Geant4 speed up options

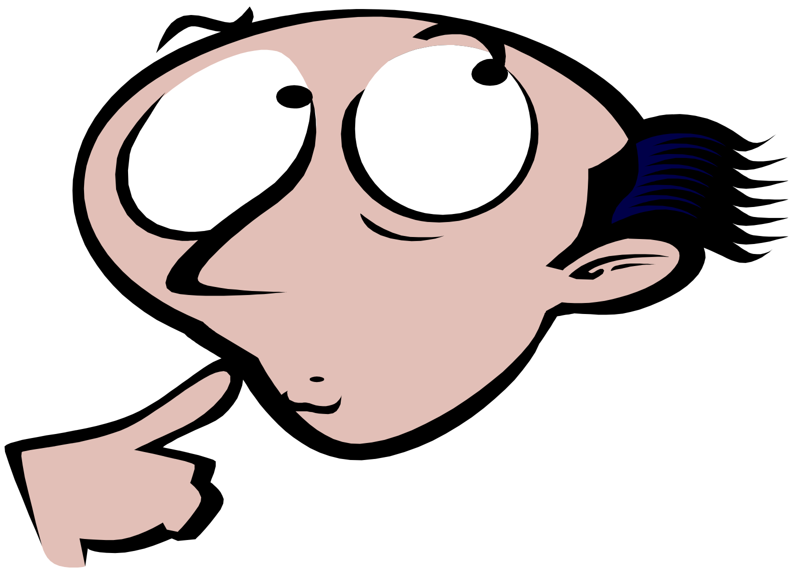
Vladimir Ivanchenko , CERN
2nd Fast Monte Carlo Workshop in HEP
14-16 January 2014, DESY, Zeuthen , Germany

Outline

- Introduction
- CPU performance of Geant4 simulation for CMS
- Geant4 electromagnetic (EM) physics options
- Russian Roulette (RR) method
- Geant4 multi-threaded
- Summary

Introduction

- Geant4 simulation of any HEP experiment is the most time consuming step of offline data processing
  - Even for 50 PU the digitisation step of CMS Monte Carlo is significantly faster than the SIM step
- For each specific experiment performance of Geant4 simulation can be optimized
  - CMS uses in FullSim production several optimisations:
    - QGSP_FTFP_BERT_EML Physics List (equivalent QGSP_FTFP_BERT_EMV)
    - Parametersation of Forward detectors responses
    - New: Russian roulette method
- In this talk I will try to discuss how to speedup full simulation in general and what limitations exist



CPU PERFORMANCE OF GEANT4 SIMULATION FOR CMS

Full Sim profiling for 8 TeV Geant4 9.6p02 (D. Nikolopoulos)

	Functions / Events	MinBias Cummulative / Self cpu-time (%)	ZEE Cummulative / Self cpu-time (%)	TTBar Cummulative / Self cpu-time (%)
1	G4Mag_UsualEqRhs:: EvaluateRhsGivenB	3.29 / 3.29	3.32 / 3.32	3.33 / 3.33
2	G4PolyconeSide:: DistanceAway	2.66 / 2.34	2.72 / 2.30	2.55 / 2.02
3	G4Navigator:: LocateGlobalPointAndSetup	5.39 / 1.84	5.66 / 2.00	6.21 / 2.29
4	SimTrackManager:: idSavedTrack	1.20 / 1.20	4.03 / 4.03	1.06 / 1.06
5	G4CrossSectionDataStore:: GetCrossSection	15.06 / 1.54	13.70 / 1.60	14.14 / 1.61
6	G4UrbanMscModel95:: ComputeCrossSectionPerAtom	2.84 / 1.70	2.18 / 1.30	1.80 / 1.09
7	G4ClassicalRK4:: DumbStepper	6.65 / 1.55	6.72 / 1.60	6.33 / 1.58
8	G4hPairProductionModel:: ComputeDMicroscopicCrossSection	1.01 / 0.22	0.37 / 0.08	0.20 / 0.05
9	G4PhysicsLogVector:: FindBinLocation	0.62 / 0.11	0.79 / 0.14	0.93 / 0.17
10	G4BGGNucleonInelasticXS:: CoulombFactor	3.43 / 0.53	2.84 / 0.44	3.10 / 0.47
11	G4ElasticHadrNucleusHE:: HadrNucDifferCrSec	1.34 / 0.26	0.63 / 0.12	0.47 / 0.09
12	G4InuclSpecialFunctions:: G4cbrt	0.88 / 0.04	0.72 / 0.03	0.76 / 0.04

The 12 most time consuming methods:

- **1 from CMSSW**
- **11 from Geant4**

Full Sim profiling for 13 TeV Geant4 9.6p02 (D. Nikolopoulos)

	Functions	Cumulative / Self time spent (%) (MinBias13TeV)	Cumulative / Self time spent (%) (MinBias13TeV + RR)	Cumulative / Self time spent (%) (TTBar13TeV)	Cumulative / Self time spent (%) (TTbar13TeV + RR)	Short Comments
1	SimTrackManager:: idSavedTrack	18.62 / 18.62	16.65 / 16.65	10.08 / 10.08	10.32 / 10.32	The most consuming function. Recursive.
2	G4Mag_UsualEqRhs:: EvaluateRhsGivenB	2.38 / 2.38	2.58 / 2.58	2.82 / 2.82	2.88 / 2.88	-
3	G4PolyconeSide:: DistanceAway	2.41 / 2.19	2.38 / 2.18	2.44 / 2.01	2.21 / 1.85	About 0.3% of cumulative is used by atan2.
4	G4Navigator:: LocateGlobalPointAndSetup	6.67 / 1.76	6.70 / 1.80	6.63 / 2.20	2.27 / 2.12	Calls several functions
5	G4ClassicalRK4:: DumbStepper	5.24 / 1.17	5.64 / 1.25	5.89 / 1.46	5.88 / 1.38	sim::Field:: GetFieldValue and function #2 of the table are called the most.
6	G4ElectroNuclearCrossSection:: GetIsoCrossSection	1.33 / 1.10	1..45 / 1.20	1.49 / 1.23	1.57 / 1.29	About 0.25% of cumulative spent in log

*(clicking the numbers redirects to web profile)

The 6 most time consuming methods:

- **1 from CMSSW (but significantly increased from 2 to 15 %)**
- **5 from Geant4**

CPU profiles for 8 and 13 TeV are similar for Geant4 functions

Requirements for Geant4 10.0

- CPU hot spots in Geant4 for CMS are in simple functions:
  - G4PhysicsVector
  - Cmath library
  - Hadronic cross sections
  - G4Polycone
- Geant4 Collaboration accepts and implement requirements
  - Introduced G4Log and G4Exp
    - Extracted from VDT library (T.Hauth, V.Innocente, D.Piparo)
  - Introduced G4Pow
  - G4PhysicsVector was updated

	Functions	Cumulative time spent (%) (MinBias13TeV)	Cumulative time spent (%) (TTbar13TeV)
1	log	2.55	2.65
2	exp	3.08	3.26
3	atan2	2.85	2.87
4	log10	1.30	1.51
5	pow	0.89	0.96
6	sincos	0.54	0.54
7	atan2f	0.23	0.26
8	atanf	0.11	0.12
9	cos	0.10	0.09
+	TOTAL	11.65 %	12.26 %

*(clicking the numbers redirects to web profile)

Summary on CMS simulation time studies

- Simulation of 13 TeV run will require about 25 % more CPU for the same number of events as for 8 TeV
- For Geant4 9.6p02 main part of CPU time is splitted between
  - Mathematical functions
  - Calculation of hadronic cross sections
  - Tracking in field and CMS geometry
  - EM physics
- For Geant4 10.0 about 5% speed up is achieved by
  - Using fast functions G4Log, G4Exp, G4Pow
  - Optimisation of computation of cross sections
- CPU speed is mainly depend on number of simulation steps of all simulated tracks
  - To reduce number of steps one should increase length of each step or reduce number of tracks

