Software for operating the magnet protection systems

GSI - CERN workshop

Marc-Antoine Galilée
TE-MPE-MS

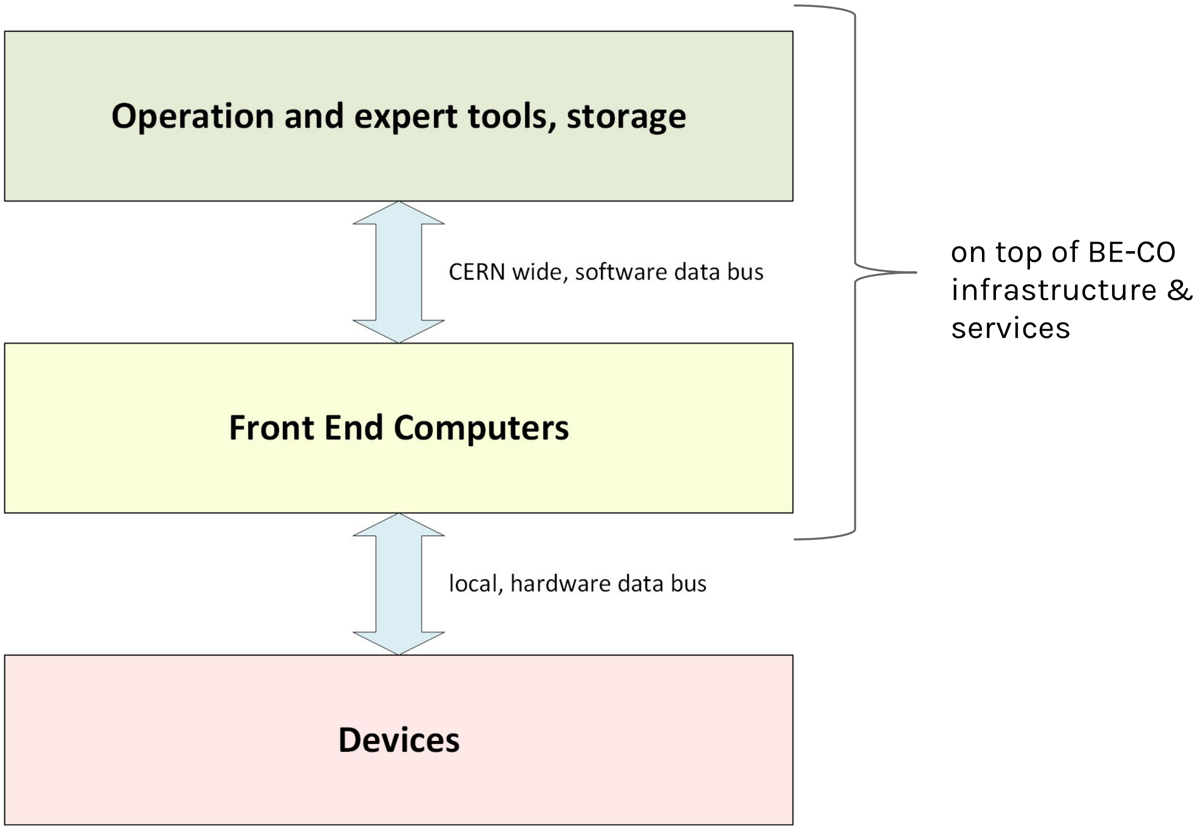
Overview

- Software stack & architecture
- Tools for operators and experts
- Collection of data
- Known shortcomings & planned evolutions

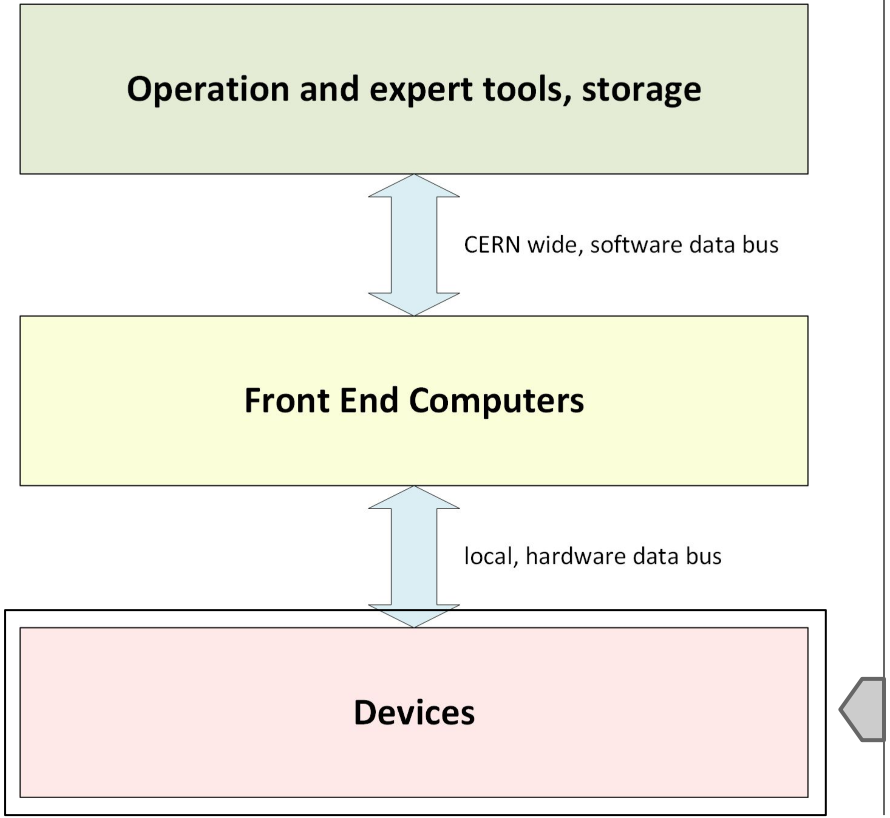
Overview

- Software stack & architecture
- Tools for operators and experts
- Collection of data
- Known shortcomings & planned evolutions

