SETTINGS MANAGEMENT WITHIN THE FAIR CONTROL SYSTEM BASED ON THE CERN LSA FRAMEWORK

J. Fitzek, R. Mueller, D. Ondreka, GSI, Darmstadt, Germany

Abstract

A control system for operating the future FAIR (Facility for Antiproton and Ion Research) accelerator complex is being developed at GSI. One of its core components is the settings management system.

At CERN, settings management and data supply for large parts of the CERN accelerator complex is done using the LSA (LHC Software Architecture) framework. Several concepts of the LSA framework already fit the FAIR requirements: Generic structures for keeping accelerator data; modular design; separation between data model, business logic and applications; standardized interfaces for implementing the physical machine model. An LSA test installation was set up at GSI and first tests were performed controlling the existing GSI synchrotron SIS18 already with the new system.

These successes notwithstanding, there are issues resulting from conceptual differences between CERN and FAIR operations. CERN and GSI have established a collaboration to make LSA fit for both institutes, thereby developing LSA into a generic framework for accelerator settings management. While focussing on the enhancements that are necessary for FAIR, this paper also presents key concepts of the LSA system.

FAIR

The international FAIR facility with its nine new accelerator installations will be built at GSI, using the existing linac and synchrotron SIS18 as injectors (see Fig. 1).

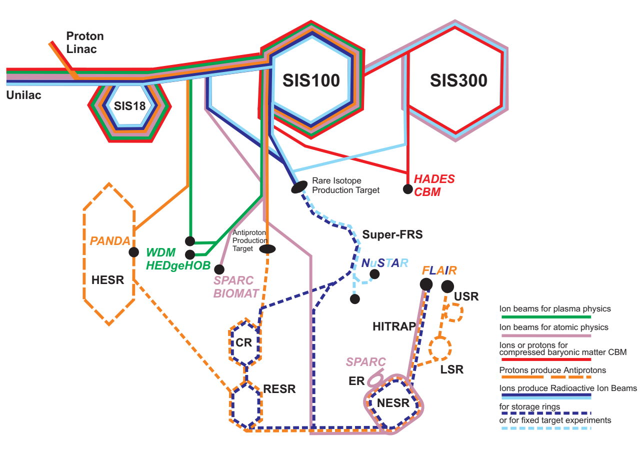

Figure 1: GSI/FAIR beamlines, P. Schuett, GSI 2010.

Central aspect is an increased number of research programs resulting in up to five beams in parallel with pulse-to-pulse switching between different particle types. The future facility will be controlled by a new control system which will be able to support all aspects of the complex GSI/FAIR operations on a common technical basis [4]. The future control system is designed at the moment, keeping well working and proven principles while adopting new methodologies where beneficial.

Important aspects of the control system are generation of settings and data supply. It was evaluated and decided to use the existing LSA framework from CERN for settings management and data supply within the FAIR control system. A collaboration with CERN was set up with joint development effort put into future LSA development [1].

LSA - THE LHC SOFTWARE ARCHITECTURE

LSA was developed at CERN starting in 2001 and is now the core controls software component for settings management and data supply within the CERN control system. For a detailed description of LSA see [2].

LSA - Functional Overview

The LSA system was designed in a generic way and provides clear separation between data model, business logic and applications. Its modular structure allows institute specific implementation to be easily plugged in.

The system covers all important settings management aspects: optics (twiss, machine layout), parameter space, settings generation and management, settings modification (trim), propagation from physics to hardware parameters, operational and hardware exploitation (equipment control, measurements), and beam based measurements.

An accelerator within LSA is modeled by defining its parameter hierarchy – from top level physics down to hardware parameters. Using the optics, the LSA system can already calculate good initial settings. Corrections can be applied to any level of the hierarchy, resulting in a consistent change of many devices at the same time. As an example for a part of such a hierarchy at GSI, see Fig. 3. The LSA system consists of different functional building blocks, which among other benefits entitle physicists to implement the machine model themselves in a structured and simple way.

LSA - Technology Stack

LSA is written in Java and uses the Spring framework, that provides a light-weight container for the Java platform, dependency injection, aspect oriented programming (AOP), testing framework, remoting and transactions. An overview of the LSA software stack is shown in Fig. 2.