GEANT4 EM PHYSICS OPTIONS

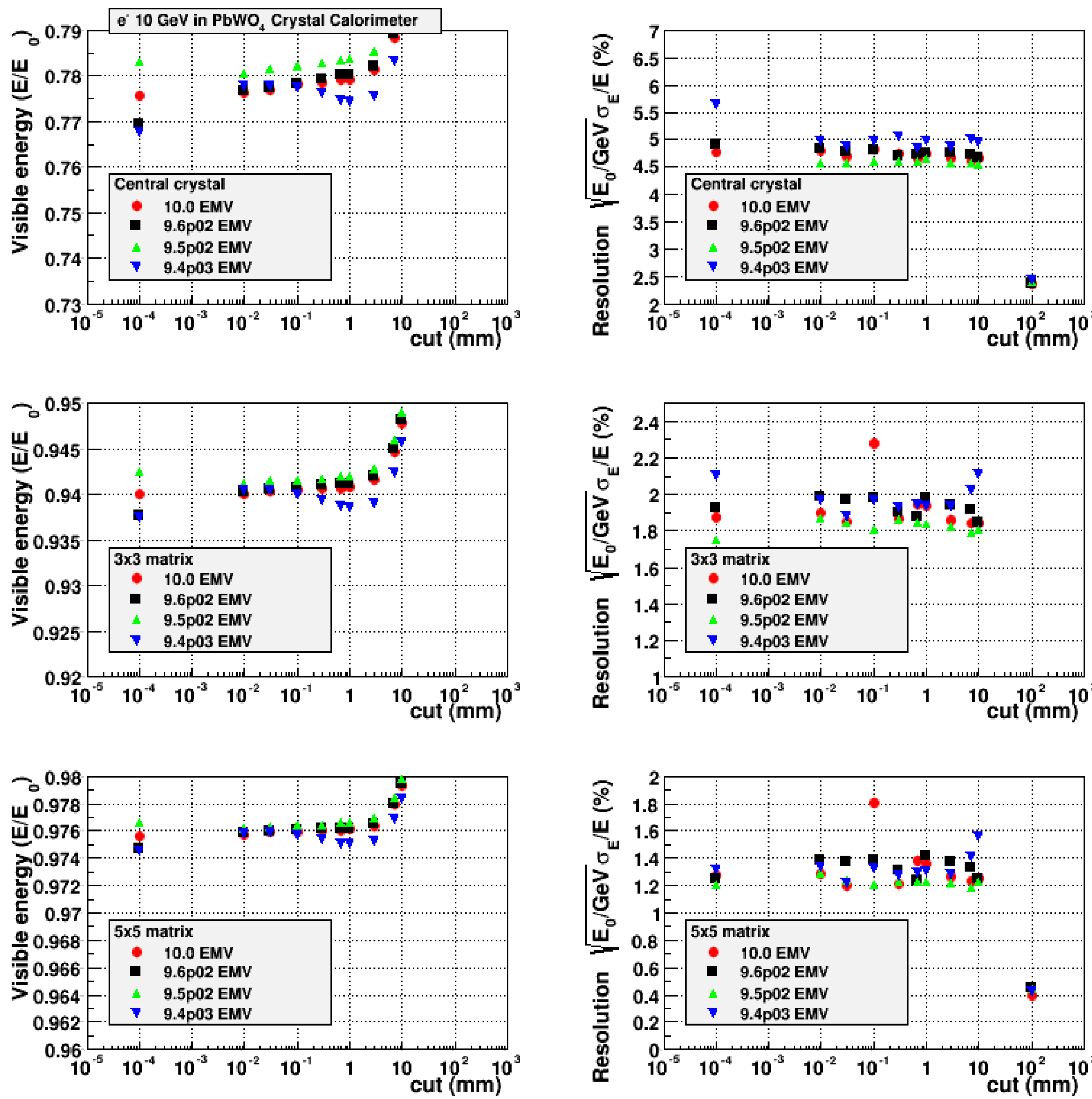
EM options for CMS FullSim production

- CMS are using several specific EM options which improve CPU performance and do not compromise accuracy
  - Cuts per G4Region
  - «Simple» step limitation for multiple scattering
  - «ApplyCuts» for gamma processes
  - Parameterisations in forward detectors (Gflash andshower libraries)
- QGSP_FTFP_BERT_EML custom Physics List
  - In recent Geant4 releases equivalent to QGSP_FTFP_BERT_EML
- Optimisation of cuts is a typical task for any experiment
  - Examples of results for simplified standalone calorimeters will be in following slides

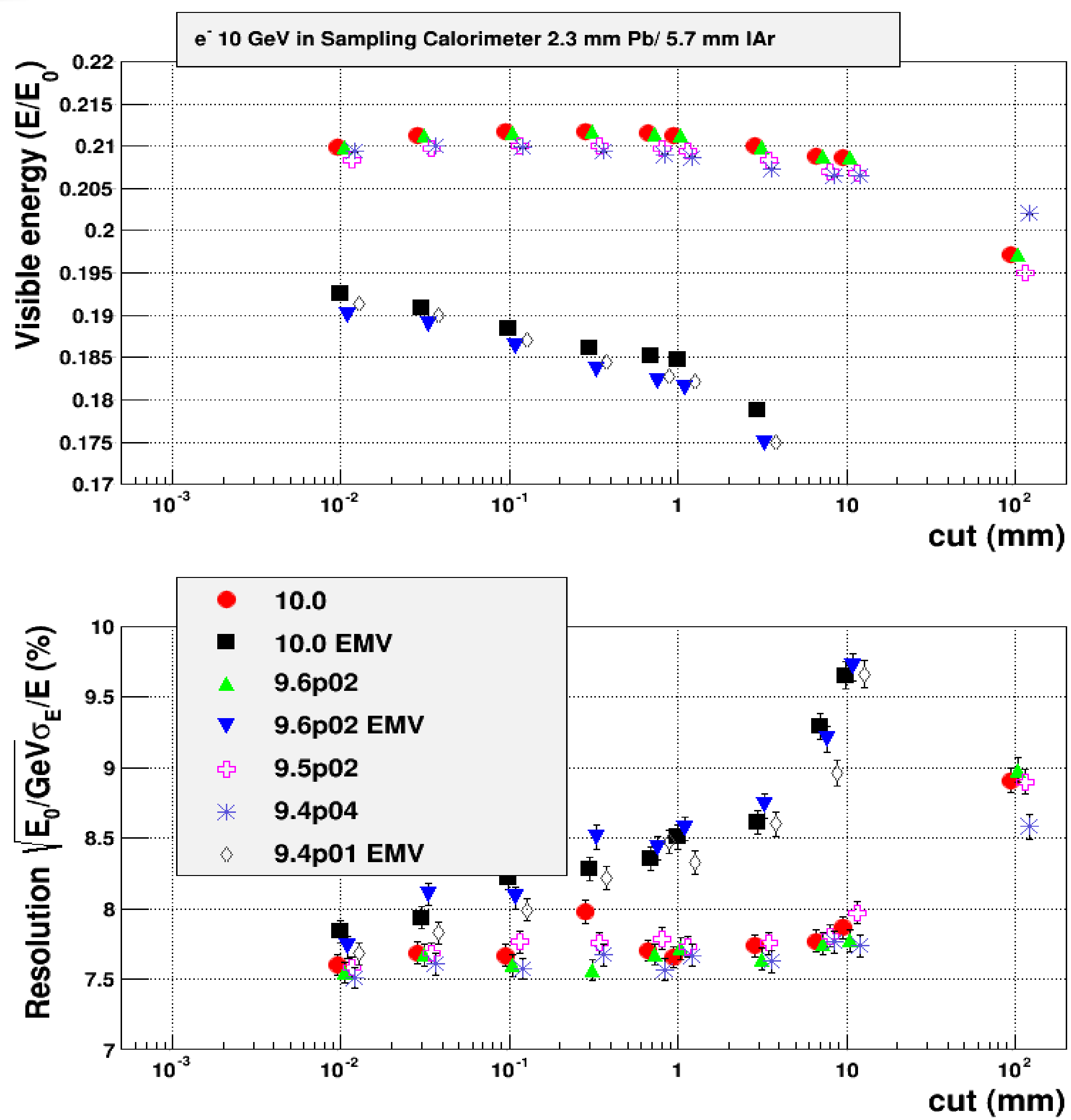
CMS-ECAL type calorimeter

EMV Physics List
«Simple» msc

cut 1 mm used in ECAL and HCAL
E_e = 1.1 MeV
E_γ = 89 keV
EM shower shape is stable



ATLAS-barrel type calorimeter



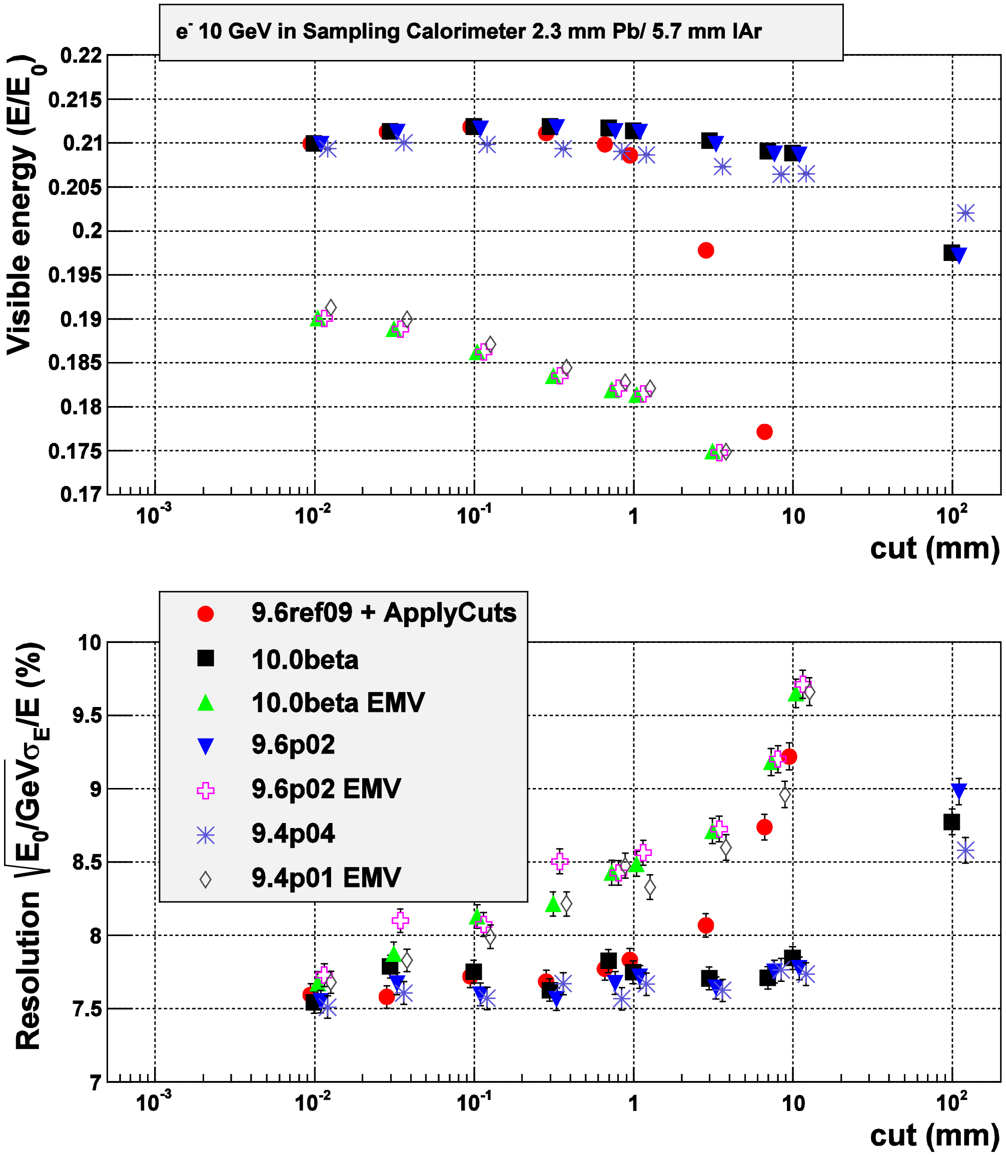
Default Physics List
«UseSafety» msc
step limit
EM shower shape
is stable

for cut 1 mm in Pb
E_e = 1.4 MeV
E_γ = 102 keV

for cut 1 mm in lAr
E_e = 345 keV
E_γ = 6.2 keV

EMV EM Physics
provides low response
and biased RMS for
for any cut value
this configuration is
faster by factor 2

