

The G4 Computing Performance Task



Back in February 2010, J. Apostolakis asked me to organize a G4CPT within the G4 Collaboration

Not a "Task Force" but an open ended activity organized as a group with regular meetings every ~ 6-8 weeks

Charge

- (a) Profiling to identify bottlenecks in Geant4 based on main stream applications. We need to discuss profiling tools, what we want to measure, metrics. EM, Geometry and hadronics are the areas more involved in CPU usage.
- (b) Code reviews geared towards improving computing performance and coding practices.
- (c) Establish computing performance activities with the medical and space G4 communities.
- (d) Identify issues in multi-core-multithread G4. ←

Not discussed beyond the first meeting - See Tuesday plenary

Ideas discussed during the first meeting



- Define a set of profiling tests/metrics.
- Profile simple examples and big experiments regularly, i.e. benchmark every candidate release/patch.
- Store information in a database for consistency and to track history
- Important to have a profiling tool with few dependences that can also be used outside of HEP.
- Identify unique features, advantages/disadvantages of each profiling tool to define profiling strategy and guide new users

List of Top Problems to Investigate



1. Memory Allocation

- * Navigation (G. Cosmo working on a fix - ATLAS, CMS testing)
- * Bertini (Mike Kelsey working on mem/speed improvements)

2. EM Physics Package

- * Optimization of parameters in applications
- * Revisit physics algorithms in Geant4 code: optimizations, approximations
- * Multiple scattering
- * Code review

3. Navigation speed and memory use in Voxel geometries and when handling large numbers of materials (brought up by the medical community)

4. Ion-ion inelastic models speed and memory use (medical).

5. Propagation in Magnetic Fields

- * Code review (done - no low hanging fruit from the programming side)
- * Testing/validation/profiling new steppers (ATLAS is testing Nystrom)

6. Hadronic cross-sections

- * Code review

7. Precompound/de-excitation

- * Code optimization. Many log/power functions are called. Many classes.

Challenge: define, staff, initiate remaining projects

Summary and Outlook



- Significant progress **testing and automating different profilers** on simple and complex *G4* applications
- Slow progress on reviewing and fixing code based on what is learned.
- List of top problems not fully covered - man power issues.

What's next?

- Finish automation of profiling tools for use in *G4* as well as documentation on usage
- Define a set of profiling/performance tests to be run frequently on some subset of ref/candidate/public releases - staff the effort.
- Cover all elements in list of top problems to investigate and those that arise from the systematic and frequent profiling of *G4* applications
- Do we want a hypernews list or similar to post/follow progress?