Quality assurance of the LHCb Simulation

Dmitry Popov [1] on behalf of the LHCb Simulation Project

[1]University of Chinese Academy of Sciences

Abstract

Monte Carlo simulation is a vital tool for all physics programs of particle physics experiments. Their accuracy and reliability in reproducing detector response is of the utmost importance. For the LHCb experiment a full suite of validations has been put in place for its simulation software to ensure the quality of the data samples produced.

This poster describes various testing procedures that have been put in place, from simple checks that are performed as soon as new software is submitted for integration in the code repository to more in-depth performance regression [1] checks handled by the dedicated LHCbPR [2] infrastructure and tools for simulation data quality shifters to check for anomalies and alert developers in the case of unexpected behaviour.

LHCb Simulation software stack

- **Gauss** [3] is the simulation software framework for LHCb, which binds together all necessary pieces from Monte Carlo (MC) event generators to particle propagation and interaction mechanisms with the detector.
- **Gauss-on-Gaussino** will replace Gauss, based on **Gaussino** [4] project.
- **Boole** [5] models detector response, digitising MC hits into RAW data, propagating MC truth information.

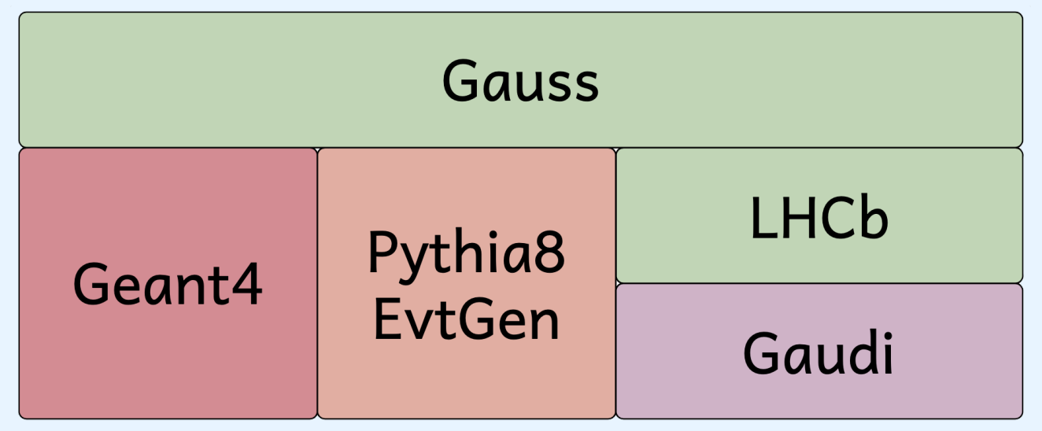

Figure 1. Gauss structure in Run1 and Run2 data taking periods.

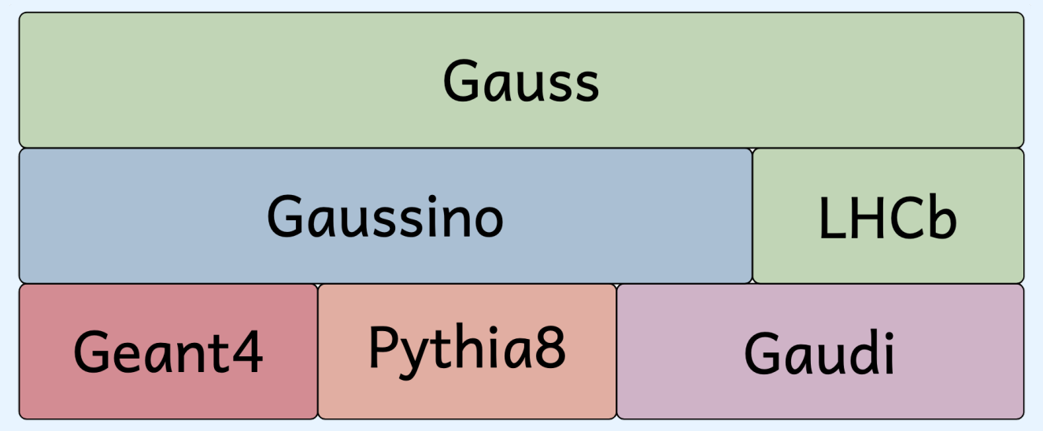

Figure 2. Gauss-on-Gaussino structure in Run3 data taking.

