ADVANTAGES AND CHALLENGES TO THE USE OF ON-LINE FEEDBACK IN CERN'S ACCELERATORS CONTROLS CONFIGURATION MANAGEMENT

Z. Zaharieva, S. Jensen, J. Rolland Lopez De Coca, A. Romero Marin
CERN, Geneva, Switzerland

Abstract

The Controls Configuration Service (CCS) provides the Configuration Management facilities for the Controls System for all CERN accelerators. It complies with Configuration Management standards, tracking the life of configuration items and their relationships by allowing identification and triggering change management processes. Data stored in the CCS is extracted and propagated to the controls hardware for remote configuration. The article will present the ability of the CCS to audit items and verify conformance to specification with the implementation of on-line feedback focusing on Front-End Computers (FEC) configurations. Long-standing problems existed in this area such as discrepancies between the actual state of the FEC and the configuration sent to it at reboot. This resulted in difficult-to-diagnose behaviour and disturbance for the Operations team. The article will discuss the solution architecture (tailored processes and tools), the development and implementation challenges, as well as the advantages of this approach and the benefits to the user groups – from equipment specialists and controls systems experts to the operators in the Accelerators Controls Centre.

INTRODUCTION

The Configuration Management is nowadays an indispensable part of the operation and maintenance of any modern engineering system, especially when considering large scale Controls Systems.

The amount of technical data, necessary for the control of the CERN accelerator complex is enormous (more than 4000 computers, 80000 devices and 2000000 parameters to control). A common description (configuration), in a centralized storage, of all objects needed for the control of the accelerators is an essential prerequisite for maintaining the integrity of the Controls System and the correct functioning of the accelerators.

The Controls Configuration Service (CCS) provides a collection of processes, tools and a common repository for all configuration data (a relational database) with the objective to present the up-to-date state of the components (configuration items) of the Controls System at a given moment in time (baseline), systematically tracks the components' changes with time and keeps an overall consistency of the different components from the view point of the coherent functioning of the Controls System [1].

The CCS implements the best practices for Configuration Management formalized by different standards such as ITIL (Information Technologies Infrastructure Library) [2], IEEE standards [3], Control Objectives for Information and Related Technology (COBIT) [4], etc. The CCS provides configuration management functionalities such as the unique identification of the configuration items and presents their authorized configurations, complying with predefined criteria, in addition to controlling configuration changes and status accounting of the items. The CCS provides the relationships between the configuration items and their dependencies too.

The Controls Configuration Database (CCDB) is the central part of the CCS and the heart of the CERN Accelerators Controls System. It caters for the configuration management of the components of the Controls System itself, for example the Controls Front-End Computers (FECs), the Controls Timing, the Controls Diagnostics and Monitoring (DIAMON) [5], the Controls Middleware (CMW) and many other essential Controls components, as well as the configuration of the accelerator elements as seen by the Controls System and their software representation, e.g. power converters [6], vacuum systems, etc., for all accelerators: the Large Hadron Collider (LHC), the Super Proton Synchrotron (SPS) Complex, the Proton Synchrotron (PS) Complex, and the CLIC Test Facility (CTF3).

Currently the relational database model of the CCDB comprises of 960 tables and more than 70000 lines of PL/SQL code, which supports the specific business logic for the configuration management processes. Today, the CCDB stores over 15GB of current reference data in addition to about 85GB of historical versioned data.

THE NEED FOR CONFIGURATION FEEDBACK

A recently implemented, new functionality of the Controls Configuration Service is the availability of a specific type of status accounting of the different configuration items, called configuration feedback. The new feature also includes the usage of the configuration feedback for the purposes of auditing and reporting on the configuration items and their compliance to the predefined configurations.

An overview of the configuration data flow is presented in Figure 1. The predefined configuration models of the configuration items are extracted by different APIs from CCDB and provided to the objects being configured, e.g. FECs, drivers, etc., in the format best matching their requirements (XML or other text files, binaries, etc.).

ISBN 978-3-95450-139-7

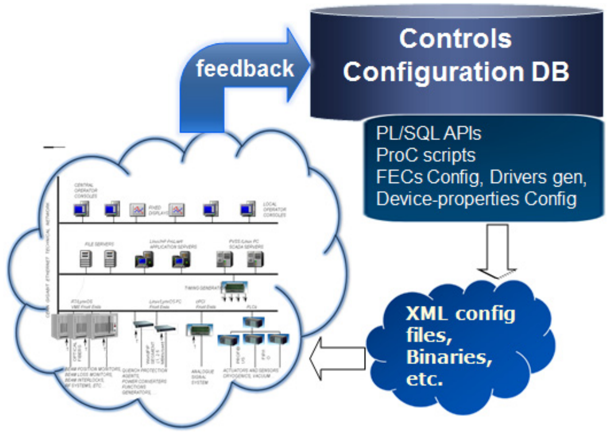

Figure 1: Overview of the configuration data flow.

