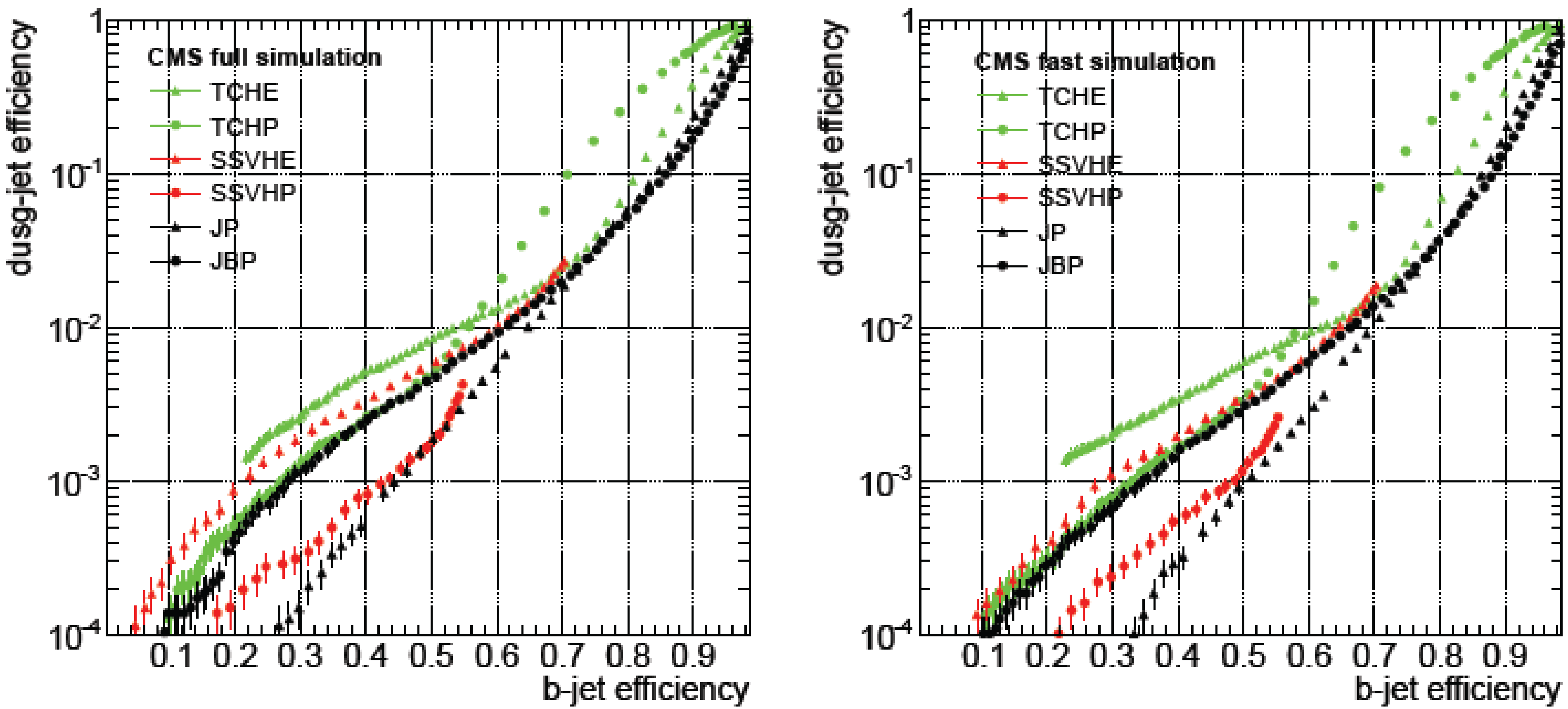


Figure 22: Light flavor mistag efficiency versus b-tagging efficiency in comparison for several b-tagging algorithms. On the left: full simulation, on the right: fast simulation.

- Same b-tagging modules are used as in data and in FullSim

b-tagging

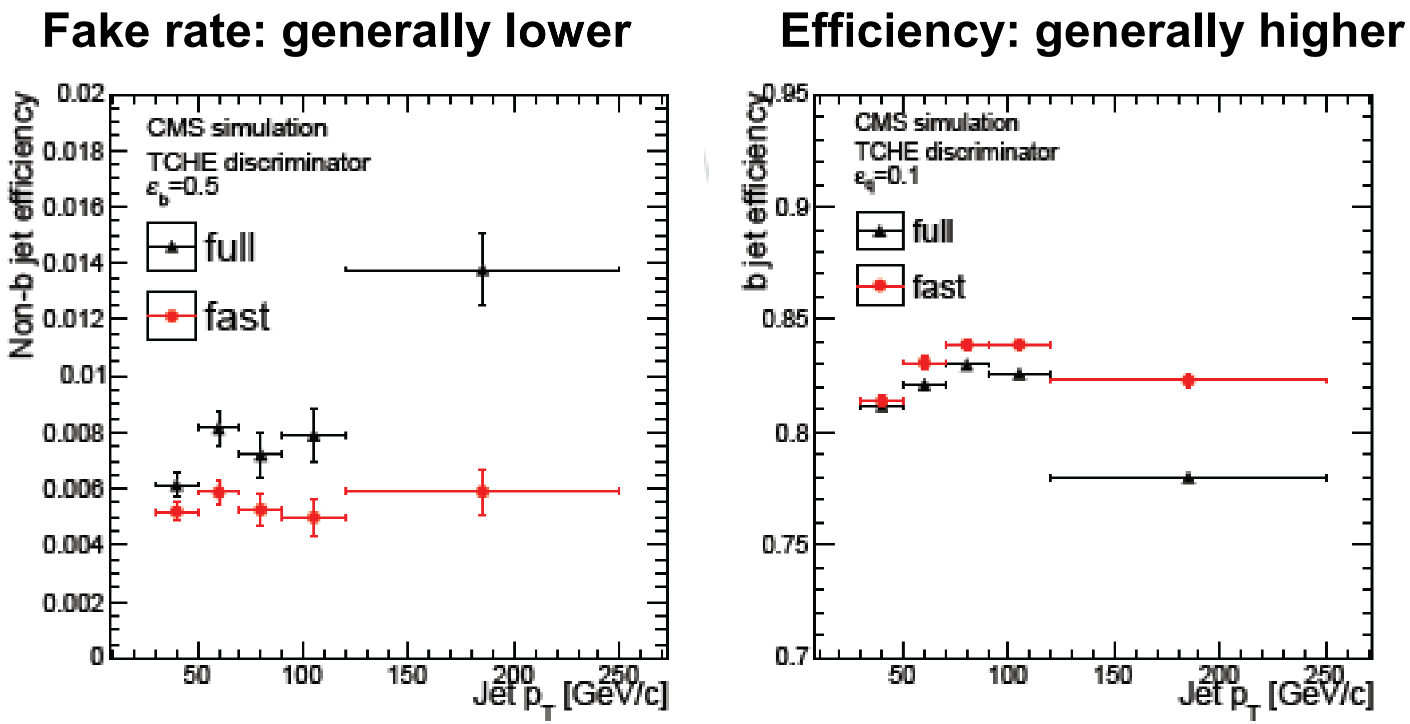
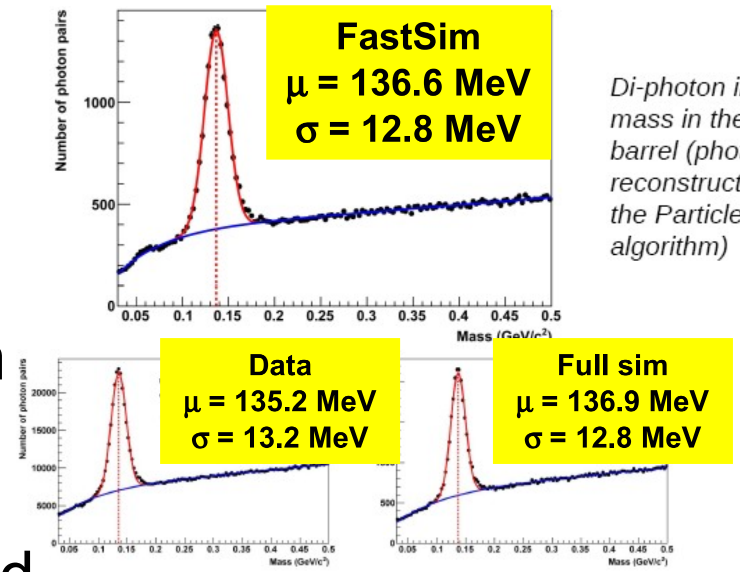


Figure 24: Comparison of the b-tagging performance between full and fast detector simulation for the track counting high efficiency algorithm. Left: mistag rate versus jet p_T at fixed b-tag efficiency of 50%. Right: b-tag efficiency versus jet p_T at fixed light flavor mistag rate of 10%.