Software stack

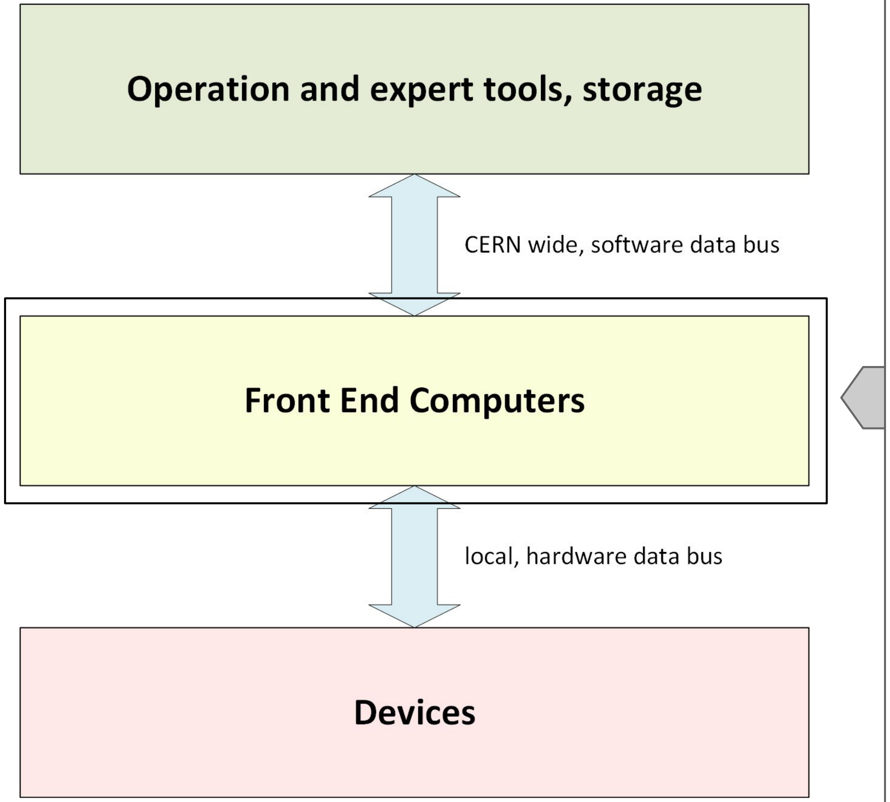


Software stack - Devices



Software
- **firmware & drivers (VHDL, C, ...)**
- hardware development and validation tools

Software stack - Front End Computers (FECs)



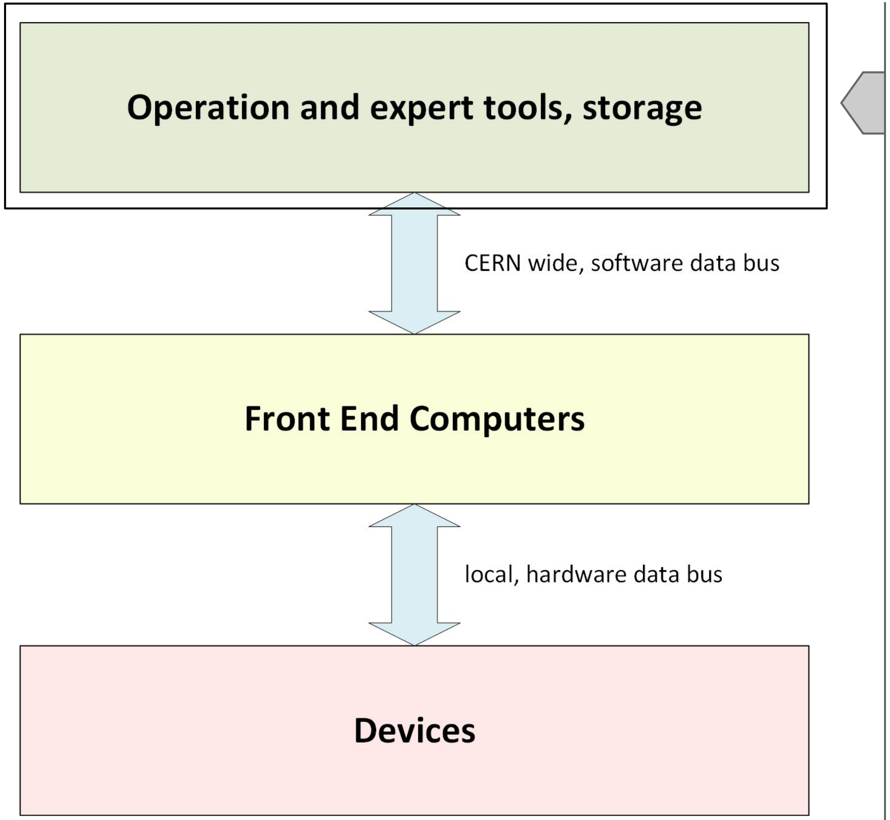
Software

- **C++**
- **Real Time**
- ‘Thin’ layer to abstract over the device(s)

The **‘FESA’ (Front End Software Architecture)** framework

- CERN standard
- authentication
- deployment
- configuration
- software data bus