For many years, one of the biggest problems faced by CERN's Controls Configuration Management was how to be sure that once the configuration data is extracted, it is successfully loaded and actually used by the different components of the Controls System. Another problem was related to the fact that it is possible to have a mismatch between the hardware configuration of a given FEC and its state in reality (e.g. missing module in the crate, additional module in the crate, etc.), which means that the software configurations sent to the computer could not be used by it.

The previously existing one-way mechanism of data propagation for the configuration of the Controls components was crucial for the initialization of the different layers of the Controls System; however it was not enough for the flawless operation of the System.

It is for the aforementioned reasons that the feedback functionality was designed and implemented as part of the Controls Configuration Service during the last 2 years.

The status data comes in the form of an automated feedback, sent from the configured components, identifying the currently loaded configuration. This information provides the possibility to:

- ✓ Verify the conformance of the real component with respect to the model configuration:
  - To detect installation errors, such as a hardware module not being installed, or hardware module installed into the wrong position in the crate
  - To identify the removal of a previously installed component, an action not reflected into the central configuration model
- ✓ Confirm that the configurations generated from the central repository are successfully loaded into the target configuration item (e.g. Front-End computer).
- ✓ Know at every moment in time, which configuration version is currently loaded into the configuration item.

As seen from the examples listed above, the configuration feedback is a powerful mechanism that provides the means to diagnose the configuration discrepancies between the configuration model and the actual state of the configuration items.

SCOPE OF THE FUNCTIONALITY

The configurations items described in the Controls Configuration Database could be grouped into several logical domains depicted in Figure 2.

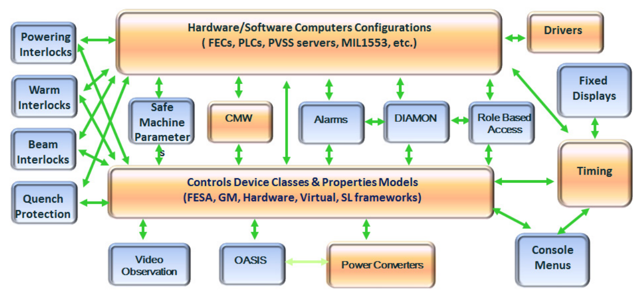

Figure 2: Overview of the logical configuration domains.

The Configuration feedback could be beneficial for the configuration items in any of the domains presented in Figure 2. The implementation of the feedback however is being undertaken based on a priority rating determined by weighing the severity of the experienced problems, the impact to the users of the service and the ease with which the feedback could be introduced.

The first Configuration domains where the feedback was introduced successfully were the configuration of the Controls Middleware, the Power Converters and the Hardware configurations of the FECs. It is previewed that the feedback functionality will be extended to cover other Controls Configuration domains in the near future, e.g. the Drivers, the software devices configurations as well as the Timing.

OVERVIEW OF THE ARCHITECTURE

The CCDB model had to be extended in order to implement the new dedicated tables, the configuration feedback-specific processes and the user interfaces. The configuration feedback architecture is presented in Figure 3. The feedback mechanism is implemented as a persistent process that:

- ✓ Receives the direct, on-line feedback messages in a predefined, standard format from the FECs.
- ✓ Extracts and stores the feedback data into a set of dedicated tables in the CCDB
- ✓ Processes the data and performs comparisons between the feedback data and the existing configurations
- ✓ Presents the results from the comparisons in a user-friendly way through different reports
- ✓ Triggers a notification to the end users in the case of important discrepancies (errors).

ISBN 978-3-95450-139-7

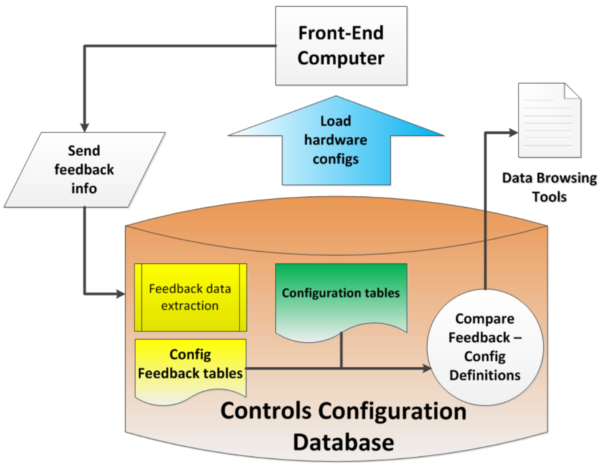

Figure 3: Configuration Feedback Architecture.