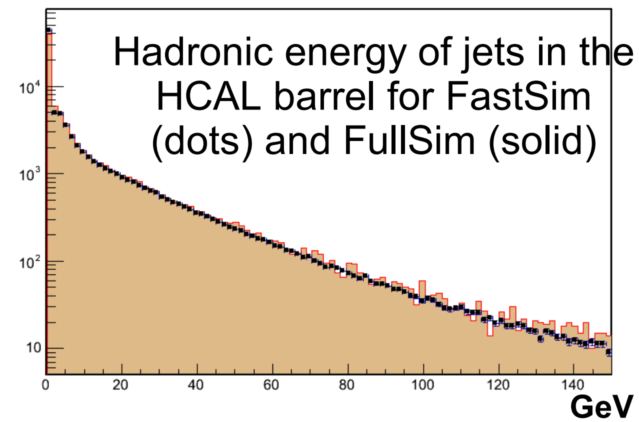
- Discrepancies attributed to:
  - No fake tracks
  - No cluster merging (important source of track inefficiency in dense high-momentum jets)

Calorimeters

- Showers simulated à la GFLASH
- ECAL:
  - Treated as a homogeneous medium
  - Cracks, leakage, magn.field as FullSim
- HCAL:
  - Hadronic response and resolution tuned to single pions in FullSim
  - Extensively validated with test-beam data and isolated tracks from 2010 data
  - New tuning in 2013 (versus FullSim) after transition to digitizers: at SimHit level instead of RecHits, to decouple from electronic effects

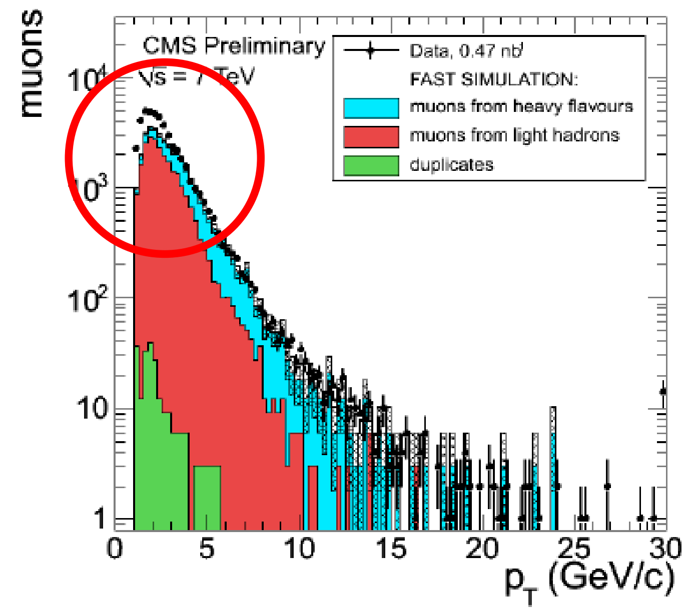


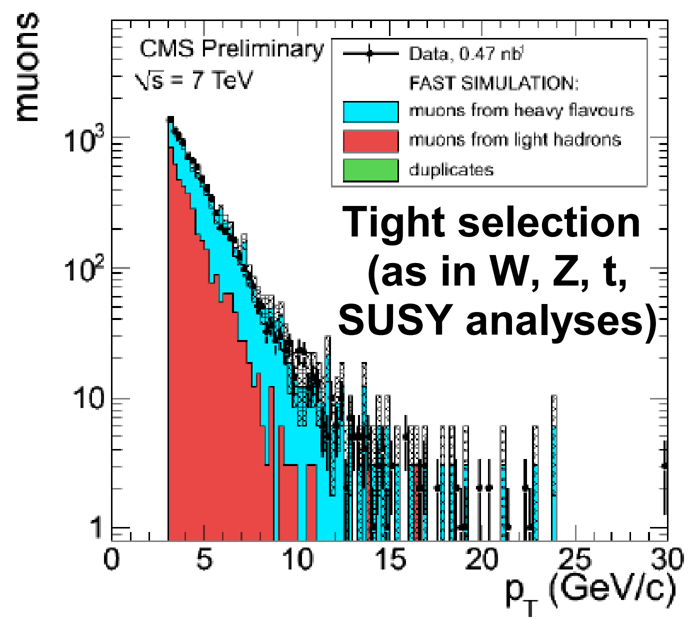
Di-photon invariant mass in the ECAL barrel (photons reconstructed with the Particle Flow algorithm)



Muons

- Muons are the only generated particles propagated to the muon chambers
  - Mult.scattering and dE/dx by ionization
  - Muons from hadronic decays propagated only if the decay is in the tracker volume; no late decays and no punch-through
  - Calo deposits are parametrized
  - No bremsstrahlung, no delta rays
- Same geometry as FullSim
- Standard digi+reco is applied
  - No need for short-cuts in outer tracking: multiplicity is low





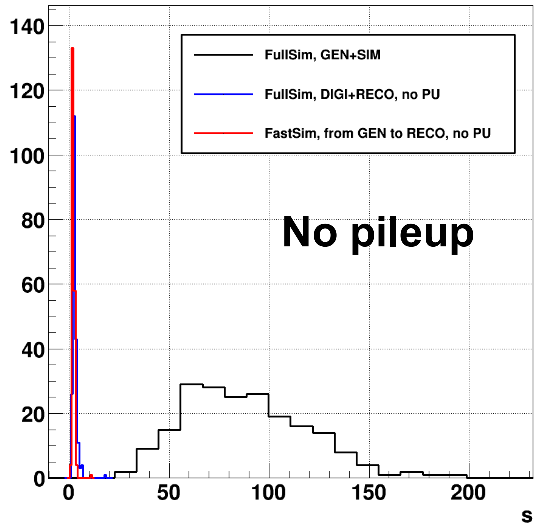
Pile-up

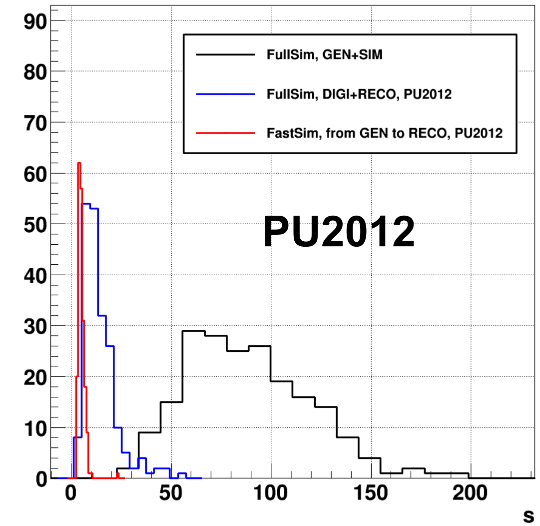
- Number of pile-up collisions can be diced from a Poissonian or from a histogrammed distribution, as in FullSim (see Mike's talk)
- Difference with FullSim: they add SimHits, we add generated particles and we treat signal and PU particles at the same time
  - Original rationale (when we expected <PU>~5 in Run-I): rerunning everything for all PU particles is affordable with FastSim; advantageous because we can use very slim minimum-bias files
- Future scenarios w/ large <PU> make us reconsider late mixing
  - Plan: mix calo & muon SimHits (as FullSim does), and already reconstructed tracks (saving considerable cpu time and memory)
  - Appealing possibility: mix FastSim signal with FullSim minimum-bias
- So far, only in-time PU is simulated
  - Ongoing work to include out-of-time PU; now possible thanks to integration with the digitizers in calorimeters and muon chambers
  - Pixels and strip: narrow pulse shapes minimize OOT effects

