LSA- THE HIGH LEVEL APPLICATION SOFTWARE OF THE LHC AND ITS PERFORMANCE DURING THE FIRST 3 YEARS OF OPERATION

D. Jacquet, R. Gorbonosov, G. Kruk, P. Pera Mira

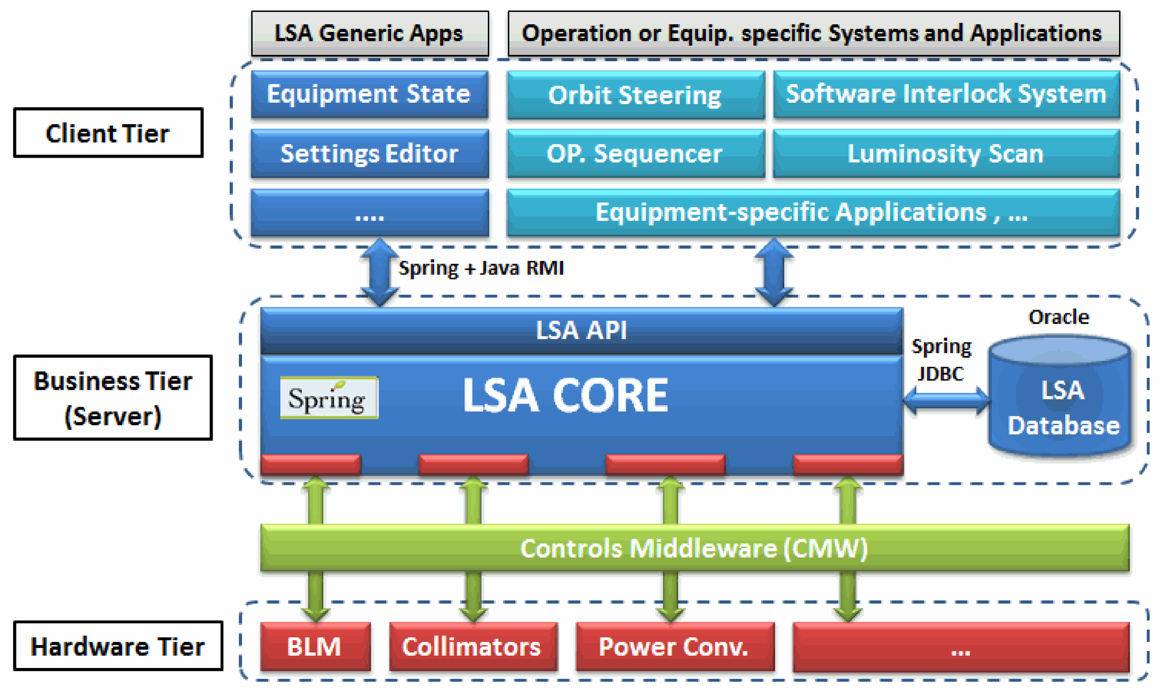
INTRODUCTION

The LHC Software Architecture (LSA)[1] project was started in 2001 with the aim of developing the high level core software for the control of the LHC accelerator. As LHC was under construction and many operational aspects not defined, the initial design was based on SPS requirements and on operational experiences with the LEP accelerator, with the idea that much of accelerator control functionality is common to all particle accelerators. The main principles and functionality of LSA can therefore be re-used and extended to any machine.

First tested in LHC transfer lines and SPS in 2003, it has now been deployed widely across the CERN accelerator complex and has been largely successful in meeting its initial aims.