ATLAS-barrel type calorimeter



Default Physics List
«UseSafety» msc
step limit
EM shower shape
is stable

«ApplyCuts» option
significantly affect
CPU, response
and resolution

Remarks on EM option optimisation

- Optimal set of cuts and other options depends on detector geometry and experiment requirements to simulation accuracy
  - It is possible to get about factor 2 more effitient simulation tuning cuts and options
  - It is not possible to to get more
- In typical calorimeters optimal thresholds
  - for e- production is about 0.2-1.0 MeV
  - for gamma production
  - this limits on thresholds are the same for all shower energies from GeV to TeV
- To get more significant speedup we need to try to reduce number of steps
  - Increase mean step size
  - Reduce number of tracks
  - Use parameterisations instead of detailed simulation

Parameterisations for CMS FullSim

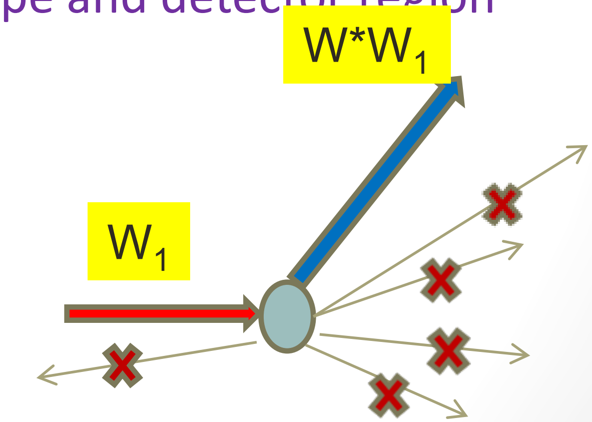
- CMS uses parameterisations for all forward detectors
  - GFlash (Geant4 based)
  - Shower libraries (produced by geant4)
- GFlash parameterizations of EM and hadronics was also developed for central calorimeters
  - Used in FastSim
  - similar for forward calorimeters
  - In CMSSW this can be applied on top of any Physics List
  - is enabled for high energy e^-, p > 1 GeV/c
  - Eta regions between the barrel and the endcap calorimeters are excluded
  - Unfortunately, overall CPU effect is about 10%

Russual Roulette Method

Well established method for neutron transport :
Lewis, E. E.&Miller, Jr.,W. F. [1984]. Computational Methods of Neutron Transport, John Wiley & Sons, New York.
Stephen A. Dupree, S. K. Fraley [2004] A Monte Carlo Primer: A Practical Approach to Radiation Transport, Volume 2
http://www.oecd-nea.org/tools/abstract/detail/nea-0387/
http://www.oecd-nea.org/tools/abstract/detail/ccc-0754/

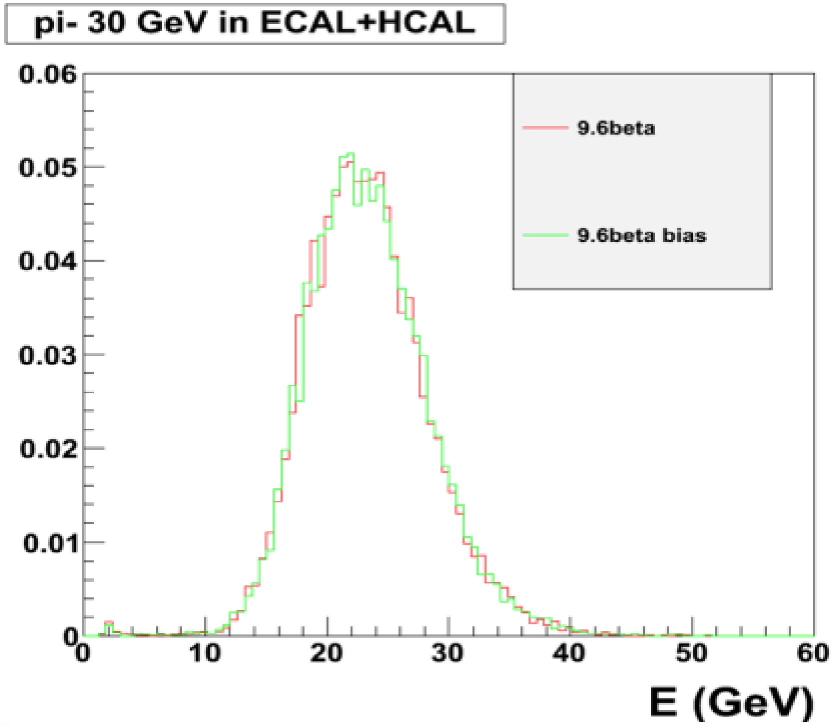
Russian Roulette in CMS

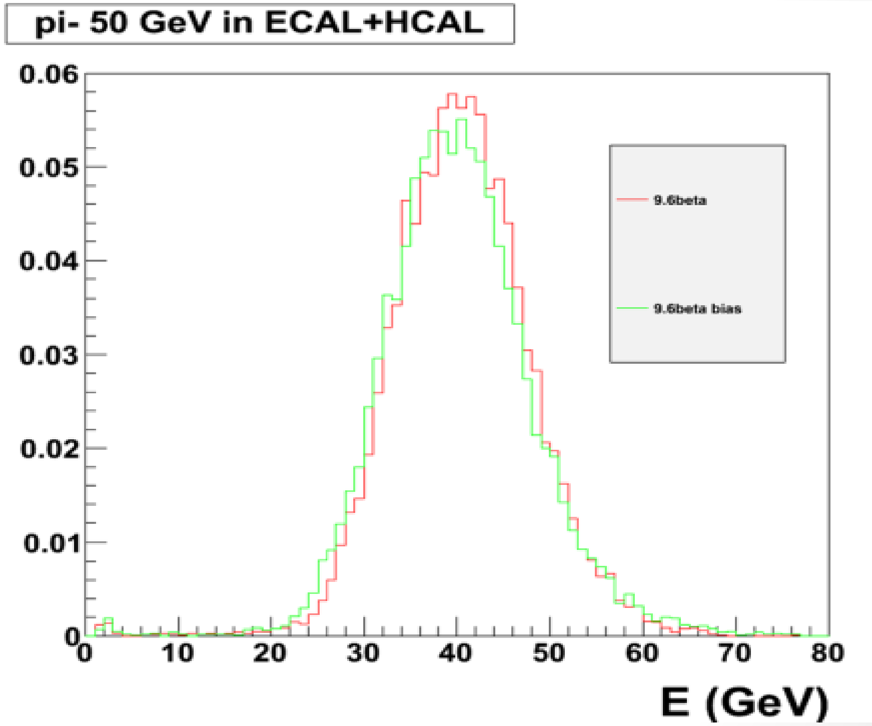
- Method used in neutron shielding calculations for many years
  - Not necessary to track all low-energy particles in a shower
- Some fraction of low-energy particles are killed but remainder get higher weight
  - not suited for tracker, muon systems
  - direct CPU savings (for calorimeter simulation)
  - geometry independent
- RR may be enabled separately per particle type and detector region
  - n, γ - allow significant CPU savings for CMS
  - p, e^- - no visible effect so far
- Two parameters per particle
  - RR factor (1/W)
  - Upper energy limit



