Recent developments and plans of Geant4 kernel

Makoto Asai (SLAC)
On behalf of the Geant4 Collaboration
Calorimetry for High Energy Frontier 2019
Hakata, Japan
Outline

• Overview of planned developments for 2019~
  – Kernel modules
  – Physics (see talks by V.Ivantchenko)

❖ Detailed patch release notes:
  • [http://cern.ch/geant4-data/ReleaseNotes/Patch4.10.5-1.txt](http://cern.ch/geant4-data/ReleaseNotes/Patch4.10.5-1.txt)

❖ List of planned features for 2019:
  • [http://cern.ch/geant4/support/planned_features](http://cern.ch/geant4/support/planned_features)

• Current version
  – Version 10.5-patch01 (released on April 17th)

• Next release
  – Version 10.6 (scheduled on December 6th)
Infrastructure

- Migration of HyperNews fora to Discourse •
- Migration of web site to Drupal-8
  - Upgrade from existing Drupal-7 site
- Testing infrastructure in Jenkins •
  - Adoption of Docker containers for testing
  - Versioning of builds through pipelines
- Enhancements to Geant4 GitLab workflow •
  - Addition of code formatting hooks; integration with Coverity analysis; ...
  - Adaptation to new features in future versions of GitLab; study of GitLab CI use
- Build and publication of Docker/Singularity images for releases
- Migration of static preprocessor –D flags to #define/undef directives •
- Modularization of Geant4 Libraries •
  - Global/granular/optional
- Optimization of Data Libraries
  - Simplify data library configuration/location using layered lookup via self-location, single environment variable, UI commands/C++ API
  - Provide C++ API for accessing/parsing data libraries
  - Optimize file access patterns and formats to minimize number of small files opened
Recent developments and plans of Geant4 kernel

M. Asai (SLAC)

<table>
<thead>
<tr>
<th>Topic</th>
<th>Replies</th>
<th>Views</th>
<th>Activity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rpm based installation in fedora 30</td>
<td>2</td>
<td>119</td>
<td>12m</td>
</tr>
<tr>
<td>Getting Started</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9 keV cutoff on Gamma spectrum with G4MuBremsstrahlungModel!</td>
<td>1</td>
<td>26</td>
<td>11h</td>
</tr>
<tr>
<td>Physics Processes, Models and Cross Sections</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Geant4 Install Folder doesn’t populate</td>
<td>1</td>
<td>27</td>
<td>2d</td>
</tr>
<tr>
<td>Getting Started</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Biasing of neutron production cross section</td>
<td>1</td>
<td>121</td>
<td>2d</td>
</tr>
<tr>
<td>Particles, Track, Event, Run and Biasing</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Memory leak in SteppingAction in mutithread mode</td>
<td>2</td>
<td>78</td>
<td>2d</td>
</tr>
<tr>
<td>Particles, Track, Event, Run and Biasing</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Is it safe to change event ID outside of RunManager?</td>
<td>1</td>
<td>18</td>
<td>2d</td>
</tr>
<tr>
<td>Particles, Track, Event, Run and Biasing</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LUT Davis optical model</td>
<td>13</td>
<td>119</td>
<td>3d</td>
</tr>
<tr>
<td>Physics Processes, Models and Cross Sections</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Record deposited energy in a process class</td>
<td>2</td>
<td>35</td>
<td>3d</td>
</tr>
<tr>
<td>Recording, Visualizing and Persisting Data</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OpenGL error during ExampleB1</td>
<td>5</td>
<td>59</td>
<td>4d</td>
</tr>
<tr>
<td>Recording, Visualizing and Persisting Data</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Geometry & Transportation

