GeantV code sprint report

Fermilab 3-8 October 2015

A. Gheata

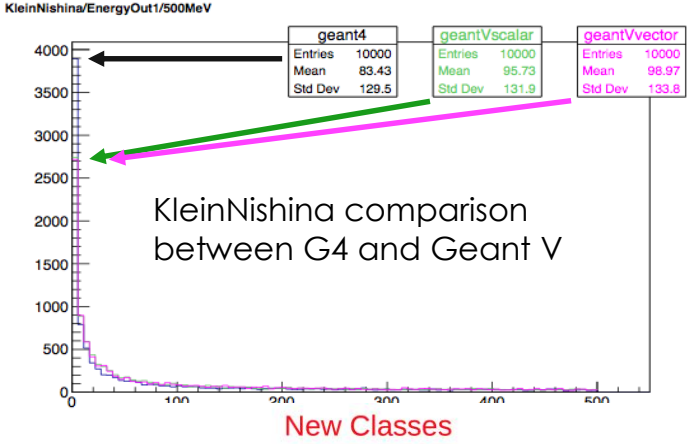
Scope & attendance

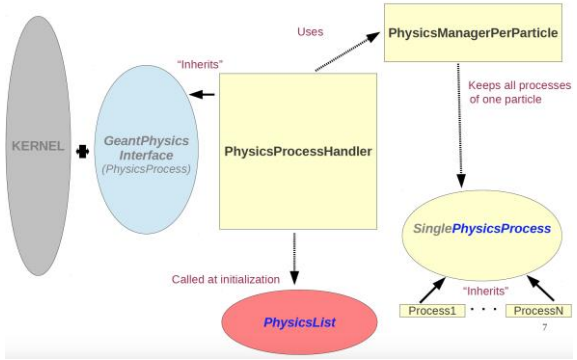
- Biannual workshop CERN/FNAL aiming to:
  - Make the point on the current project status and review individual reports (morning sessions)
  - Make joint code sprints in small groups on different hot subjects (afternoon sessions)
  - Decide on short to medium term goals and corresponding tasks
    - Definition of the “smoking gun” demonstrator
- Hosted by Fermilab the week after the Geant4 collaboration meeting
  - Big thanks for their hospitality!
- Participants
  - FNAL: P. Canal, D. Elvira, S. Yung Jun, G. Lima
  - CERN: J. Apostolakis, F. Carminati, A. Gheata, M. Novak, A. Ribon, S. Wenzel

The starting point briefly

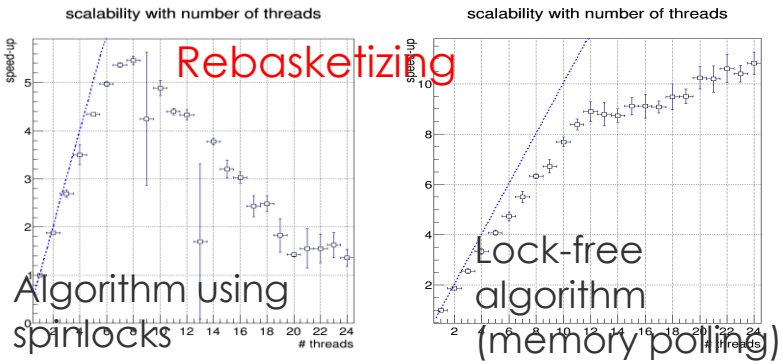
- Geometry
  - Scalar navigation demonstrated with CMS setup
  - Work on different solids ongoing
  - Voxel navigation in production
- Physics
  - New (scalar) MS model ready to integrate with Geant4
  - Compton KN model ready (modulo a bug to be fixed)
    - -> Need for a general validation procedure
  - GeantV physics design/interfacing on the board

Shape	VecGeom	Multi-Track	Internal SIMD
Box	yes	yes	
Trap + Trd	yes	yes	
Tube[s]	yes	yes	
Cone[s]	yes	(incomplete)	
GenericTrap/Arb8	(yes)	(yes)	(yes)
Tet			(targeted)
Polycone	yes		(targeted)
Polyhedron	yes		yes
Torus	yes	yes	
Parallelepiped	yes	yes	
Extruded solid			(targeted)
MultiUnion			(targeted)
Tesselated Solid			(targeted)
Composites	yes		
Templat. Composites	(yes)	(yes)	
Hype,Ellipsoid, Parab	yes	yes	
Orb/Sphere	yes	yes	
... the rest ...			

