Strategies for Modeling Extreme Luminosities in the CMS Simulation

Elizabeth Sexton-Kennedy & Mike Hildreth

Fermilab & Université de Notre Dame du Lac

Representing the CMS Simulation Group

Overview

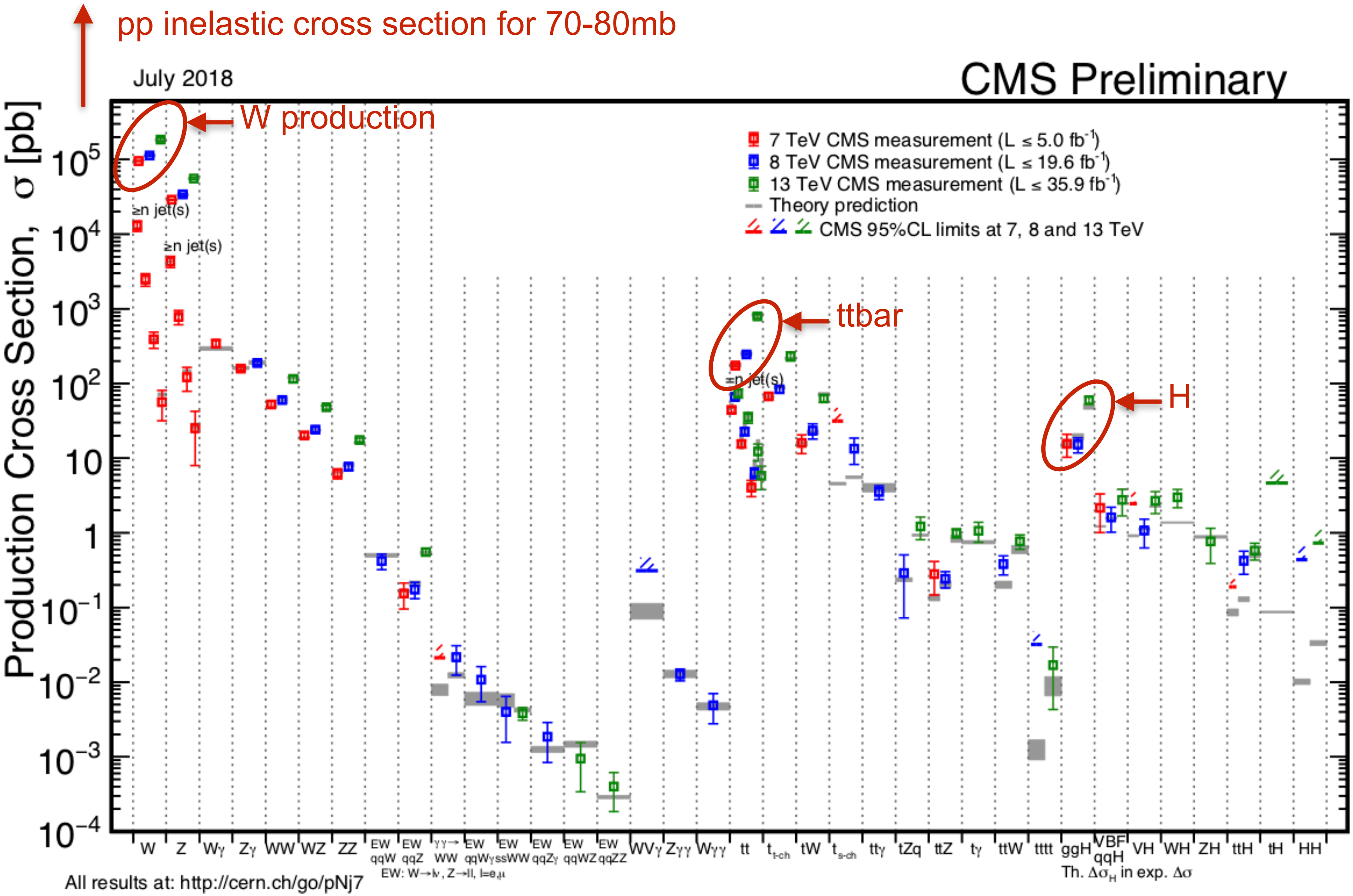
- The physics problem
- Simulation Data Flow
- Geant Optimizations
- Reworking Digitization
- Simulation Technical Performance

Physics Problem

- The fundamental physics problem at hadron colliders like the Large Hadron Collider (LHC) is that the total inelastic cross section is orders of magnitude larger than the cross section for interesting physics worth studying.
- LHC thus must be extraordinarily bright, so bright that many proton pairs interact during the same crossing.
  - Crossings occur 40 million times per second (25ns spacing)
  - “trigger system” selects about 1000 of these per second as interesting
- Triggered crossings contain the other uninteresting pairs as well
  - The interesting pair is called the “hard scatter interaction”
  - The noise pairs that came along for the ride are called in time “pile up” interactions.
- Detector Effects
  - The signal integration time of most detectors is greater then 25ns bunch spacing leading to out of time pile up
- Our simulations must match a triggered crossing including detector effects.

