QEDjl Event Generator Hackathon

Report of Contributions

Contribution ID: **1** Type: **not specified**

Welcome & Setup

Monday 20 October 2025 09:00 (30 minutes)

Open the day, align expectations, short overview of the agenda, ensure everyone can run the code locally, and ensure the local rest environment works.

Presenter: Dr HERNANDEZ ACOSTA, Uwe (Center for Advanced Systems Understanding)

Contribution ID: **3**   Type: **not specified**

Code review - Part 1: General & Core

Monday 20 October 2025 09:30 (1 hour)

Introduction to event generation stages and general requirements. Discussion of the current core design choices and coding patterns.

Presenter: Dr HERNANDEZ ACOSTA, Uwe (Center for Advanced Systems Understanding)

Contribution ID: 4   Type: **not specified**

Code review - Part 2: Examples & Infrastructure

Monday 20 October 2025 11:00 (1 hour)

Introduction of the current examples, test environment, and benchmark suite.

Presenter: Dr HERNANDEZ ACOSTA, Uwe (Center for Advanced Systems Understanding)

Contribution ID: **5**

Type: **not specified**

Discussion: Code & API

Monday 20 October 2025 13:00 (45 minutes)

Discussion about the general event generator pipeline. Agreement on the public API and the package layout. Definition of naming conventions and testing/benchmark pipelines.

Presenter: Dr HERNANDEZ ACOSTA, Uwe (Center for Advanced Systems Understanding)

Contribution ID: **6**   Type: **not specified**

Planning Session: Refactoring

Monday 20 October 2025 13:45 (45 minutes)

Fill the milestone' Event Generator Hackathon 2025' with issues and todos and assign them.

Identify integration breaks in QEDjl packages and write issues in the respective repo, label them with an overall label migration: RejectionSamplers.jl

Presenter: Dr HERNANDEZ ACOSTA, Uwe (Center for Advanced Systems Understanding)

Contribution ID: **7**   Type: **not specified**

Refactoring Sprint 1

Monday 20 October 2025 14:45 (2 hours)

Remove dead code and obsolete QEDjl patches. Perform overall cleanup. Remove add_dev.jl and adapt to dev branches of all QEDjl packages.
Migrate unit test CI by using the IntegrationTests.jl framework and a Gitlab mirror.