THE LHC SEQUENCER

Reyes Alemany-Fernandez, Vito Baggiolini, Roman Gorbonosov,

Denis Khasbulatov, Mike Lamont, Pascal Le Roux, Chris Roderick,
CERN, Geneva, Switzerland

ABSTRACT

The Large Hadron Collider (LHC) at CERN is a highly complex system made of many different sub-systems whose operation implies the execution of many tasks with stringent constraints on the order and duration of the execution. To be able to operate such a system in the most efficient and reliable way, the operators in the CERN control room use a high level control system: the LHC Sequencer. The LHC Sequencer system is composed of several components, including an Oracle database where operational sequences are configured, a core server that orchestrates the execution of the sequences, and two graphical user interfaces: one for sequence edition, and another for sequence execution. This paper describes the architecture of the LHC Sequencer system, and how the sequences are prepared and used for LHC operation.

THE LHC SEQUENCER ARCHITECTURE

The LHC Sequencer Architecture is made of **two core components**: the **Sequencer Executor** and the **Database**. Two Graphical User Interfaces (GUI), one to interface the executor and another one to interface the database, provide the operators in the CERN Control Centre (CCC) with the required control on running sequences and on the creation or modification of sequences, respectively.

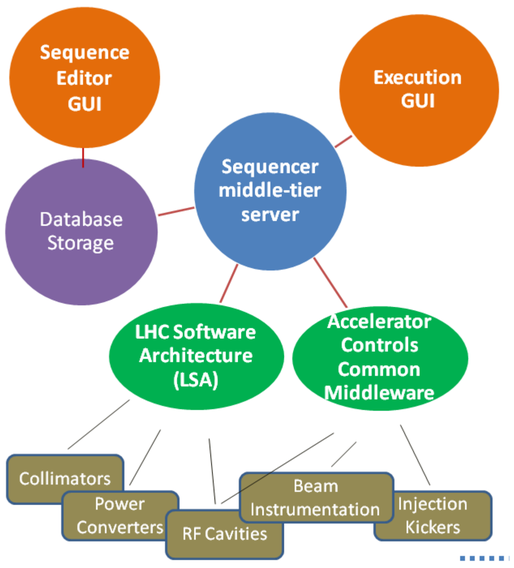

Figure 1: LHC Sequencer Architecture.

Sequencer Server and Sequencer Client

The architecture and technology used by the Sequencer [1] follows the CERN accelerator controls software standards: a 3-tier architecture implemented in Java using the Spring Framework [2]. This architecture is shown in Figure 1. The **sequencer middle-tier server** (running Linux) contains the core functionality, mainly, sequence execution, temporary sequence storage and interface via Java libraries to the controls of the different LHC sub-systems, i.e. the **LHC Software Architecture (LSA)** [3] and the accelerator controls **Common Middleware (CMW)** [4].

The client tier consists of an **execution GUI** from which the execution of the sequences is controlled by the LHC operators in the Linux or Windows consoles at the CCC. Many GUIs may connect to the same middle-tier server.

Database Persistent Storage of Sequences

All sequences are persistently stored in the **Oracle database**. The database schema consists of a series of tables that map the sequences representation as shown in Figure 2.

Sequence Editor and Sequence Executor

Every of these tables and columns is filled using the **Sequences Editor**, a Java program that access directly the database via SQL statements. The editor is protected with Role Based Access [5] in order to restrict the edition of sequences to authorized people only. Figure 3-left shows a picture of the Sequences Editor.

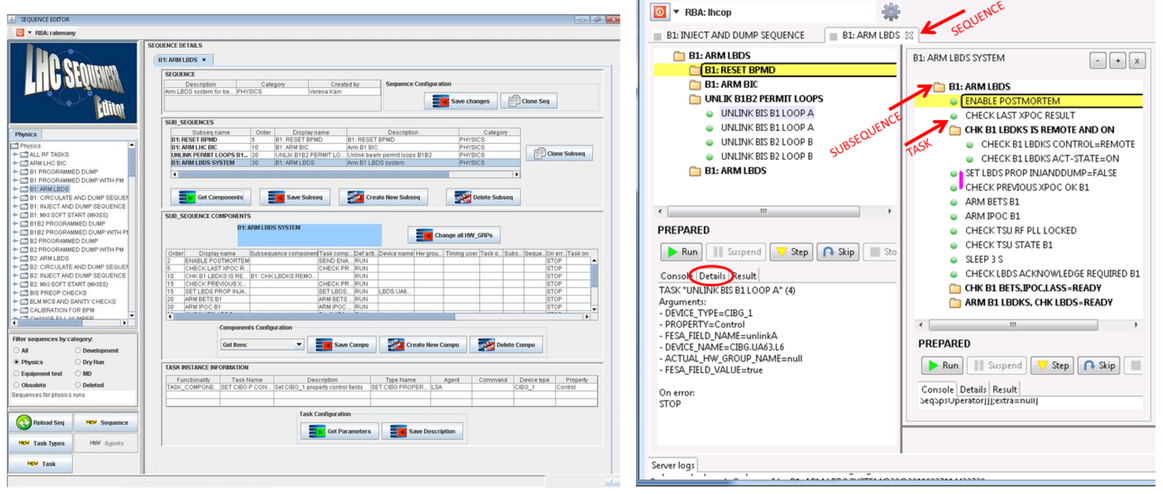

Figure 3: Left: Sequences Editor GUI; right: Sequences Executor GUI. Both GUIs show the same sequence, B1: ARM LBDS.