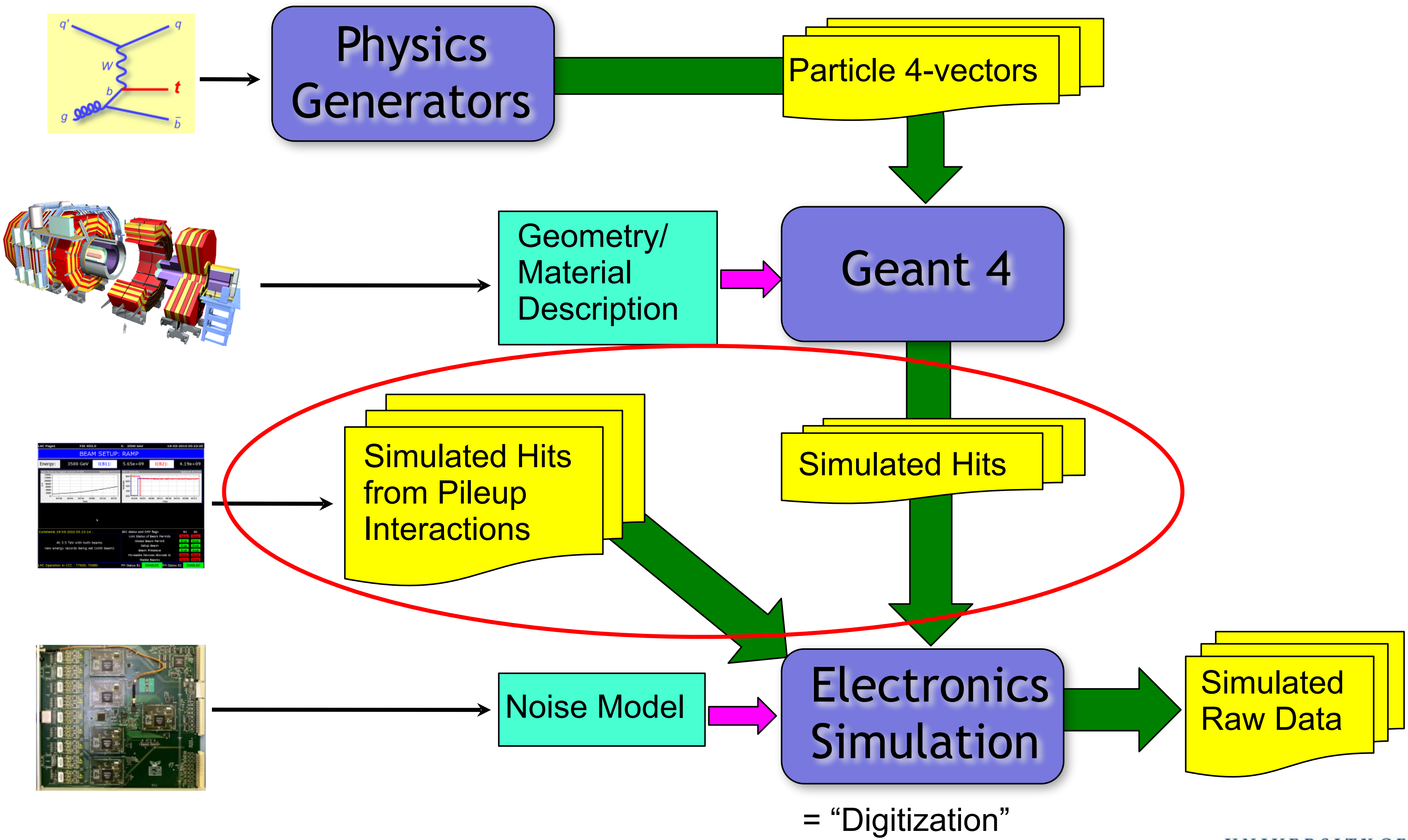
Collider Physics Cross Sections

pp inelastic cross section for 70-80mb



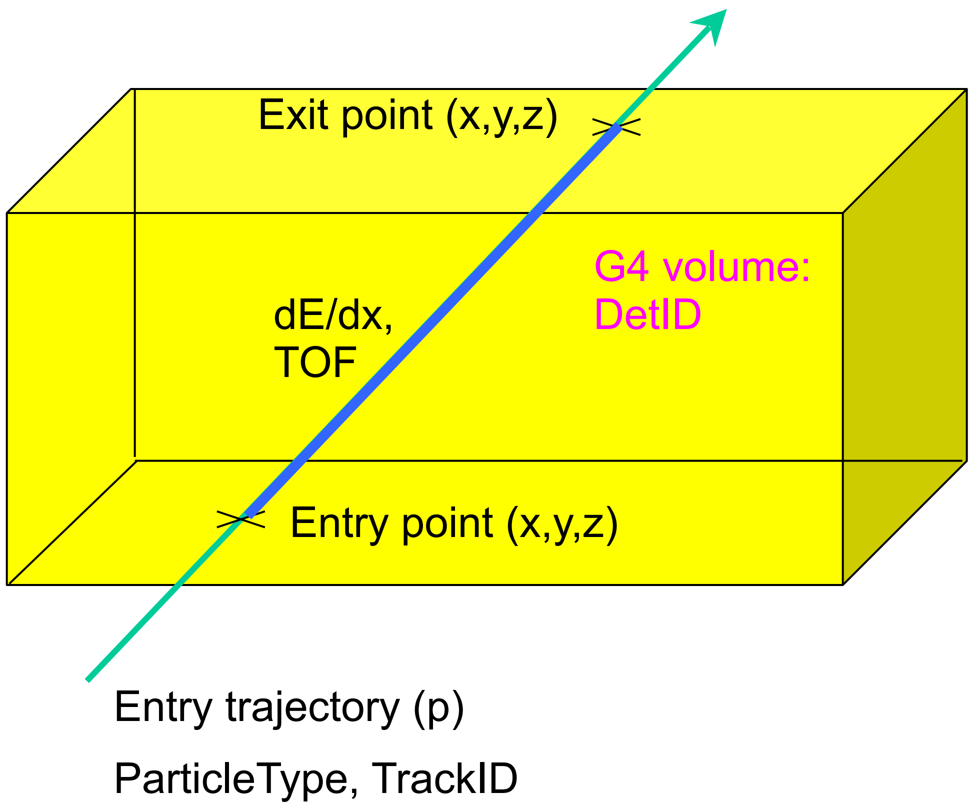
University of Notre Dame
Fermilab

Classic Pileup Simulation Data Flow



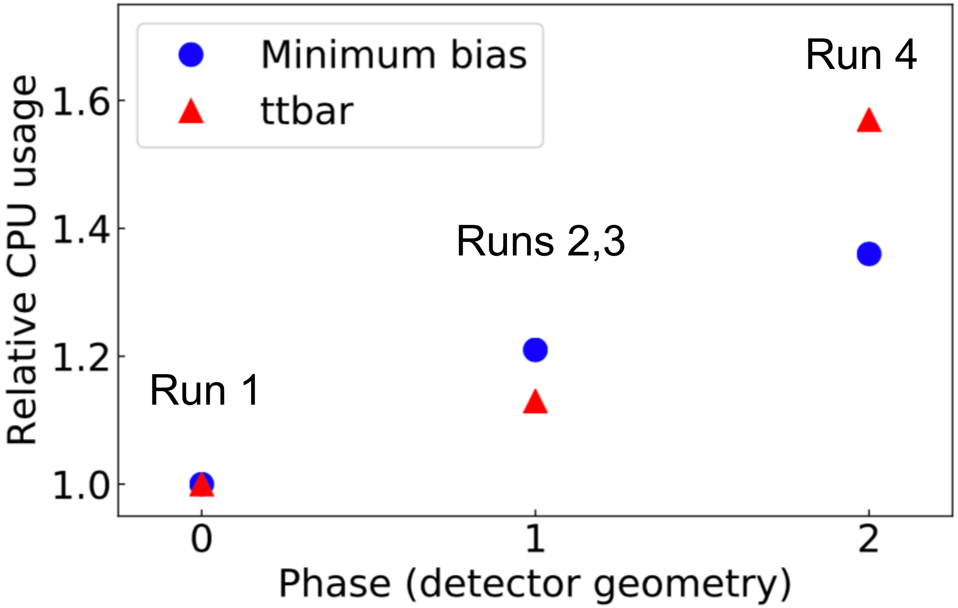
SimHits: inputs to Digitization

- SimHits are the quantum of Simulation information, generated by Geant4
- Each SimHit contains (more or less) the same information about a particle traversing a simulation volume (calorimeter hits carry a bit more info.)



Geant4 Optimizations

- The time per event needed to generate Geant4 simulation hits does however depend on the complexity of the detector geometry being modeled:



- Over the years CMS has adopted and physics validated a number of optimizations in order to reduce the time per event by almost a factor of 5.
- The table below, from the community white paper*, summarizes the effect of each.

Configuration	Relative CPU usage Minbias	$t\bar{t}$
No optimizations	1.00	1.00
Static library	0.95	0.93
Production cuts	0.93	0.97
Tracking cut	0.69	0.88
Time cut	0.95	0.97
Shower library	0.60	0.74
Russian roulette	0.75	0.71
FTFP_BERT_EMM	0.87	0.83
All optimizations	0.21	0.29