Software stack - Tools & storage



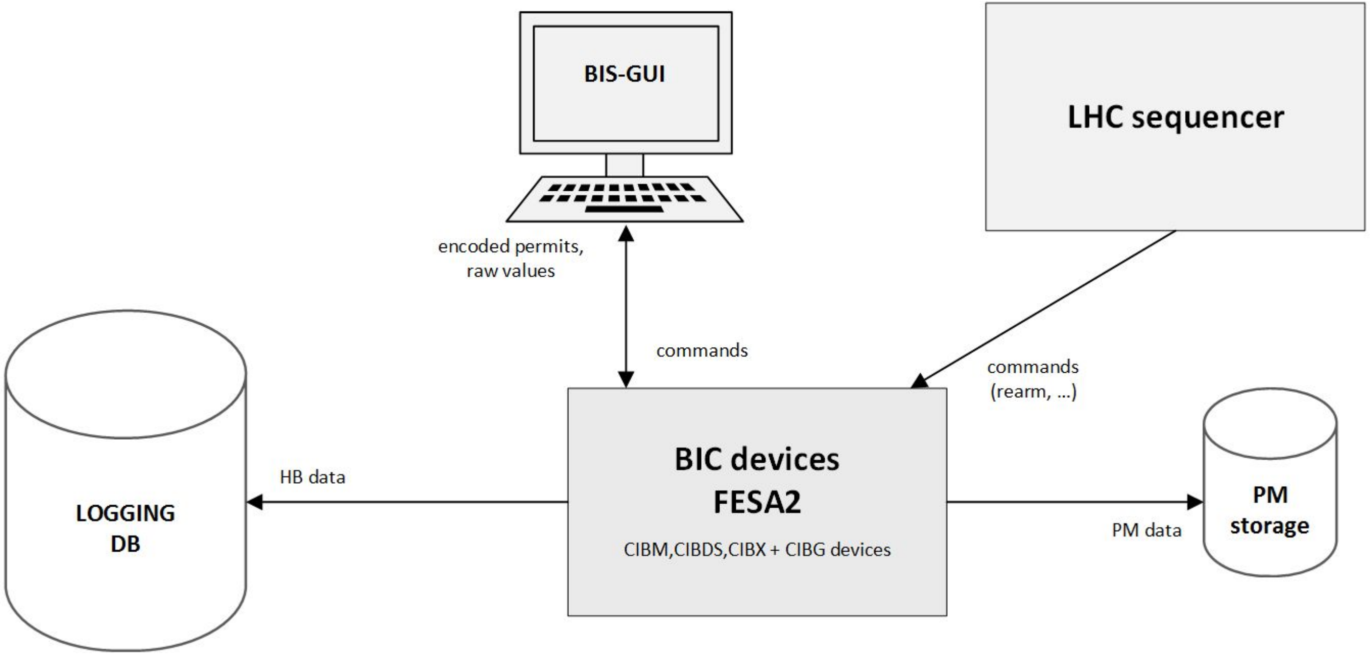
Software

- **Java**
- **Labview**
- **WinCC OA**
- and more
- various DBs and storages

Usages

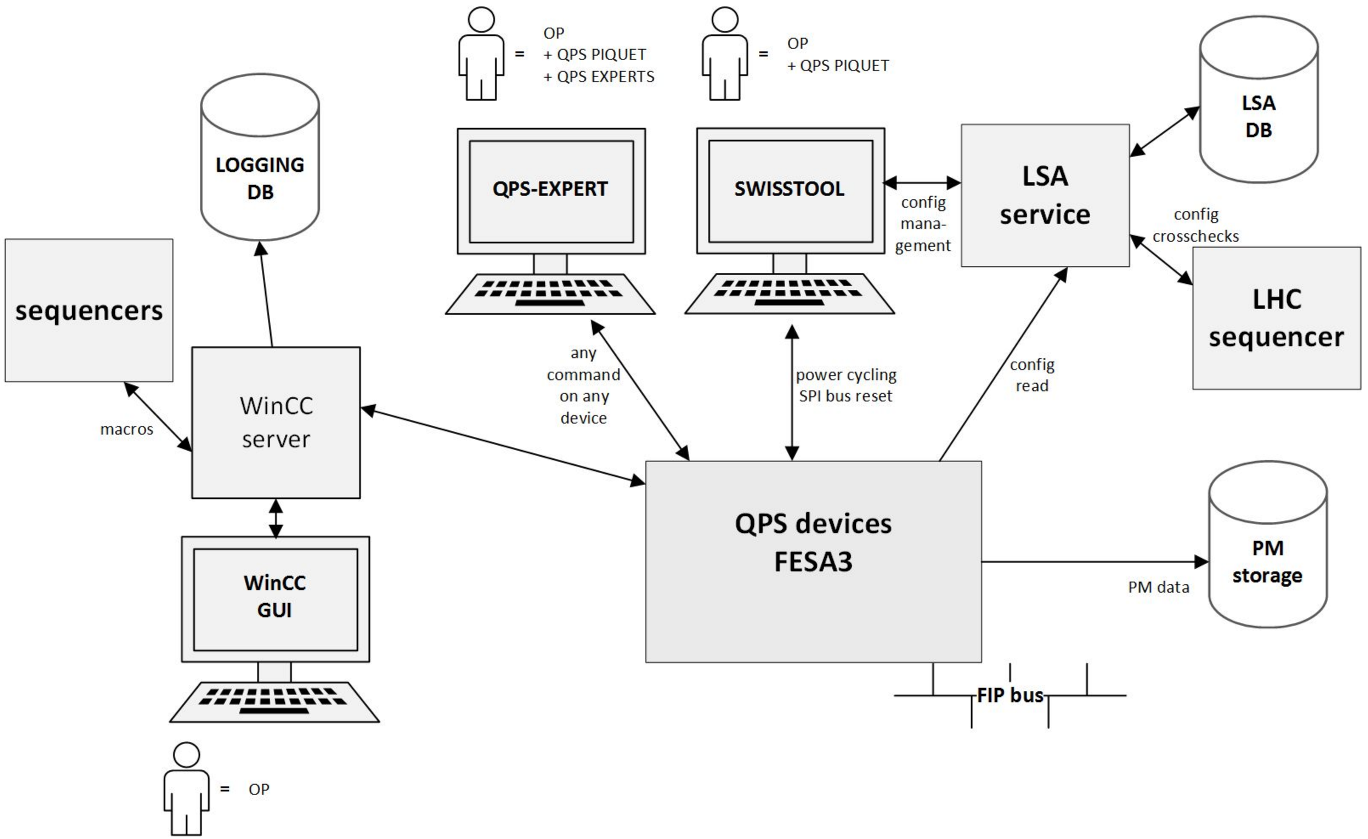
- Operation
- Analysis
- Commissioning
- Development
- Diagnostics

Example: Beam Interlock System architecture



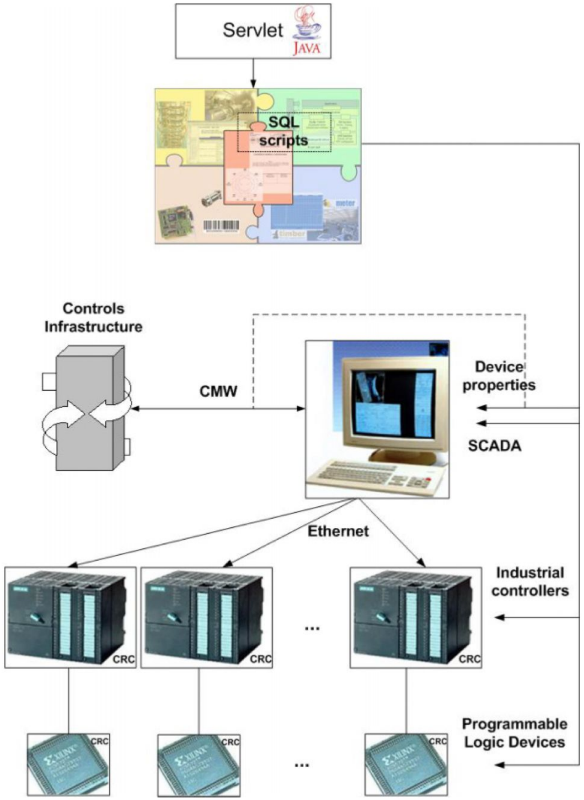
(not a magnet protection system per se, provided here as an example of a lower complexity system following a very similar architecture)