LSA MAIN PRINCIPLES

LSA is composed of a data model, a set of software modules based on that model, providing accelerator control services, and a set of generic applications using these services



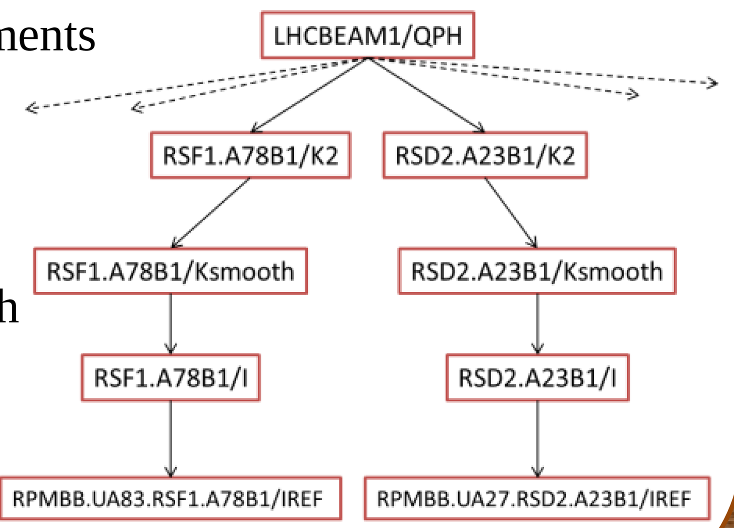
LSA is based on 3-tier architecture. The logic is implemented on the server side that exposes a simple interface used by LSA generic applications and many other client applications. All equipment and database accesses are made through the server.

LSA CORE MAIN MODULES

- Accelerator layout definition
- Devices and properties definition
- Optics and twiss parameters
- Device parameters and settings