* J. Apostolakis et al. (HEP Software Foundation) (2018), 1803.04165

Relative Importance of Workflow Steps

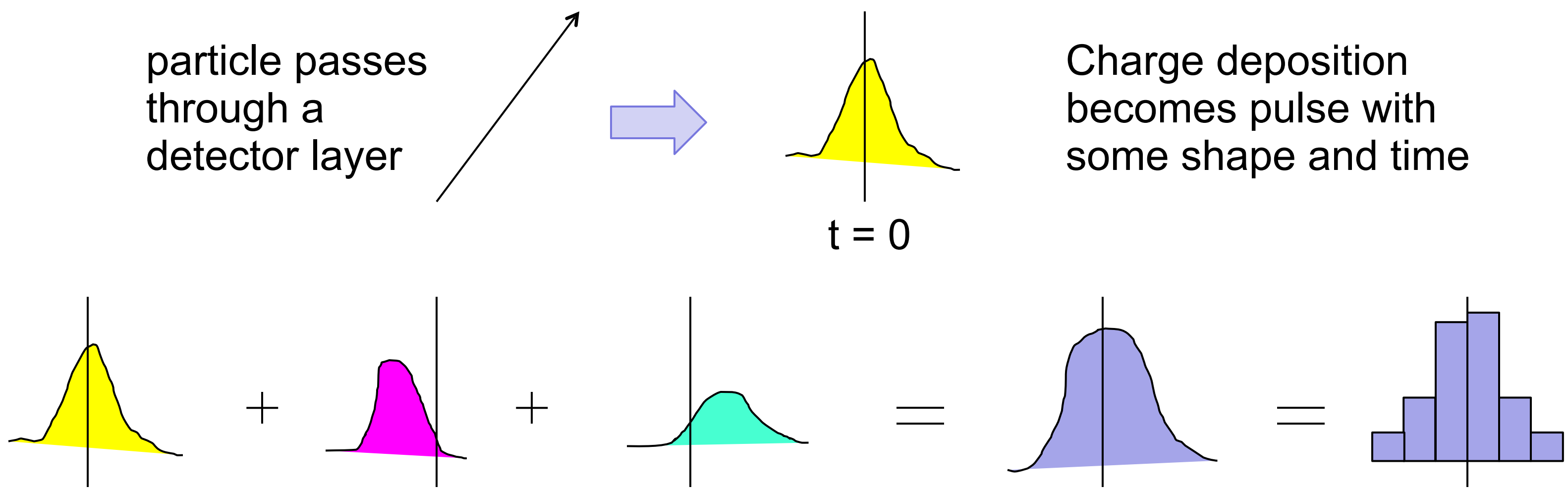
- The time per event needed to generate Geant4 simulation hits in the simulation workflow does not depend on pileup by construction.
- For our benchmark process, in our master branch:
  - ttbar: 48 s/ev/core [was 54 s/ev/core in 2017] at εCPU=98% using 8 cores
- The rest of the steps in the workflow do.
- These benchmark performance numbers illustrate this:

tt [vs 10.1X*] in s/evt	PU20	PU35	PU50	PU70	PU100	PU140
pre-PREMIX	27 [31]	44 [51]	61 [68]	83 [84]	116 [119]	89
DIGI+HLT	11 [14]	13 [20]	14 [29]	18 [46]	25 [19]	38
RECO	10.1 [9.9]	15.5 [17.7]	25.8 [30]	58 [52]	296 [308]	1061
miniAOD	2.1 [2.3]	2.5 [3.1]	3.1 [4.3]	4.1 [4.9]	5.1 [6.7]	16
RECO+miniAOD	11.5 [11.0]	17.4 [19.4]	29 [32]	61 [56]	303 [316]	1081

- Note that that by pileup 50 Geant is no longer the slowest step. Pileup library generation is. At 70 Reconstruction also takes longer.

Digitization (= Electronics Simulation)

- "Digitization" as CMS defines it is the process of collecting all of the energy depositions of the SimHits, including their arrival time structure, and forming appropriate representations of the electronic signals that would be generated.
- In pictures:



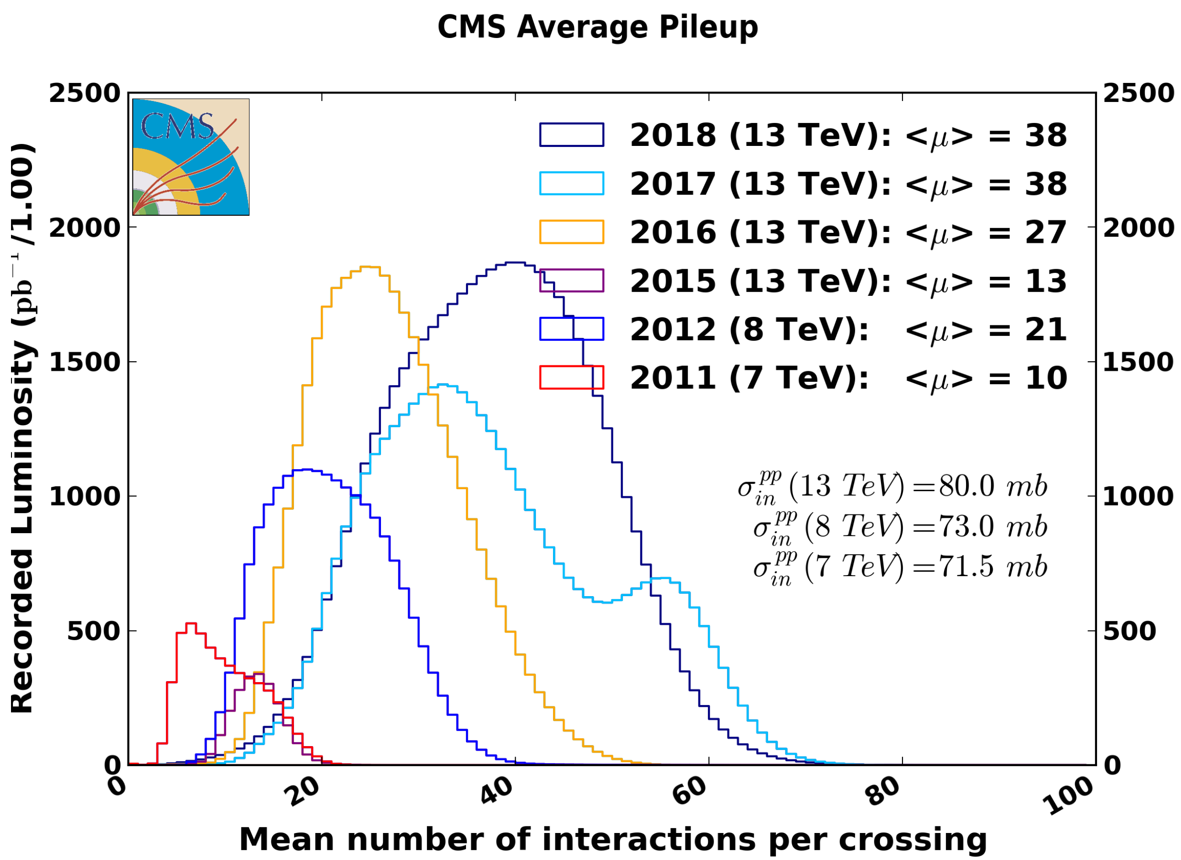
summation usually done in "natural" units (photo-electrons, fC, other representation of charge)

