Upgrades for the CMS simulation

David Lange,
Mike Hildreth, Vladimir Ivantchenko
for the CMS collaboration

CHEP 2016

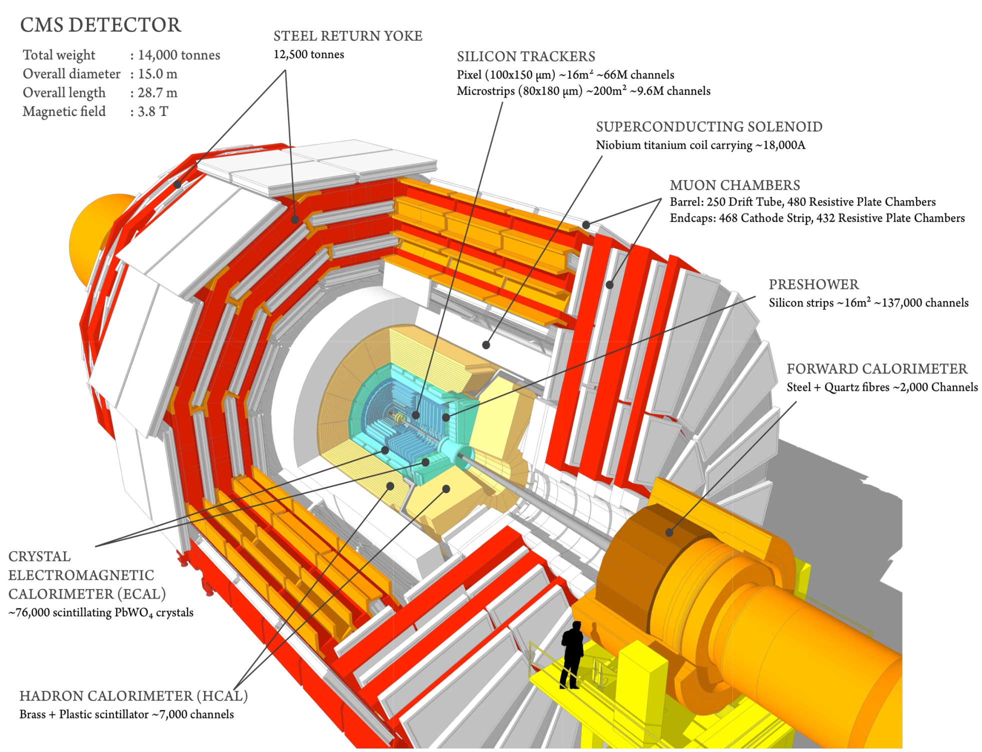
CMS simulation faces significant challenges for both today and tomorrow

Higher LHC luminosity means:

- Need for more accurate simulations
- Need for more events (ideally more events/CHF)
- More demanding pileup simulation requirements

Major detector upgrades in 2017, during LS-2 and for HL-LHC mean:

- New detector concepts to develop, benchmark and validate
- The need to make reliable simulations for HL-LHC luminosities

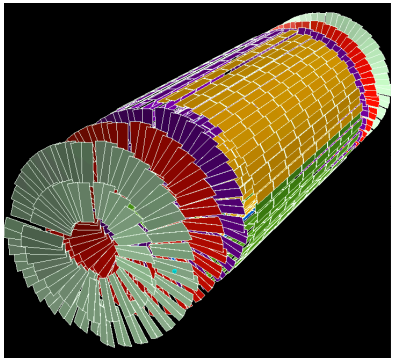


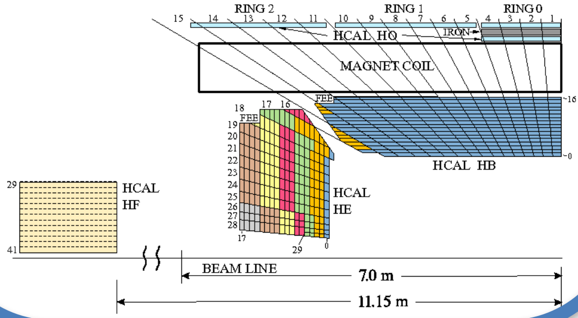
Increases in LHC luminosity and CMS detector upgrades push our simulation capabilities