CPU time, an example

- Latest CMSSW pre-release; numbers are in **seconds/event**
- "PU2012" is the PU profile of 2012 data; only in-time for FastSim, also OOT for FullSim
- Machine: 64 bits, Scientific Linux 5.9, Intel(R) Xeon(R) CPU L5640 @ 2.27GHz

ttbar @ 13 TeV	FullSim no PU	FastSim no PU	FullSim PU2012	FastSim PU2012
Generator (Pythia)	0.02	same	same	same
Detector simulation	88	0.20	same as no PU	0.88
Digitization	0.7	0.24	3.2	0.30
Reconstruction	1.9	1.2	10.6	2.8



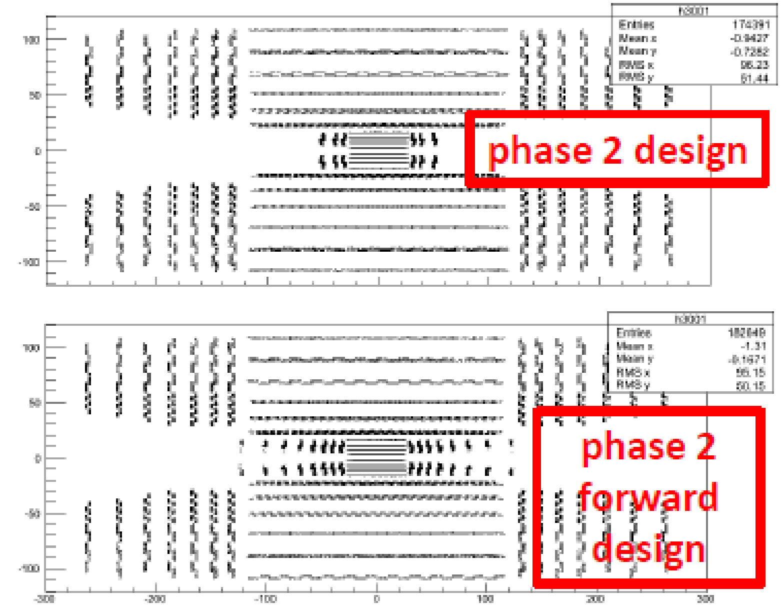


Examples of usage in CMS

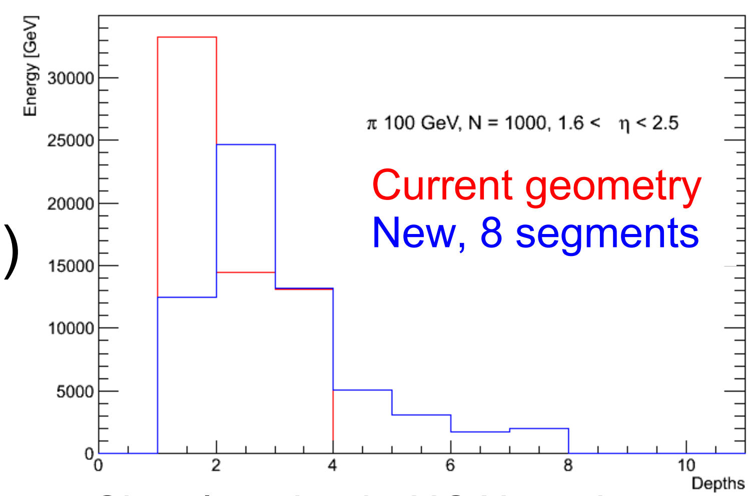
- Centralized high-statistics productions
  - SUSY “simplified model scans”, used as signal samples for all publications of the CMS SUSY group since 2011 (FullSim used to validate a few points in the scan)
  - Top (pair and single), W+jets, Z+jets samples with non-default parameters (masses, Pythia tune, QCD scales), for the evaluation of systematics
- Private productions
  - Signals of interest of one/few groups (e.g., black holes)
  - Template extraction for scans of parameter of interest
  - MVA training
  - Enhance statistics of interesting events inside large backgrounds; filtering at RECO level is possible
  - Upgrade studies: see next slide

Upgrades

- Using the geometry files from FullSim or Reco: better realism but flexibility is tough
  - Important activity in 2012-2013 towards generalizations to arbitrary geometries
- Flexible tracking geometry
  - The tkLayout tool (link) is used to generate arbitrary Reco geometries; from that, we automatically derive the interaction geometry
- Endcap and forward calorimeter upgrades
  - Effects of radiation damage have been studied with FastSim (amplitude degradation)
  - Made geometry, segmentation and material properties fully configurable



Tracker hit “radiography”



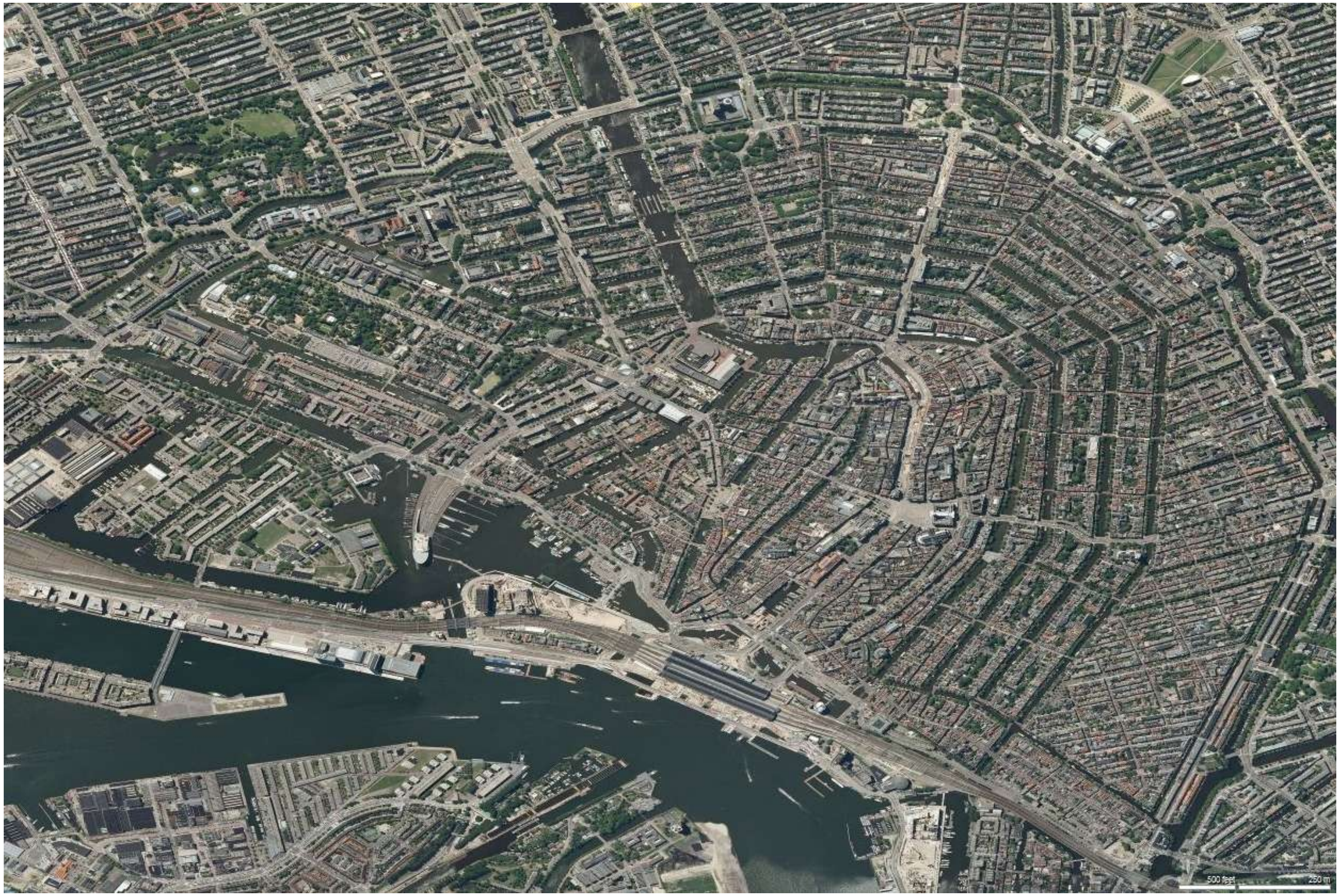
Signal vs depth, HCAL endcap

Conclusions

- The CMS Fast Simulation is designed to achieve a O(%) accuracy with O(second/event) execution time
- Current direction of development is towards a tighter integration with FullSim (common use of the digitizers, similar treatment of pileup) and more flexibility
- An important tool, complementary to GEANT4
- Bibliography:
  - Data/FastSim comparisons: *CMS-DPS-2010-039*
  - B-tagging FullSim/FastSim comparisons: *CMS-BTV-11-002*
  - Description as of 2012 (CHEP2012 proceedings): *R.Rahmat and A.Giammanco, J. Phys.: Conf. Ser. 396 (2012) 062016*