Pileup Simulation Overview (in words)

- For each event, the instantaneous luminosity is chosen from the input distribution at random
  - The number of in- and out-of-time interactions to be overlaid are selected individually from a poisson distribution based on the chosen luminosity and the total inelastic cross section
    - σ_{tot} is a function of center of mass energy
    - Pool of pythia minbias (single interaction) events is used as input
  - Out-of-time interactions are simulated for each beam crossing that is "in scope" for a given production run
    - can do any arbitrary bunch configuration in 25ns steps
      - times of Geant SimHits are shifted to match bunch assignment
      - Digitization simulation considers hit times for pulse shapes
- Collection of Geant SimHits from all of the minbias events and hard-scatter "signal" event are merged, then processed by digitization/ electronics simulation

Pileup Simulation Overview

- Pure MC inputs used to simulate pileup interactions
- Distribution of the number of interactions per beam crossing chosen in advance to simulate a desired luminosity profile:

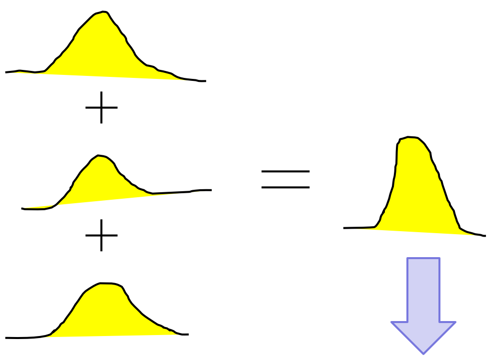


UNIVERSITY OF NOTRE DAME
Fermilab

PreMixing in Pictures

First Step: combination of signals from minbias events, no noise

PreMixing:



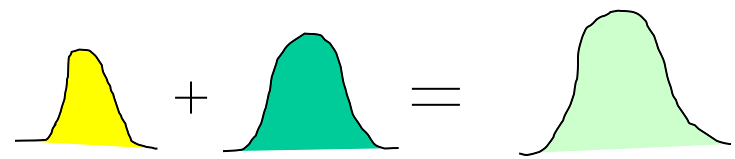
Special Digi format (special "lossless" packing required)

PreMixing workflow

This is a crucial piece! Physics studies required to understand how low-level signals need to be saved

Second Step: signals from minbias, hard scatter event combined

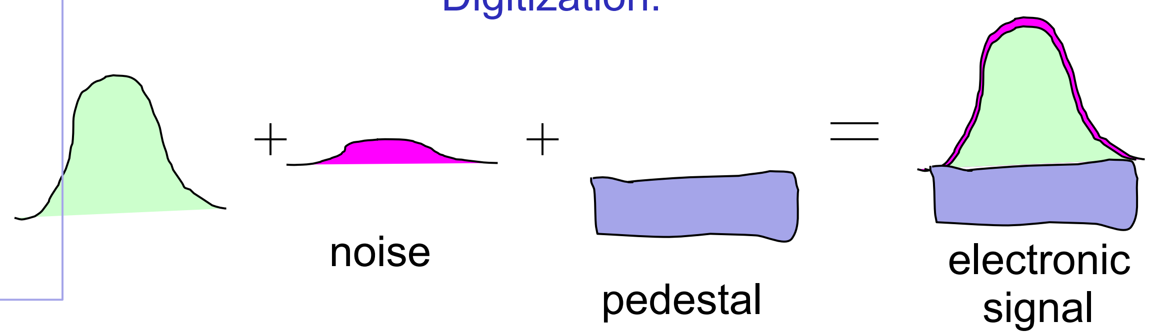
Signal Combination:



Third Step: signals are combined with noise, pedestals in electronics simulation to mimic a "real" electronic signal for each channel ("normal" digitization)

