

Test-driven software upgrade of the LHC beam-based feedback systems

Diogo Alves, Kajetan Fuchsberger, Stephen Jackson and Jorg Wenninger

Abstract—The beam-based feedback system is essential for the operation of the LHC. It comprises two C++ servers: a FESA-based (framework for real-time systems developed at CERN) acquisition and configuration proxy, and a non FESA-based controller which sanitises the acquisition data and feeds it to multiple real-time feedback algorithms (orbit control, radial-loop control and tune control) ensuring a stable orbit of the LHC’s beams. Responsibility for the further development and maintenance of the servers was recently transferred to a new team, who have made considerable efforts to document the existing system as well as improve its operational reliability, performance, maintainability and compliance with CERN’s software and operational standards. Software changes are accompanied by rigorous unit-testing with future releases tested outside the operational environment, thus minimizing the potential for beam downtime. This approach has proven very effective during re-commissioning for LHC’s run 2, where the systems underwent significant changes.

In a bid to homogenize operational procedures for configuring LHC systems, a demand to improve the real-time configuration of the system’s feedback references and optics was identified. To replace the existing ad-hoc method of real-time configuration, a new waveform-based server, pre-configured with sequences of N-dimensional values versus time, autonomously ensures that the system is re-configured at precisely the correct time. This paper describes the design choices, software architecture, integration and preliminary testing of the new waveform-based server. In particular, considerable effort was put into reducing the impact of changing already established and tested behaviour.

I. INTRODUCTION

Active feedback control is essential for the operation of the LHC. The beam-based feedback system [1] is responsible for controlling, simultaneously, the beam orbit and the machine tune [2]. The former controls the average trajectory of the beam particles with respect to a reference orbit whereas the latter adjusts the machine tune¹, a crucial quantity in order to control particle losses. Quadrupole magnet misalignments are the major source of disturbances for the orbit of the particles in the LHC. This effect depends on factors such as ground motion, earth tides and optics changes. It comprises a mix of reproducible and random effects. Tune changing transients occur during dynamic phases such as injection and

whenever the energy and/or the optics change. In the LHC, the phase during which the energy of the beam is increased from the 450GeV to 6.5TeV is particularly delicate as tune transients may lead to significant particle losses.

In order to be able to control these important quantities, the beam-based feedback system in the LHC gathers position measurements from over 2’000 Beam Position Monitor (BPM) sensors [3] and 12 tune measurement sensors [4]. The more localised control of the beam orbit is performed by individually adjusting the current in over 500 dipole magnets² whereas the average radial position of the beam (radial-loop) is controlled by changing the frequency of the RF cavities accelerating the beam. The real-time adjustment of the machine tune is accomplished by controlling the current in over 250 quadrupole magnets. The central role played by the beam-based feedback system in the LHC is depicted in Fig. 1.

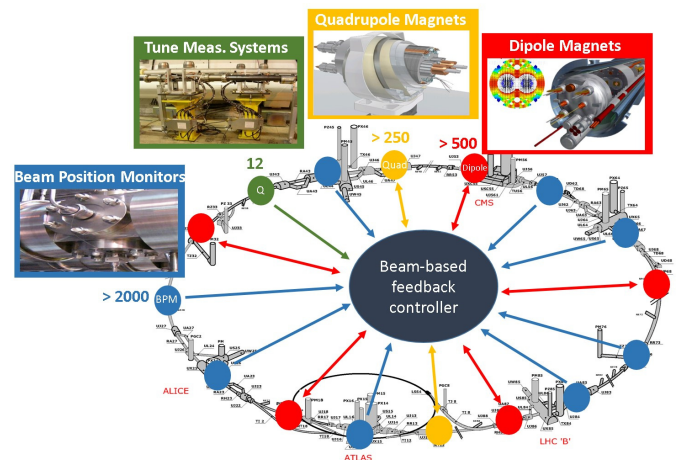


Fig. 1. LHC beam-based feedbacks overview

In a bid to homogenize operational procedures for configuring LHC systems, a need to change and improve the real-time configuration of the system’s feedback references and optics was identified. Unlike the majority of systems in the LHC, feedback references and optics are pre-loaded as discrete tables into the service unit and played-back (pushed to the controller) with the aid of an intricate mechanism which relies on the generation of dedicated timing events and payload handling, in real-time, in order to extract the table indexes of the required value(s). The aim is to simplify and homogenize the operational procedure over all LHC systems by relying on three *universal* timing events (ARM, START and ABORT)

Manuscript received June 1, 2016.