Detector upgrades push continued simulation development in CMS

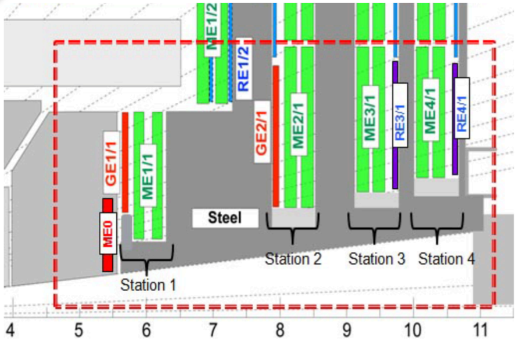
2017 upgrades

HL-LHC upgrades

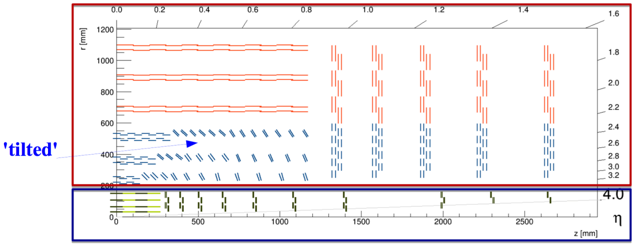




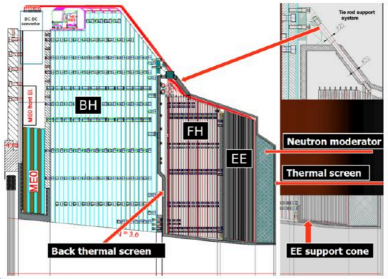
Muon ID



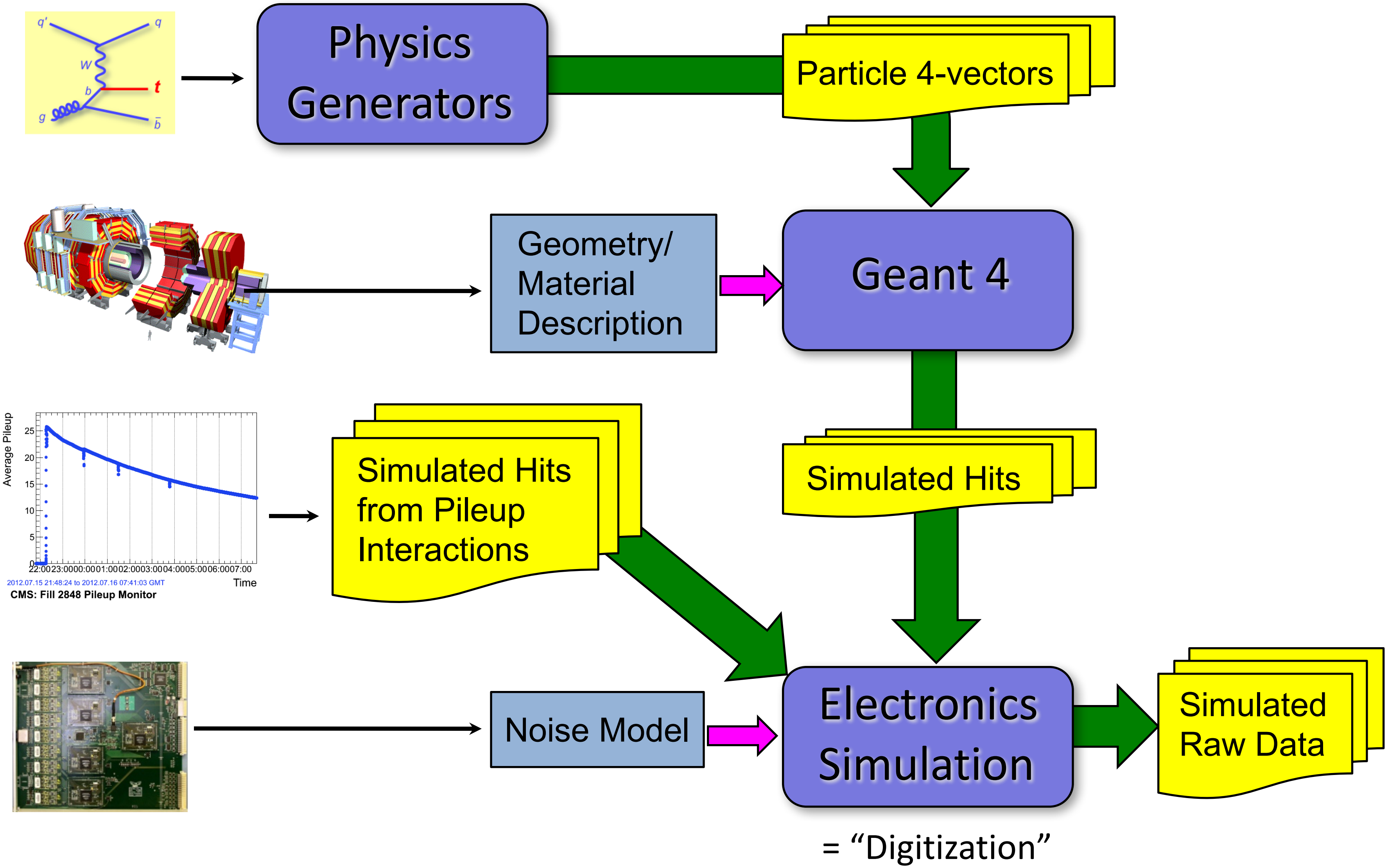
Tracking



Calorimetry



CMS Monte Carlo Simulation approach



I focus this presentation on progress in CMS simulation and digitization

- Upgrades to Geant4 configuration for upcoming major simulation production
- Integration of software for new detectors which will be installed in 2017 and proposed for HL-LHC upgraded CMS detector
- Deployment of premixing scheme for pileup interaction simulation performed at the digitization level

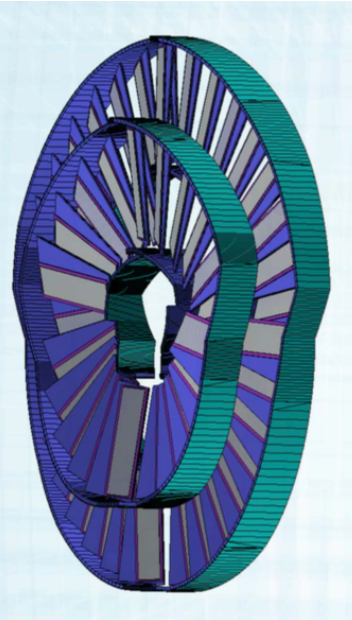
Recap from CHEP 2015: Simulation improvements for Run 2

- Geometry updates: Improvements to Run 1 detector model and integration of Run 2 detector changes
- Migration from Geant4 9.4p03 to 10.0p02
- Introduced Russian roulette method for sampling low energy particles
- New forward hadron calorimeter parameterized simulation (shower library instead of GFlash)
- Improved simulation library packaging
- Improved mixing module for high pile-up simulation
- Developed simulation for multi-threaded CMSSW framework

Together these improvements gained a factor of 2 in simulation time/event for 2015/2016 MC

What have we changed since then?

- New Geant4 10.2 built in multithreaded mode.
- Minor changes applied by CMS:
  1. FTF model parameters defined as in 10.1p03
  2. Fix to G4SubtractedSolid
  3. Fix to G4CutTubs
     - Our first use of this shape to model support structure of new pixel detector