Development life cycle for simulation software

LHCb simulation software repository in on GitLab, managed by CERN IT. Hosts all software development, maximising available tools and ecosystem. During the active development phase of major new application versions, multiple nightly builds [6] are produced to integrate and verify changes to the code across all involved software projects.

- **Variations of simulation software stacks** are configured in a specialised project to define required dependencies and their versions, patches to the code which must be applied, etc.
- **Nightly builds** are handled by Jenkins build system through various nightly slots set up to validate different aspects of the built software stacks, e.g. behaviour of new versions of MC generators and their tuning, Geant4, compilers, platforms, etc.

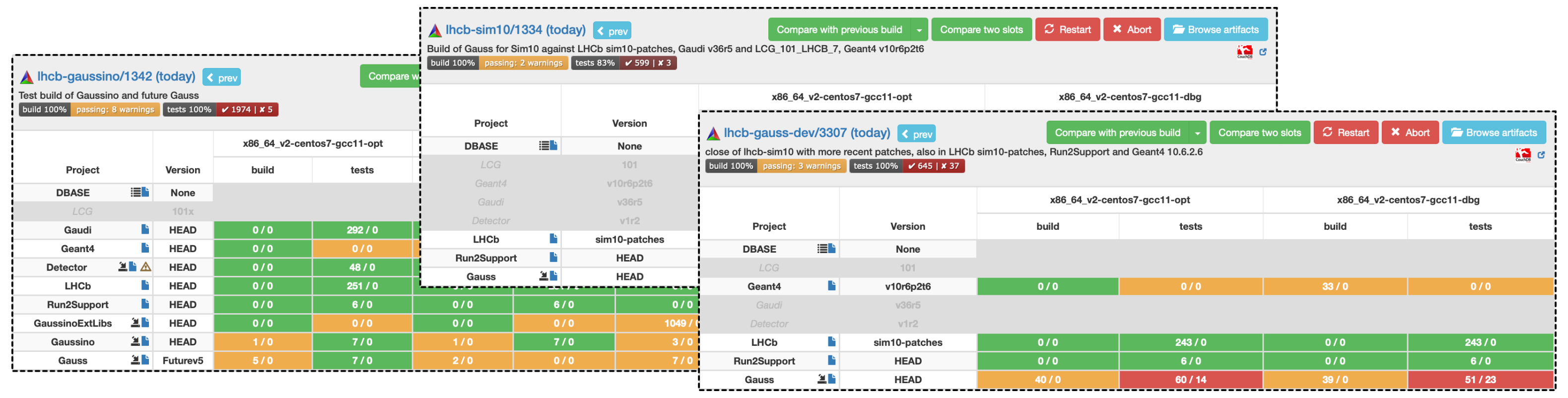

Figure 3. Example of nightly builds slots for LHCb simulation software stacks.

Continuous Integration and Nightly Tests

Changes to the code base in all related projects of the simulation software stack are automatically checked during submission and nightly builds are verified for integrity.

- **CI tests** verify submitted changes to the code, ranging from formatting compliance and syntax errors validation to static code analysis, accepted changes are applied in nightly builds.
- **Nightly tests** are designed to be simple, fast to run and ideally check just a single property. Their main goal is to verify that the built software works: the application starts, runs and finalises successfully, correct libraries are picked up, input data located, underlying frameworks and toolkits are exactly of the versions which developers intended to utilise.