Diogo Alves (e-mail: Diogo.Alves@cern.ch) and Stephen Jackson (e-mail: Stephen.Jackson@cern.ch) are with the Beams Department, Beam Instrumentation Group, European Organization for Nuclear Research, CERN, F-01631 Cedex, France.

Kajetan Fuchsberger (e-mail: Kajetan.Fuchsberger@cern.ch) and Jorg Wenninger (e-mail: Jorg.Wenninger@cern.ch) are with the Beams Department, Operations Group, European Organization for Nuclear Research, CERN, F-01631 Cedex, France.

¹The number of oscillations that particles describe in a complete revolution.

²Responsible for bending the particle beam into a closed orbit.

and the pre-loading of time-based reference waveforms to be played strictly as a function of time.

II. HARDWARE ARCHITECTURE

The beam-based feedbacks in the LHC comprise two servers [5]: 1) the feedback controller which is responsible for handling all real-time input and output communications, checking and sanitising input data and running the control algorithms, and 2) the feedback service unit which acts as a proxy, thus shielding the controller itself from interactions with the outside world, performing the management of the settings, the proxying of the acquired and processed data, as well as monitoring and logging activities. These servers are hosted on two Hewlett-Packard Proliant Gen8-based systems with 24 (hyper-threaded) cores spread over two CPU units (Xeon E5-2630) running at 2.6GHz and holding 32GB of Random Access Memory. In each of these machines is also installed a PCI-based a Control Timing Receiver (CTR) module (CERN in-house development standard), used for receiving LHC events in real-time. The two servers are connected via a private link on 1Gb/s Network Interface Cards (NIC). Both settings management and data proxying (controller pushing data to the service unit) are performed via a serialization mechanism based on ROOT (a scientific library developed at CERN) objects.

All real-time input-output is ethernet-based (1Gb/s) over CERN's switched technical network using the User Datagram Protocol (UDP). The beam position measurements, used as the observable for the control of the orbit and radial-loop (baseline orbit) are sent to the controller at a rate of 25Hz, from 67 Front-End-Computers (FECs) spread over 27Km, and adding up to a data rate of approximately 2MByte/s. These measurements consume most of the network's bandwidth taken up by the beam-based feedbacks in the LHC.

III. SOFTWARE ARCHITECTURE AND PRELIMINARY TESTING

Both the feedback controller and the feedback service unit run on version 6 of the Scientific Linux operating system (64-bit) using RedHat's Message Real-time Grid (MRG) kernel for enhanced deterministic behaviour.

In order to improve the real-time configuration of the system's feedback references and optics, several implementation options were considered: whether to add this functionality to the code-base of the controller, the service unit or whether to develop a dedicated application from scratch. In order to establish the strategy to adopt, several factors were considered, and in particular, whether to change already tested software components which were crucial from an operational point-of-view. The required functionality was designed in such a way that it could easily be incorporated into a standalone dedicated application (based on CERN's ubiquitous FESA [6] framework) or into already existing code-bases. Three main components were developed (see Fig. 2): 1) a state-machine (along with auxiliary classes for defining state transition rules and providing event and state change subscription services); 2) a class to orchestrate the play-back of the various pre-configured

waveforms and 3) an interface to be implemented by the various waveform categories to be pre-configured, e.g. tune references, orbit references, optics, etc. State machine states and events were implemented as singleton class instances allowing for type safety checks at compilation time. As part of the development process, various unit-tests were developed in order to test each of these software components, individually, as well as the overall functionality. The GoogleTest³ and GoogleMock⁴ frameworks, popular in the community and supported at CERN, were selected for this purpose. Furthermore, a fake timing source, implementing the interface for the operational timing source was also developed in order to allow full testing without the need to include hardware devices whatsoever. Tests span, from the simply checking state machine configurations (e.g. state transition rules), checking the event and state change notification services, to finally checking the full chain: from timing handler to the actual objects performing the appropriate time-based reference selection.

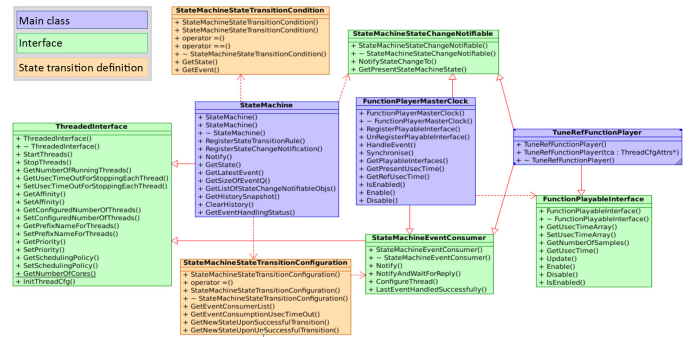


Fig. 2. UML diagram of the developed software

Finally, it was decided to incorporate the developed functionality into a new FESA-based server (the LHC feedback function player) running on the same machine as the feedback service unit, see Fig. 3.