• VecGeom
  – Implementation of missing shapes/constructs: ellipsoid, elliptical cone, etc.
  – Enhancements to specialized navigators, neighbor volume detection
  – Addition of replicas/divisions – proposed summer project
  – Use of Embree library for tessellated shapes – proposed summer project
  – GDML writer & ROOT I/O persistency – proposed summer project
  – Generation of polyhedral meshes for shapes
  – Overlaps checking – proposed summer project
• Implementation of a prototype navigator based on VecGeom
• Separate safety computation and state from navigator
  – Implement strategy for a light-weight base navigator class not holding navigation state
• Profiling and optimization of multiple navigation
  – Revise design and implementation of multiple navigation and coupled-transportation
• Revision of transportation processes
  – Specialized transportation processes for neutral and charged particles
Tessellated solids

- **G4TessellatedSolid**
  - Generic solid defined by a number of facets (**G4VFacet**)
  - Facets can be triangular (**G4TriangularFacet**) or quadrangular (**G4QuadrangularFacet**)
  - Constructs especially important for conversion of complex geometrical shapes imported from CAD systems
  - But can also be explicitly defined:
    - By providing the vertices of the facets in *anti-clock wise* order, in *absolute* or *relative* reference frame
    - GDML binding

- **G4ExtrudedSolid** is re-implemented to internally use **G4TessellatedSolid**.
Geometry updates – New solid library

- An important effort was begun in the last few years to write a new solid library, reviewing at the algorithmic level most of the primitives and provides an enhanced, optimized and well-tested implementation to be shared among software packages.
- In most cases considerable performance improvement was achieved.
  - For example, the time required to compute intersections with the tessellated solid was dramatically reduced with the adoption of spatial partitioning for composing facets into a 3D grid of voxels.
- Such techniques allow speedup factors of a few thousand for relatively complex structures having of order 100k to millions of facets, which is typical for geometry descriptions imported from CAD drawings.
  - Consequently, it is now possible to use tessellated geometries for tuning the precision in simulation by increasing the mesh resolution, something that was not possible before.

<table>
<thead>
<tr>
<th>Method</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inside</td>
<td>2423x</td>
</tr>
<tr>
<td>DistanceToIn</td>
<td>1334x</td>
</tr>
<tr>
<td>DistanceToOut</td>
<td>1976x</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Information</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of facets</td>
<td>164.149</td>
</tr>
<tr>
<td>Number of voxels</td>
<td>100.000</td>
</tr>
<tr>
<td>Memory saved compared with original Geant4</td>
<td>22% (51MB)</td>
</tr>
</tbody>
</table>
New “multi-union” solid

- In addition to a full set of highly optimized primitives and a tessellated solid, the library includes a new "multi-union" structure implementing a composite set of many solids to be placed in 3D space.
- This differs from the simple technique based on Boolean unions, with the aim of providing excellent scalability on the number of constituent solids.
- The multi-union adopts a similar voxelization technique to partition 3D space, allowing dramatically improved speed and scalability over the original implementation based on Boolean unions.

New in v10.4
G4MultiUnion

G4MultiUnion* munion_solid = new G4MultiUnion("UnitedBoxes");

for( int i=0 ; i < nNode ; i++)
{
    G4Box* aBox = new G4Box(...);
    G4ThreeVector pos = G4ThreeVector(...);
    G4RotationMatrix rot = G4ThreeVector(...);
    G4Transform3D tr = G4Transform3D(rot, pos);
    munion_solid -> AddNode( *aBox, tr );
}

munion_solid -> Voxelize();

Note : G4MultiUnion is a solid. Use it to create a logical volume.
Field Propagation

- Enable default use of interpolation in intersection calculation in field propagation
- Revise protocol between transportation and tracking to better cope with particles looping in field
- Review accuracy of boundary crossing in field
  - Recent ALICE and CMS requirement
Materials & Biasing

- Implementation of an extension for multiple particle type biasing
- Enrich event biasing options
  - leading particle biasing, DXTRAN-like biasing, implicit capture
- Extend generic biasing scheme for at rest case
- Prototype implementation of generic biasing techniques:
  - biasing of charged particles (with cross-section changing over the step)
  - occurrence biasing (continuous density change inside a same volume)
  - material/isotope biasing; Woodcock tracking
- Adoption of the new elastic differential cross-section class in hadron elastic
- Implementation of new elastic differential cross-section class to be used in DXTRAN biasing option
- Investigation on potential difficulties in propagating tiny weights for large cross-section change (neutrino interactions)
- Revision of the GFlash fast-simulation model
Persistency & Analysis