Example: QPS architecture



Example: Powering Interlock System architecture

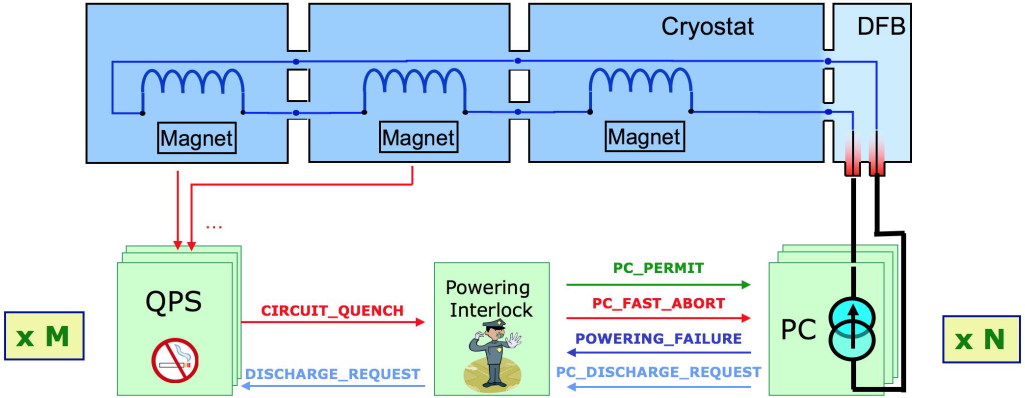
- Example of another flavour of software architecture & technologies
- PLC based system, managing all (boolean) interlock conditions between QDS, converter, cryogenics, ancillaries, … to assure safe operation of magnet powering system
- 36 individual systems around the circumference, using Profibus to connect remote I/Os
- Synchronised via NTP



Tools for operators and experts

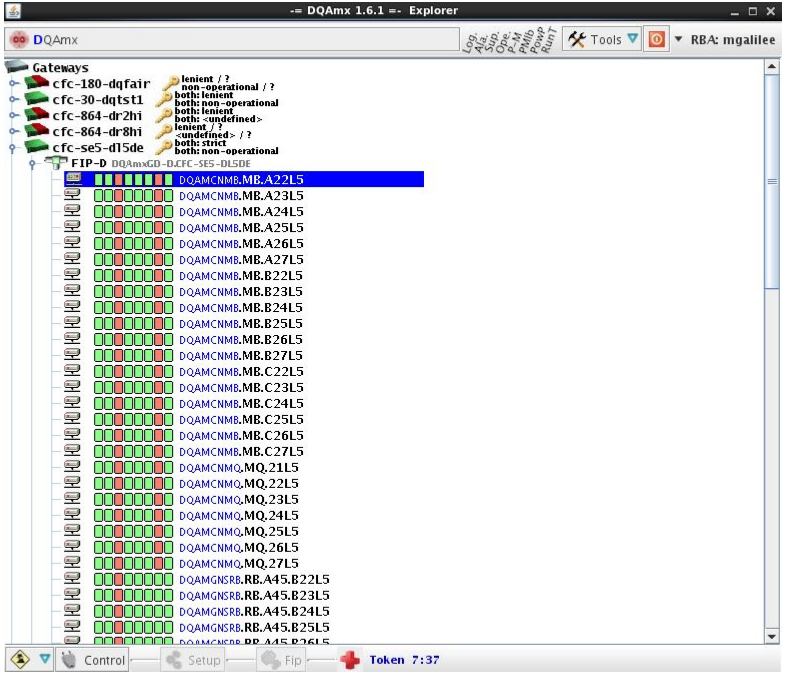
- Software stack & architecture
- Tools for operators and experts
- Collection of data
- Known shortcomings & planned evolutions