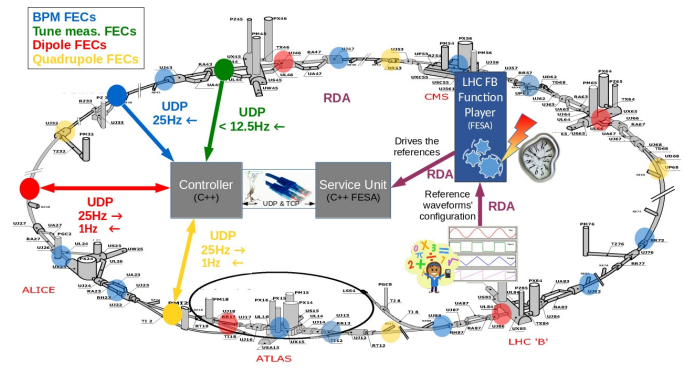


Fig. 3. Architecture of the overall feedback system

The three main reasons for supporting this choice were: 1) the fact that neither the feedback controller nor the feedback service unit code-bases had to be changed, 2) the usage of CERN's standard FESA framework (tightly integrated

³<https://github.com/google/googletest>

⁴<https://github.com/google/googletest/tree/master/googlemock>

into the accelerator control infrastructure) which allows, via the remote data access component on the middleware tier, a standard for the automatic configuration of the feedback service unit as the references are played-back as a function of time. The integration of the developed functionality into a FESA-based application was done with the help of a newly developed class that provides access to FESA components from non-FESA objects.

Preliminary tests were carried out using the configuration of the tune references as a use-case. An example of these tests is shown in Fig. 4. Here, one can see that the configured references which are being played-back by the new function player system are received by the feedback service unit at the expected times.

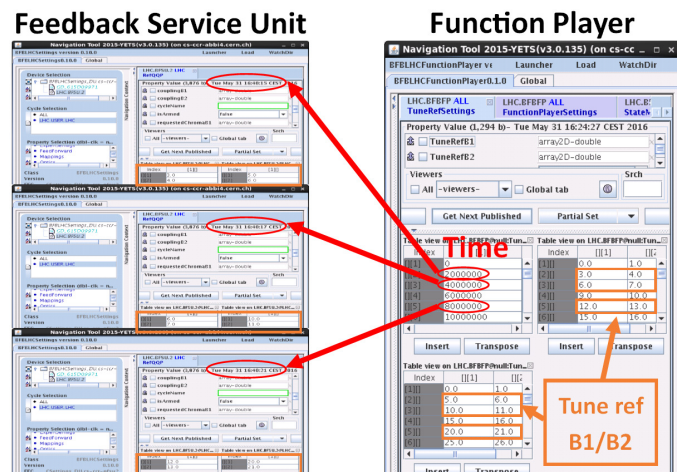


Fig. 4. Preliminary tests on the LHC beam-based feedbacks' function player system

IV. CONCLUSIONS

The required functionality has been designed, developed and unit-tested, in advance, even before the implementation strategy had been decided. Highly modular, and generic components were developed providing a high degree of flexibility in terms of configuration options. Seamless integration into CERN's standard FESA framework, was performed with the development of a simple and generic interface class. The system underwent successful preliminary tests and it is expected that its formal commissioning, initially with the tune reference as a use-case, will be performed this year, after which other interfaces will be added to configure e.g. the orbit reference and the optics.

REFERENCES

- [1] R. Steinhagen et al., *Real-Time Beam Control at the LHC*, PAC 2011.
- [2] Edmund Wilson, *The Physics Of Particle Accelerators*, Oxford University Press, first edition, 2001.
- [3] O. R. Jones, *LHC Beam Instrumentation*, PAC 2007, Albuquerque, New Mexico, USA.
- [4] M. Gasior and O. R. Jones, *High Sensitivity Tune Measurement By Direct Diode Detection*, PAC 2005, Lyon, France.
- [5] L. K. Jensen et al., *Software Architecture for the LHC Beam-Based Feedback System at CERN*, ICALEPCS 2013.
- [6] M. Arruat et al., *Front-End Software Architecture*, ICALEPCS 2007.