Detailed analysis with PR tests

Performance Regression (PR) tests handled by the LHCbPR system – comprehensive validation of technical aspects and physics results. Includes full physics analyses to validate distributions over a wide spectrum of physics observables. PR tests are flexibly configured to cover a wide variety of simulation cases: MC generators, LHC beam conditions, detector geometry, specific decay channels, etc. They may run for hours and produce larger data samples suitable for physics analysis.

GEANT4 PR tests	Gauss PR test
Hadronic cross-section	Gauss simulation validation
Sampling calorimeter	Radiation length and absorption map
Multiple scattering in thin layer	Muon multiple scattering
Simplified RICH simulation	Detailed timing in detector volumes
Gamma to di-lepton conversion	CPU and memory consumption

Table 1. Example of some of deployed PR tests for LHCb simulation.

Results of PR tests are processed and stored by LHCbPR and are available in its front-end web application for further analysis.

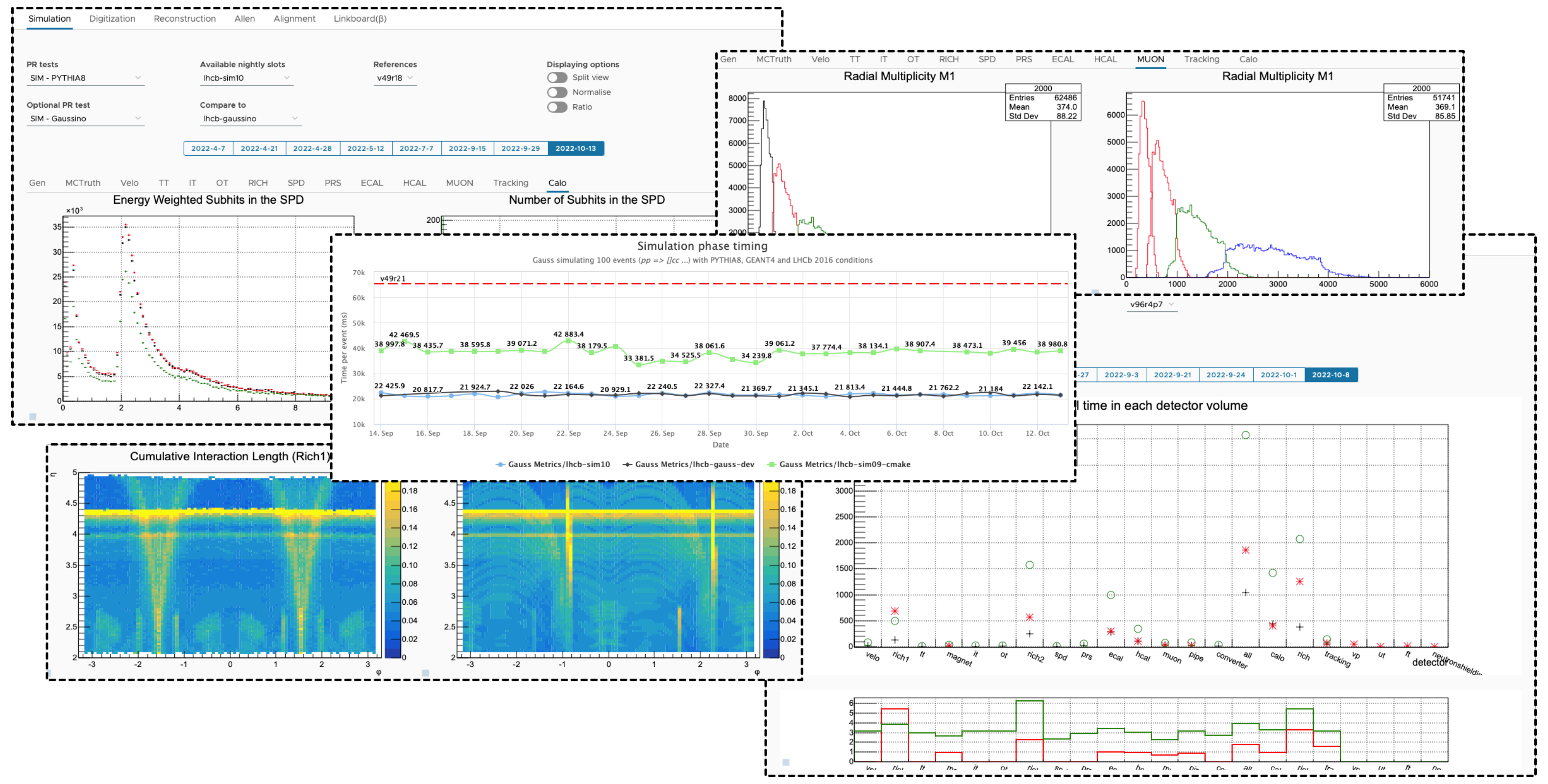

Figure 4. Examples of visualisation in LHCbPR web application.

Simulation data quality

LHCb introduced a new simulation data quality shifter role to help maintain high quality simulation software and simulated data samples. There are two main tasks defined for this role.

Early validation of technical changes

Monitor the evolution of the developed simulation software in time and spot unexpected behaviour. Results of the nightly and PR tests are used as input. The particular observables may range from CPU timings and memory consumption to significant discrepancies in specific distributions cross-checked with corresponding references. Weekly summaries are reported at simulation development meetings.

Approval of large scale central MC productions

Some nontrivial problems can be spotted only in large data samples. Finding problems late in productions leads to massive waste of CPU resources, where samples simulate dozens of different decay modes, or hundreds of millions of events. LHCb has implemented an automated safety check system to identify such flaws and avoid scenarios caused by simple misconfiguration or human error. Prior to proceeding with the total number of requested events, a small-scale ($\mathcal{O}(100k)$) produced events production is launched. The shifter will check histograms made during the production in a web-based monitoring system Monet [7] comparing them to the references and either pause the production for further investigation or approve and let it run the full course.