Tools for operators and experts



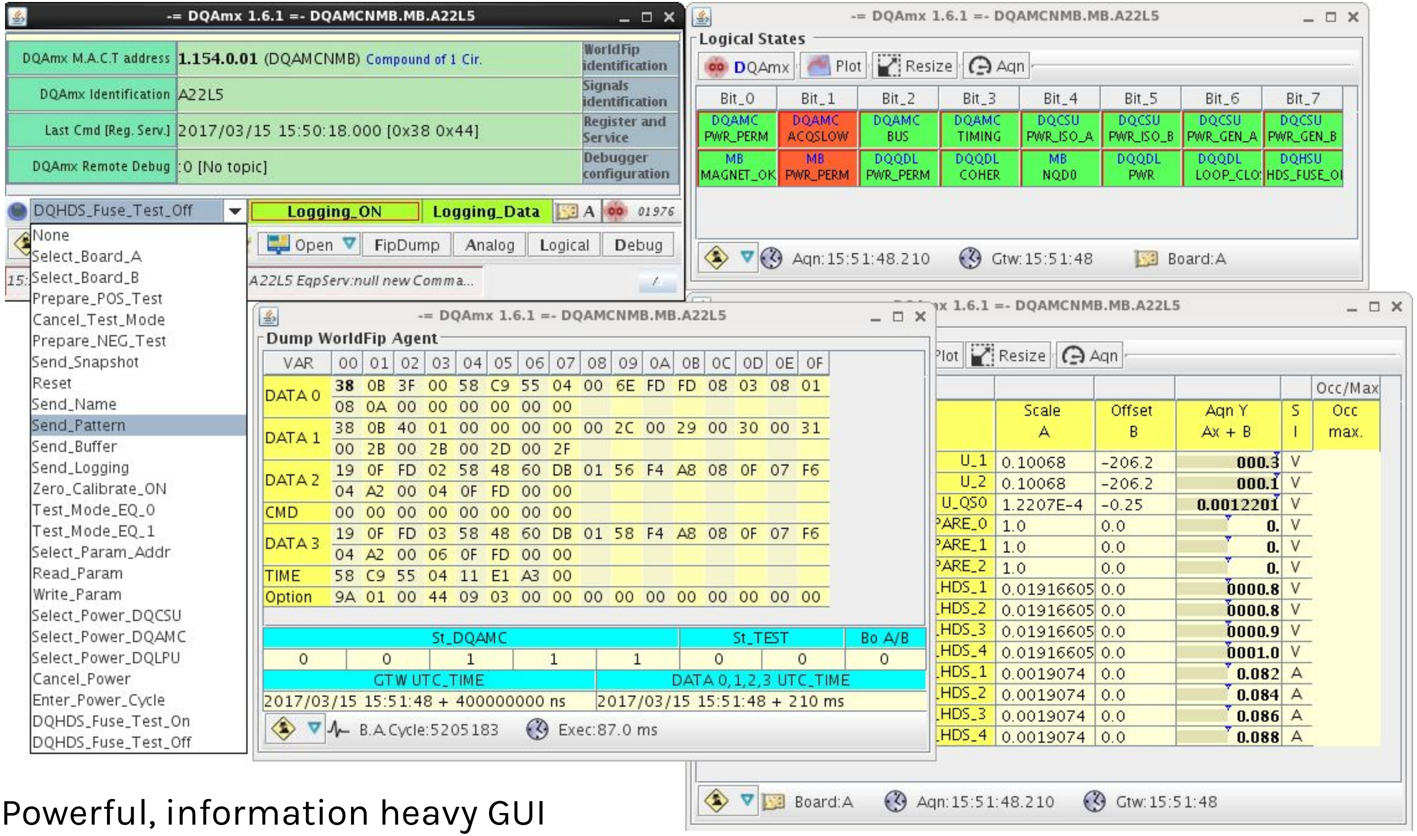
- Magnet protection system (for the LHC) relies on interaction of quench detection system, power converter and powering interlock system (+ ancillaries such as CRYO, emergency stops, UPS, access, ...)
- Systems use (very) different architectures and technologies
- Expert tools (system by system) are important for initial commissioning, development, maintenance, …
- Operational tools focus on global system view (and provide limited functionality)
- For the LHC a set of rather heterogeneous tools is used for operation of the magnet powering system

Tools: QPS Expert, overview



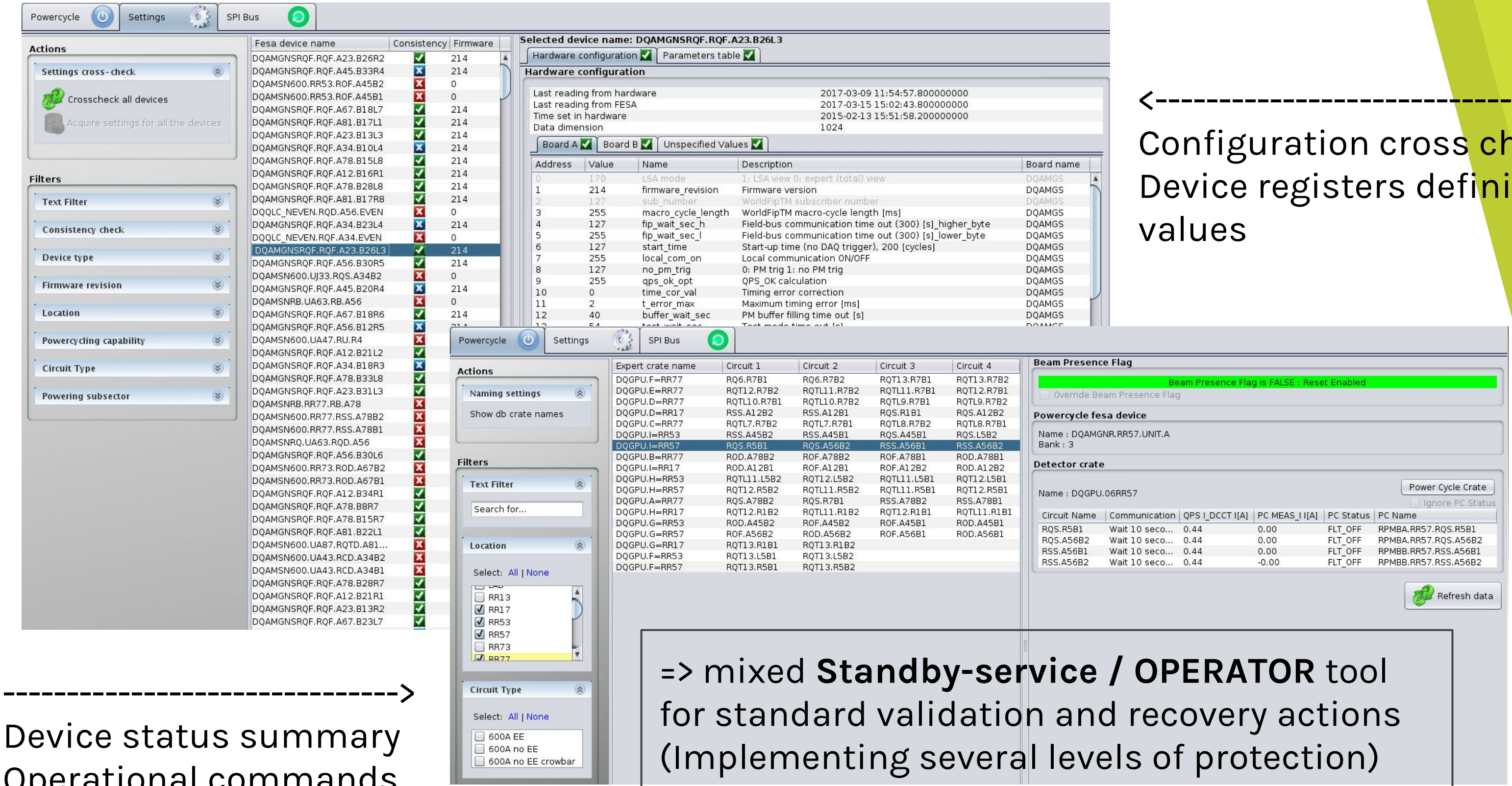
- Expert tool, providing overview of all devices and allowing necessary (expert) interactions with devices
- One level of granularity: a single protection device
- Typically such applications provide little operational context or interactions with connected equipment systems