RR: pion beam in simplified Ecal+Hcal

- Checks on total energy and resolutions

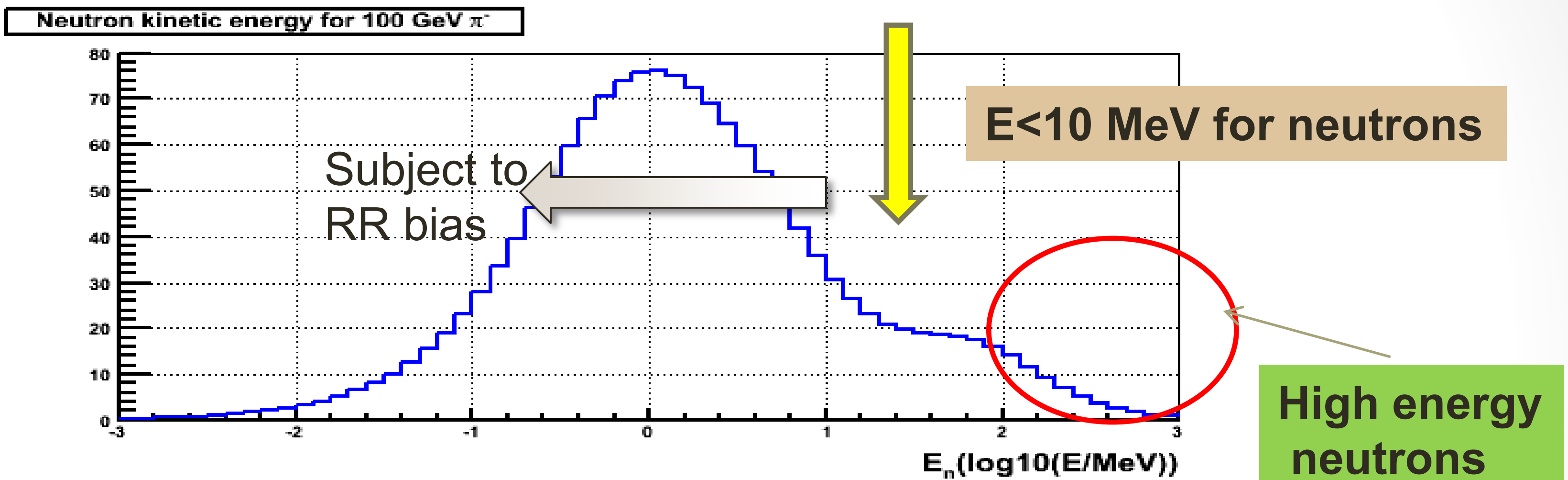


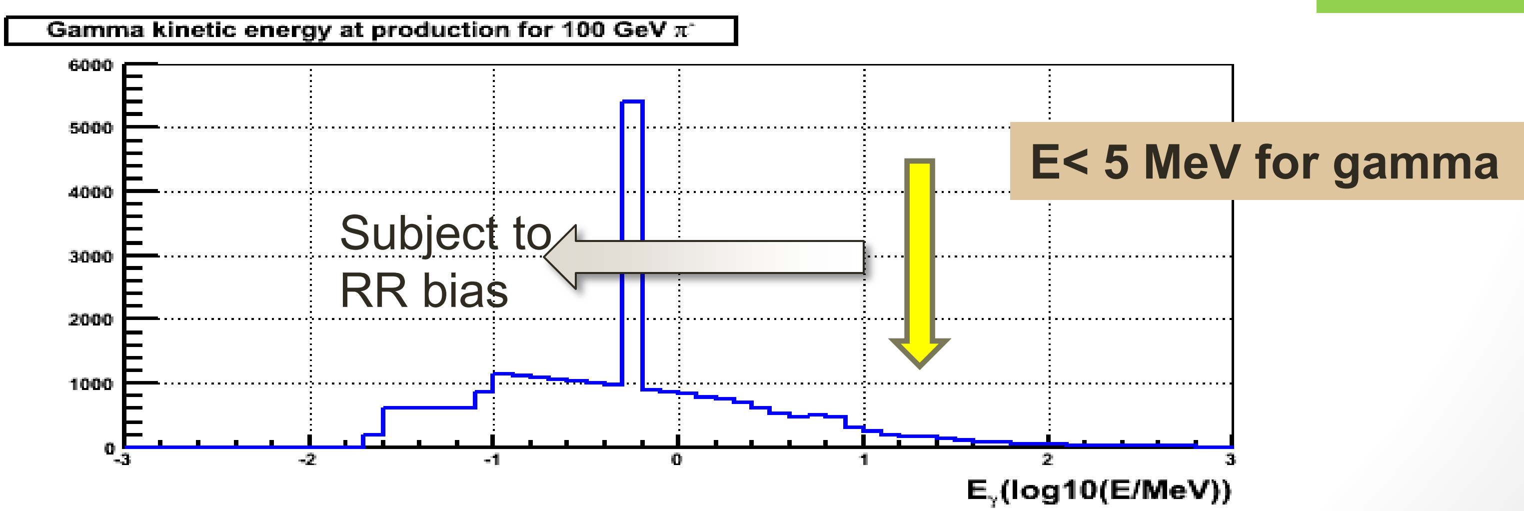


Electrons and gamma are biased below 5 MeV - 6% CPU saving

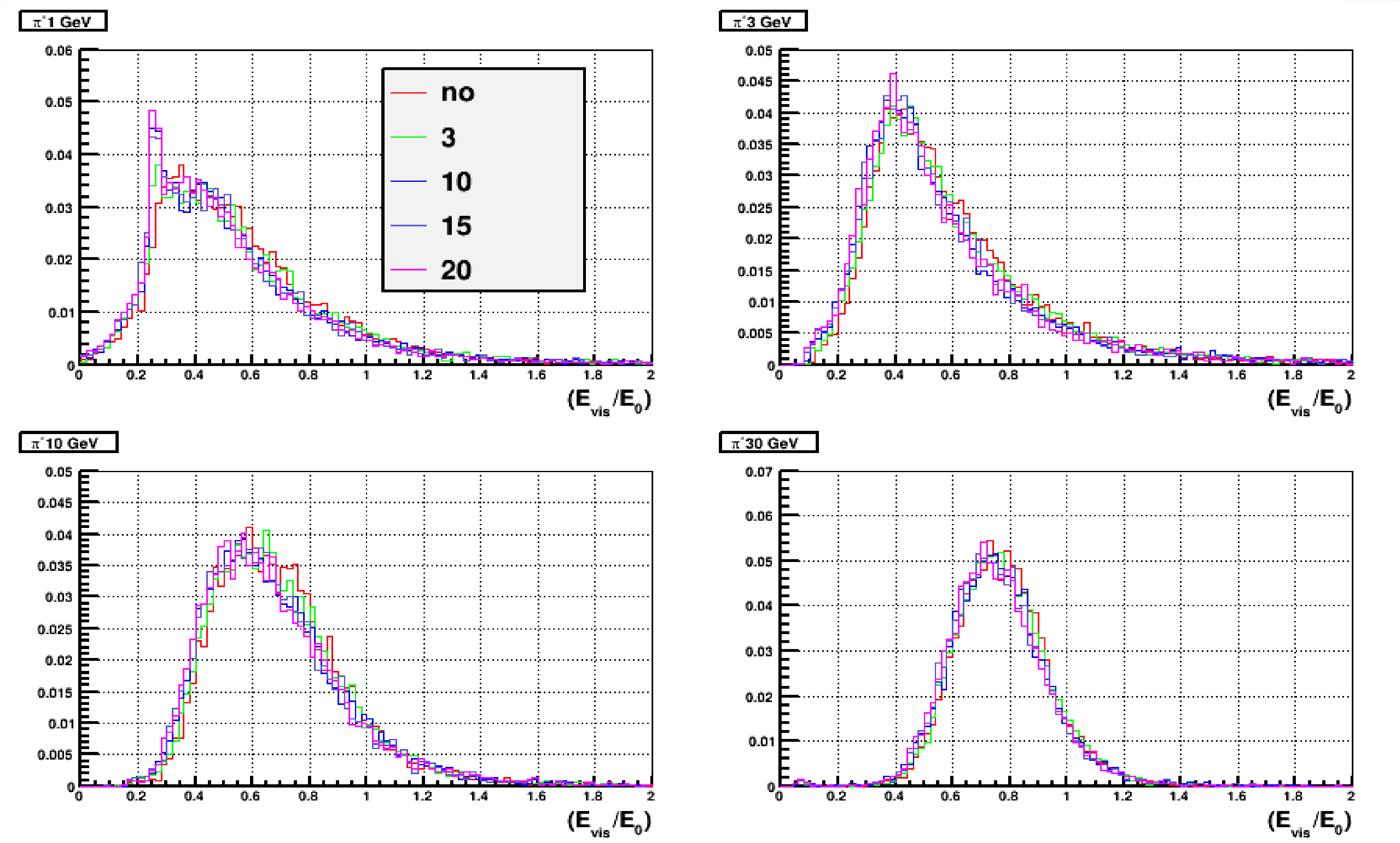
Neutrons are biased below 10 MeV - 40% CPU saving

RR: Selection of Energy limits





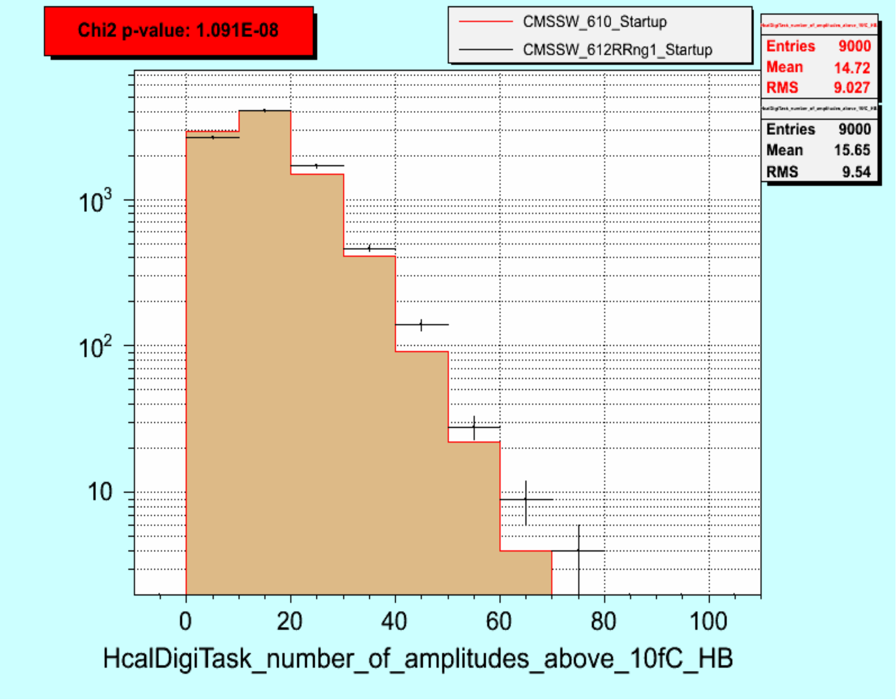
Reconstructed pions with different RR factors in ideal combined calorimeter