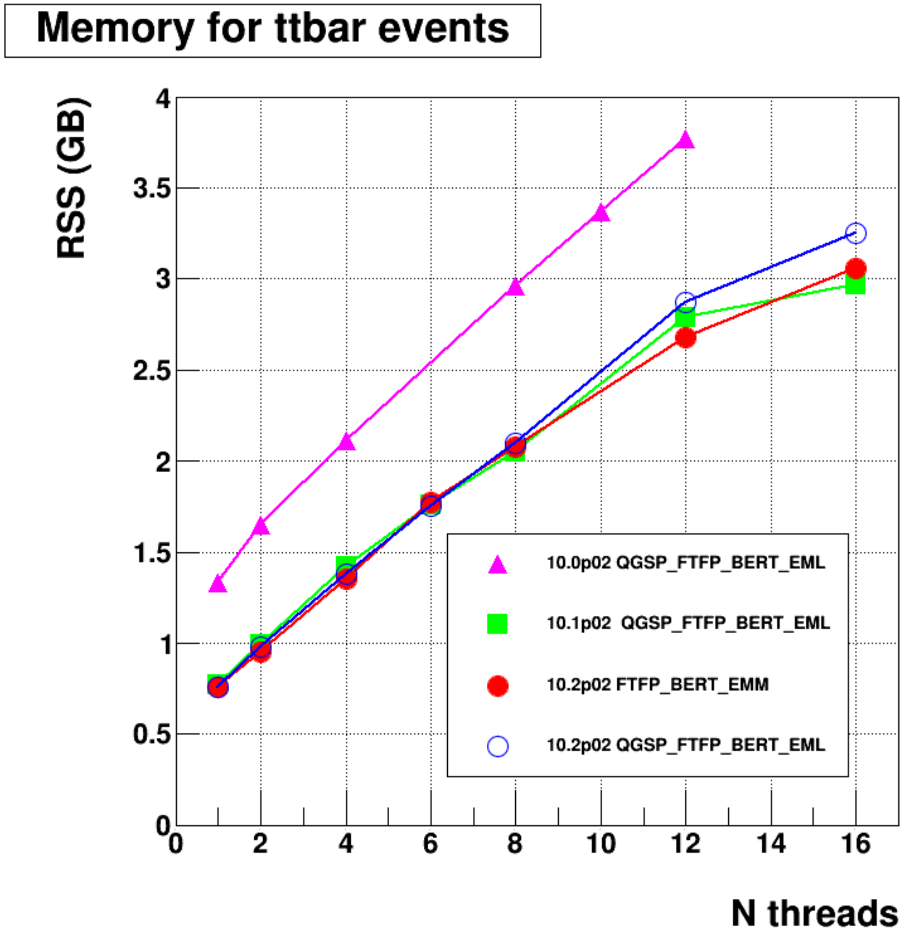
- Production platform slc6_amd64_gcc530

What have we changed since then?

- Default physics list: FTFP_BERT_EMM
  - FTFP_BERT: Standard hadronic physics list
    - Bertini-style cascade for hadrons < 5 GeV
    - FTF (Fritiof) model for high energies (> 4 GeV)
  - EMM means:
    - Default EM physics in HCAL (“Option 0“) in order to get an unbiased response of the sampling calorimeter
    - EM “Option 1“ physics in other all detectors (increased multiple scattering step length limit)
  - This physics list choice and our choice to revert changes in FTFP made in recent 10.2 patches was based on comparisons of single particle response using test-beam (2006) and run-2 data (See details in S.Banerjee talk)