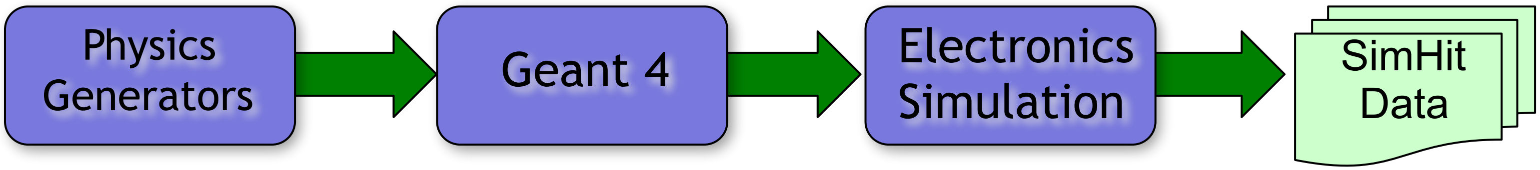
Mixing/Digi workflow

Digitization:



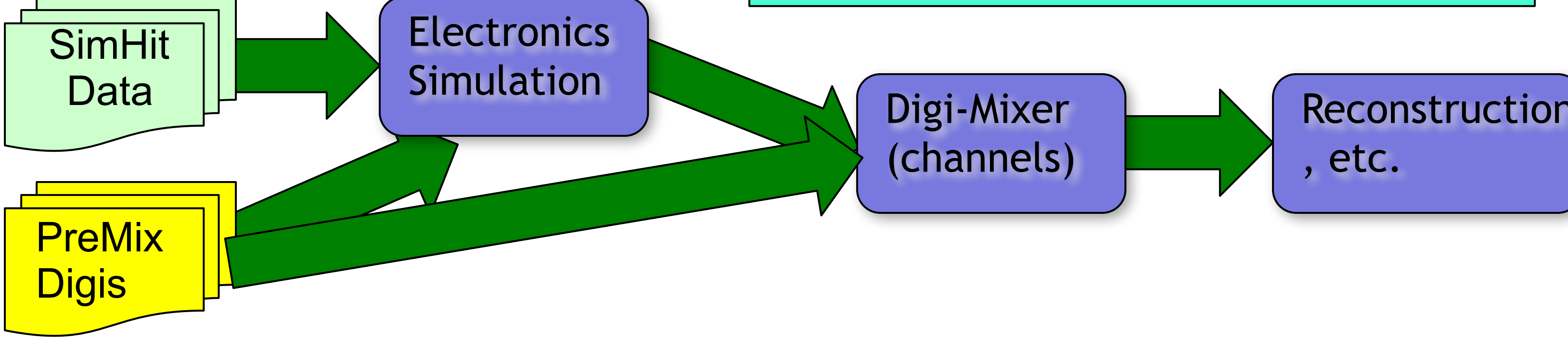
Pre-Mixing (in words)

- Next, “hard-scatter” sample is created and processed through the SIM step



- Finally, streams are merged

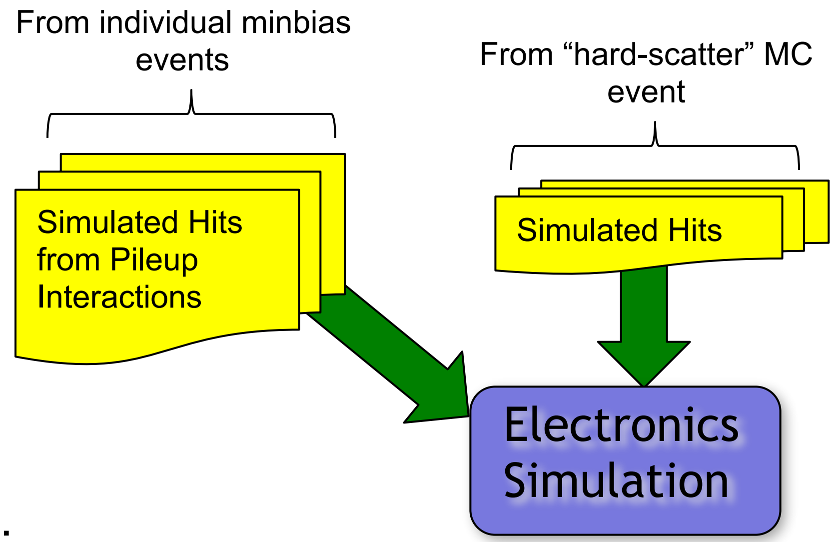
FastSim PreMixing mixes fully reconstructed tracks from Pileup and Hard Scatter events



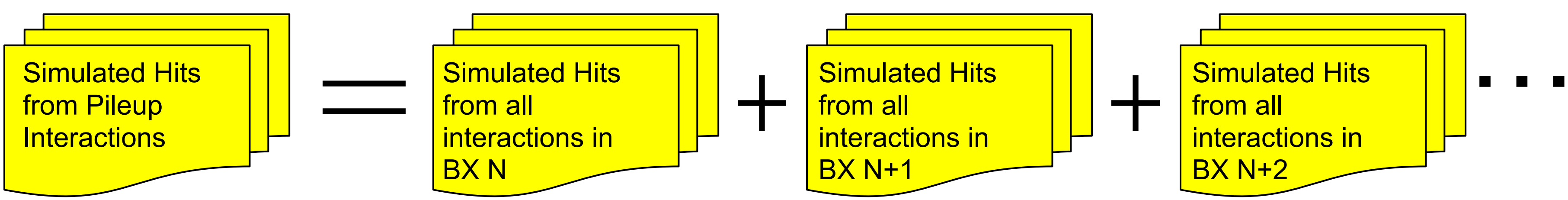
  - employs “DigiMixing” scheme where Digis are combined at the individual channel level; pileup signals are inserted as “noise” in majority of electronics simulations, combined directly in others
  - only 1 pileup event is needed for each “hard scatter” MC event
    - much simpler for computing infrastructure

Simulating Extreme Luminosities

- Details of Simulation Structure become important:
  – Reminder of the algorithmic structure for pileup simulation:



  – In more detail:



Sum contains **all** interactions in **all** beam crossings – all in memory simultaneously!