- Enabling of import/export of assemblies envelopes in GDML
- Improvements to n-tuple merging in row-wise mode
- Addition of analysis "executive" to provide possibility to choose output type at run-time
- Additional flexibility in resetting/deleting histograms
- Review support for writing same histogram/profile on file several times
- Handling of more files by analysis manager

Particles & Track

- Improvements and update of G4IonTable and G4ParticleTable to cope with muonic atom and hyperons
- Review of production thresholds
Run, Event & Detector Response

• Multi-threading:
  – Workspace and memory cleanup in MT
  – Finalize new design of threads though tasking mechanism (allow threads to join/leave workers pool)
  – Implement hooks for allowing sub-event level parallelism

• Implementation of phase-space file interface to GPS

• Revision of production thresholds

• Command-based scoring for real-world volumes
Extending command-based scorer

• Current command-based scorer works as following.
  1. Construct a parallel world with mesh geometry and register a dedicated G4ParallelWorldProcess to ProcessManager of all particle types
  2. Instantiate thread-local G4MultiFunctionalDetector and register requested PrimitiveScore(s)
  3. Set thread-local G4MultiFunctionalDetector to logical volume of the mesh cell in the parallel world
  4. HitMap of each PrimitiveScore is summed up after each event, and merged over the worker threads after the event loop

• Idea is to provide the same command-based scoring mechanism for volumes defined in mass world.
  – Steps 2. and 4. can be reused. The only addition is setting thread-local G4MultiFunctionalDetector to a logical volume in the mass world
  – Thanks to the recent developments, a logical volume may have more than one sensitive detectors

• Also, we can utilize the G4Analysis package to create histograms, plots, n-tuples on the fly without writing any single line of C++ code.
Command examples

- Construct geometry via reading GDML or ordinary detector construction, etc.
- Command samples
  
  ```
  /score/create/realWorldLogVol <logVolName> <geoLevel>
  /score/quantity/energyDeposit <primitiveScorerName> <unit>
  /score/close
  /score/analysis/1D/create <logVolName> <primitiveScorerName> <copyNo>
  /score/analysis/1D/set <nBin> <low> <high> <unit>
  /score/analysis/1P/create <logVolName> <primitiveScorerName> <nBin>
  /score/analysis/1P/set <low> <high> <unit>
  /score/analysis/plot
  /run/beamOn 1000
  /score/dumpAllQuantitiesToFile <logVolName> <fileName>
  ```

  \(<logVolName>\) could be the name of any logical volume used in the mass geometry, either hard-coded or imported from CAD/GDML

  Note: some command names are still tentative
Recent developments and plans of Geant4 kernel

M. Asai (SLAC)

```
/score/create/realWorldLogVol Gap 1
/score/quantity/energyDeposit eDepRW MeV
/score/close
/score/analysis/1D/create Gap eDepRW
/score/analysis/1D/set 100 0. 50. MeV
/score/analysis/1P/create Gap eDepRW 10
/score/analysis/1P/set 0. 50. MeV
/score/analysis/plot
/run/beamOn 1000
```
Recent developments and plans of Geant4 kernel

M. Asai (SLAC)

- Entries: 1000
- Mean: 15.4872
- RMS: 9.23518
Recent developments and plans of Geant4 kernel - M. Asai (SLAC)
Plan

- Version 10.6 will include some limited functionalities
  - `/score/create/realWorldLogVol` and related commands to define scorers to mass-world volume
- Year-2020 release will have full functionalities including histogram/plot/n-tuple
  - histogram/plot/n-tuple will also be available for parallel-world scoring mesh defined through current command-based scorer
  - Part of the code will be inside Geant4 source, and other part in an example
GUI & Visualization