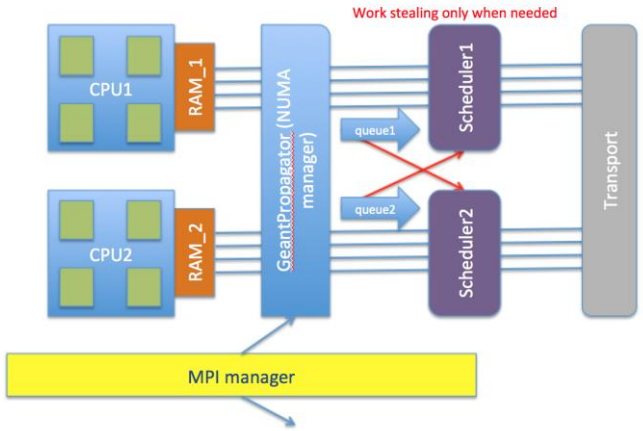


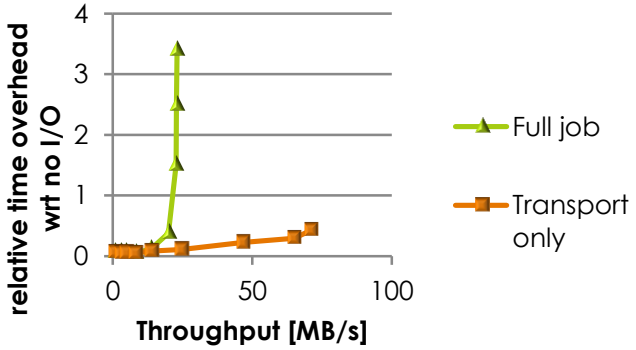


The starting point briefly



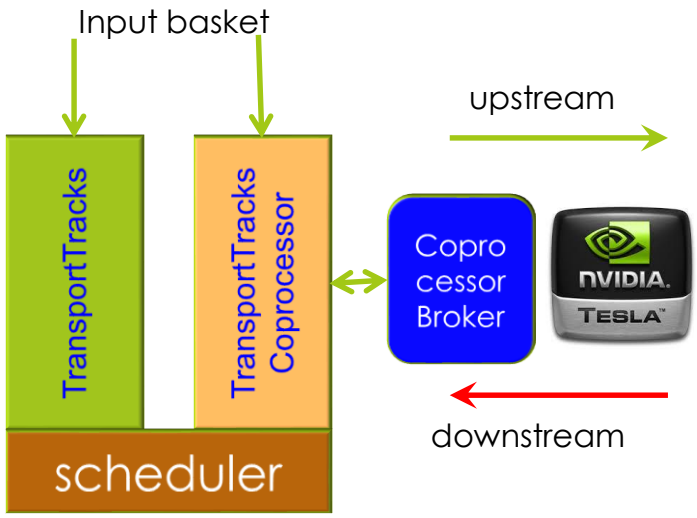
- Scheduler
  - Concurrency issues even in lock-free rebasketizer
  - Non locality effects observed
    - NUMA awareness to be addressed
    - Re-use basket in next step
- I/O
  - Concurrent streaming of hits demonstrated
    - Multiple threads pushing hit blocks in concurrent queue
    - Single thread writing to tree
    - Bottleneck in ROOT serializing & unzipping





The starting point briefly

- Cuda transport kernel
  - Coprocessor broker implemented
    - Offload track propagation for step
    - Still to be debugged
- Vector physics on CUDA
  - Issues for reproducing previous results (FNAL CUDA prototype)