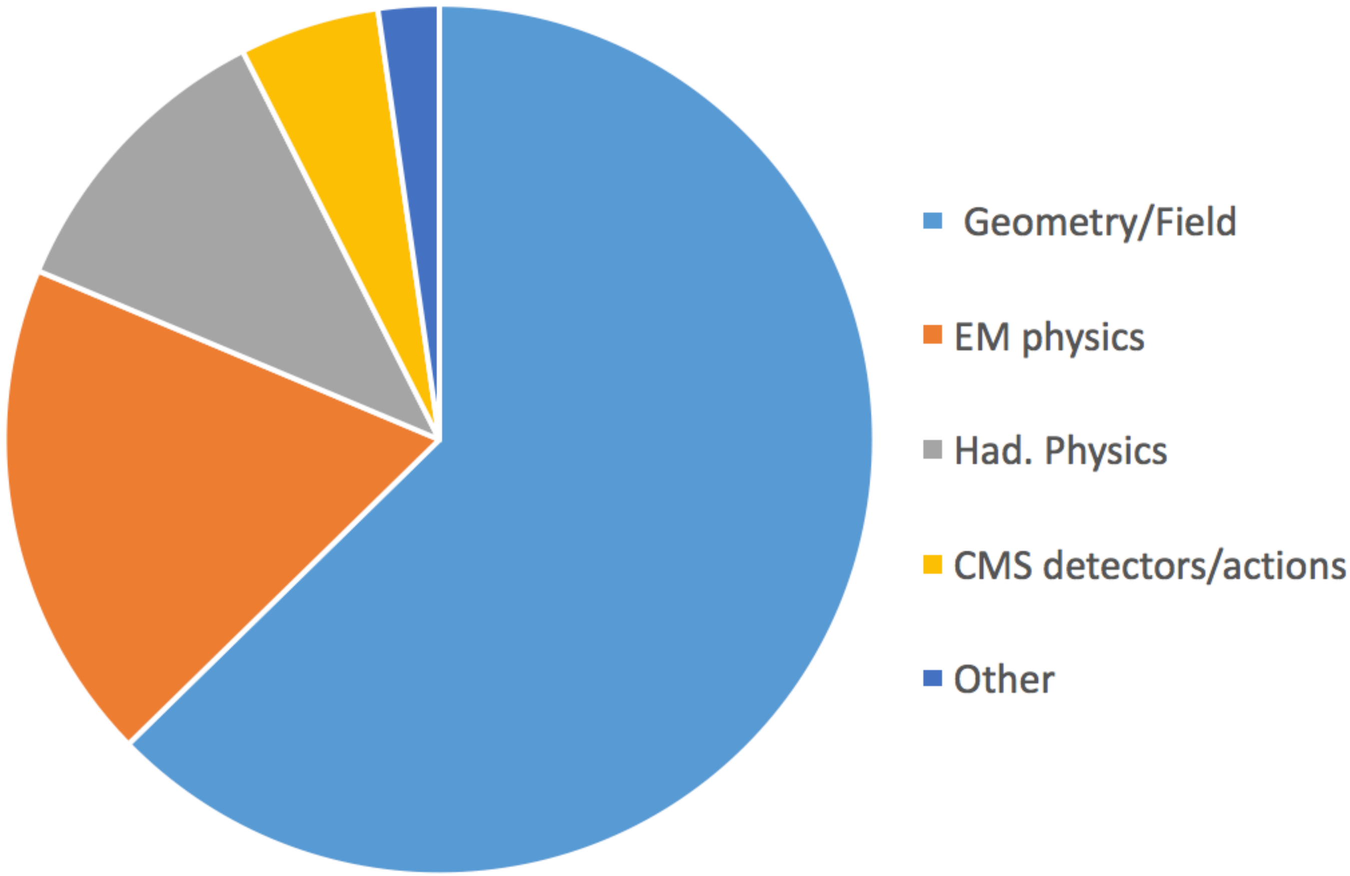
Memory for CMS run with Geant4 10.2p02

- A node with 12 Intel cores was used to study memory utilisation
- 13 TeV hard scattering event were simulated
  - Results after 1000 events are shown
  - CMS private patches to 10.0 include backports of fixes of memory leak and memory optimisation
  - Results for 10.1 and 10.2 are practically the same
  - No dependency on Physics List
- No problems to run CMS SIM production in the MT mode



Release	1st thread (GB)	Delta per thread (GB)
10.0p02+CMS patches	1.33	0.23
10.2p02	0.76	0.19

Where our simulation CPU goes



CMS CPU performance for 2017

Geant4 version and configuration	Number of threads	CPU for ttbar (events/thread relative rate)	CPU for QCD (events/thread relative rate)
10.2p02 seq.	1	1	1
10.2p02 MT	1	1.07	0.94
10.2p02 MT	4	1.1	0.89
10.2p02 MT	8	1.13	0.89

