CLHEP
Usage status & scenarios

Gabriele Cosmo

Geant4 Collaboration Workshop 2006 - Lisbon

Outline

- Status in Geant4
- Status in experiments and other client software
- Concerns & critics
- Future evolutions and scenarios
  - *If and when end of support for CLHEP will become effective*

Usage status in Geant4

- CLHEP 1.9.2.3 and 2.0.2.3 supported
  - Since release 8.0; no migration required
  - Support for 1.9/2.0 branches assured in future as long as compatibility assured by CLHEP
- Limited set of CLHEP classes used in Geant4
  - Physics vector (3-vectors, 4-vectors, simple rotations)
  - Geometrical vectors and transformations (3D-vectors, 3D-points, etc...)
  - Random numbers with static interface
  - System of units and constants
  - Other minor uses of CLHEP are restricted to examples

Usage status in experiments and other software clients

- CLHEP more or less heavily used in the software of 3 LHC experiments
  - ATLAS, CMS, LHCb
- CLHEP heavily used in BaBar
- CLHEP heavily used in the MC generators community
  - new generation of C++ MC generators (Herwig++, Pythia-8, etc…)
- CLHEP used also in other applications or clients of Geant4

Concerns in Geant4 …

- Difficulties dictated by CLHEP installation
  - Complicated configuration/build system introduced since version 1.9.0
  - Limited (or not-existing) support for some platforms compilers regularly supported by Geant4
    - Windows/VC++, Linux/Intel-icc, MacOS/gcc, 64bits systems, ...
  - Consistent amount of support requests from users coming from failures or incorrect setup in installing CLHEP
- Slow turn-around for development and introduction of new features (mainly additions of new units/constants)
- Difficulties in the synchronisation of releases CLHEP/Geant4
  - Consequent overload to system-testing and release teams to guarantee support to new CLHEP versions

More concerns …

- Complicated/confusing interface for vectors/randoms
  - Since the introduction of new features and merging with FNAL ZOOM classes
- Interest expressed by LCG (LHC Computing for the Grid) representatives of the LHC experiments to migrate in future to MathCore, the new mathematical library under development within LCG
  - currently implementing a part of the functionality provided in CLHEP: physics vector (3-vectors, 4-vectors) matrices, transformations, geometrical vectors (3D-vectors, etc...).
  - In future also a revised random-numbers module foreseen to be included

Future evolutions & scenarios for Geant4 - (a)

- Integration of CLHEP classes (only used classes) in the Geant4 distribution
  - involving extension of the current Geant4 build-system to allow OPTIONAL dependency on external official CLHEP library
- ***Pros***:
  - Geant4 standalone system, no external dependencies, build Geant4 in one go
  - Assured support for platforms/compilers verified for the Geant4 software
  - Better control of classes used by Geant4 software and faster turnaround for development if changes/additions required
  - Simple to integrate in the Geant4 configuration system.
- ***Cons***:
  - Require synchronisation with official CLHEP releases
    - Classes/files integrated must be exactly the same and refer to a -precise- version of CLHEP
  - Extra verification for system-testing
    - required to verify embedded classes PLUS external usage of CLHEP

Future evolutions & scenarios for Geant4 - (b)

- Redistribution of CLHEP (specific version) and packaging in Geant4
  - Still allowing OPTIONAL dependency on external official CLHEP library
- ***Pros***:
  - Geant4 standalone system, limited external dependency, build Geant4 in one go
- ***Cons***:
  - Same as for scenario (a)
  - May limit support for platforms/compilers verified for the Geant4 software
    - shared-libs, DLLs may not work anymore, ...
  - May cause extra complications for the Configuration/Installation of Geant4

Future evolutions & scenarios for Geant4 - (c)

- Integration of CLHEP classes (only used classes) as Geant4 classes
  - With simplification of interfaces and internal maintenance
- ***Pros***:
  - Same as for scenario (a)
- ***Cons***:
  - Extra maintenance required of the new imported classes
  - Require revision of interfaces and implementation of converters wherever necessary
  - Will change any persistent schema, affecting users

Future evolutions & scenarios for Geant4 - (d)

- Evaluate new foundation packages (e.g. MathCore)
  - Once ready and if providing all the necessary ingredients required by Geant4
- ***Pros***:
  - New, more modern, eventually lighter and more performant (?) implementation, simpler interface
  - Synchronised (?) with future use done in the LHC experiments
  - Easier and more efficient (?) installation
  - Better support for platforms/compilers
- ***Cons***:
  - Complete migration and revalidation of the Geant4 software in all its internal aspects (basic transformations, physics), exposed interfaces and performance
  - Geant4 still dependent on external software
  - May clash with different choice made by the MC generators community
  - Will clash with other non-LHC clients of Geant4

Future evolutions and scenarios for Geant4 - (e)(f)

- Same as [b] and [c] but for MathCore classes

Up to us the choice !