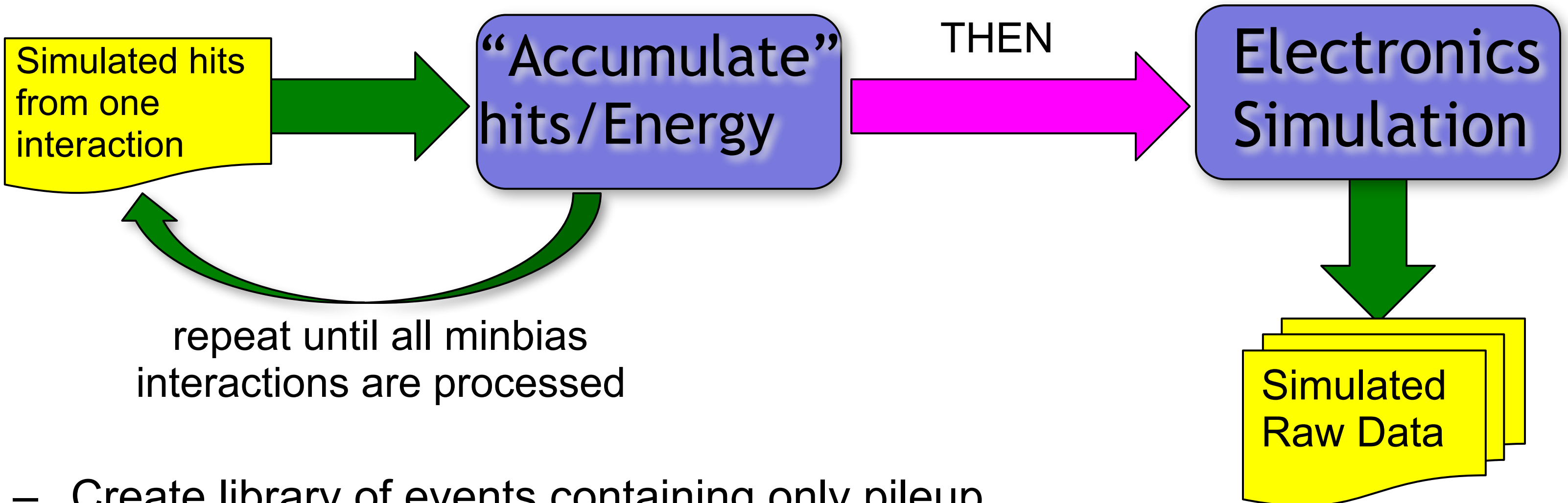
⇒ unsustainable at HL-LHC luminosities: ~140 interactions x 16 BXs = **2240** events in memory

Simulation meets Computation

Even if the events are read sequentially, more than 2000 minbias events to produce a single MC event with appropriate pileup at HL-LHC luminosities is a nightmare for computing infrastructure

- huge minbias event files have to be made available to each compute node for MC production

- Our Solution: Accumulation mixing for "Pre-Mixing" preparation step
  - For the pure minbias pileup simulation,



  - Create library of events containing only pileup contributions, following pre-determined luminosity profile to calculate how many interactions to include

Results

- Accumulation Mixing
  - dramatic effect on memory requirement
  - Memory Reduction for a sample with 100 interactions/crossing:

Pileup Configuration	VSIZ (MB)	RSS (MB)
Old Mixing Software	2520	2020
New Mixing Software	1283	933

- Current premixing implementation very successful
  - saves approximately 10x of input/output operations relative to the classic pileup simulation used previously.
  - the CPU time required to digitize has been reduced by approximately a factor of 2 for run2 and much more for run4.