The configuration feedback for the FECs hardware configurations and drivers' configurations is sent at boot time thanks to recently introduced functionalities in the drivers, generated through the standard Controls tools. The mechanism to send the configuration feedback messages from the FECs to the CCDB is the one developed for the Controls System Tracing facility [7].

Once the configuration messages arrive in the CCDB, there is a process that runs periodically and extracts the feedback data and logs it into dedicated tables per configuration domain. At that moment the data analysis of the feedback data is automatically triggered to start comparing the status of the configuration items to the feedback data.

The results are displayed through different reports, integrated into the Controls Configuration Data Browser, which is built using Oracle APEX technology [8].

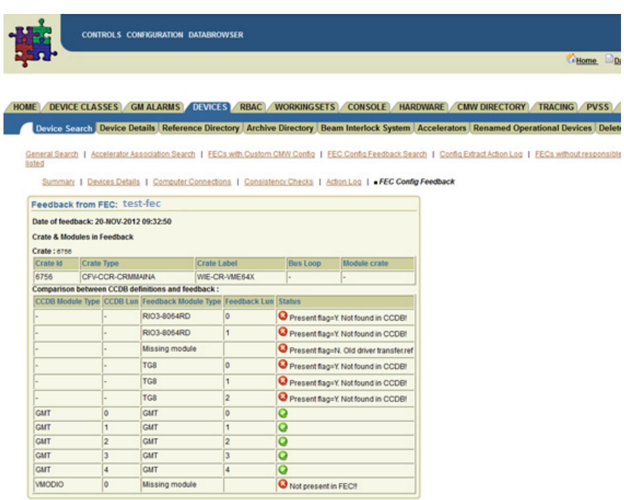

Figure 4: A report showing the results of the analysis of the configuration feedback for the hardware modules for the given Front-End computer.

The report shown in Figure 4 presents one of the analysed use cases, which detects if there are discrepancies between the installed modules in a FEC and the configured modules for the same FEC in the CCDB. The possible options are: 1. the module feedback data and the pre-defined configuration match; 2. the module is present in the FEC and it is not configured in the CCDB – error raised; 3. the module is not present in the FEC and it is configured in the CCDB – error raised.

Apart from presenting the results through reporting tools, a notification mechanism is also implemented to inform the responsible of a given FEC about the errors (inconsistencies) found during the analysis of the feedback.

IMPLEMENTATION CHALLENGES

Some of the challenges, which were encountered and overcome in order to implement the configuration feedback as part of the Controls Configuration Service, are listed below.

Extraction of Configurations – Data-Driven Controls System

In order to be able to implement the configuration feedback mechanisms, all related configuration extractions from the CCDB had to be modified in order to be compliant with certain criteria, e.g. to carry the version of the configuration and the timestamp of its extraction in order to be able to identify each extracted configuration in a unique way. This is quite a challenging task as it needs to be implemented across different Configuration areas and sometimes there are many history layers that need to be taken into consideration.

Standardization of the Feedback Message Format

In order to be able to introduce in a unified way the diverse feedback data from different configuration items and areas into the CCDB, it was important to provide a standard for the feedback messages. This ensures that the data is structured and can be easily analysed later on. On the other side the feedback message format had to be flexible enough to support the needs of the diverse configuration items and adapt to them.

A specification of the feedback message was prepared in collaboration with the Accelerator Controls Exploitation Tools project (ACET) as well as with people responsible for different Controls components, e.g. CMW, Drivers, etc.

The feedback message is a string, comprising of multiple key-value pairs. Some of the keys are obligatory and have predefined possible list of values, while others are optional. The keys were split into three groups: header keys (e.g. message-type, sender-domain, sender-sub-domain), configuration keys (e.g. sender-user-id) and project specific keys (e.g. drivers specific keys – time-generated, time-compiled, time-loaded, etc.).