Tools: QPS Expert, device GUIs



Powerful, information heavy GUI
=> ~microscope equivalent => pure **Expert** tool

Tools: QPS Swisstool



<-------------------------------------

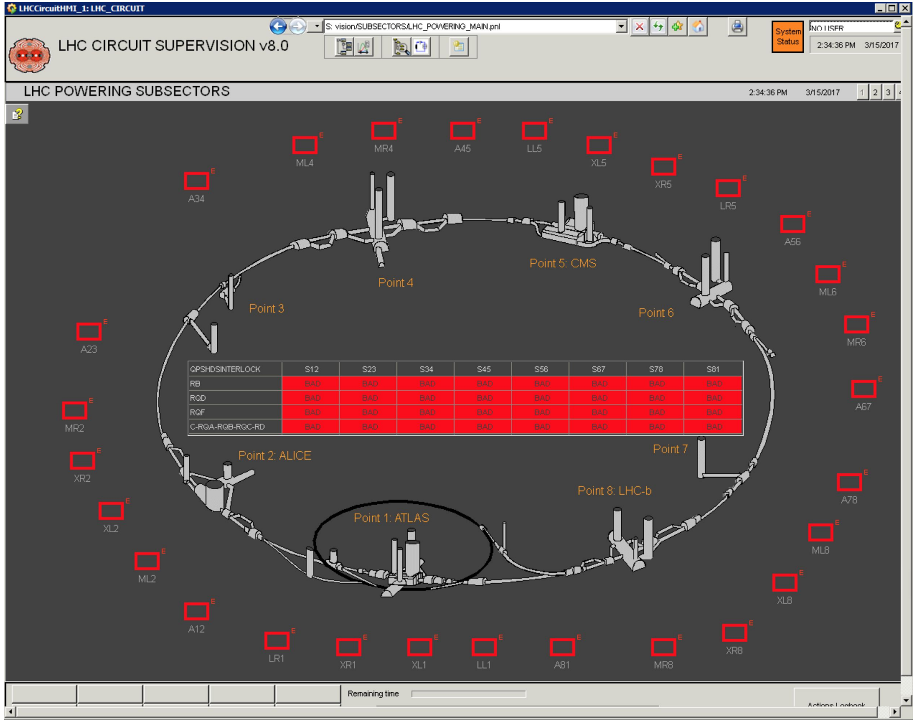
Configuration cross checking
Device registers definitions + values

-------------------------------->

Device status summary
Operational commands
Beam context

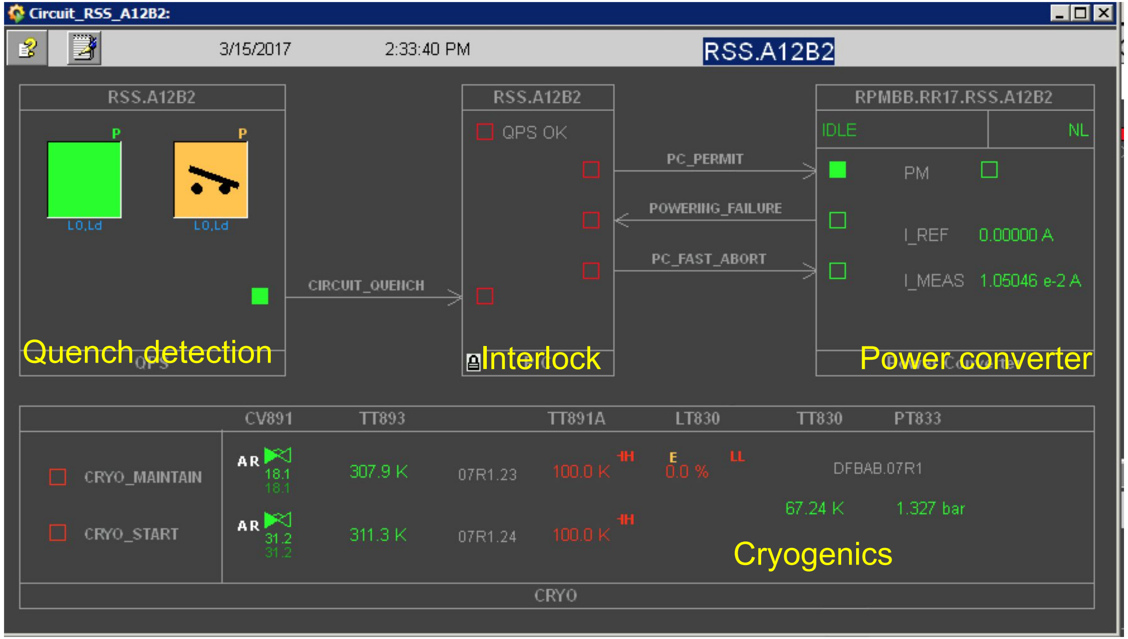
=> mixed **Standby-service / OPERATOR** tool for standard validation and recovery actions (Implementing several levels of protection)

Tools: LHC circuit supervision



- Operational tool, intended to provide an LHC wide synthetic view for the >1600 magnet circuits
- Main aim is to quickly find and identify the root cause of events
- First level of granularity: sector (then circuits, powering/protection systems and finally individual electronic cards)

Tools: circuit supervision



- Second level of granularity: the magnet circuit
- Comprehensive circuit view, includes devices and ancillaries
- Allowing for easy identification of root cause

