Common Geometry Primitives library

WP3 – 15/12/2016

G.Cosmo, M.Gheata (CERN EP/SFT)

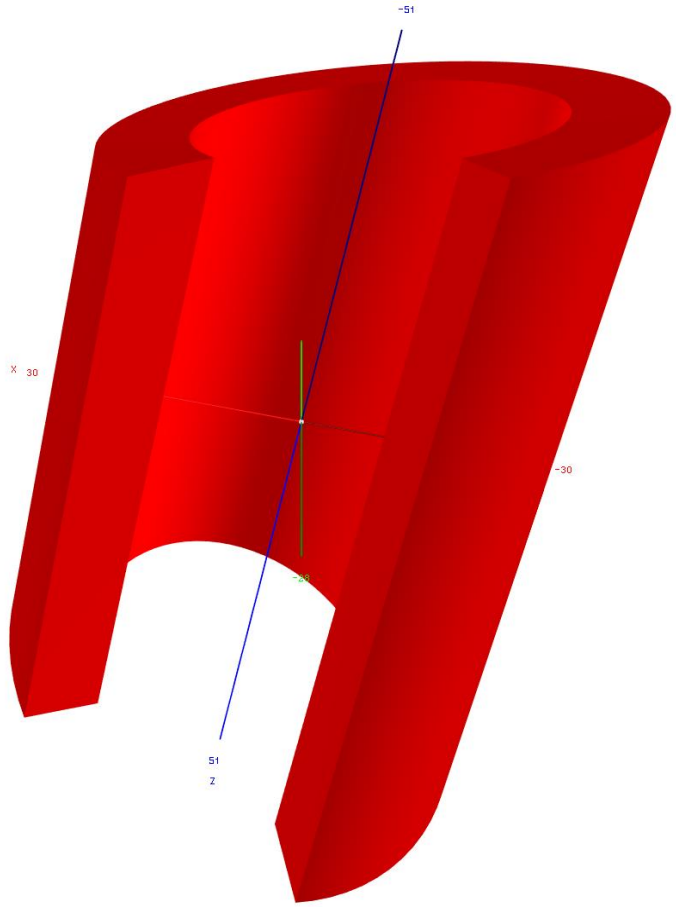
VecGeom 00.03.00

- New VecGeom release meant to be used with latest release of Geant4 10.3 deployed last week
- Primitives included:
  - Box, Orb, Trapezoid (Trap), Simple Trapezoid (Trd), Sphere (+ sphere section), Tube (+ cylindrical section) , Cone (+ conical section), Generic Trapezoid (Arb8), Polycone, Polyhedron, Paraboloid, Parallelepiped (Para), Hyperboloid, Ellipsoid, Torus (+ torus section), Simple Extruded Solid, Cut Tube, Scaled Solid, Boolean (addition, subtraction, intersection)
- Still to convert from USolids:
  - Tetrahedron (Tet) (*), Multi-Union, Tessellated Solid, Generic Polycone, Extruded solid
- Not yet implemented:
  - Elliptical Cone(*), Elliptical Tube(*), Half-planes, *Twisted shapes (box, trap, tube)*

(*) Can be implemented as either scaled shape or as specialization of an existing shape