• Integration of G4Py module to CMake build.
• OpenGL drivers:
  – New driver OGLFile to produce image files in batch jobs
  – Improvements to toolbar in OpenGL Qt
  – Adaptors to newer OpenGL versions, exploit new functionalities and replace deprecated calls.
• Other drivers:
  – New Apple/Metal driver for MacOS and native Qt driver.
  – New driver for export to format readable by Paraview
  – New driver G4DAE exporter for export in Collada format
  – Updates to OpenInventor Extended Viewer
  – Development of visualization solutions for iOS and Android devices
  – Change from flat format to hierarchical format in VRML
• New tool to support high resolution transparent visualization with ability to rotate and zoom
• Visualisation of geometry overlaps.
• Integrated visualization of field lines.
Novice & Extended Examples

• New example "dnadamage" for Geant4-DNA •
  – Simulation of a DNA chromatin segment with molecular definition
• New cross-sections for gas materials in the "icsd” DNA example
• New example illustrating generic biasing
  – "DXTRAN" MCNP-like option and implicit capture
• Extended biasing examples •
  – Fix overlap among B02, B03 and GB03 examples
• Updating selected examples with usage of G4Accumulable •
• Porting of Geant4e and related example to multi-threading
• Extension to the DICOM reader to support RT Dose format
• Complete migration to MixMax in EM examples •
• Review of examples macros and tests •
• Complete application of coding guidelines & code review •
Following up

http://geant4.org/

- If you identify an issue.
  - Geant4 Bugzilla https://bugzilla-geant4.kek.jp
  - You may also use Geant4 Discourse to start with
- If you have a requirement / concern.
  - Geant4 Technical Forum
    https://geant4.web.cern.ch/collaboration/technical_forum
  - A few times per year, publicly open to any user to discuss requests, requirements and priorities that may concern functionality, performance, user support or any other Geant4-user related aspects.
  - Next Technical Forum will be held at CERN in January where
    - Contact me, Vladimir or Alberto if you want to make a presentation.
- If you want to make a contribution
  - For small fixes/enhancements : Geant4 GitHub
  - Large scale contribution : present at Geant4 Technical Forum and/or contact Geant4 WG coordinators.
New physics – new opportunities

- Physics of O(100TeV)
- Specialized EM model for noble liquid (e.g. liq.Xe)
- Neutrino interactions
  - Should come with enriched event biasing options
- Electron/hole drift in semiconductor
- More phonon physics
- Channeling effects and physics with crystal structure in general
  - X-ray diffraction, neutron scattering in crystal
- Single atom irradiation
- Target material polarization
- Chemical reactions of radicals in DNA-scale
- New domains?

- Note: Geant4 kernel is robust enough over 20 years of evolution. This stability enables risk-free extensions to new physics.
New computing trends

• HPC and cloud friendliness
  – Seamlessly combining MPI and MT
  – Smart data collection from millions of threads

• Code re-engineering
  – Solid library, EM physics
  – Splitting transportation process
  – Sub-event level parallelization

• GPU as a co-processor
  – Off-loading some calculations to GPU, e.g. EM physics, thermal neutron physics, DNA physics and chemical processes, etc.

• Will be integrated into Geant4 with (hopefully) minimum API changes
To sum up

- Geant4 is a general purpose Monte Carlo simulation tool for elementary particles passing through and interacting with matter. It finds quite a wide variety of user domains including high energy and nuclear physics, space engineering, medical applications, material science, radiation protection and security.

- 2019 is the 20\textsuperscript{th} year anniversary of Geant4 public releases. After 20 years with several architectural evolutions, Geant4 is still steadily evolving.
  - Latest major evolution was Geant4 version 10.0 released in December 2013 that is the first fully multithreaded general-purpose large-scale physics software in the world.
  - New physics models for coming experiments, e.g. hadronic model for multi-TeV regime, neutrino physics model, physics with crystal structure.
  - Given Geant4 is nowadays mission-critical for many users in all of above-mentioned domains, Geant4 is to be kept maintained and still evolving for at least next decade.