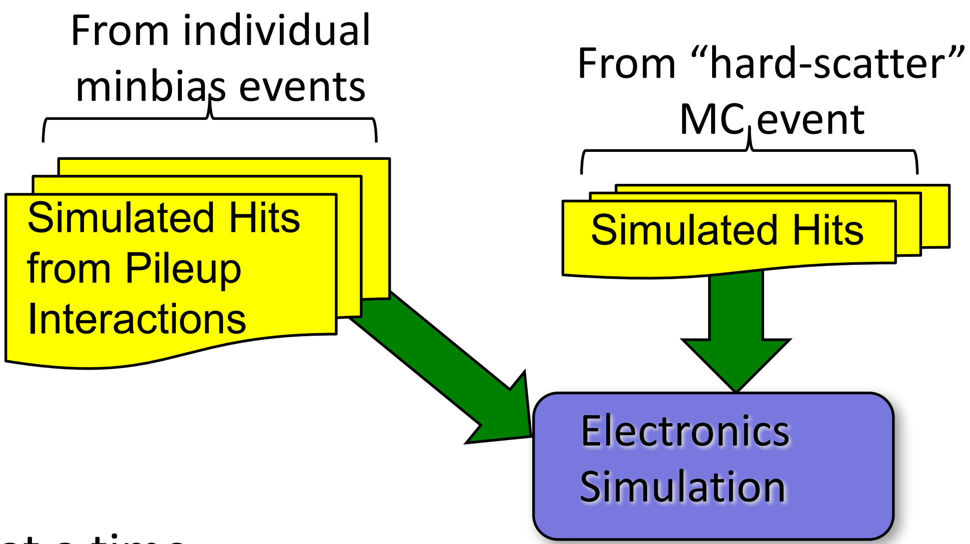
CMS 2015/6

CMS 2017

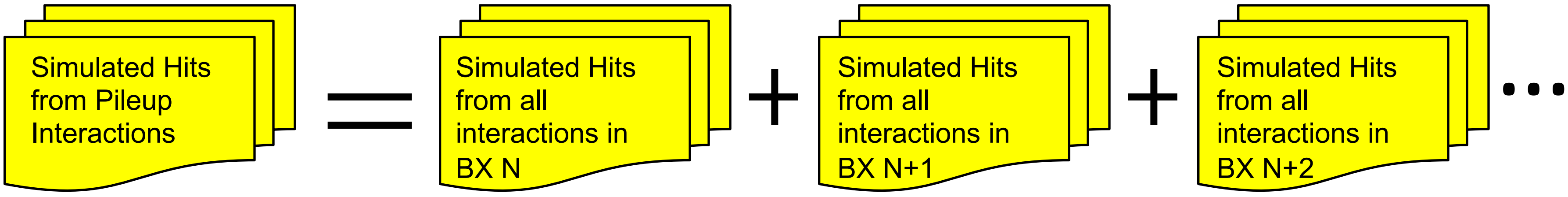
- Overall, no substantial changes in our technical performance since 2015
  - Modest improvements in CPU performance in Geant4 10.2p2 and in CMS sensitive detector code offset by other changes
- No noticible performance loss (or gain) with threading (up to 8 threads)

Simulating Extreme Luminosities

- Model pileup by including G4hits from MinBias events generated separately from the hard-scatter event
  - Hits are loaded one interaction at a time, processed and accumulated for the final digitized output



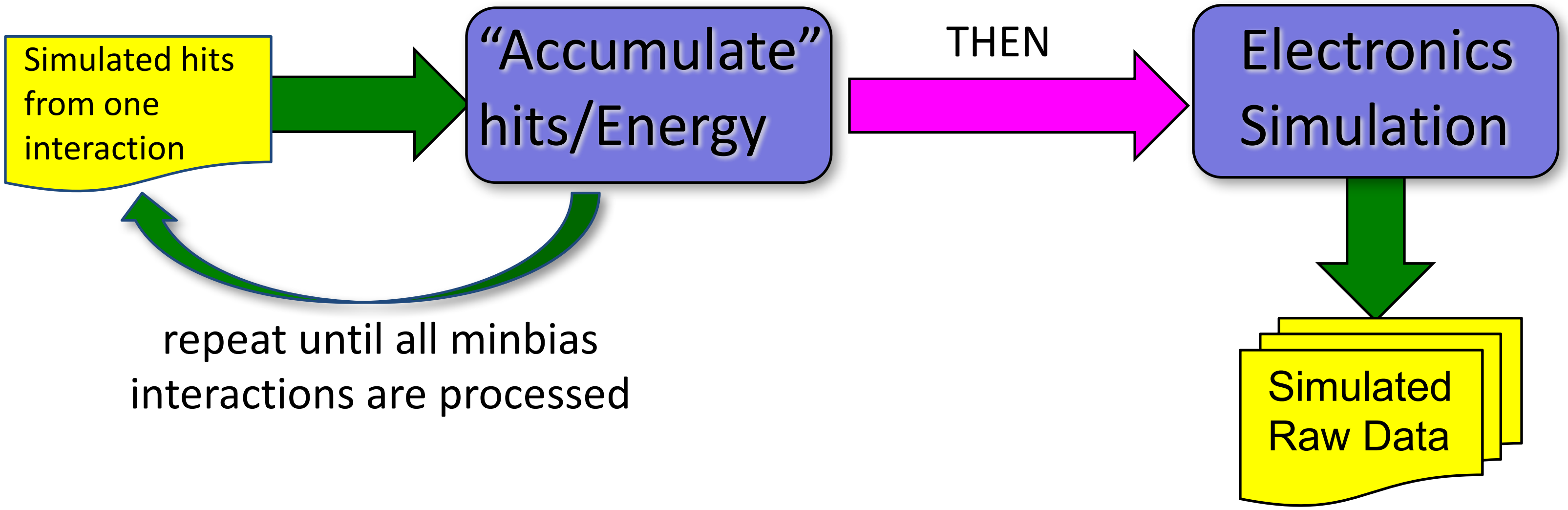
- The pileup interaction simulation is the sum of many interactions



Tens or hundreds of interactions per bunch crossing

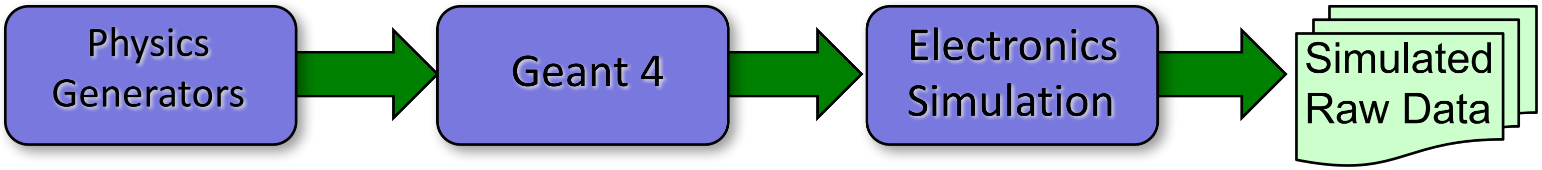
This method of pileup simulation is very I/O intensive: 1000s of minbias to read per hardscatter event