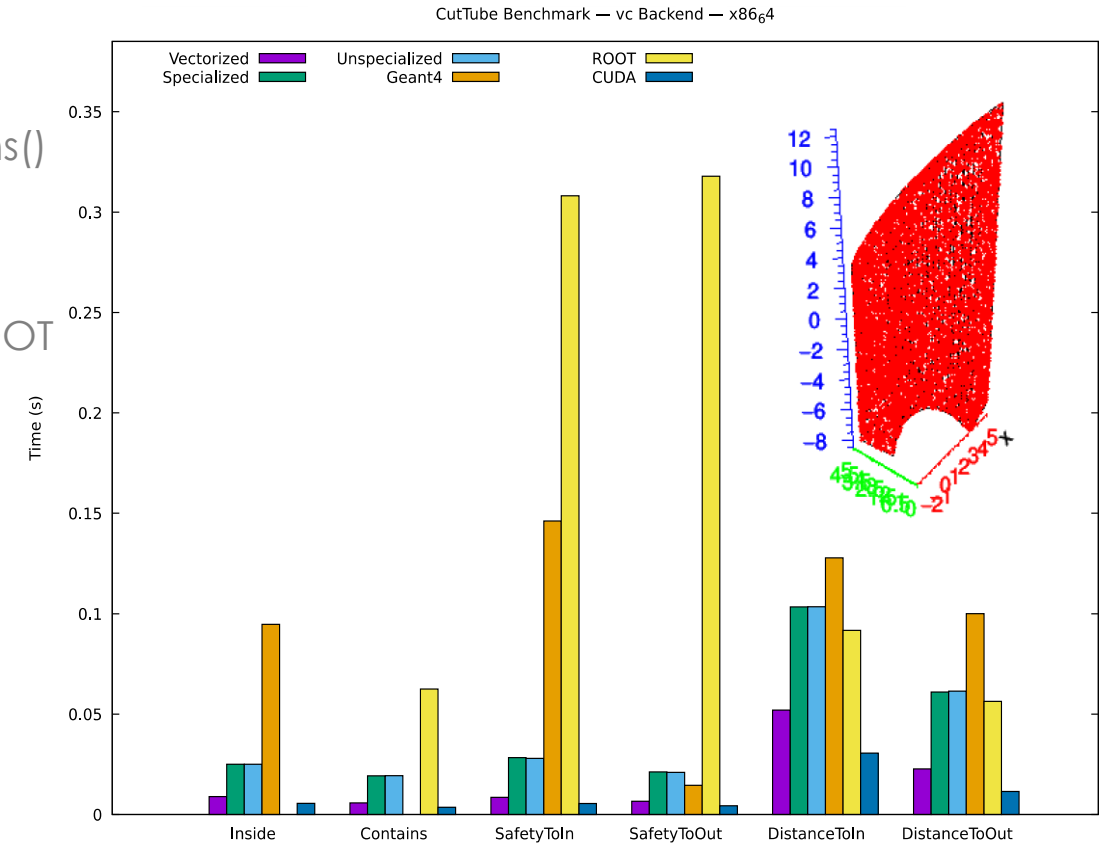
New added primitive: cut tube

- A tube cut by two planes
  - Planes cutting Z axis at +/- Z
  - Defined by normal vectors oriented outwards the solid
- Available both in ROOT and Geant4 geometry packages
- Adopted in ALICE and new CMS tracker geometry
- Implemented using as primitives the tube and plane implementations
  - Plane implementation to be reused for description of half-space primitive



Cut tube performance

- Overall scalar performance comparable to ROOT/Geant4
  - Much better for Inside(), Contains() and SafetyToIn()
- Capacity and surface area computed analytically
  - Just estimates in Geant4 and ROOT
- SIMD and CUDA support
  - Implemented using VecCore interfaces
  - Vectorization gain for AVX in the range 2x-3x
- Visualizer, shape-tester and benchmarker provided for 4 topologies:
  - with/without Rmin
  - with/without phi cut



Ongoing activity…

- Code robustness & correctness
  - Reviewing/fixing issues on existing shapes detected by the ***ShapeTester*** testing suite
  - Shapes particularly under exam: Torus, Cone, Polycone
  - Extending coverage to different possible topologies
- Completing refactoring of code to VecCore
  - Currently addressing Polyhedra
  - Preparing for deploying VecCore as independent library
- Verification of correctness in complex geometry setups
  - Analyzing issues from solids/compositions in the CMS setup
- Extending testing coverage by adding new shape topologies

Resources

- Assuming current resources sum up to ~1.5 FTE, adding up contributions from PH/SFT:
  - John Apostolakis
  - Gabriele Cosmo
  - Andrei Gheata
  - Mihaela Gheata (AIDA PJAS)
  - Evgueni Tcherniaev
  - Sandro Wenzel