GEANT4 11.1 highlights

kernel modules

Gabriele Cosmo, CERN EP-SFT

for the Geant4 Collaboration

Outline

- Features and fixes introduced in release 11.1
  - Kernel modules
  - EM Physics (see talk by V.Ivantchenko)
  - Hadronic physics (see talk by A.Ribon)

➢ *Detailed release & patches notes:*

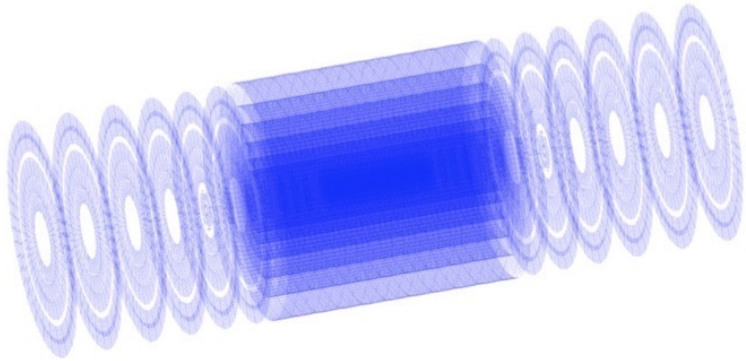
- **http://cern.ch/geant4-data/ReleaseNotes/ReleaseNotes.11.1.html**

➢ *List of planned features for 2022:*

- **http://cern.ch/geant4/support/planned_features**

Geometry

VecGeom



- Updated VecGeom library, VecGeom v1.2.1
  - Selection for enabling use made at configuration
    - https://gitlab.cern.ch/VecGeom/VecGeom/tree/v01.02.01
  - Improvements and optimisations to BVH acceleration (for CPU and GPU)
  - Added surface area heuristic for BVH construction
  - Added implementation of marching cubes algorithm
  - Extended GDML reader to support all existing shapes
  - Modernised Cmake usage and settings; switched to C++17 by default
  - Improved CUDA support in configuration and memory allocation
  - Bug fixes
- First implementation of a Surface Model demonstrator
  - Conversion of solids to framed surfaces – currently: box, tube, trapezoid
  - First raytracing benchmarks

Geometry

Navigation, Volumes, Transportation

- Added ability to optionally check for overlaps in parallel geometries
  - Through **`/geometry/run/test`** UI command
- Improved computation of surface area and cubic volume in specific solids
- Revised implementation of G4CoupledTransportation
  - Now inheriting from G4Transportation; consolidated common variables and methods
- New specialised transportation class for multiple-scattering process
- Introduced new class G4TransportationParameters to enable fine grain control of parameters for killing charged particles looping in a field
  - Optional, but if created it applies to all stable charged particles
- Revised G4FieldManager to ensure robust behaviour of the integration
  - Ensuring that **`epsilon_min`**/**`_max`** parameters are between 'minimum and maximum accepted' accuracy values
- New 2nd order symplectic integration method G4BorisDriver
  - Targeting better control of energy conservation and conservation of phase space volume

Materials, Analysis & Digits/Hits

- Materials:
  - Added and updated material descriptions in the NIST material data-base
    - G4_PHOSPHORIC_ACID, G4_ADENINE, G4_GUANINE, G4_CYTOSINE, G4_THYMINE and G4_DEOXYRIBOSE
  - Fixed density effect data for 8 materials
    - G4_Tm, G4_Be, G4_Mg, G4_Fe, and G4_Y
- Analysis:
  - Added support for writing analysis objects (histograms, profiles, n-tuples) in a file multiple times
  - Added new UI commands for analysis file management
    - **`/analysis/openFile`**, **`write`**, **`closeFile`** and **`reset`**
  - Added new analysis manager List functions and associated commands
    - **`/analysis/list [onlyIfActive]`**, **`/analysis/xy/list [onlyIfActive]`** where **`xy = h1, h2, h3, p1, p2, ntuple`**
- Digits/Hits:
  - Added support for n-tuple merging in scoring

Run, Tracking, Global

- Run:
  - Retired **`tasking`** module. Contents merged in **`global`** and **`run`** categories
- Tracking:
  - Improved diagnostic for problems with primary track parameters or with secondary tracks in particle-change classes
  - Introduced methods in G4Track for providing information on short-lived parent hadronic resonances
- Global:
  - New scheme to locate required data-sets, making use of environment variables optional
  - Fixed compilation warnings on new compiler versions and when building through the latest macOS/Xcode tool

Visualization/Interfaces

- Updated ToolsSG (TGS) drivers for X11, Xt and Qt:
  - Now fully functional and supporting histogram plotting
  - TGS driver now being built by default
- Improved rendering of complex meshes (special mesh rendering) for most drivers
  - See advanced examples ICRP110 and 145.
- "Twinkling" of volumes when centring
  - **`/vis/viewer/centreOn`**, etc.
- Cutaway now available for all visualization drivers
  - **`/vis/viewer/addCutawayPlane`**
- Qt drivers require the Qt-5 platform. Qt-6 is not yet supported

Data sets

- New data set versions:
  - **`G4EMLOW-8.2, G4NDL-4.7`**
- In order to use ParticleHP for charged particles (protons, deuterons, tritons, He3 and alphas), an optional data set is required, and can be optionally downloaded in addition:
  - **`G4TENDL-1.4`**

Configuration & Externals

- Cmake:
  - New script to automatically query/check module interfaces for dependencies
  - Added support for modular builds
    - Only full granularity for the moment
  - Added “Sanitizer” build mode for memory/threads debugging
  - Added required configuration to be able to generate DEB/RPM packages with Cpack
  - Removed deprecated GNUmake build system for kernel libraries
  - Removed Geant4Py from toolkit build with distribution and support moved to upstream repository
  - CMake 3.16 is the minimum version required to build Geant4
- CLHEP - Version 2.4.6.2/3
  - Fixed inconsistencies in Evaluator internal set of units
  - Fixed compilation warnings on new supported compilers

Extended examples

- New set of extended hadronic examples, implementing different setups showing how to score **particle fluences**
- **analysis/AnaEx03** - New example , showing usage of new analysis commands for file management, writing histograms and n-tuples in a file multiple times
- **medical/dna/jetcounter** - new example for simulation of a typical experiment with the Jet Counter nano-dosemeter
- **medical/dna/moleculardna** - new example for the simulation of physics, physico-chemistry and chemistry processes in DNA geometries
- The example **field/field01** now demonstrates how to use the new G4TransporationParameters class; showing also a full set of ways to control the parameters for killing looping stable charged particles

Advanced examples

- **eFLASH_radiotherapy**
  - New application simulating the beamline and energy spectra based on the Triode Electron Gun Equipped ElectronFlash Manufactured by Sordina Iort Technologies S.p.A. available at the Centro Pisano Flash Radiotherapy (CPFR) in Pisa
- **ICRP145_HumanPhantoms**
  - New advanced example implementing a ICRP145 human phantom

Platforms for 11.1

- Linux CentOS8/Stream
  - gcc-8.3.1 to 12.2, 64 bits (Intel or AMD)
- macOS 13 Ventura
  - Apple Clang-14 (XCode 14.x), 64 bits (Intel or Apple Silicon)
- Windows 10
  - Visual C++ 14.33 (Visual Studio 2022)

❖ Also tested (sequential/MT):

  - Linux CentOS7, icc-2021, icx-2022, clang-10/11/13/14
  - Linux Ubuntu 22, gcc-11.3
  - macOS 12 Monterey, Apple Clang-14
  - macOS 11 Big Sur, Apple Clang-12

Thanks!