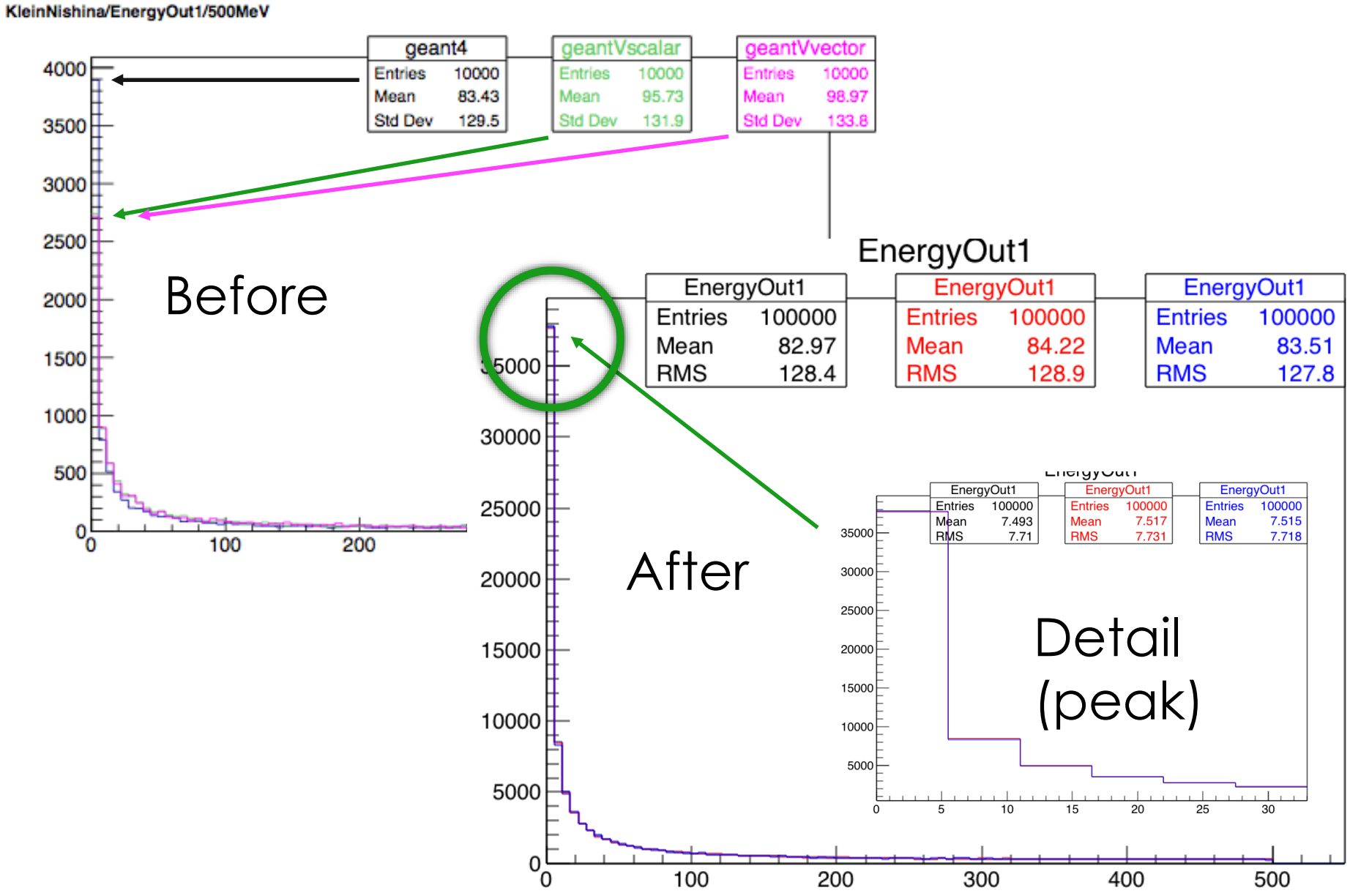
Progress on Design & Physics

- Agreed with proposed design for interaction of kernel with physics process (classes & key methods)
- Started to investigate (with help from Sandro) on the use of templates to avoid (or at least reduce) the virtual function calls
- Decided to give priority to the following 3 electromagnetic processes for electrons:
  - Multiple scattering
  - Ionization
  - Bremsstrahlung
- Complex from the point of view of the structure and interfaces for the physics
- They will allow to refine the design of the whole physics, and then test the template mechanism
- We did not discuss about SIMD Random Number Generation, but we will have to confront this at some point

Progress with GPU

- Vector physics
  - Understood the bias seen from Compton (first bin sampling)
  - Found solution to reduce the bias (re-weighting the sample position instead of using the mid-point)
  - Understood performance of vector physics models wrt old FNAL GPU prototype and recovered reasonable gain in CPU/GPU (x2-3 compilation flag, x2-3 number of tracks)
  - Understand the design of the proposed physics interface (physics processes)
  - Identified number of items to improve the current implementation of vector physics models
- Coprocessor: several fixes in setup, compilation, data structure transfers
  - Fix scheduling issues in the Coprocessor (missing setup)
  - Add physics to CUDA code path (run on CPU for now)
  - Fix issues with data structure alignment dissonance on CPU and GPU.
  - Resolve build/linking issue on GPU in master.
  - Tracking down issue with volume pointer value on GPU side.