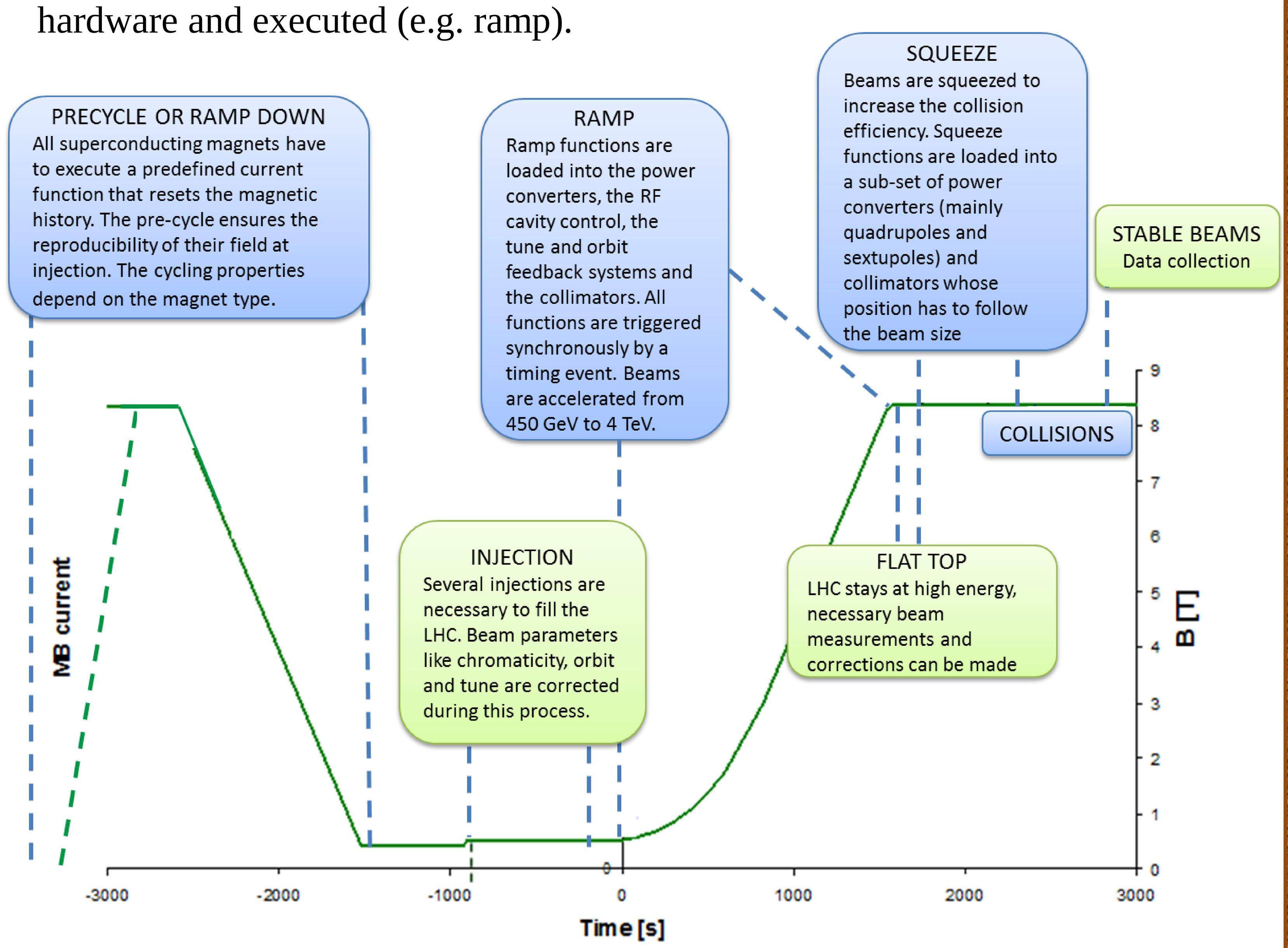
LSA SERVICES

- Access to the configuration data (Oracle database [2])
- Abstraction of the equipment access layer
- Hardware control and measurements
- Management of device settings
  - Generation of initial settings based on optic
  - Coherent settings changes with history and rollback
  - Parameters hierarchies and relationships



THE LHC CYCLE

The LHC cycle is composed of several phases necessary to bring beam from injection to luminosity production. Phases of arbitrary length (injection for example) alternate with phases where pre-defined functions are loaded into the hardware and executed (e.g. ramp).



Some of the LHC equipment, such as the power converters, the RF cavity controllers or the collimators, have been designed to support **two types of settings:**

SCALAR: during phases of undefined length like injection or flat top, scalar values are loaded to hardware. These values are set synchronously when beam parameter corrections are applied.

FUNCTION: during phases like ramp or squeeze, function settings are loaded to the hardware. Functions are triggered by timing events, no modification can be made while a function is executed by the equipment, although real time tune and orbit feedback are operational.

The LHC cycle is driven **by services provided by LSA** as:

- loading of function or scalar settings to the hardware
- control of the settings change
- triggering of timing event.