- Newly deployed solution: "Pre-Mixing" which proceeds in 2 steps
    1. Upfront I/O intensive step: Create library of events containing only pileup contributions

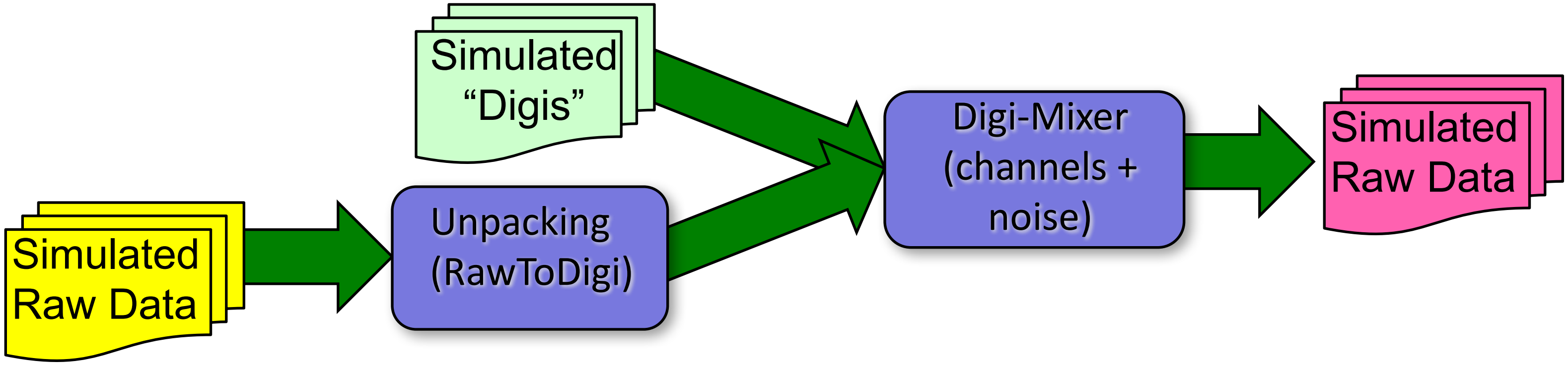


Then hard-scatter and pileup events are combined as “RAW” data format events

- The hard-scatter sample is created and processed through the digitization step with no pileup, convert to our raw data format



- Then the two streams are merged



- Only 1 pileup event is needed for each “hard scatter” MC event
  - Much easier to process through computing infrastructure once the premixing sample is created

Premixing lessons learned

- After a long development and validation process, **premixing deployed in CMS MC production** since this summer

- Issues and benefits we found
  - Extended "raw" format extended to ensure **sufficient precision** for closeby interactions
  - **Event reuse**: We now potentially re-use entire pileup events instead just individual minimum bias events in Monte Carlo production
  - **Flexibility** considerations: Generating multiple pileup configurations is now more time consuming
  - **Major CPU savings**: At current pileup, our digitization+reconstruction processing runs ~2x faster (with a one time cost of the up front production of the premixed library)

Premixing lessons learned

- The premixing library is big (~0.1*N(PU) MB/event)
  - Must save sufficient information from the pileup events to allow an accurate digitization
  - This is still a huge savings over the I/O seen in our old approach (**~40x**). We can run production using remote reads of premixing library

Premixing has brought a substantial operational improvement to our operations!

Conclusion: Looking forward

- CMS simulation continues to improve physics accuracy and to make better use of CPU resources
  - Latest Geant4 developments going into production after extended physics studies
  - Pileup simulation has been reimplemented in more techically efficient (CPU and I/O) manner while maintaining physics performance