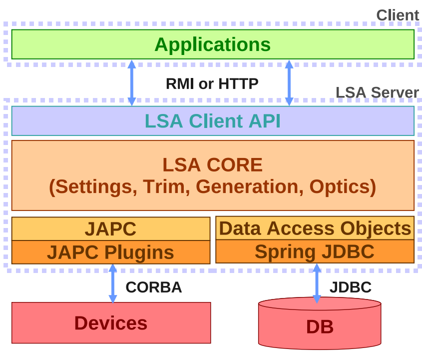

Figure 2: LSA software stack, G. Kruk, ICALEPCS 2007.

LSA is visible to the applications trough a series of façade classes that group the functionality of LSA by topics (setting management, hardware access etc.). They represent a stable and backward compatible interface which separates applications from business logic, thus applications can concentrate on presenting information.

Communication with the devices is done through a powerful abstraction layer called JAPC (Java API for Parameter Control) [3], that hides middleware specifics and thus allows access to all devices through the same interface.

LSA AT GSI

The collaboration on LSA started in 2007 with two software developers from GSI working for 1.5 years on site in the LSA team during the LHC commissioning and startup. Since then the collaboration is well established. In 2008, an LSA test system was set up at GSI.

Setting up an LSA test system

The LSA system runs out-of-the-box given an empty LSA database with just a few tables prefilled. First steps include setting up an Oracle database instance and an LSA test server, which is a standalone Java process. For the small number of missing software references (e.g. to CERN's online model server or role based access system), a dummy implementation needs to be provided which fulfills the interface. Since the LSA system is data-driven, the next step is to import the accelerator layout into the database, such as static information about accelerators, beamlines and devices. As a result of this initial setup, generic LSA applications deployed via Java WebStart are already running.

After a new JAPC plug-in for the existing GSI middleware was developed, first calls to devices proved that the environment was correctly set up.

Implementing GSI accelerators within LSA

A project team consisting of machine physicists and software developers from different groups at GSI started modeling the existing synchrotron SIS18 within the institute specific part of LSA: defining the parameter hierarchy, implementing propagation rules, importing optics and defining test cycles. Since the implemented rules were written in a generic way, even test cycles for the future FAIR synchrotron SIS100 have already been successfully generated. Next step will be to look at the existing GSI storage ring and its representation within the new system.

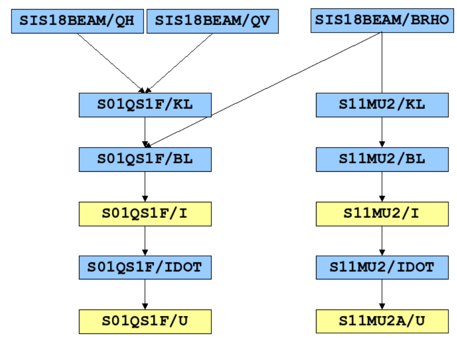

Figure 3: Example of an LSA parameter hierarchy at GSI.

While modeling the existing accelerators in LSA, the project team mainly focusses on the LSA concepts and the question, whether they are really generic enough to fit all needs of complex and parallel accelerator operations.

First test with beam

Keeping in mind the challenge of putting the new control system for FAIR in place while the existing facility is running, it is however vital, that the new system will be commissioned with parts of the existing machine.

For the settings management and data supply part with LSA, this includes testing the new system already with the existing accelerators. As a first milestone, a successful test with beam was performed in march 2010 with the existing synchrotron SIS18. Several scenarios were tested: operations with one and two cavities and one and two shot extraction using a cycle with fast extraction.

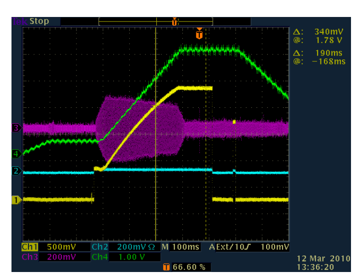

Figure 4: SIS18 test: single shot extraction.

Extending LSA

The flexibility of accelerator operations at GSI put a new view on the LSA system. In particular while modeling the SIS18, certain restrictions were found within the LSA machine model, resulting from the rather static operation of the accelerators at CERN. Based on these observations, requirements were collected. For some of these topics a final solution has been agreed upon and implementation has already started.