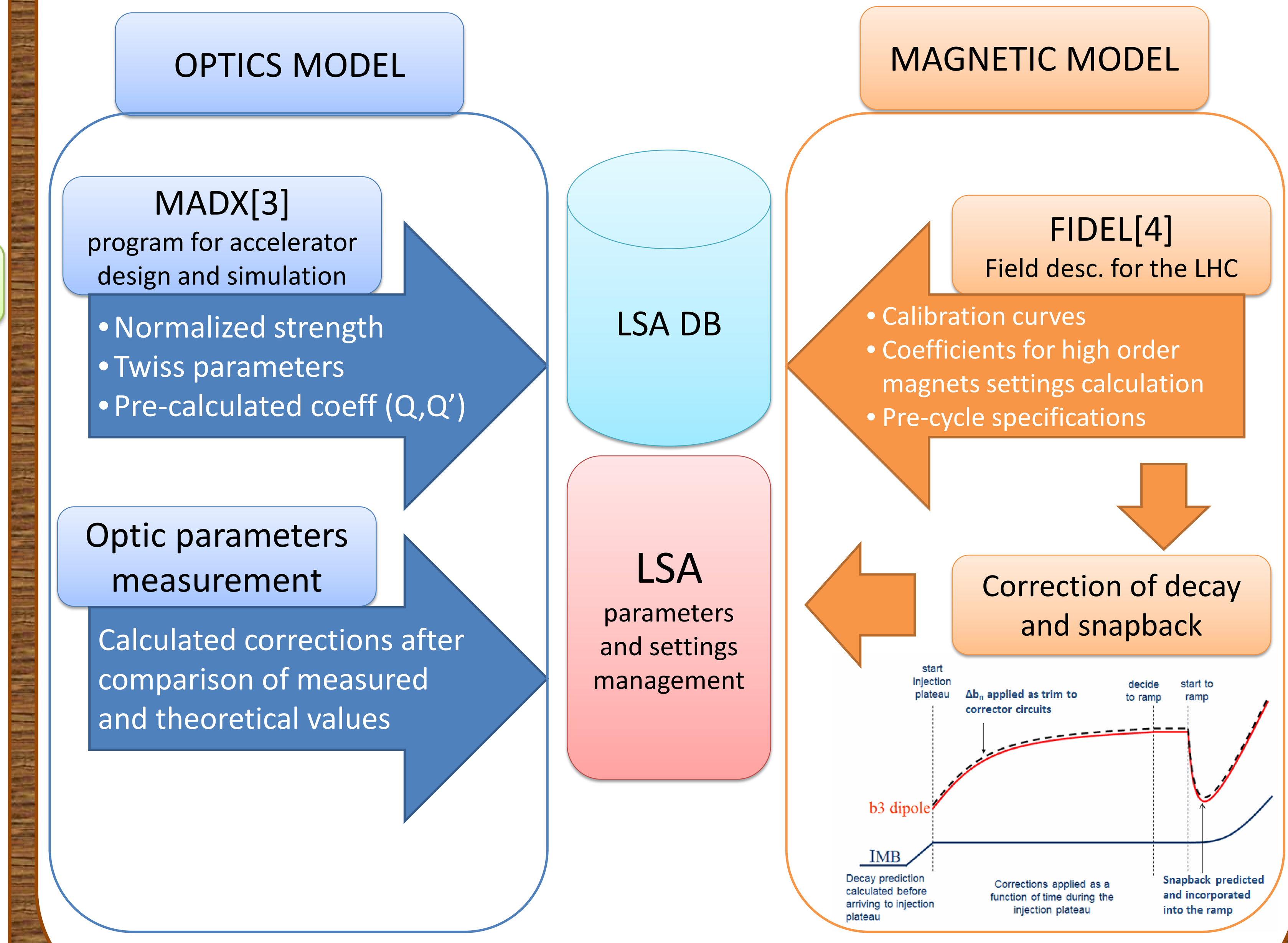
LSA ensure the **settings continuity** along the cycle:

Whenever a scalar value is modified for a given parameter at phases like injection and flat top, LSA incorporates the new setting into the function of the following phase

LSA also manages the **asynchronous parameters settings:**

For parameter whose values are independent of the cycle phase

THE LHC MODELS



CONCLUSIONS

Universality

- Basic principles common to all accelerators
- Used for cycling machine and colliders
- Deployed in all CERN accelerators

Completeness

- Covers every aspect of accelerator control, including optics and magnetic models
- Used for the control of all equipment (RF, collimators, power converters…)

Reliability

- 24/24 hours, 7/7 days, no downtime
- Strict release policy
- Intensive testing

LSA PERFORMANCE

Flexibility

- Modular architecture
- Adaptable settings management functionalities
- Support many types of parameters: cycle dependant or constant, standalone or in complex hierarchies

Reproducibility

- Good magnetic model for static and dynamic corrections of errors
- Systematic pre-cycle
- Real time corrections[5] & feed-forward[6]

REFERENCES

[1] G. Kruk, S. Deghaye, M. Lamont, M. Misiowiec, W. Sliwinski "LHC Software Architecture[LSA] – Evolution toward LHC beam commissioning", ICALEPCS'07, Knoxville, TN, USA.
[2] C.Roderick, R.Billen "The LSA database to drive the accelerator settings", CERN-ATS-2009-100, ICALEPCS'09, Kobe, Japan.
[3] H.Grote, F.Smidt "MAD-X An upgrade from MAD8", PAC'03, Portland, OR, USA
[4] N. Sammut "Magnet Modelling", LTC extended meeting, Mar 2008, CERN, Geneva, Switzerland.
[5] R. Steinhagen "Real time control at the LHC", PAC'11, New York, NY, USA
[6] M. Pereira, X. Buffat, K. Fuchsberger, M. Lamont, G.J. Müller, S. Redaelli, R.J. Steinhagen, J. Wenninger, "Feed forward in the LHC", ICALEPCS'11, Grenoble, France