Project Proposal:

Improving testing procedures for prompt data processing at CMS

Mycola Kolomiiets

June 27, 2023

1 Project Summary

1.1 Problem statement

The CMS Tier0 is responsible for the creation and permanent storage of CMS RAW data, as well as the first reconstruction which is used for physics studies and publications. It is based on a workload management system to deal with the large flow of data and depends on many services ranging from data reception and storage, as well as many intermediate steps that steer and configure the output of the data.

Users of the collaboration often request to deploy changes in the Tier0 configuration. Due to the complexity of the code involved, such changes could potentially break the system, preventing Tier0 from further processing and leading to eventual data loss. Therefore, integrated tests are exploited for testing such configuration changes, prior to the deployment in production. Currently, Tier0 uses simple syntax testing and automated triggering of replays. These replays are full scale tests based on dedicated representative data, that use a substantial part of the production infrastructure resources.

In order to facilitate more automated testing, a finer granularity of tests is desired that can run locally or on a small subset of the production environment. Such unit tests are currently not implemented and would greatly improve the testing capabilities of Tier0. Moreover, it will reduce the resources needed and lead to a faster discovery of potential errors or bugs.

1.2 Solution

To address the shortcomings, it is crucial to establish additional automated tests within the Tier0 system. This has to be integrated within the Tier0 code and made flexible enough for further improvements and the addition of unit tests. By implementing unit tests and smaller functional tests, developers can quickly assess the state of their code during the development phase. These tests will provide faster feedback, allowing developers to identify and address issues early on, resulting in more frequent contributions and ultimately improving the development progress.

1.3 Goals

Main goals of the project are:

- set up automated small-scale functional tests for the T0 repository, which are comparatively smaller in scope compared to the existing "replay" test.

- establish automated unit tests for the Tier0 repository

- ensure that the tests implemented in the Tier0 repository are well-documented.

Additionally, an possible task of the project is to modularize sections of the T0 code. This involves breaking down the codebase into smaller, independent modules or components, each responsible for a specific functionality or task. By modularizing the code, it becomes easier to maintain and test.

2   Project plan

Week	Tasks
1-2	- Set up the development environment, including necessary tools, libraries, and dependencies. - Familiarize yourself with the Tier0 repository and the specific code segments where most contributions are made. - Gain an understanding of the existing testing infrastructure and code structure.
3-10	- Identify key areas in the Tier0 repository that require testing and modularization. - Create automated small-scale functional tests for code segments with frequent contributions. - Develop automated unit tests. - Continuously integrate and execute the developed tests to ensure they work as expected. - Write documentation for tests while working on the implementation. - Possible: modularize sections of the Tier0 code to improve maintainability and testability.
11	- Document the developed tests, including instructions for execution and expected results. - Create guidelines and best practices for future test development and maintenance.
12	- Prepare a report summarizing the work done during the project. - Present any challenges faced, lessons learned, and recommendations for future improvements.