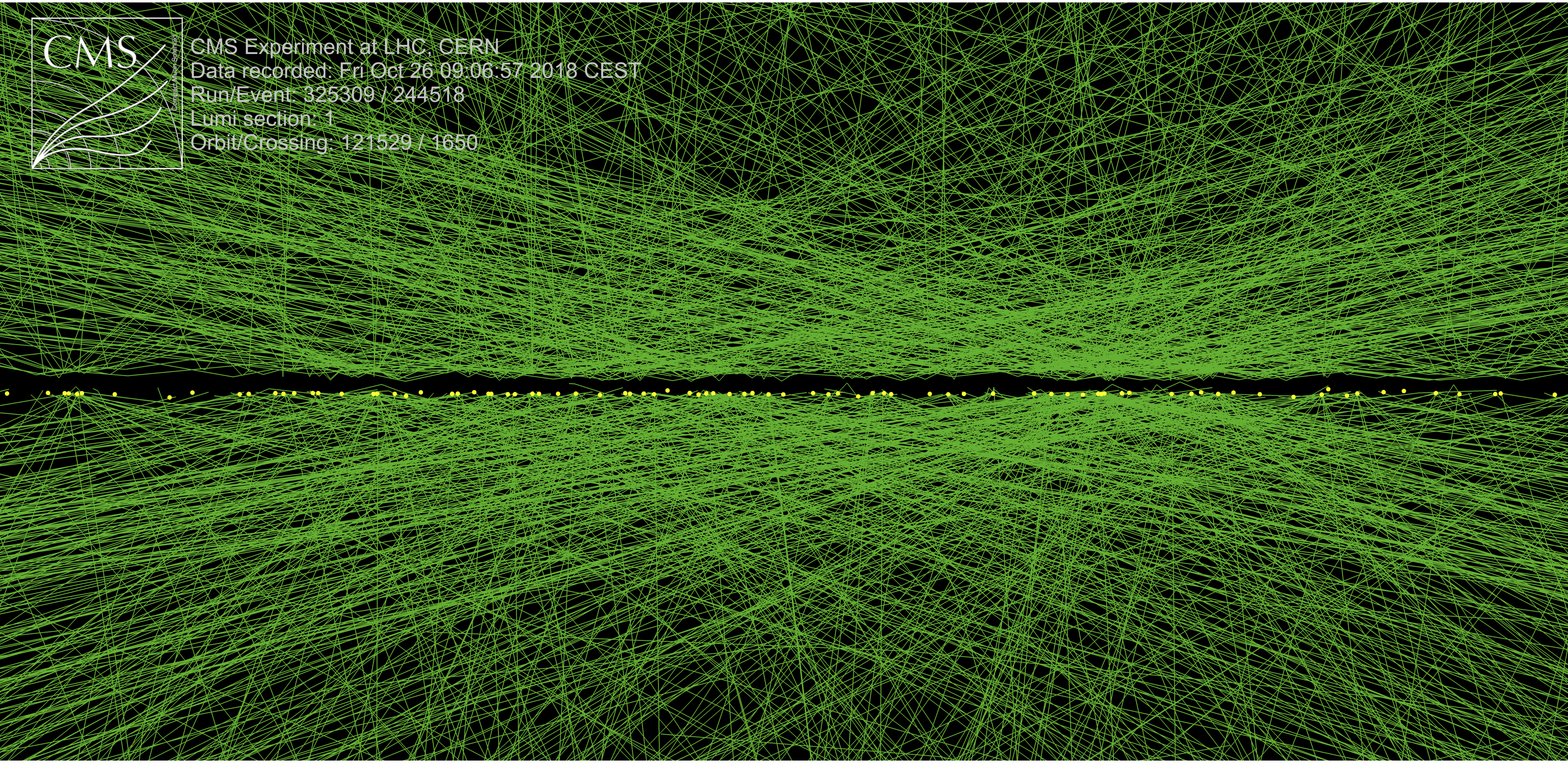
- A single premixed pileup library is large and can’t be stored at every site. The library data is streamed over the xrootd federation. Streaming over the WAN is only made possible by the large reduction in data.

Conclusions

- Pileup Simulation is not an easy problem
  - The next challenge will be controlling event sizes for the highly granular run 4 detectors
  - CPU/Memory consumption will continue to be problematic
    - especially for HL-LHC
    - constant vigilance required to keep computing under control
    - out-of-time pileup is difficult to study/quantify - detector expertise is required to get summation correct

- Simulation already confronting the High Luminosity LHC challenge
  - more events and more accuracy,
  - with a more complicated geometry,
  - while using a relatively smaller fraction of the total CPU budget.

- A major reworking of mixing infrastructure has been necessary to confront the challenges of high(est) luminosity simulation
  - with these modifications, ready for 13-14 TeV at HL-LHC luminosities



The future is here! This event has 136 vertices and was collected last Friday, 26-Oct

University of Notre Dame
Fermilab

Slide 7's table listings - Backup

- **Static library:** all simulation code in CMSSW is compiled into one static library file, to avoid calls to the procedure linkage table that would occur if libraries were loaded dynamically.
- **Production cuts:** a requirement on the range value to produce secondary particles is customized for each detector region. For the pixel system, the cut is 0.01 mm; for the strip tracker, 0.1 mm; for the ECAL and HCAL, 1 mm; for the muon systems, 0.002 mm; and for the support structure, 1 cm.
- **Tracking cut:** within the vacuum chamber, between the interaction point and the start of the pixel system, charged particles with energy less than 2 MeV are rejected. This avoids the possibility of looping electrons or positrons.
- **Time cut:** the maximum propagation time considered is 500 ns.
- **Shower library:** a library of pre-generated showers is used for the forward calorimeter. Particle multiplicities are highest in the forward region, so this relatively small volume can take a disproportionate amount of CPU to simulate completely.
- **Russian roulette:** an algorithm which discards, at random, N − 1 neutrons with energy less than 10 MeV or photons with energy less than 5 MeV in the calorimeter system. The Nth particle is retained and assigned a weight of N to account for the energies of the discarded particles.
- **FTFP_BERT_EMM:** a modified physics list with a simplified model of multiple scattering used in most regions, except for HCAL and HGCal.