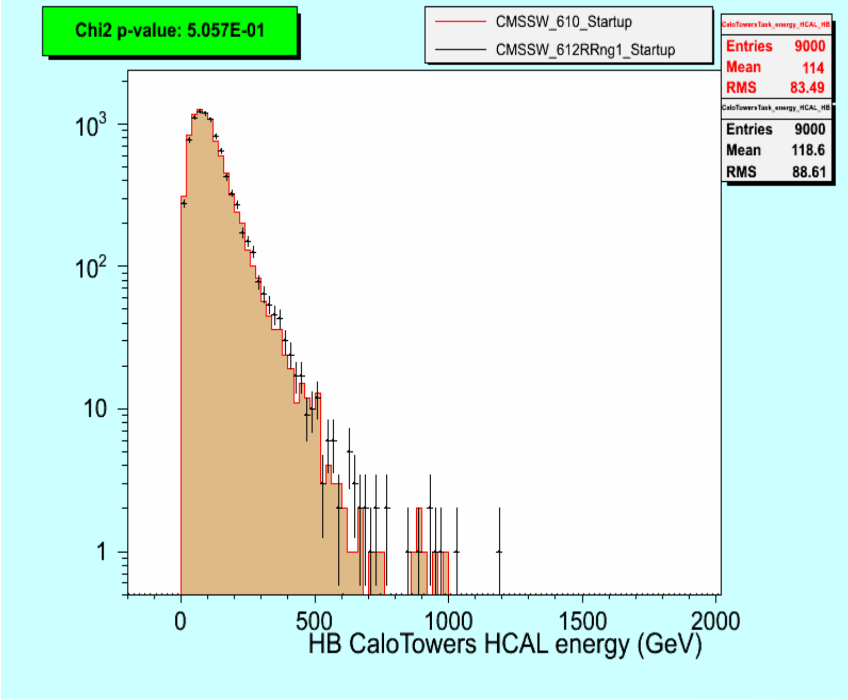


RR increasing fluctuations but not changing the shape

HCAL response for RR (J.Dittman, S.Abdoullin)

CMSSW_6_1_2_RR

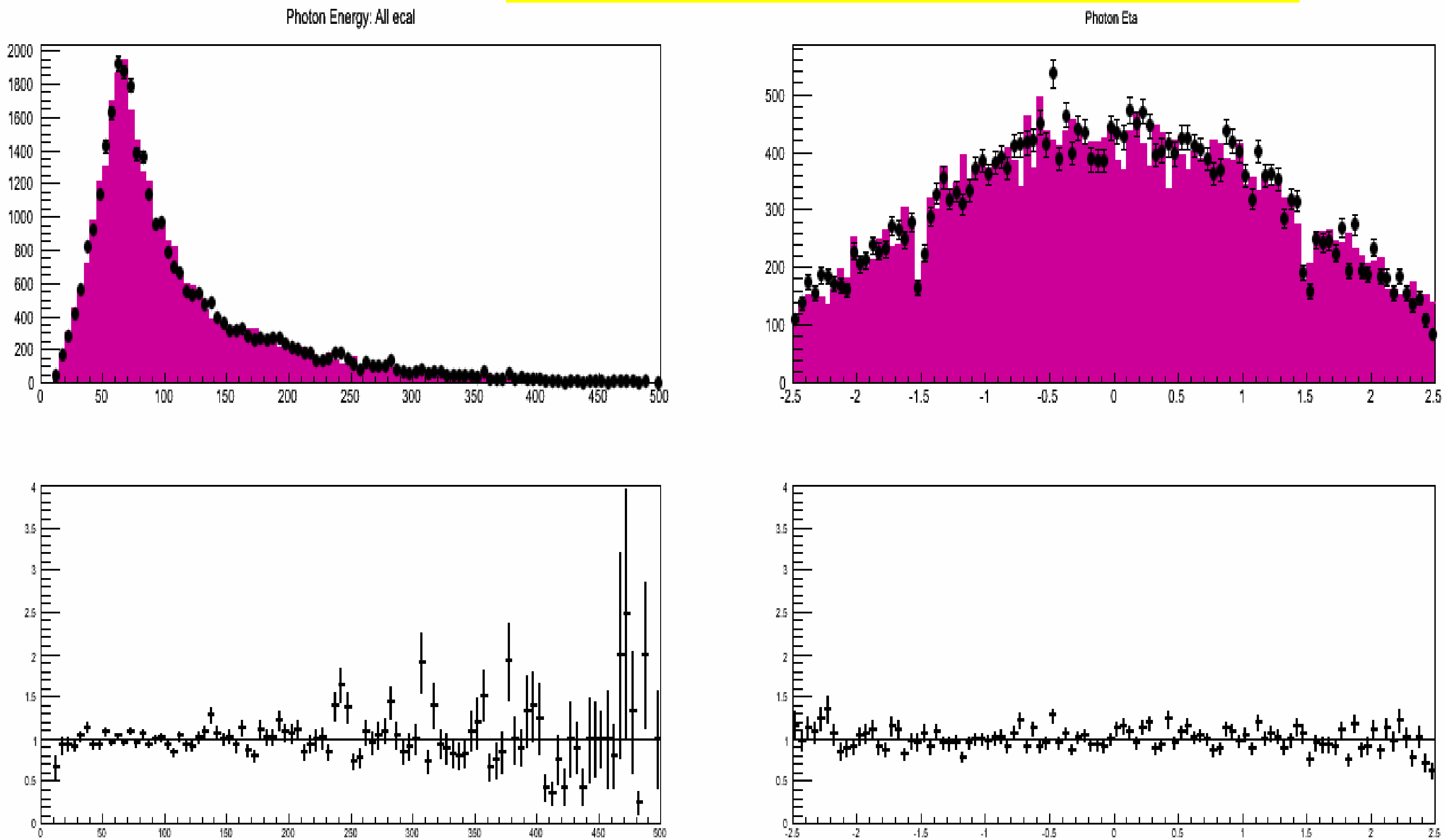




Possible problem of RR method – outliers
Very high energy deposition event in HCAL tower are not seen

Photon energies
(N.Marinelli)

Purple – 6_1_2; black dots – 6_1_2_RR
No visible difference

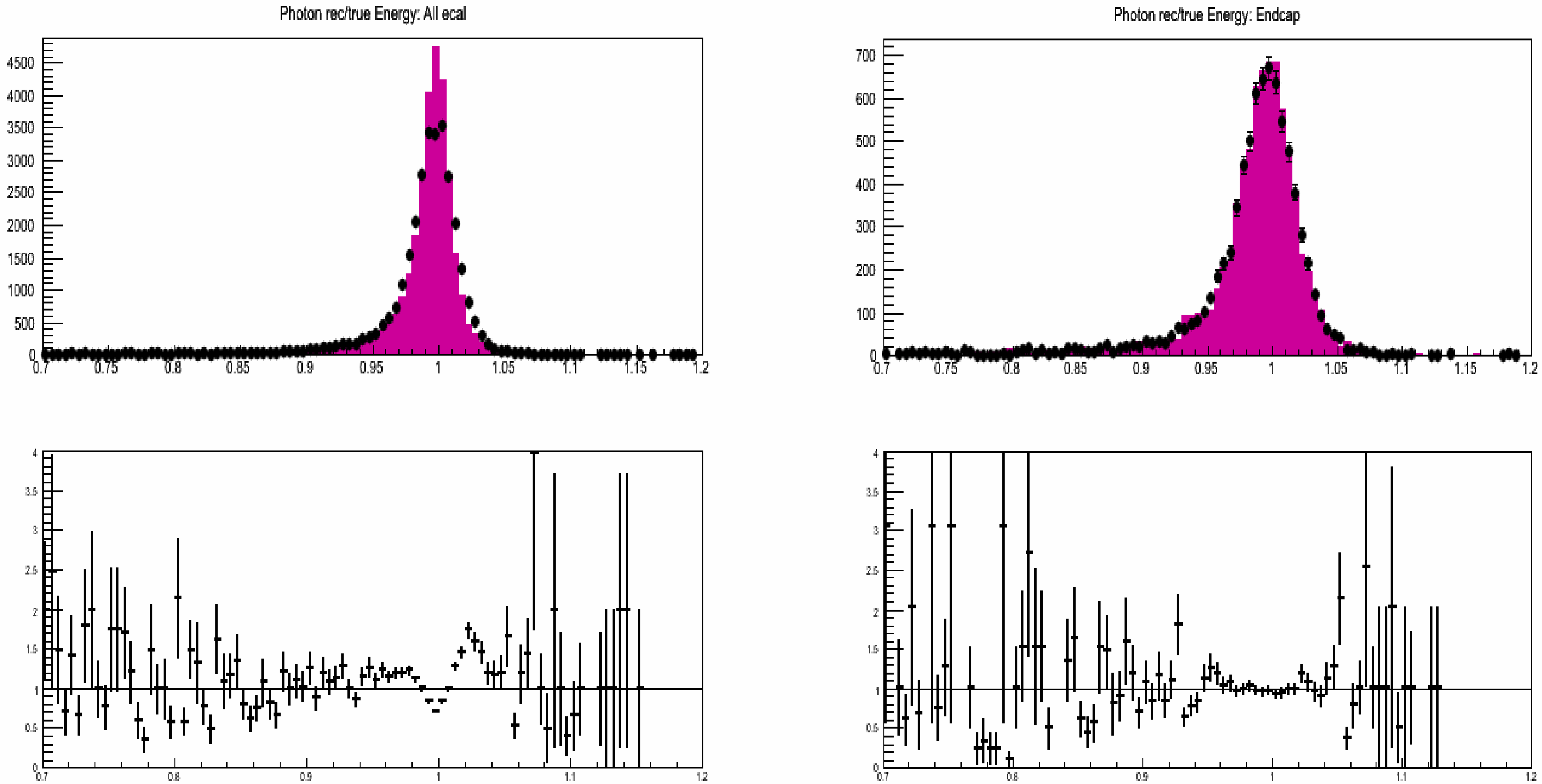


Photon energy/true energy (N.Marinelli)

