

Impact of different compilers and build types on Geant4 simulation execution time

Thursday, 3 October 2019 10:45 (3 minutes)

Experimental observations and advanced computer simulations in High Energy Physics (HEP) paved way for the recent discoveries at the Large Hadron Collider (LHC) at CERN. Currently, Monte Carlo simulations account for a very significant amount of computational resources of the Worldwide LHC Computing Grid (WLCG).

In looking at the recent trends in modern computer architectures we see a significant deficit in expected growth in performance. Coupled with the increasing compute demand for High Luminosity (HL-LHC) run, it becomes vital to address this shortfall with more efficient simulation.

The simulation software for particle tracking algorithms of the LHC experiments predominantly relies on the Geant4 simulation toolkit. The Geant4 framework can be built either as a dynamic or a static library, the former being the more widely used approach. This study focuses on evaluating the impact of both having libraries statically vs dynamically linked and compiler optimization levels, on the simulation software's execution time.

Multiple versions of the more widely used versions of compilers for UNIX-like systems have been used for these investigations. Both compiler optimization levels (e.g. O3, O2 on GCC) and link-time optimization (LTO) have been studied. Initial results indicate that significant execution time reductions can be obtained by switching from static to dynamic linking.

Primary author: ELEN, Einar Alfred (Lund University (SE))

Presenter: ELEN, Einar Alfred (Lund University (SE))

Session Classification: Thu morning session 2