Once a (sub)sequence is created or modified by the user, the sequencer server retrieves it from the database and it is converted into a Java source file. The right compilation of the source code is a way of ensuring, to a great extent, the coherence of the sequence.

The Sequencer Executor is a graphical user interface (GUI) developed using the Standard Widget Toolkit[6]. Figure 3-right shows a picture of the Executor.

Once a (sub)sequence is selected, the GUI shows the corresponding tree structure of the sequence where the component names displayed are the ones defined by the user as displayed names in the database tables. Clicking on a particular task, the parameters of the tasks, as read from the database, are shown in the tab called “Details”. The GUI allows for drag and drop of subsequences.

REFERENCES

[1] V. Baggiolini, R. Alemany Fernandez, R. Gorbonosov, D. Khasbulatov, M. Lamont, “A Sequencer for the LHC era”, ICALEPCS’09, Kobe, Japan, 2009, Conference Proceedings.

[2] http://www.springframework.org

[3] G. Kruk, S. Deghaye, M. Lamont, M. Misiowiec, W. Sliwinski, “LHC Software Architecture [LSA] – evolution toward LHC beam commissioning”, ICALEPCS’07, Knoxville, Tennessee, 2007, Conference Proceedings.

[4] http://proj-cmw.web.cern.ch/proj-cmw/documents.ht m

[5] S. Gysin, A.D. Petrov, P. Charrue, W. Gajewski, V. Kain, K. Kostro, G. Kruk, S. Page, M. Peryt, “Role-based Access Control for the Accelerator Control”, ICALEPCS’07, Knoxville, Tennessee, 2007, Conference Proceedings.

[6] http://www.eclipse.org/swt

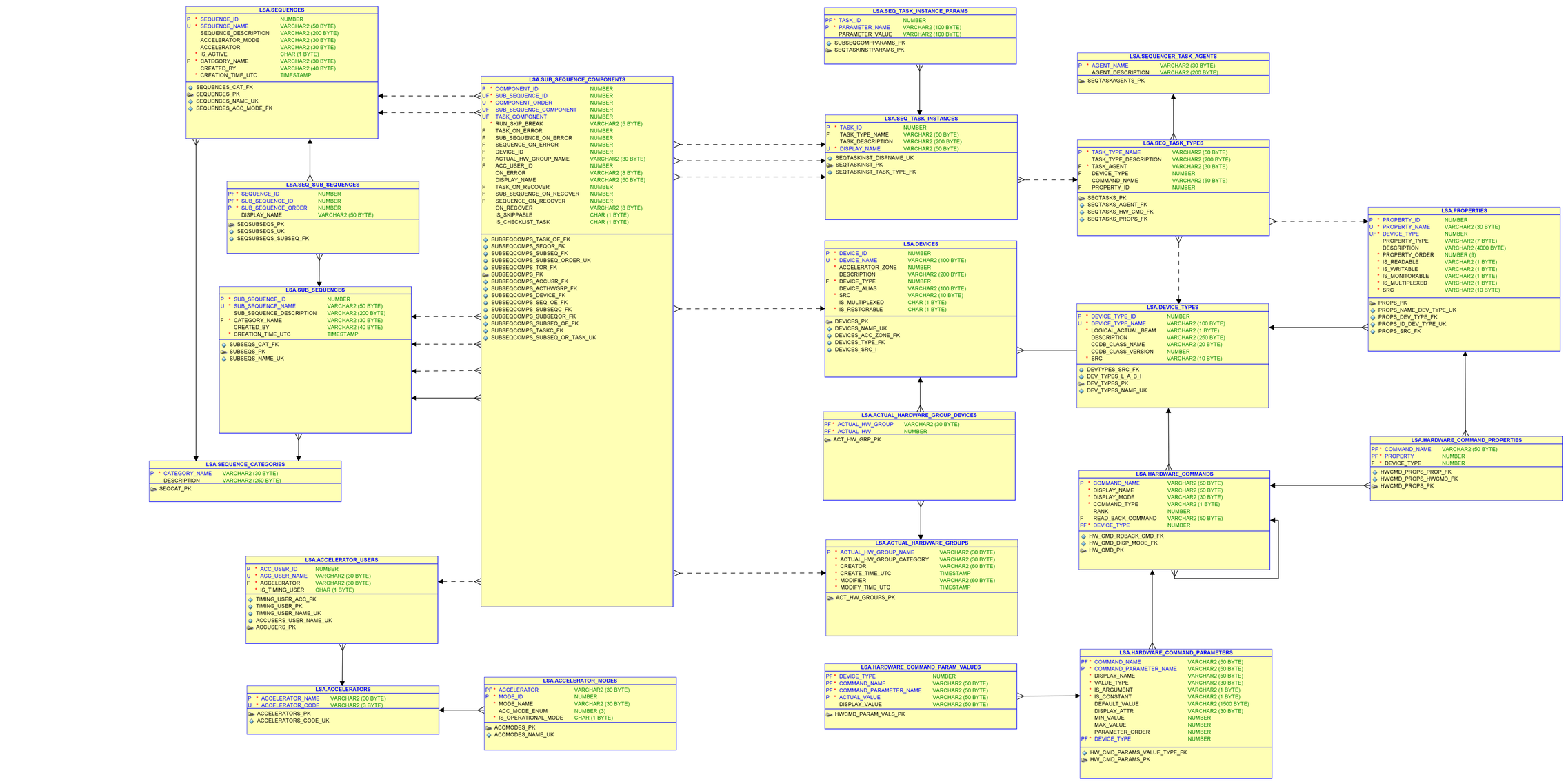

Figure 2: LHC Sequences Database Schema.