EB

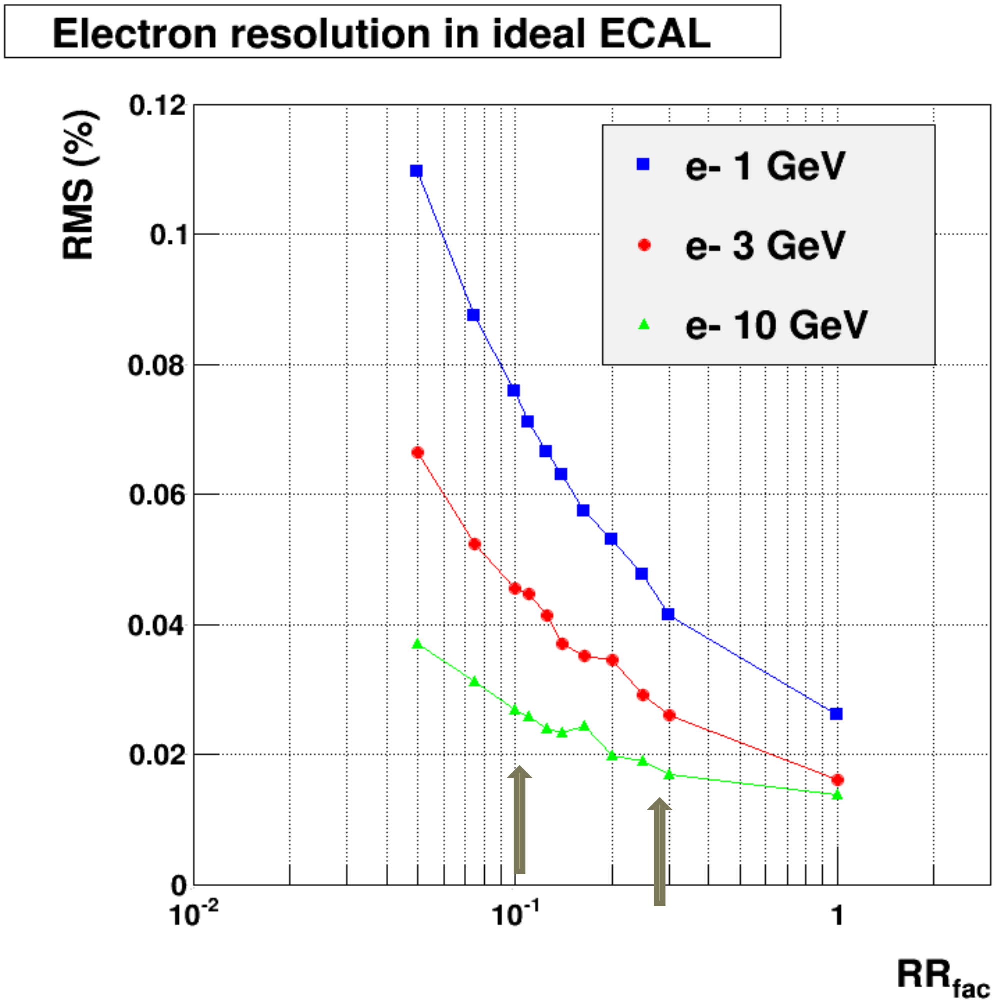
Purple – 6_1_2; black dots – 6_1_2_RR

EE



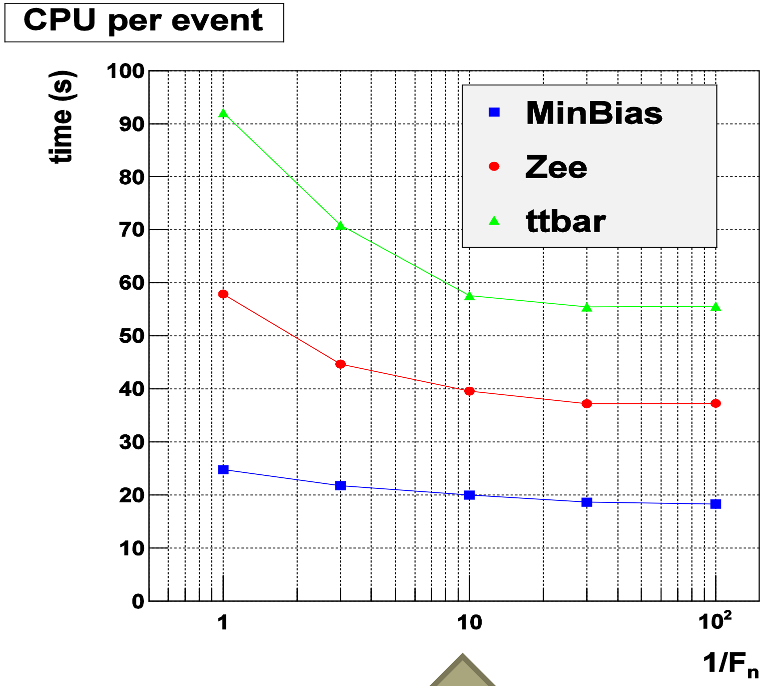
RR method may introduce some extra smearing which is seen only in ECAL Barrel due to high resolution

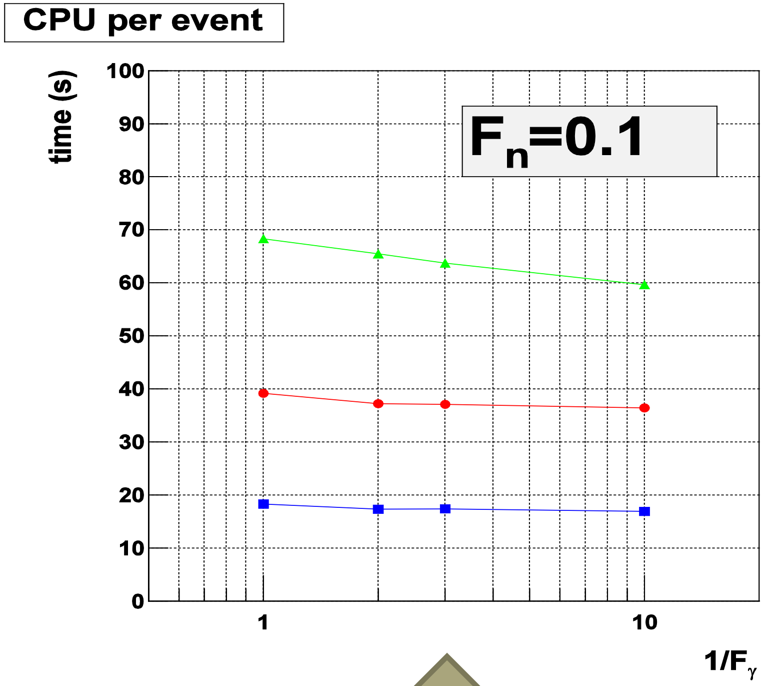
Electron energy resolution in ideal detector (standalone combined calorimeter)



- Resolution of ideal crystal calorimeter is defined by leakage
- Fluctuations of leaked energy increased if RR is applied
- Relative fluctuation increased when energy decreased
- RR factor 0.3 is more safe to ECAL gamma (not 0.1 as used before)
- factor 2 reduction of RR width in ECAL

