Summary of Vector Physics (Status) and Vector Core (Proposal)

Andrei, John, Mihaly, Soon (4/24) &

Calebe, Guilherme (A.,L.), Philippe, Sandro, Soon (4/27)

Vector Physics Model - Status

1. Revisit/improve the current implementation of alias sampling methods
   - Building table (log binning in E_in, normalized sampling ranges in [0,1], …)
   - Selecting table/sampling
   - Interpolation (uniform/linear)
2. Connectivity to the framework
   - Vectorized physics “processes”
   - Common interface for tabulated/vectorized models

Vector Physics Model - Needs

3. Core libraries/headers
   - Global (system) header: system of units, physics constants
   - Base Vector: basic VecGeom + Lorentz-vector …
   - Physics utility classes: element, material, particle object (static properties such as definition, pid …)

Other requirements

4. Physics list: postponed
5. SIMD random number generator
6. Validation and benchmarking
    - Performance and quality
7. Task control: one main JIRA + sub tasks

Vector Core: Proposal

- Requirements
  - Generic
  - SIMD Vectorization
  - Consideration for coprocessors
- Components (identified so far)
  - System header (global, physics)
  - Vector algebra (extension of VecGeom basic vectors)
  - Physics utility classes
  - Variable size objects and bit sets

Decouple VecCore from VecGeom

- Global header
  - Common macros, typedef, …
- Backend
- Base (array, vector/2D/3D, transformation, point, SOA, …) + internal vectorization
- Random
  - Vc::Random (as a starter)
  - Generic SIMD: independent state per thread
  - Reproducibility (scalar vs. vector), precision?

Vector Physics Utilities

- Vectorized material classes
  - Element
  - Material
  - (UMaterial by Ferderico?)
- Particle object

Physics Header Files

- System of units
- Physics constants
- (adapt existing CLHEP headers for now)
- Use geant repository

Remarks

- Do not consider how to package/structure them for now
  - Minimal requirements for the next milestone
- Use existing repositories for sub-components
  - vecgeom = VecGeom + VecCore
  - geant = geantV + headers + material