Thanks to: Liz Sexton-Kennedy, Fabio Cossutti, David Lange, Patrizia Azzi, Luca Malgeri, Vladimir Andreev, Alexander Ledovskoy, Dimitrios Nikolopoulos, and all FastSim developers

Backup



CPU time, two more examples

- Latest CMSSW pre-release; numbers are in **seconds/event**
- "PU2012" is the PU profile observed in 2012; only in-time for FastSim, also OOT for FullSim
- Machine: 64 bits, Scientific Linux 5.9, Intel(R) Xeon(R) CPU L5640 @ 2.27GHz

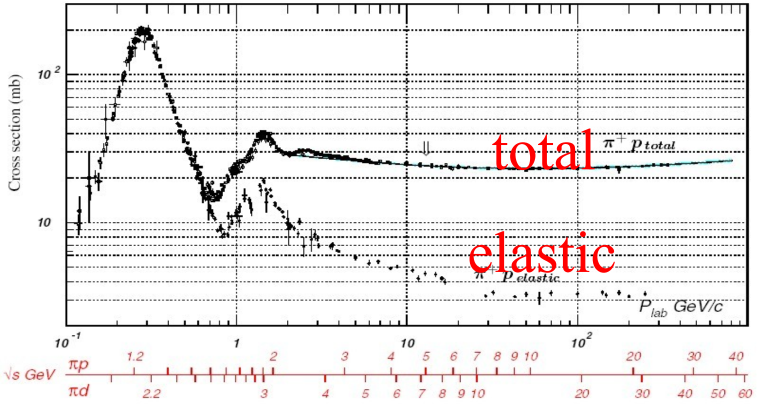
Z→ee @ 13 TeV	**FullSim no PU**	**FastSim no PU**	**FullSim PU2012**	**FastSim PU2012**
Generator (Pythia)	0.02	same	same (*)	same (*)
Detector simulation	52	0.11	52 (*)	0.78
Digitization	0.5	0.22	3.1	0.26
Reconstruction	0.9	0.67	8.9	2.1

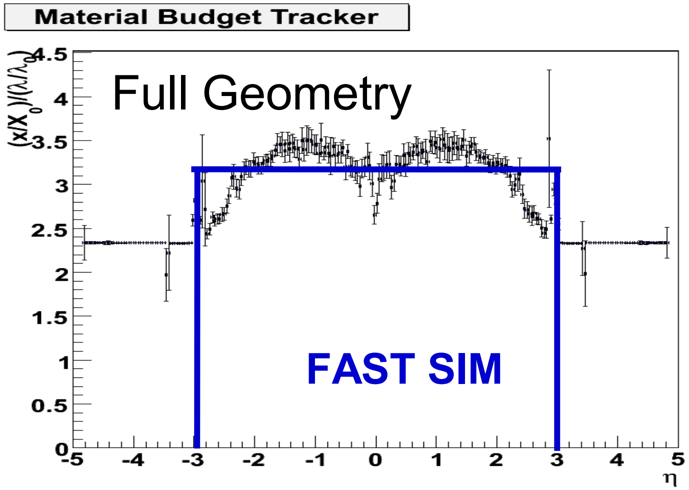
Minimum bias @ 13 TeV	**FullSim no PU**	**FastSim no PU**	**FullSim PU2012**	**FastSim PU2012**
Generator (Pythia)	0.01	same	same (*)	same (*)
Detector simulation	18.5	0.04	18.5 (*)	0.69
Digitization	0.4	0.21	3.0	0.26
Reconstruction	0.5	0.46	6.9	1.9

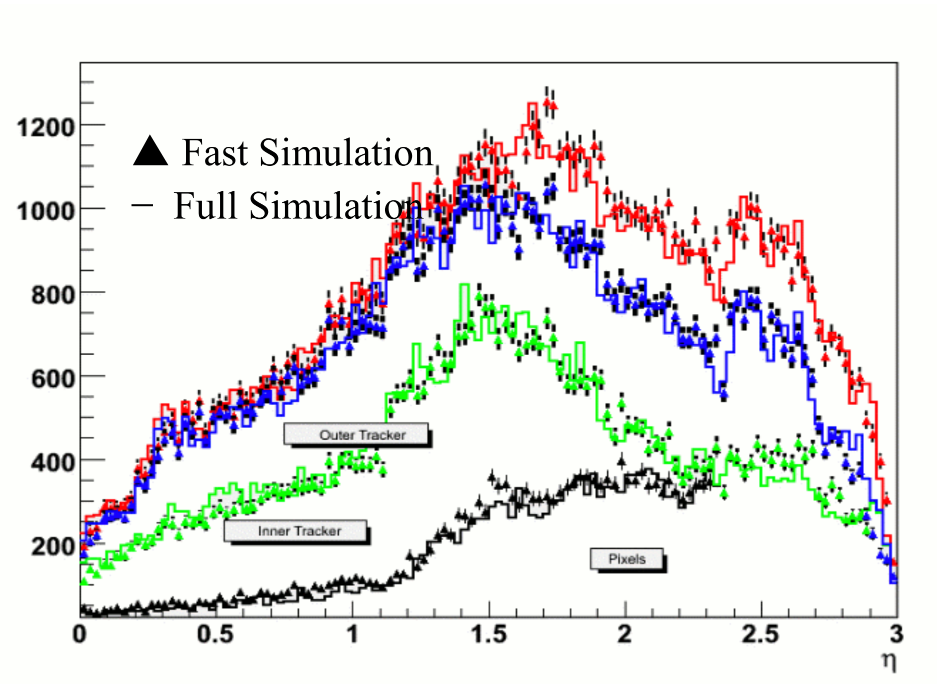
(*) unaffected by pileup because mixing is executed after this step

Nuclear interactions

- Interaction probability parametrized from PDG
- Layer thickness considered constant in η
- Shower library used for the interaction products
- Libraries available for 9 different hadrons, several bins in range 1<E<1000 GeV