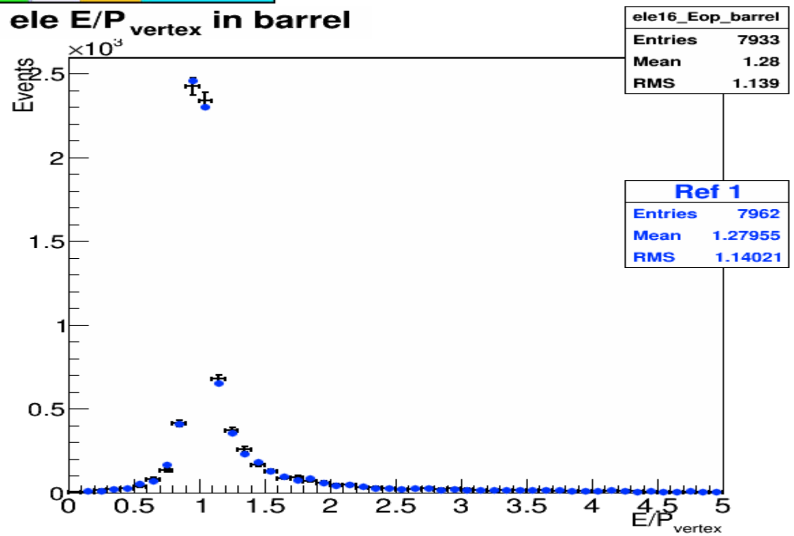
CPU for different RR factors

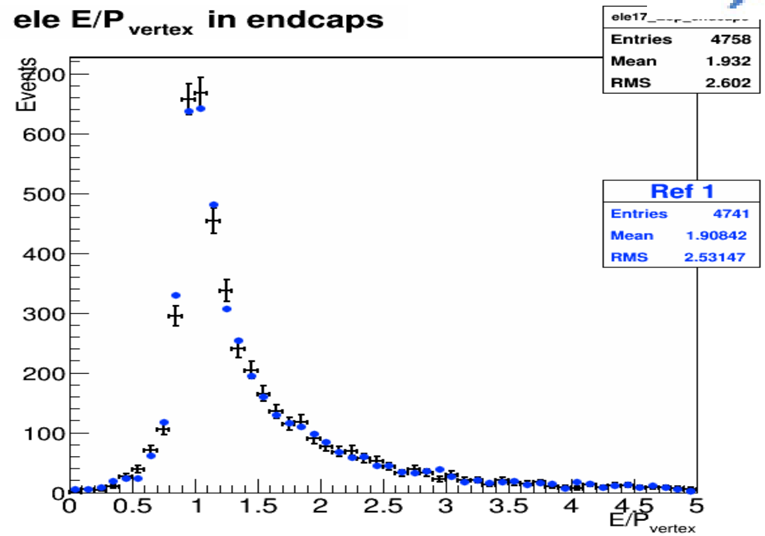


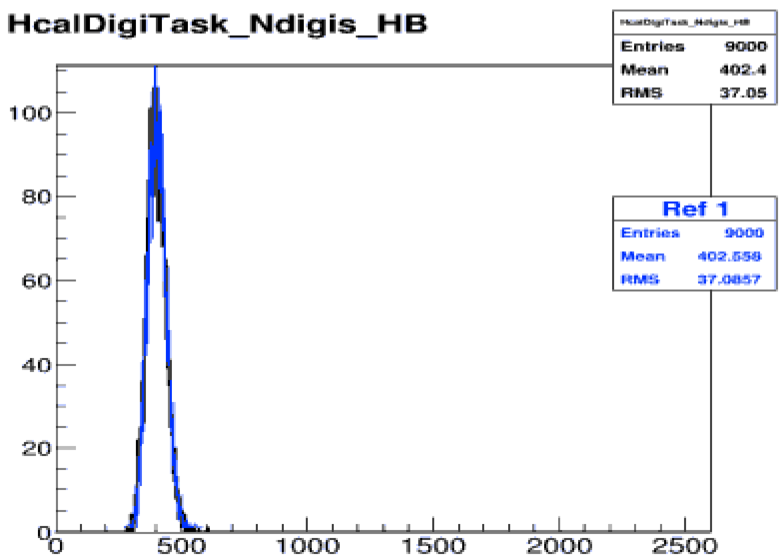


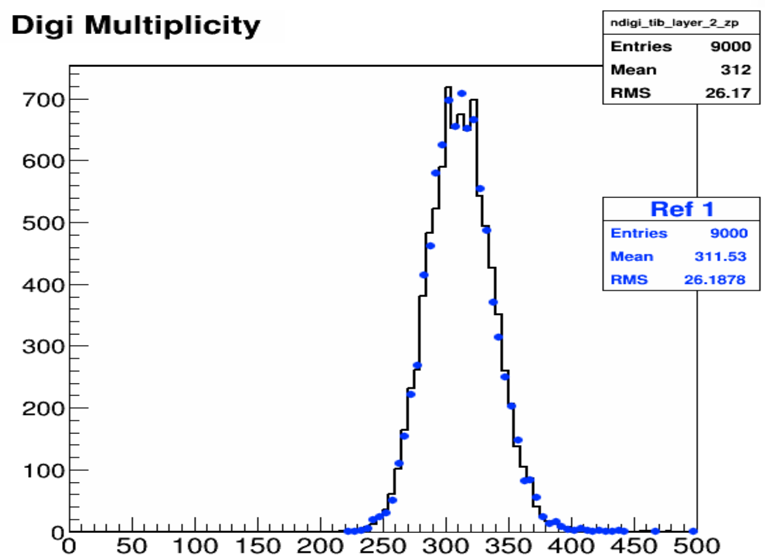
- G4Regions where RR is enabled:
  - ECAL, HCAL, Pre-Shower, IronYolk, Castor, World
- RR factors and energy limits:
  - F=0.1, E= 10 MeV for neutron
  - F=0.3, E = 5 MeV for gamma
- Special treatment for ECAl and Pre-Shower:
  - RR is not applied if primary are γ, e+, e-

Z->e+e- Electrons in ECAL and HCAL
RR is disabled for primary



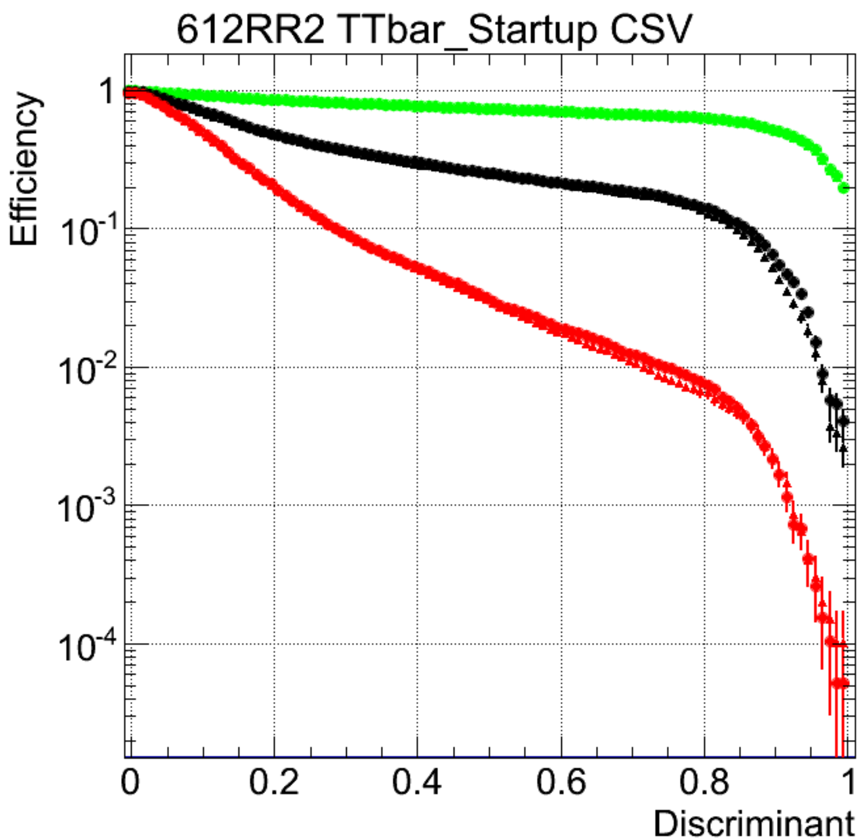


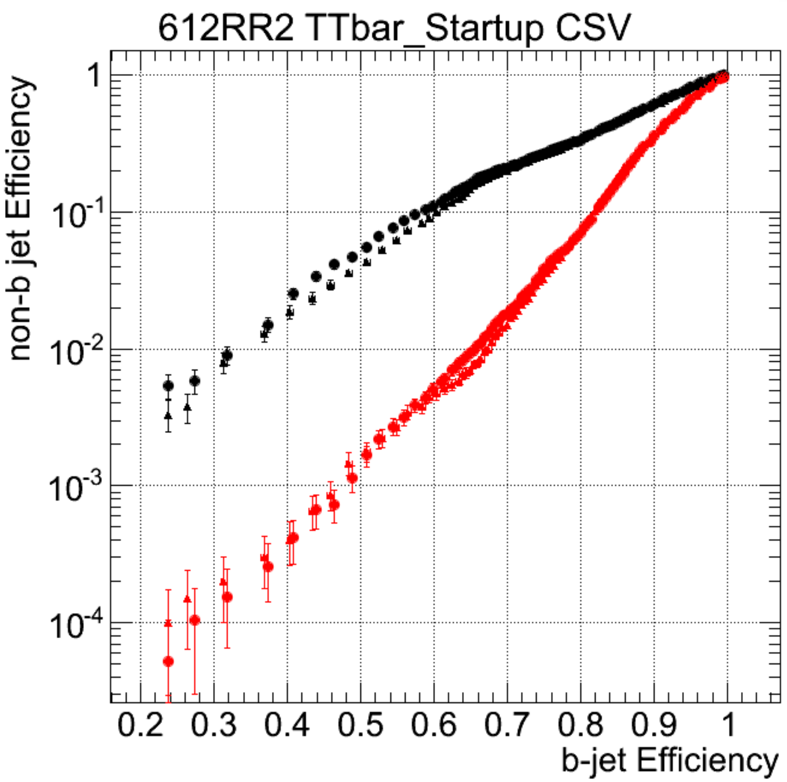




BTAG RelMon Report (A.Aubin)

If RR is used corrections to high level objects should be retuned





Russian Roulette CPU Usage

- Comparison of CPU performance between 8 TeV and 14 TeV Simulation:

Events	Energy (TeV)	No RR	RR=10	Energy (TeV)	No RR	RR=10
MinBias	8	19.3s	15.2s 78.5%	14	21.5s	16.1.s 74.2%
Z→e+e-	8	50.9s	33.4s 65.6%	14	116.9s	92.3s 78.7%
ttbar	8	87.1s	52.8s 60.6%	14	115.8s	74.3s 62.4%

Only n and γ are biased in ECAL and HCAL; RR Factor 10 is used

CMS software version CMSSW_6_2_0_pre6

CPU performance of ttbar simulation and reconstruction (Dimitrios Nikolopoulos)

- "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 7_0_0_pre4	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

CMSSW_7_0_0_pre8: RR enabled, MC truth handling optimisation, TrackingParticles and CrossingFrame disabled