One new feature within LSA which will be implemented within the collaboration is the flexibility of cycle length: length of cycles and therefore length of specific functions like the dipole current can vary due to applied trims. This feature will be heavily used at GSI, where e.g. extraction energy is frequently trimmed and the corresponding adaptation of the cycle length is indispensable for the optimization of the duty cycle. This of course presumes a flexible timing system with no predefined base cycle length.

Another feature will be the support for modeling a full chain of accelerators, especially modeling inter-accelerator dependencies, which is necessary for FAIR. Also at CERN, the focus shifts towards controlling the full accelerator chain. This change in perspective is related to the fact, that now the same control system is used for many accelerators at CERN. The idea is e.g. to connect the extraction energy of one accelerator with the injection energy of the next accelerator in the chain and automatically trim settings for the whole affected chain when changed in one place. First brainstorming on this topic has started and will be continued in the near future.

So far, it seems that all of the requirements now coming from GSI are also of interest for CERN and that their implementation will be part of the LSA core system. The goal of both involved parties is clearly to make LSA as generic and flexible as necessary to be able to really fulfill all requirements, that arise from complex accelerator operations.

Technically implementing new features in the LSA core system is encouraged by the use of the Spring framework. It easily allows plugging in test implementations by one party using XML configuration while the existing implementation remains untouched. This also supports using one repository for the LSA core system even while realizing new features.

However, also institute specific implementation like access to devices, accelerator specific physics propagation rules etc. fit into the LSA concept: they reside in institute specific modules which complement the core functionality by implementing the respective interfaces. Even though it is planned to manage those specific software modules locally at the institute site in the future, at the moment the GSI modules still reside at the CERN repository, benefitting from CERNs build and release environment.

Development of LSA based applications

In addition to the existing generic LSA applications there is the need for new applications developed at GSI which fit the operators workflow. Since the LSA business logic is well separated from the applications and encapsulated by GSI specific façade classes (where only a subset of full LSA features is made visible to application developers), applications can focus on displaying information. Additionally standard prefilled GUI elements and a stable API substantially ease application development and also encourage others to write applications based on LSA.

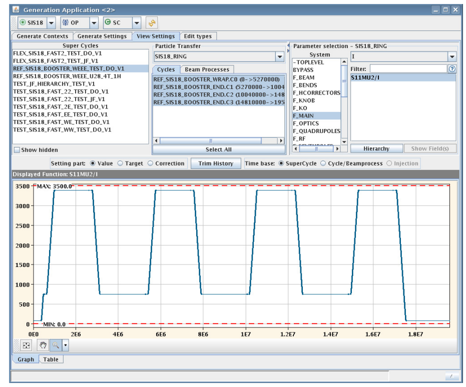

Figure 5: LSA application showing a SIS18 supercycle.

SUMMARY/OUTLOOK

The prototype installation and successful first tests with beam proved that the LSA framework already fits the requirements for settings management and data supply for single accelerators within the FAIR control system. From a technical perspective it was easy to install and to set up the system in its initial state. The biggest effort was to implement the accelerator model using the LSA framework.

New requirements arise from the flexible GSI/FAIR accelerator operations and from the necessity to model the whole accelerator complex within LSA. The corresponding enhancements of the LSA framework are implemented within the collaboration. In this way LSA evolves into a generic and flexible settings management framework for complex accelerator facilities.

REFERENCES

[1] R. Mueller, J. Fitzek, D.Ondreka, “Evaluating the LHC Software Architecture for data suppy and settings management within the FAIR control system”, ICALEPCS’09, Kobe, Japan, THP012.

[2] Grzegorz Kruk et al, “LHC Software Architecture (LSA) - Evolution Toward LHC Beam Commissioning”, ICALEPCS’07, Knoxville, Tennessee, USA, WOPA03

[3] V. Baggiolini et al, “JAPC - the Java API for LHC Timing Parameter Control”, ICALEPCS’05, Geneva, Switzerland, TH1.5-8O.

[4] Ralph C. Bär et al., “Development of a New Control System for the FAIR Accelerator Complex at GSI”, ICALEPCS’09, Kobe, Japan, TUP107.