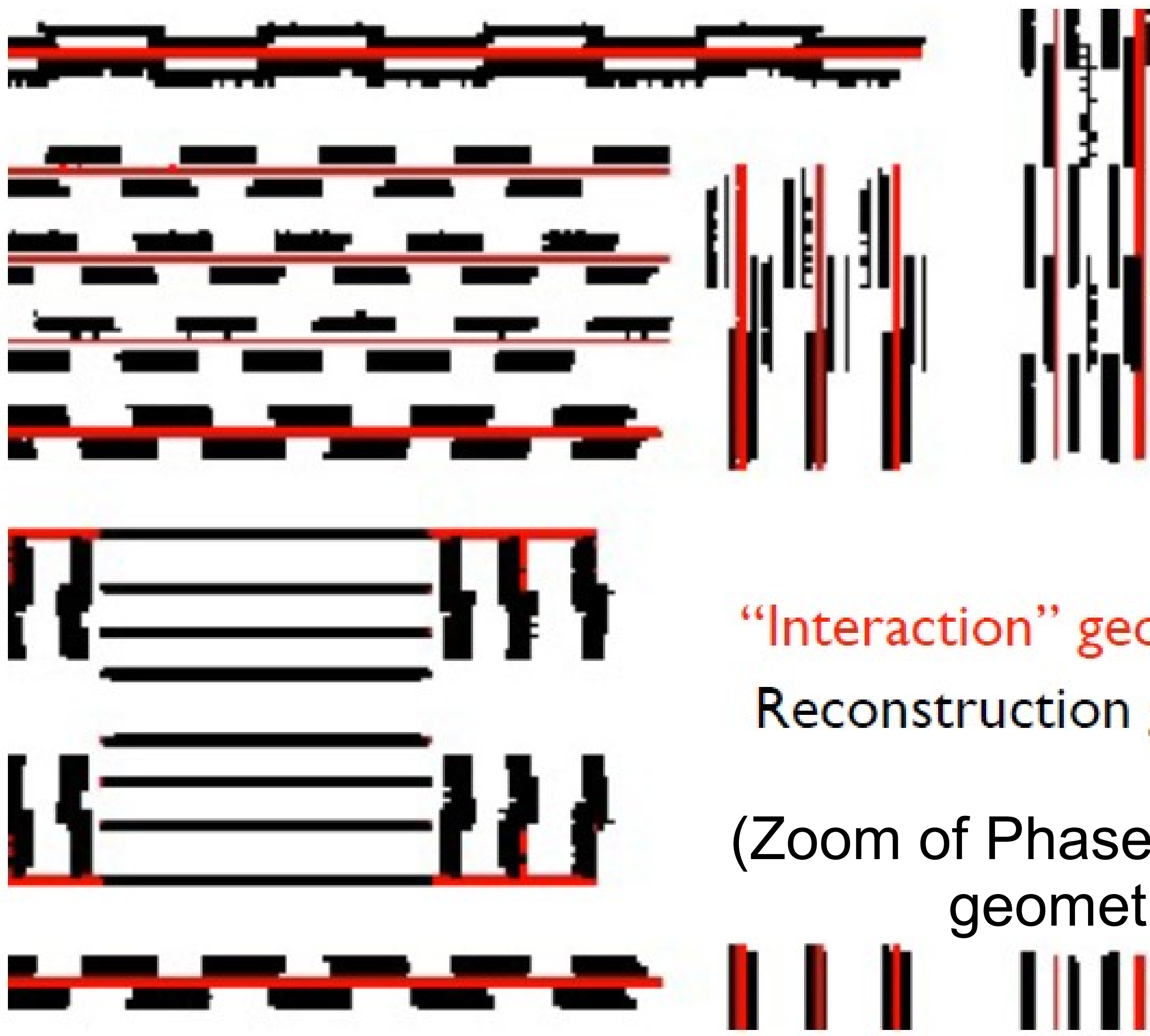






Number of nuclear interactions for 500K 15 GeV pions

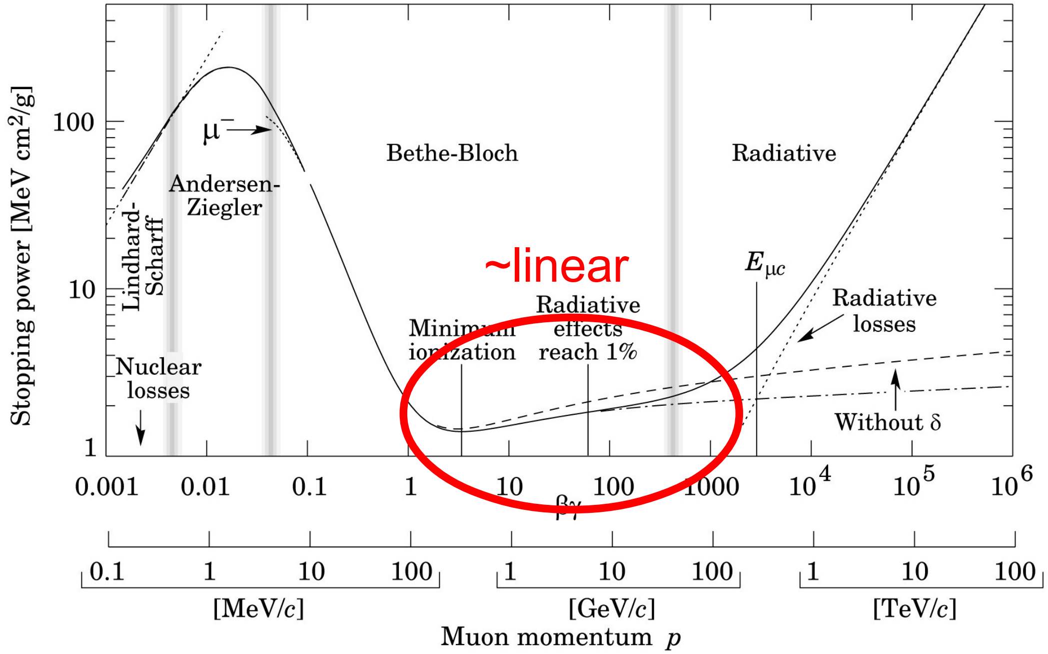
Interaction and reconstruction geometries



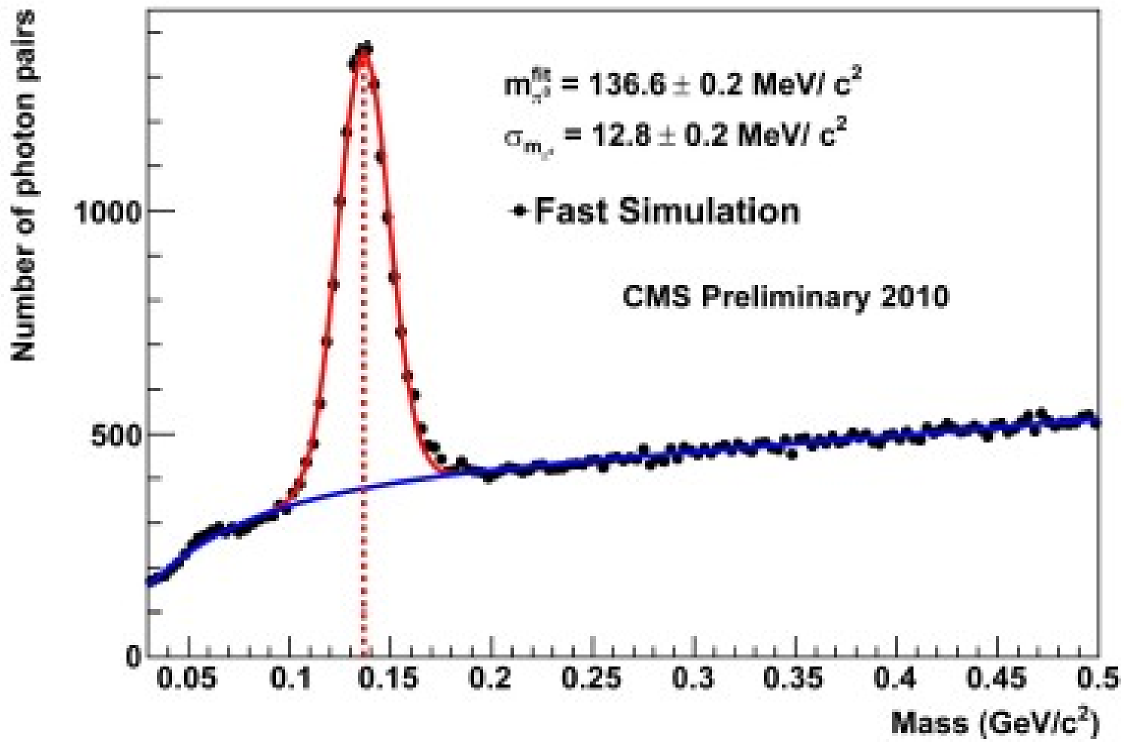
(Zoom of Phase-I Upgrade geometry)

Emulation of δ-rays effects for μ

- δ-rays emitted at the entry of a cell may cause the hit to get corrupted (⇒inefficiency) or an after-pulse (~harmless)
- Log of hit inefficiency is found to be pretty linear with log(P) for DT and CSC, as expected if the cause are δ-rays; almost no P dependence found in RPCs, as expected due to their coarser spatial resolution
- Hit inefficiency has been parametrized as a function of log(P) for DT and CSC

