Testable physics by design

C. Choi, M. C. Han, S. Hauf, G. Hoff, C. H. Kim, H. S. Kim, S. H. Kim, M. Kuster, M. G. Pia, P. Saracco, G. Weidenspointner

The validation of physics calculations requires the capability to thoroughly test them. The difficulty of exposing parts of the software to adequate testing can be the source of incorrect physics functionality, which in turn may generate hard to identify systematic effects in physics observables produced by the experiments.

Starting from real-life examples encountered in the course of an extensive validation effort of Geant4 physics, we show how software design choices may affect the ability to test basic aspects of physics functionality and to monitor their evolution in the course of software lifecycle. We document cases where inaccurate physics behaviour was hidden by software design that prevented its testability, and illustrate how improved physics functionality could be achieved by improving the transparency of software design.

Exploiting the experience with Geant4 as a playground for investigation, we discuss methods to enhance the testability of existing software systems by means of refactoring techniques, the identification of inflection points and their coverage. We discuss how these techniques differ from, but can be combined with, the traditional practice of monitoring some physics observables with regression testing.

We also present guidelines to introduce testability into the software design since the early phases of the software development, and to preserve it in the course of the product lifecycle. This issue is especially relevant in the context of ongoing R&D on future simulation systems.