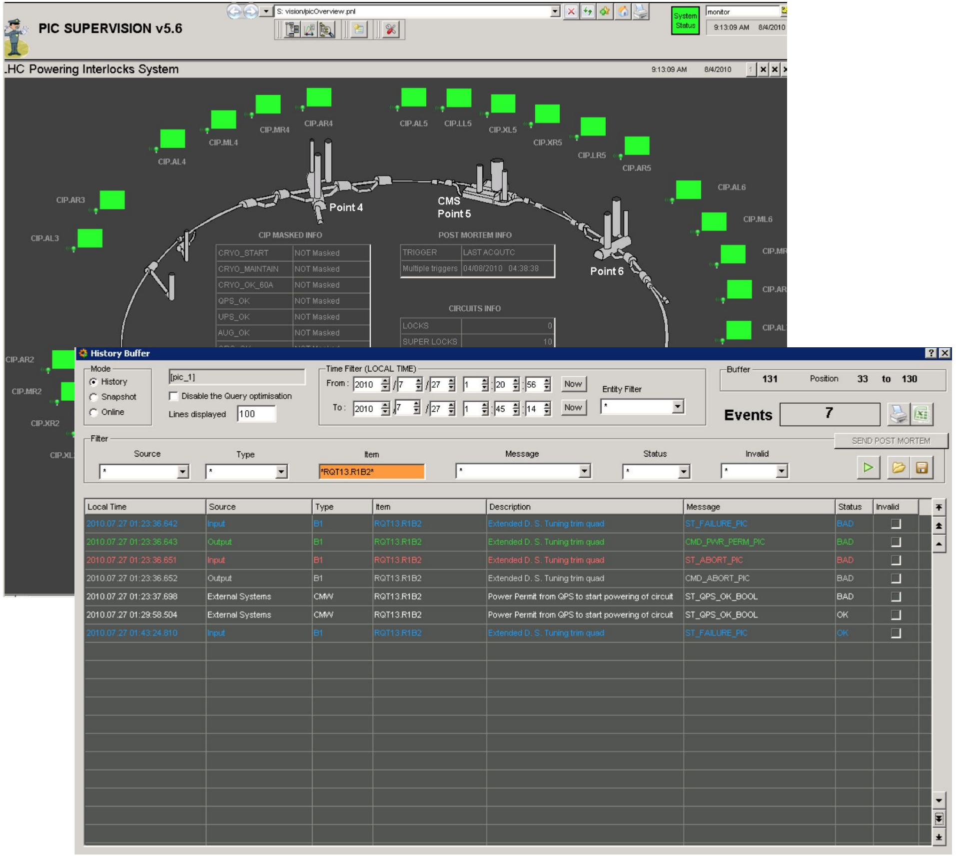
Tools: circuit supervision, into the device

- Third level of granularity: the protection device
- Detailed device status, operational commands
- Display and verify configuration parameters

Overview

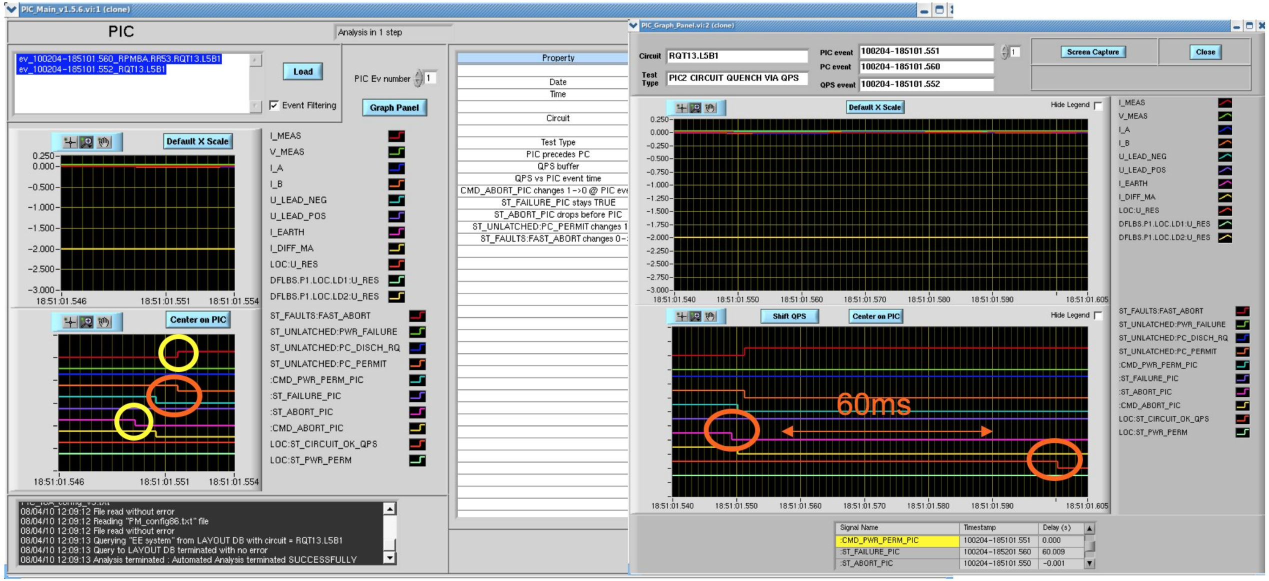
- Software stack & architecture
- Tools for operators and experts
- **Collection of data**
- Known shortcomings & planned evolutions

Collection of data



- Interlock systems play a central role in event analysis as signals from many systems are connected (and logged) by such systems
- Accurate interlock history buffers are in the LHC the sole key to reliably resolve the event sequence for powering events
- Timing accuracy of <1ms required

Collection of data

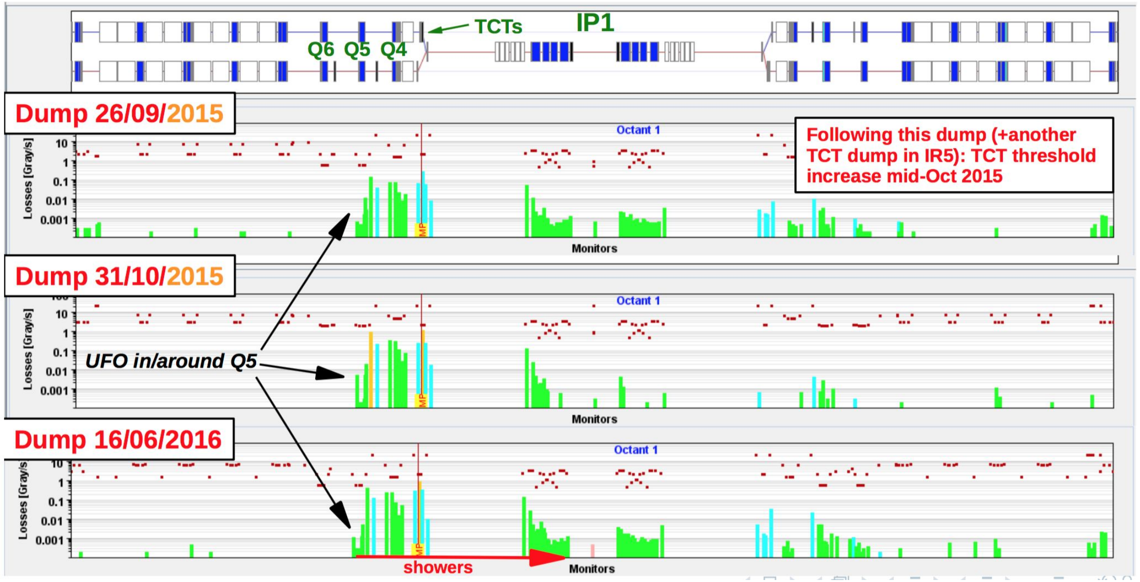


- Due to differences in timestamping source and mechanisms (local vs remote/gateway timestamping, central timing vs NTP…), identical signals can get timestamped with insufficient accuracy