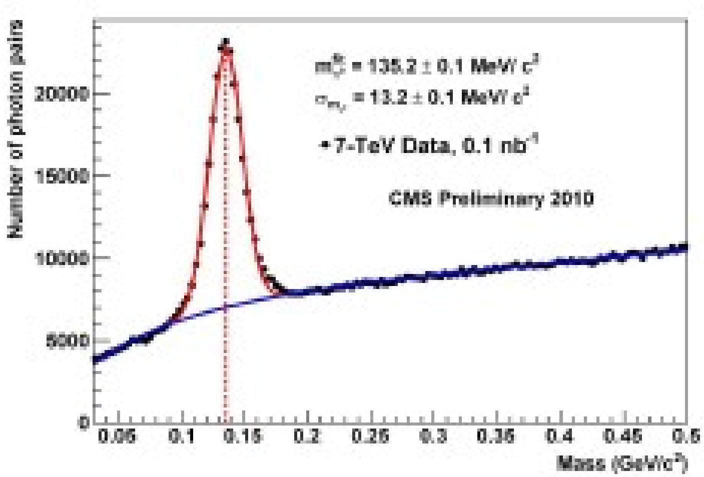


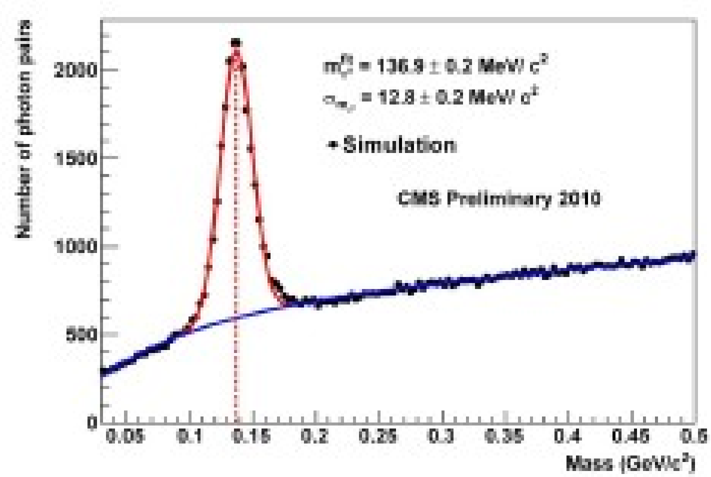
Di-photon invariant mass



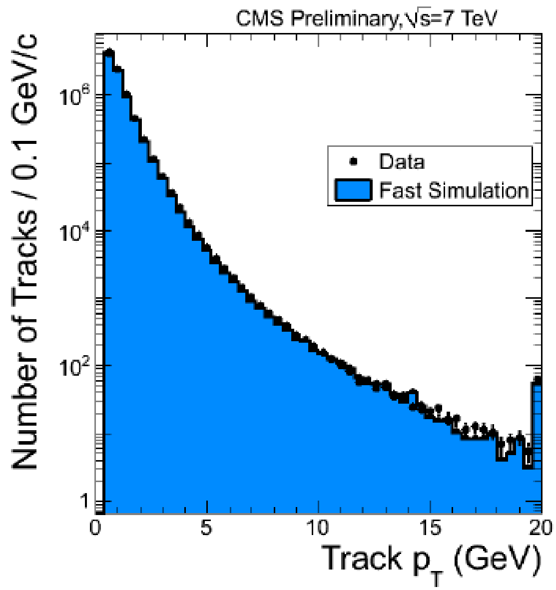
Di-photon invariant mass in the ECAL barrel (photons reconstructed with the Particle Flow algorithm)