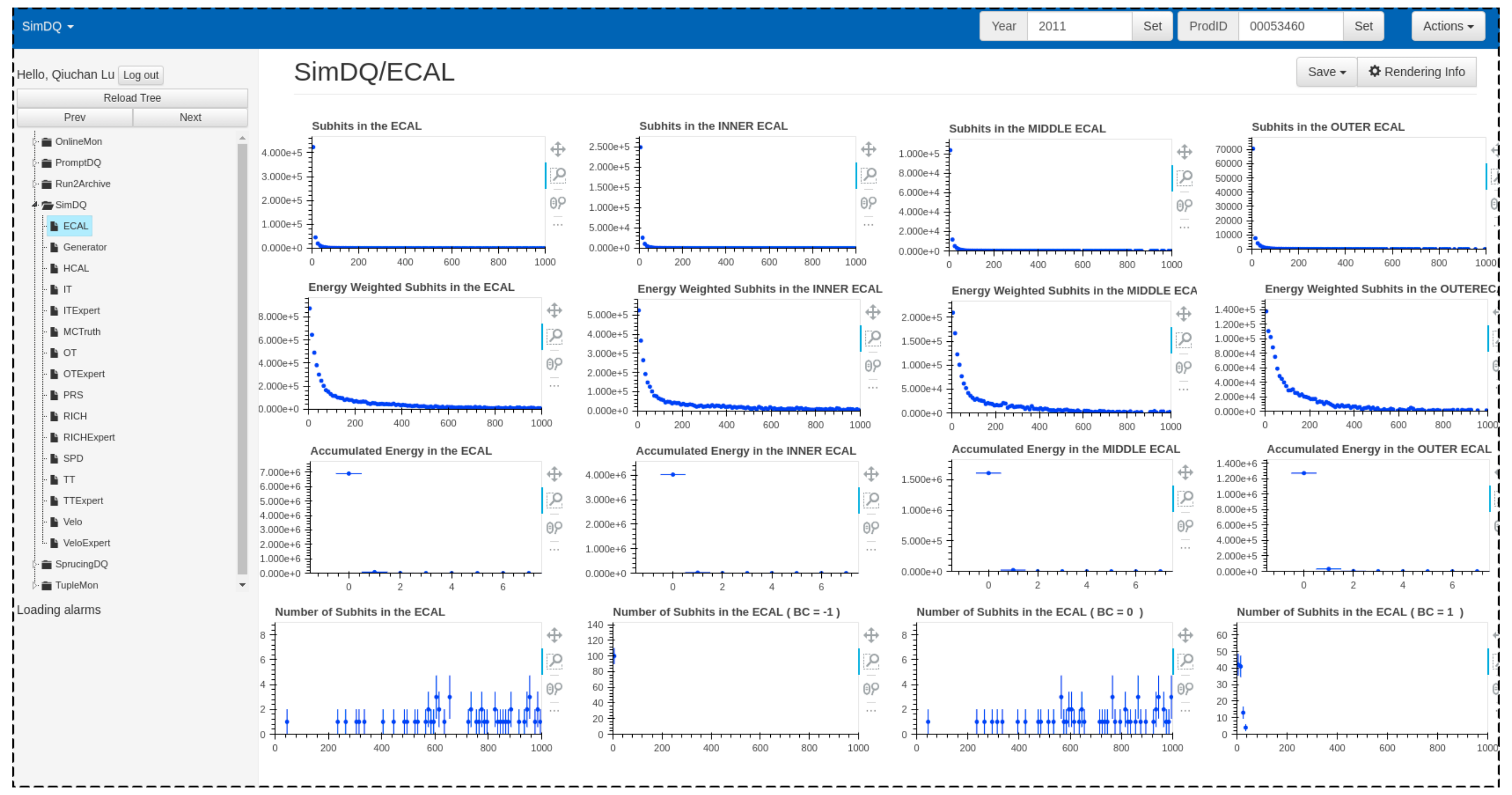

Figure 5. Example of Monet interface.

References

[1] M Clemencic and B Couturier. LHCb Build and Deployment Infrastructure for run 2. *J. Phys.: Conf. Ser.*, 664(6):062008. 6 p, 2015.

[2] Dmitry Popov. Testing and verification of the LHCb Simulation. *EPJ Web Conf.*, 214:02043. 4 p, 2019.

[3] M Clemencic et al. The LHCb simulation application, Gauss: Design, evolution and experience. *J. Phys. Conf. Ser.*, 331:032023, 2011.

[4] Benedetto Gianluca Siddi and Dominik Muller. Gaussino - a Gaudi-Based Core Simulation Framework. *2019 IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, pages 1–4. 4, 2019.

[5] Gloria Corti, Guy Barrand, Ivan Belyaev, Marco Cattaneo, Philippe Charpentier, Markus Frank, Patrick Koppenburg, P Mato-Vila, Florence Ranjard, and Stefan Roiser. Software for the LHCb experiment. *IEEE Trans. Nucl. Sci.*, 53:1323–1328, 2006.

[6] Marco Clemencic and B Couturier. A New Nightly Build System for LHCb. *J. Phys.: Conf. Ser.*, 513:052007. 5 p, 2014.

[7] M Adinolfi, F Archilli, W Baldini, A Baranov, D Derkach, A Panin, A Pearce, and A Ustyuzhanin. LHCb data quality monitoring. *J. Phys. : Conf. Ser.*, 898(9):092027. 5 p, 2017.

LHCb Experiment

21st International Workshop on Advanced Computing and Analysis Techniques in Physics Research, 23–28 Oct 2022, Bari, Italy

Dmitry.Popov@cern.ch