TTbar @ 13TeV	FullSim no PU	FastSim no PU	FullSim PU2012	FastSim PU2012
Generation step	0.042369	0.020837	0.042369	0.020607
Simulation step	55.925123	0.213183	55.925123	0.886051
Digitization step	0.632664	0.254931	3.044882	0.296844
Reconstruction step	1.942620	1.223289	10.979556	2.831909

Comments on RR method

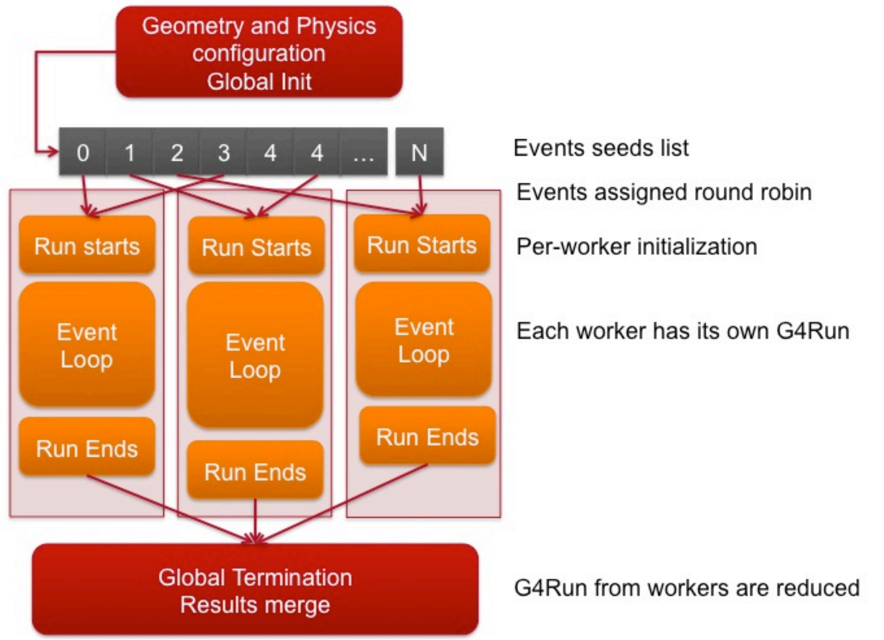
- Geant4 and CMS FullSIm is very reasonable Monte Carlo but is not ideal
  - There are several sources of systematic uncertainties due to finite accuracy of Monte Carlo
- RR method is an extra compromise of accuracy versus CPU speed
  - It cannot increase accuracy of Monte Carlo
  - However, it is proofed that is working for the main signal
- The benefit seems to be significant: ~40% CPU saving
  - This allows to simulate 13 TeV events faster than 8 TeV events for the 2012 run
- This will modify final objects, so minor re-tuning and re-training will be unavoidable
  - However new Geant4 version will require this as well

GEANT4 10.0

Multi-threading in Geant4 10.0

Event-level parallelism

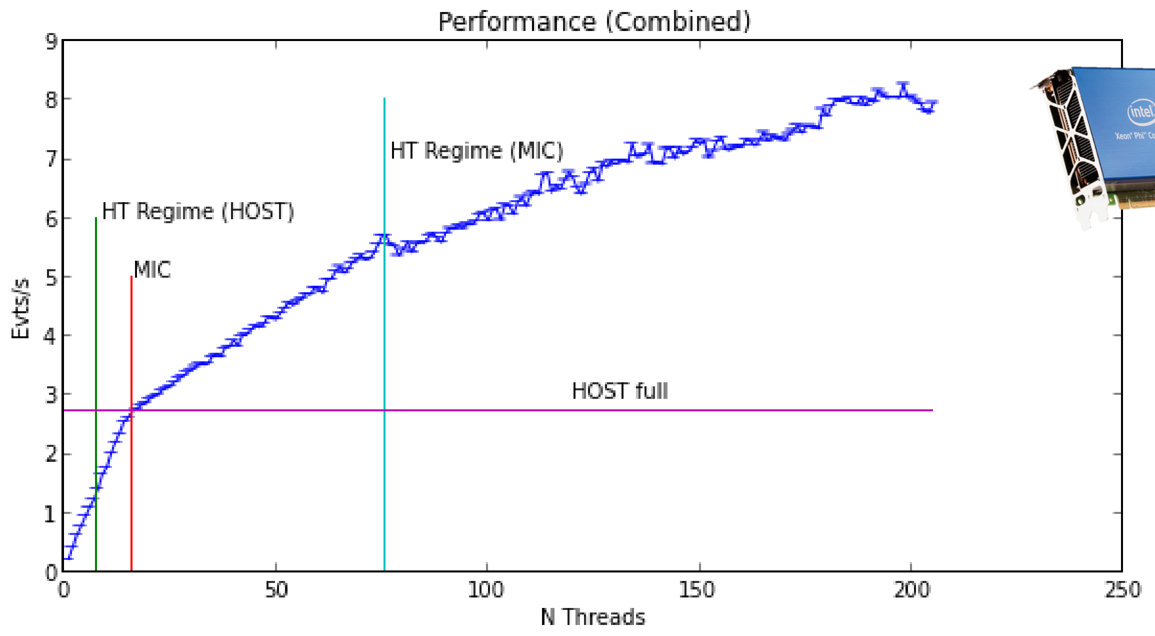
- Each worker thread proceeds independently
  - Initializes its state from a master thread
  - Identifies its part of the work (events)
  - Generates hits in its own hits-collection
  - Uses thread-private objects and state
  - Shares read-only data structures (e.g. geometry, cross-sections, …)
  - Has its own read-write part in a few ‘shared/split’ objects

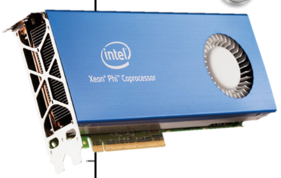


- Possibility to install/run Geant4 either in pure sequential or parallel (MT) mode
  - Choice at configuration/installation time
  - Sequential mode currently the default

Multi-threading in Geant4 10.0

- Hybrid mode: Host + Intel® Xeon Phi™ coprocessor (MIC)
  - First look at total throughput (evt/s) (*)
  - Excellent results: factor ~x3 in events produced w.r.t. host only

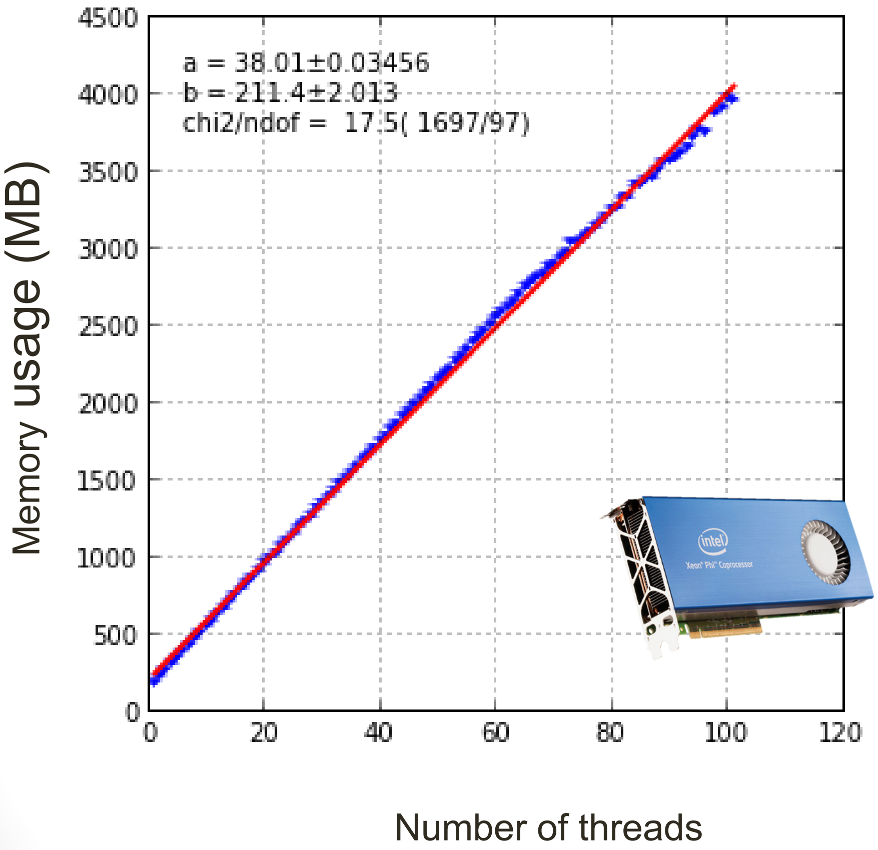




- Confirmed good scalability up to O(100) threads
- Reduced use of memory
  - (see next slide)

(*) Preliminary analysis on full-CMS benchmark by A.Dotti, SLAC

Multi-threading in Geant4 10.0



- Hybrid mode: Host + Intel® Xeon Phi™ coprocessor
  - Using out-of-the-box 10.0-beta (i.e. no optimisations)
  - 40 MB/thread
    - Baseline: Full-CMS benchmark; 200 MB (geometry and physics)
  - Speedup almost linear with reasonably small increase of memory usage

(*) Preliminary analysis on full-CMS benchmark by A.Dotti, SLAC

Summary

- Full Simulation mandate is to provide as accurate results as possible
- Full Simulation may be optimized for each concrete experiment
  - Optimal set of EM parameters may provide up to factor 2
  - Usage of parameterisation of forward detectors may significantly improve CPU
  - Russian Roulette method applied mainly for neutrons and gamma may provide about 40% speedup
- Geant4 10.0 offers new feature – event level parallelism
  - Can be used at modern hardware with better ratio $/Event
- However, analysis of LHC data cannot be effective without FastSimulation