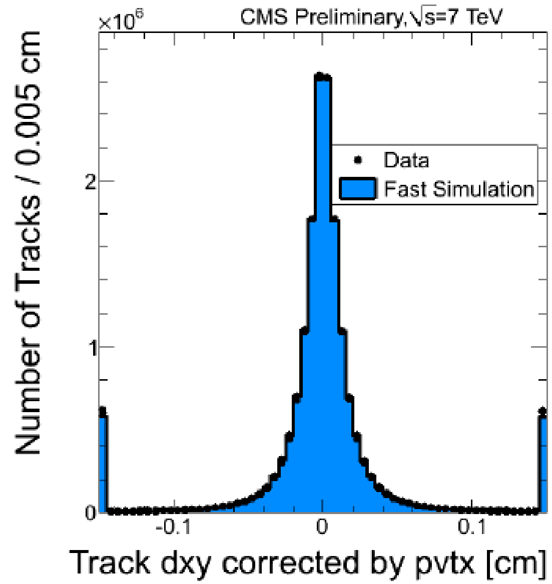
CMS-DPS-2010-039

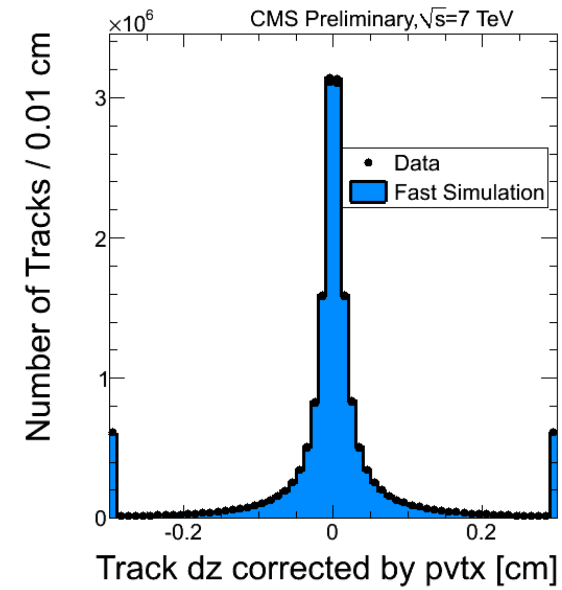


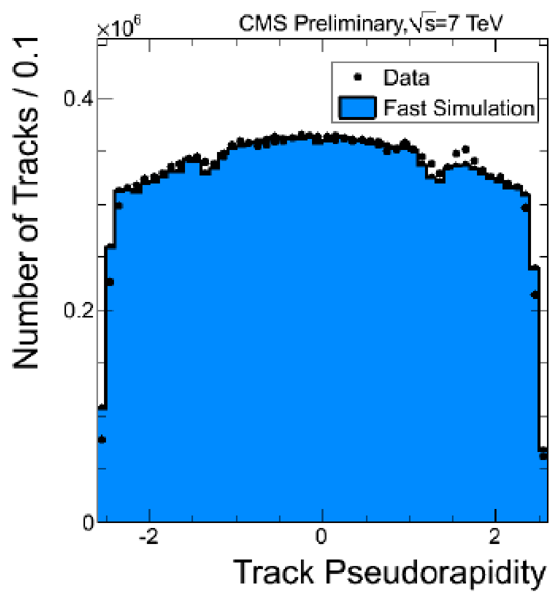


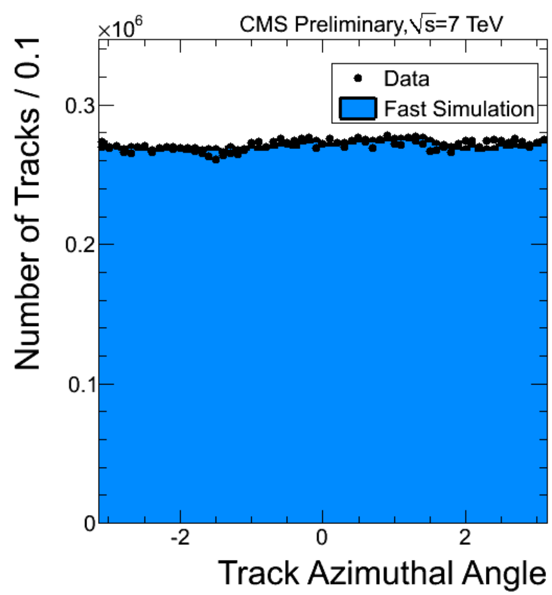
Track kinematics









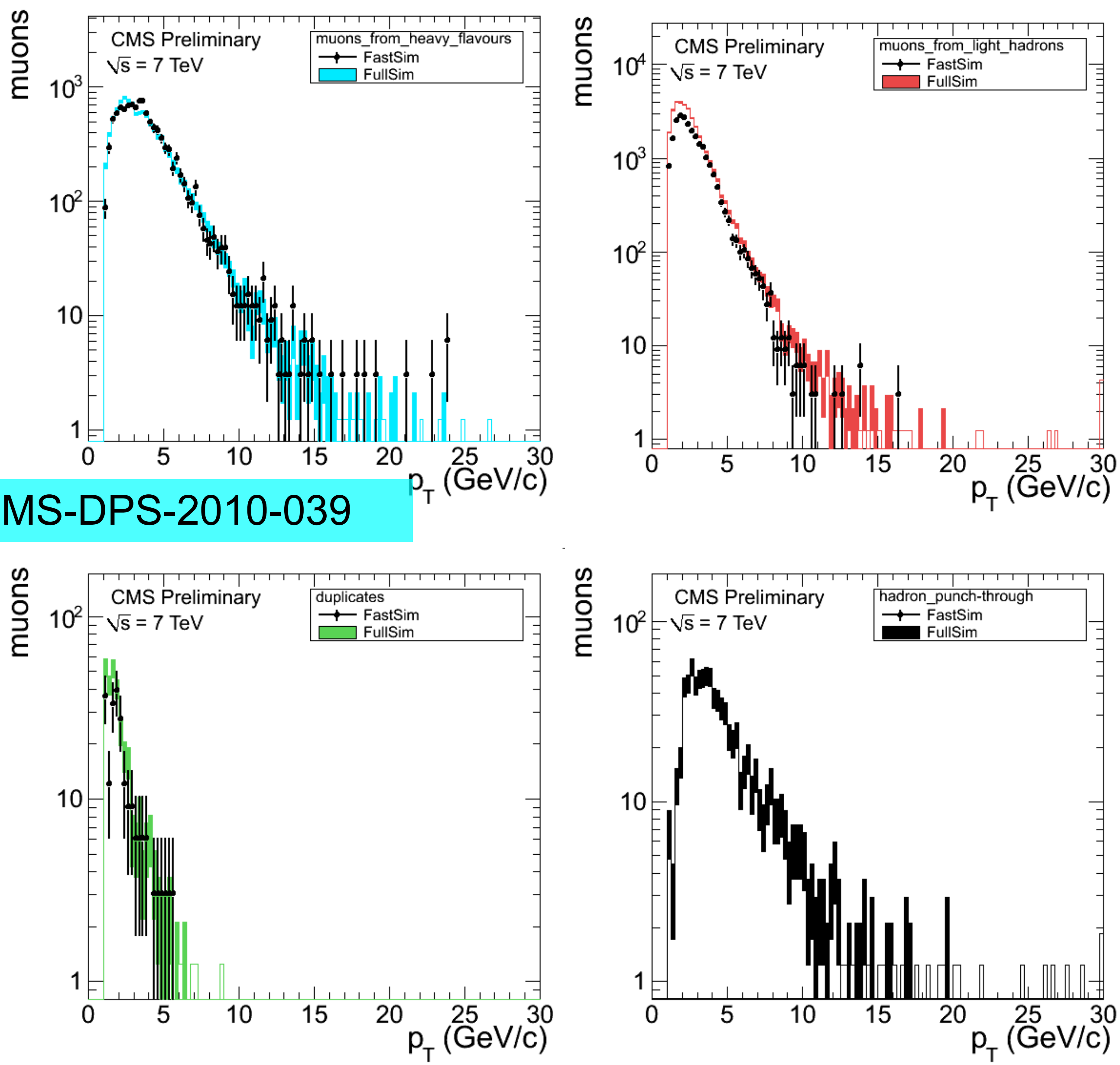


Comparison to early 2010 data collected with a minimum-bias trigger; track selection:

- p_T>0.5 GeV
- "high quality" flag
- $\Delta p_T/p_T$<5%
- within 10 sigma of beam spot

CMS-DPS-2010-039

Breakdown of the p_T distribution according to the origin of the muon



CMS-DPS-2010-039

p_T distribution (Global Muons)

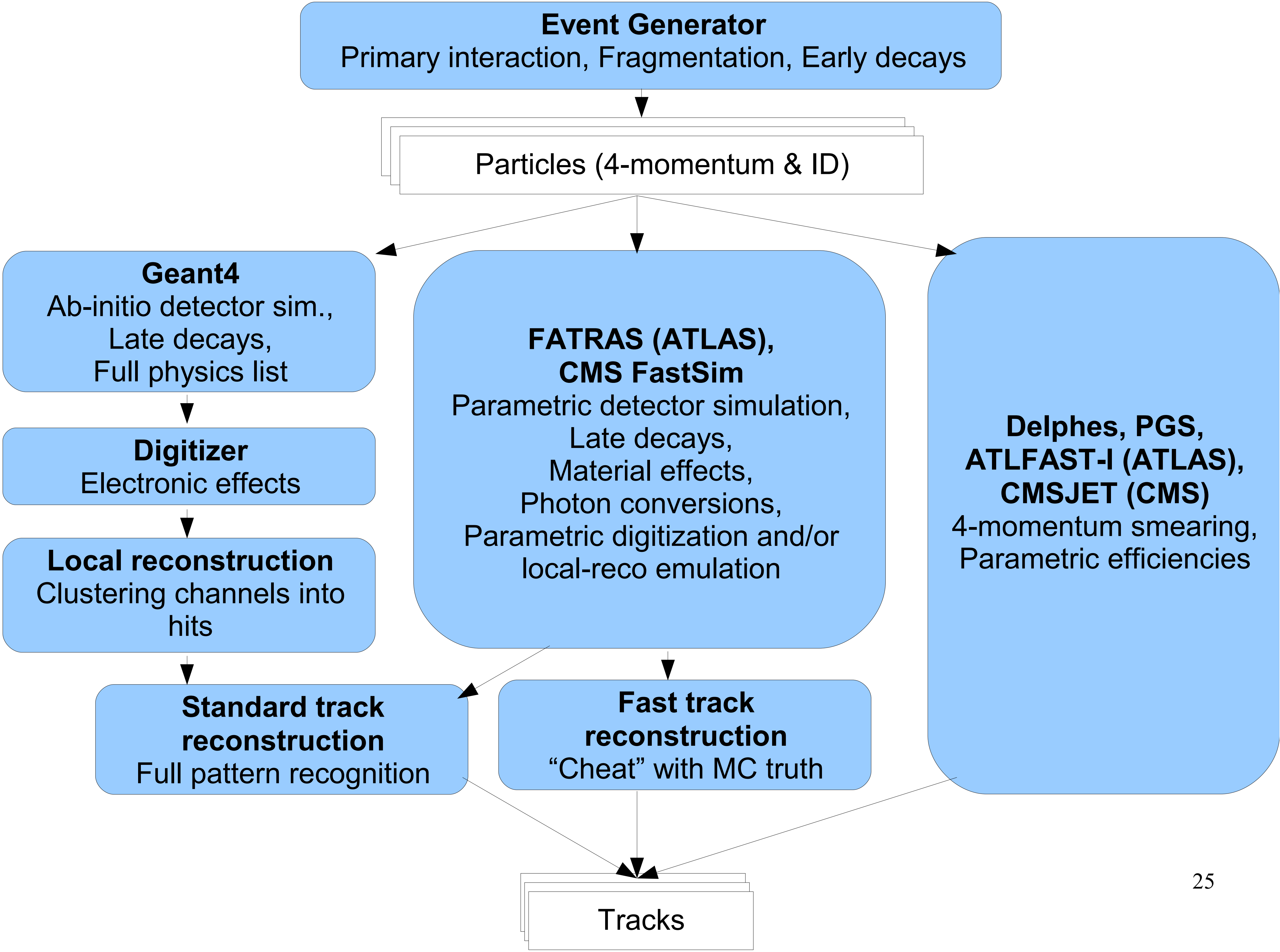
Full (histo) vs Fast (points) contributions, separated per muon type:

- Prompt muons

- Decays in flight

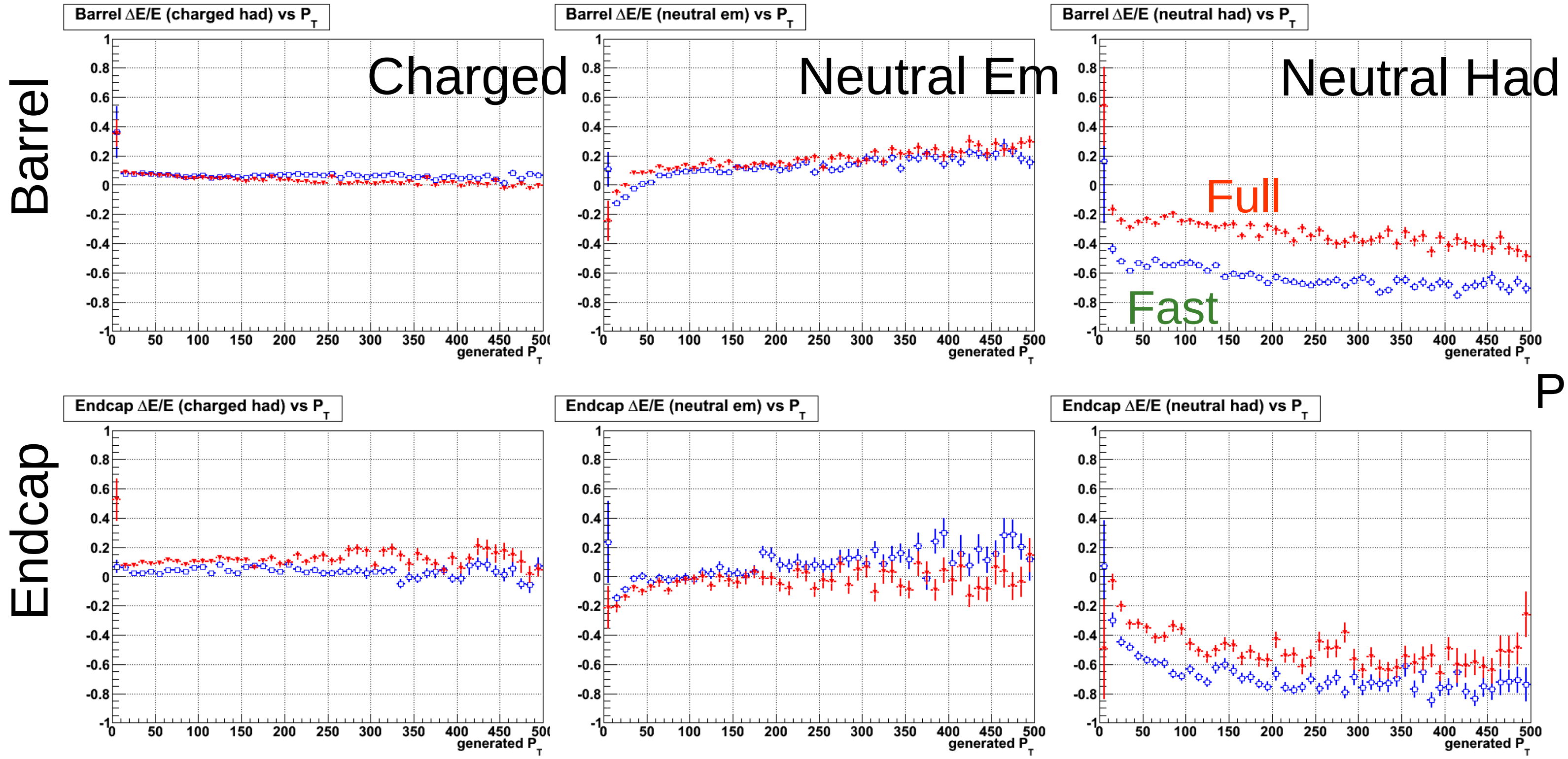
- Fakes (ghosts)

- **Punch-through** (only for the Full Simulation, as the Fast Simulation does not simulate it)


Event Generator
Primary interaction, Fragmentation, Early decays
Particles (4-momentum & ID)
Geant4
Ab-initio detector sim.,
Late decays,
Full physics list
FATRAS (ATLAS),
CMS FastSim
Parametric detector simulation,
Late decays,
Material effects,
Photon conversions,
Parametric digitization and/or
local-reco emulation
Delphes, PGS,
ATLFAST-I (ATLAS),
CMSJET (CMS)
4-momentum smearing,
Parametric efficiencies
Digitizer
Electronic effects
Local reconstruction
Clustering channels into
hits
Standard track
reconstruction
Full pattern recognition
Fast track
reconstruction
“Cheat” with MC truth
Tracks

Particle Flow

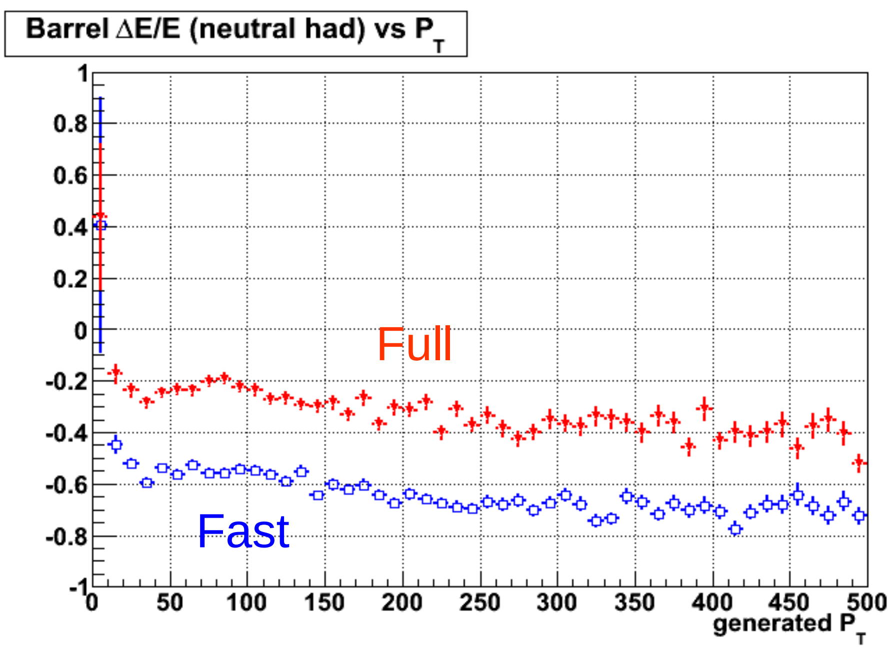
- Optimal use of tracking and ECAL for particle-id before jet clustering
- Used by default for jets, MET and isolation in high-P_T CMS analyses
- ~70% of jet energy is carried by charged, only ~10% in HCAL
- A discrepancy in the neutral hadronic component was observed



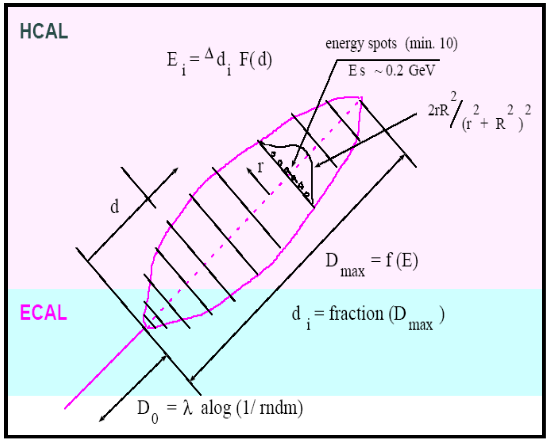
Private plots

In a nutshell

- It's not really a “missing” neutral component: it has to do with how PF accounts for far-away energy clusters, deciding whether to attribute them to a nearby charged particle or to create a “neutral had” cluster
- Lateral shower is tuned on FullSim, but cannot account for outliers
- As a short-term patch, we now can create extra clusters at PF level, fixing this variable while keeping (or improving) agreement elsewhere



Current Hadron Shower model based on a 6 years old approximation of GFlash:



Complete GFlash implementation in FastSim could be a long-term solution