Geant4 MT Migration and Examples Issues

I. Hrivnacova, IPN Orsay

17th Geant4 Collaboration Meeting,
10 - 14 September 2012, Chartres

Introduction

- As the examples are clients of all Geant4 categories they will be affected by all changes which have an impact on a user application
- As not all changes have been already agreed and decided it is not easy to evaluate what will be needed for the examples at the end
- In this talk I will
  - First summarize my (quite short) experience with Geant4 MT
  - Then discuss the impact of the items for migration of the user code to Geant4 X as they are actually explained in Geant4 MT prototype documentation
  - And finally what may be the issues in the basic examples

Experience With MT proto 9.4.p01

- Example ParN02Root
  - In Geant4 MT branch: adds Root IO to ParN02
- Geant4 VMC MT prototype
  - Most of works: singleton objects -> singletons per thread
  - Does not redefine the event loop (use G4RunManager) – can reuse G4ParRunManager for the event loop
    - But modifications in G4ParRunManager/G4ParTop.icc were needed in order to get executed all user functions (including initialization of Root threading) at the right time
  - Based on Geant4 MT branch, available in ROOT SVN (in branch)
  - More in CHEP2012 poster
  - The first proto-release of Geant4 VMC MT is planned for Geant4 MT based on 9.5.p01

Experience
With MT proto 9.5.p01

- The Geant4 libraries build ok on my laptop (FC14, gcc 4.5.1)
- All (three) examples build & run ok (on 2 cores)
- Failure to build with OpenGL ON
  - G4VisManager.hh:99:26: fatal error: G4GMocrenIO.hh: No such file or directory
- As a fast exercise B4c example (with scoring via SDs) migrated to MT
  - No visualization
  - The analysis code produces 2 Root files (1 per core) with histograms; after excluding ntuple; but fails when ntuple is included

Items for Migration of User Code

And their impact to examples

Single Threaded Applications

- No (or very minimal) changes should be required in a single threaded user application to be built against Geant4 X libraries
- This is almost true for geant4_mt_proto.9.5.p01
  - The basic examples from Geant4 9.5.p01 build (almost) ok
  - There has to be added a dummy function void my_slave_thread(void*) {} to get the examples build

Multi Threaded Applications

- Current MT prototype requires:
  - Reorganizing detector construction class (adding three function)
  - Update of hit classes (change G4Allocator to thread local)
  - Modify main() to initialize threading
- More may be needed in more complex applications
  - Which defines eg. their own singletons, other objects using G4Allocator (eg. user track information) etc.
- 4 basic + ~80 extended + 21 advanced + 7 novice -> we will have to apply each item for migration 112 times
- If this list of the items could be (still) reduced it will make end users life easier

Migration of Examples

- If the list of items for migration (to MT application) will stay as it is now, migrating all examples (to MT applications) is not realistic
  - It should not be a problem as far as the examples can be built against Geant4 X as single threaded applications
- Some examples (which use not thread-safe external libraries) will be not portable to MT at all
- Migration of all basic examples looks feasible from the point of view of the amount of work
- geant4_mt_proto.9.5.p01 includes only 3 examples: ParA01, ParN02, ParN03 all with scoring via sensitive detectors
- In basic examples: also *scoring directly in user actions classes or own data object*
- In some basic examples: *the summary statistics in the end of run* will require to recuperate the data per event sets accumulated at each thread
  - A recipe how to proceed in these cases will be needed

Documentation

- The User's Guide in the Geant4-MT prototype provides a recipe how to migrate a user application to MT
- What is not covered
  - Explanation how the Geant4 MT works, mainly which user objects are build on threads
  - The event loop in MT application, the role of G4MTTop.icc and G4MTParRunManager
- But is necessary for users who are novice in MT programming