KleinNishina comparison between G4 and Geant V



Progress with Geometry

- Bug fixes on the tube and polycone, including visualization
- Use of root visualization tools to speed up geometry debugging
- Streamlined the treatment of special cases in polycone implementation

Progress with Scheduler

- Agreed to change handling of baskets: from SOA (current) to AOS.
  - Track copying can be postponed to the moment the SOA needs to be dispatched to geometry and physics
  - Handling of AOS using track pointers is more lightweight from the point of view of memory copying (reshuffling not needed) and re-basketizing (copy pointers rather than data in concurrency mode)
- Plan is to tackle this as soon as we finish more urgent business (geom. vectorization and physics validation with VecGeom)
- Expected to improve Amdahl and general performance, but need
  - Track management as a pool instead of basket management (the basket will only be locally produced and used when dispatching)
  - Major redesign and requires many changes in the code

Validation DB

- Developing new physics models brings up validation
  - What is the reference, how reproducible, what is the procedure?
  - A simulation Validation DB seems to be the solution
  - FNAL started to develop a tool for Geant4 (Hans Wenzel et al)
    - Web application on top of a DB
      - Select experiment, make selections, …
- Content
  - Experimental results (thin target data) as reference for simulations
  - Results of previous releases (regression tests)

PoW for Validation DB

- Hans Wenzel will prepare an “export pack” of the database and the associated facilities. He will provide you all the pointers.
- We will develop a C++ API (all the code is in the GEANT svn repo, we will add this there) and we will modify some G4 example (e.g. example 19) to derive the validation data from the DB.
- Once the proof-of-concept achieved, we will start a campaign to modify all G4 examples to work from this DB and give the corresponding data to our colleagues at FNAL for inclusion.
- In general we should not include the data ourselves in the DB but rather provide them to FNAL and get back a new version of the database which is updated.
- We will surely have a number of requirements that we will forward to our FNAL colleagues for consideration. We are of course happy to help also in the implementation, if this is feasible.
- We will work at establishing and enforcing a group policy for whoever develops physics code in the group to derive the validation data from the database and to upload the relevant validation results (to be further detailed).
- When this works, we will consider the possibility to prepare a global G4 policy about validation to discuss at the appropriate level.

Decision on Units

- Internal units for GeantV will be GeV, centimeters, seconds and derived units
  - Test programs, particularly those derived from G4 can be written in any unit
  - GeantV production code will conform to the above units
  - All constexpr parameters will be initialized using units, not bare numbers
  - User code will just need to be using the right namespace (geant:: vs. clhep::)
- We will have a phys const header in GeantV and a numerical / geom const header in VecGeom
- This decision will be announced to the list

Decision on Cuts

- Allow length and energy cuts
- We will have transport cuts for all particles (energy/length and time)
- Charged particles will have the option to be “ranged out”
- Production cuts will coincide with transport cuts for all particles, potentially apart those with continuous-discrete processes

Decision on Physics interface

- A&M proposal is accepted as presented (and amended)
- We will provide feedback to the first implementation
- Waiting for the code

Decision on Code maintenance

- Code has been moved to gitlab
- We have a single *main* branch (master)
  - Small fixes are checked in directly
  - Feature developments: pull; branch (local & remote); develop; merge (rebase) master into; make merge request
- We will tag regularly (really!)
- For the release we will tag/branch/stabilize/tag/release/patch

Smoking gun definition

- Full CMS geometry
- Basketized transport
  - Run in single thread/MT & scalability
- VecGeom navigation with voxelisation & vectorisation + solids
- Scoring
- Mag Field (CashKarp)
- Parametrised solenoid field with CMS dipole nominal strength
- Tabulated physics, no MS
- GPU & Xeon Phi will be included

Major elements missing for the “smoking gun”

- Navigation in non-uniform magnetic field
- Serialization of x-sec / final states
- Make the current state working on GPU (full step taken by the coprocessor, physics + geometry)
- Move physics code on GPU
- Add missing management part to the CUDA side ("NavigationState" on the GPU must give back right GPU pointer; the same state on the CPU gives back CPU pointer)
- Put into place new navigation classes and infrastructure ("LogicalVolume has pointer to a Navigator")
- Test new infrastructure for XRay Benchmarker
- Change Geant-V side to make use of new navigation classes
- Optimize navigation (heuristic on which navigators to use when)
- Understanding phys difference between G4 and GV