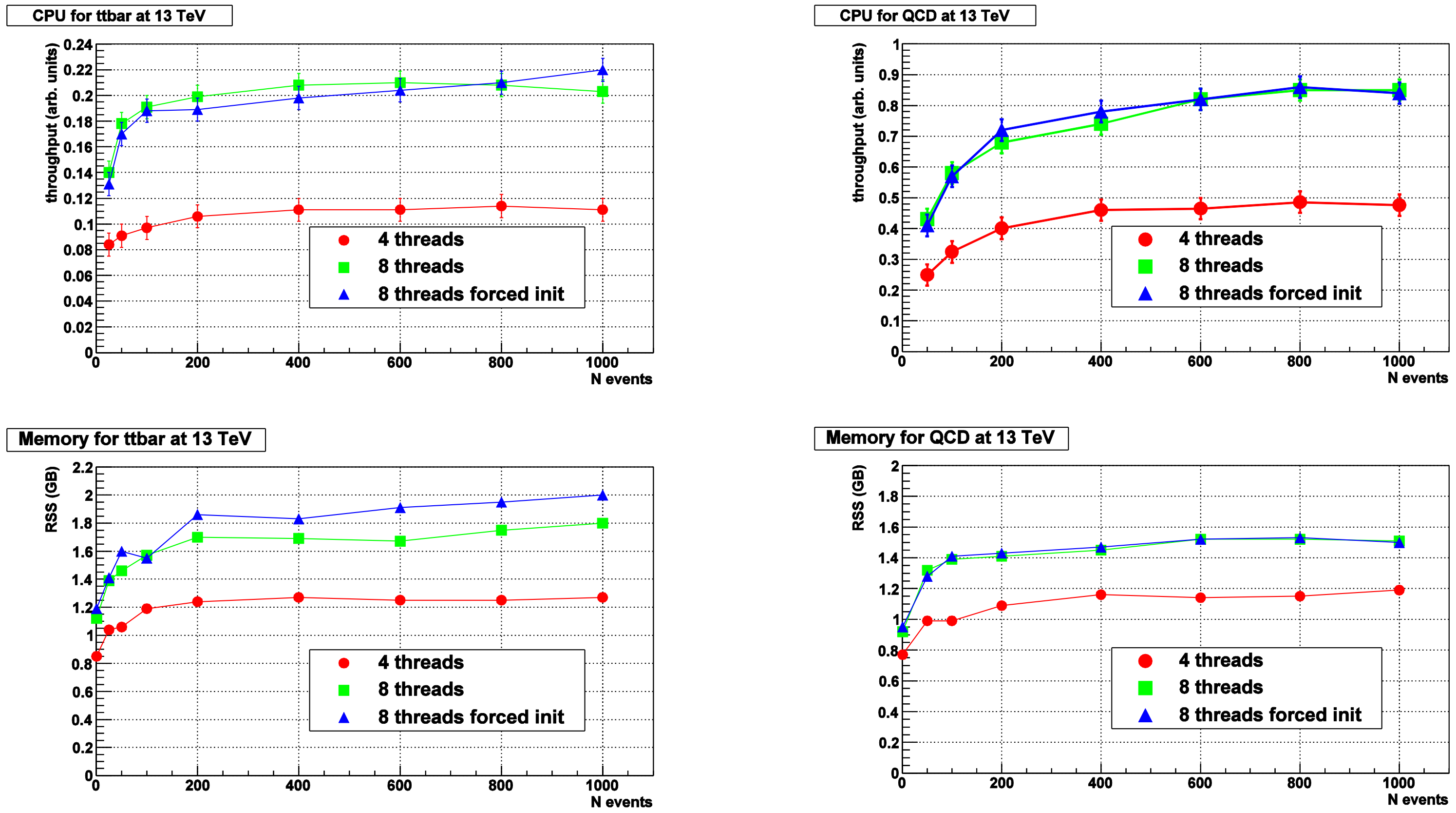
- Supporting near-term (and major) detector changes as well as Phase-II CMS upgrades pose new challenges.

Extra information

CMS multithreaded simulation

- CMS has developed and deployed a multi-threaded framework based on TBB. Our simulation workflow is an important user
  - At the same time, we need to work within the Geant-4 mutlithreading model (POSIX threads)
- Master vs worker threads
  - Master must not do any work
  - Worker threads simulate events in their own event loops (Spawned by CMSSW Framework via TBB)
- Constrains and requirements we worked through:
  - Initialization of SIM module should be done once
  - Access to geometry and field should be const
  - Both Sequential and MT simulation should be functional within a CMSSW release

Dynamic of CPU and RSS for 13 TeV CMS simulation run in MT mode for Geant4 10.2p02



- Maxmum CPU efficiency is achieved after simulation ~500 events
  - The turnon shape is expected due to
    1. CMS and Geant4 initialisation before 1st event
    2. Lazy initialization of hadron physics in Geant4