Integration into the Relational DB Model

In order to treat the feedback data inside the CCDB a specific data flow and processes were implemented, using PL/SQL. The feedback messages are first stored into a staging table, which collects all of the feedback data, not

ISBN 978-3-95450-139-7

taking into account which configuration domain this data is destined to be used in later on. Based on the existing relational tables for each domain (e.g. computer-crates-modules), analogous structures were added in order to store the feedback data.

Dedicated Extract-Transform-Load (ETL) processes are implemented between the staging table, where the raw data gets inserted, and the dedicated relational tables for the specific configuration domain. The values in the header keys of sender-domain and sub-sender-domain determine which particular ETL process to be executed.

Scalability

There are more than 3000 FECs that could send configuration feedback with more than 13000 modules, therefore the scalability of the implemented solution was an important design consideration, especially taking into account that in the near future the feedback will be extended to other configuration domains, such as the Timing. The standardization of the feedback message format helped to process in a consistent and scalable way the preliminary recorded data and to transfer it to the dedicated tables.

Another important point for the design was the modularity of the ETL processes, which could be increased if there are new use-cases added for the configuration feedback (new values for the header keys of sender-domain and sub-sender-domain).

QUALITY ASSURANCE AND TESTING OF THE FEEDBACK MECHANISMS

The CCS consists of four environments – Development, Test, Next and Production.

The new functionality for the Configuration Feedback was first tested in the Test environment within the limits of the Controls Configuration Service and afterwards transferred to the Next environment, which is part of the Controls TestBed [9]. It was possible to validate the configuration feedback and to perform integration testing with the major components of the Controls System before deploying the solution in a production environment. Stress testing was also performed in the TestBed with the help of the CMW tracing components for propagation of the feedback messages.

IMPACT FOR THE USERS

The configuration feedback benefits the complete user community of the CCS (more than 300 users) – equipment specialists, controls experts as well as operators in the Accelerators Controls Centre. Due to the implemented functionality of auditing the model configurations with the ‘real-life’ data and its presentation via the reporting tools, the users are able to quickly diagnose problems or even to prevent them before they occur. The implemented solution notifies the users in real time for hardware configuration problems thus giving them the possibility to react immediately and correct the configuration data.

A future development in order to further enhance the user experience will be to present the results of the configuration feedback analysis and integrate them in the DIAMON console as well as into the Alarms (LASER) console in the case of configuration errors. Putting in place this improvement should not be that difficult as both systems, DIAMON and LASER use the CCS as a provider for their configuration data.

CONCLUSION

The configuration feedback is an extremely important functionality provided by the Controls Configuration Service. It presents a solution to the years-old problems of discrepancies between the generated FECs configurations and the actually loaded ones, usually resulting in unexpected behaviour after a reboot (so called ‘time-bombs’).

By putting in place this specific status accounting of the configuration items and its usage for items auditing, the Controls Configuration Service answers to the latest standards in the area of Configuration Management. The feedback gives the possibility to automatically discover information about the configuration items and track changes as they happen. Different challenges were overcome in order to provide a scalable and user-friendly solution, which gives the users the possibility to quickly troubleshoot configuration problems or even to prevent them from happening thus improving the availability of the FECs and of the Controls System in general.

REFERENCES

[1] Z. Zaharieva *et al.,* “Database Foundation for the Configuration Management of the CERN Accelerator Controls Systems”, ICALEPCS’11, Grenoble, France, Oct-2011.
[2] ITIL: www.itil-officialsite.com
[3] IEEE Std. 828-2012: standards.ieee.org
[4] COBIT: www.isaca.org/COBIT
[5] Z. Makonnen, M. Buttner, Z. Zaharieva, “Challenges to Providing a Successful Central Configuration Service to Support CERN’s New Controls Diagnostics and Monitoring System”, ICALEPCS’13, San Francisco, USA, Oct-2013.
[6] Q. King, S.T. Page and Z. Zaharieva, “Automatic Inventory and Configuration Management Tools for the LHC Power Converter Controls”, ICALEPCS’09, Kobe, Japan, Oct-2009.
[7] F. Ehm, A. Dworak, “A Remote Tracing Facility for Distributed Systems”, ICALEPCS’11, Grenoble, France, Oct-2011.
[8] Z. Zaharieva, R. Billen, “Rapid Development of Database Interfaces with Oracle APEX, used for the Controls Systems at CERN”, ICALEPCS’09, Kobe, Japan, Oct-2009.
[9] J. Nguyen Xuan, V. Baggiolini, “A Testbed for Validating the LHC Controls System Core before Deployment”, ICALEPCS’11, Grenoble, France, Oct-2011.

ISBN 978-3-95450-139-7