Collection of data

- Accurate data collection not only important within magnet powering system, but certain events require correlation with (even more accurate) beam diagnostic data

Collection of data



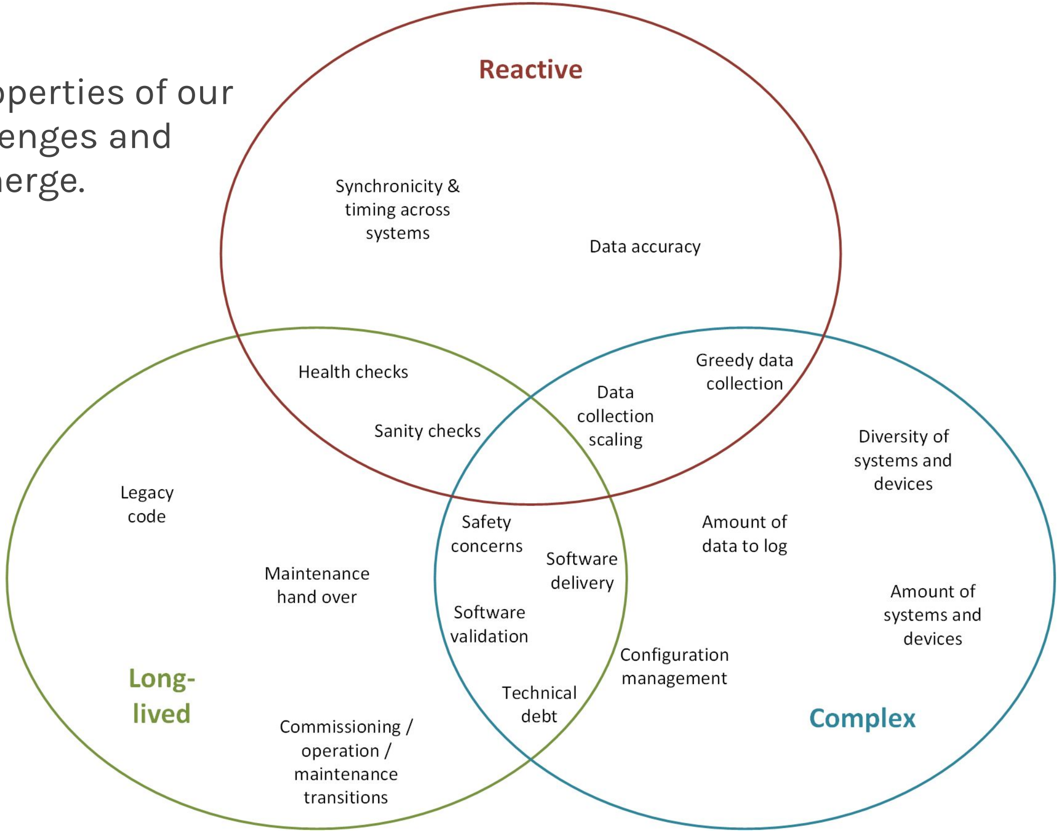
- Typical LHC example: Losses from (dust) objects interacting with beam - did the beam losses trigger the quench or did the quench trigger the beam losses?
  For a deeper look, see presentation by Zinur later this morning

Overview

- Software stack & architecture
- Tools for operators and experts
- Collection of data
- Known shortcomings & planned evolutions

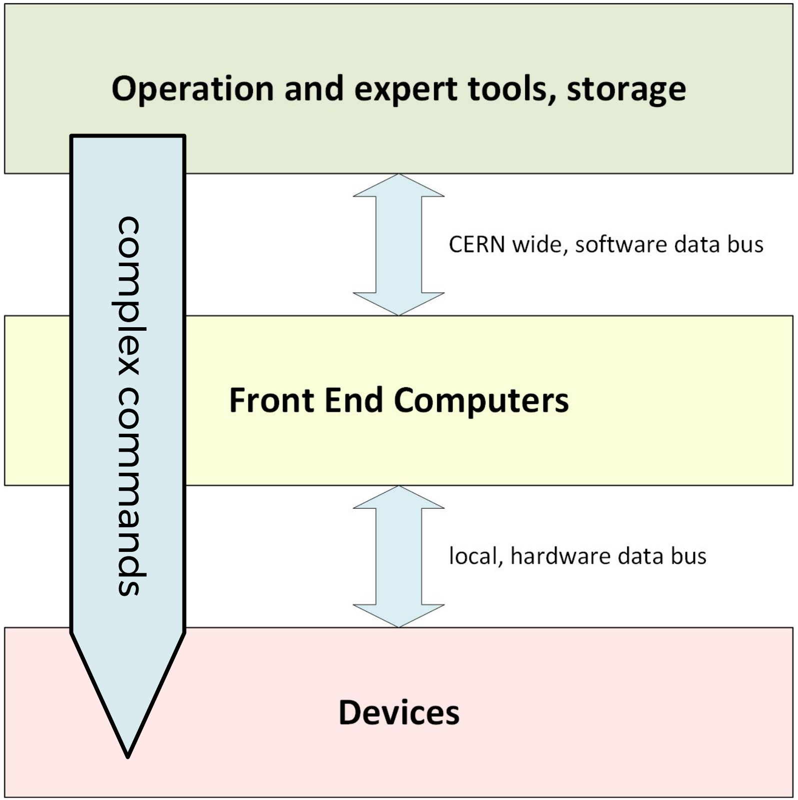
Shortcomings & challenges

From 3 inherent properties of our system, many challenges and potential issues emerge.



Note: non exhaustive, non quantitative representation

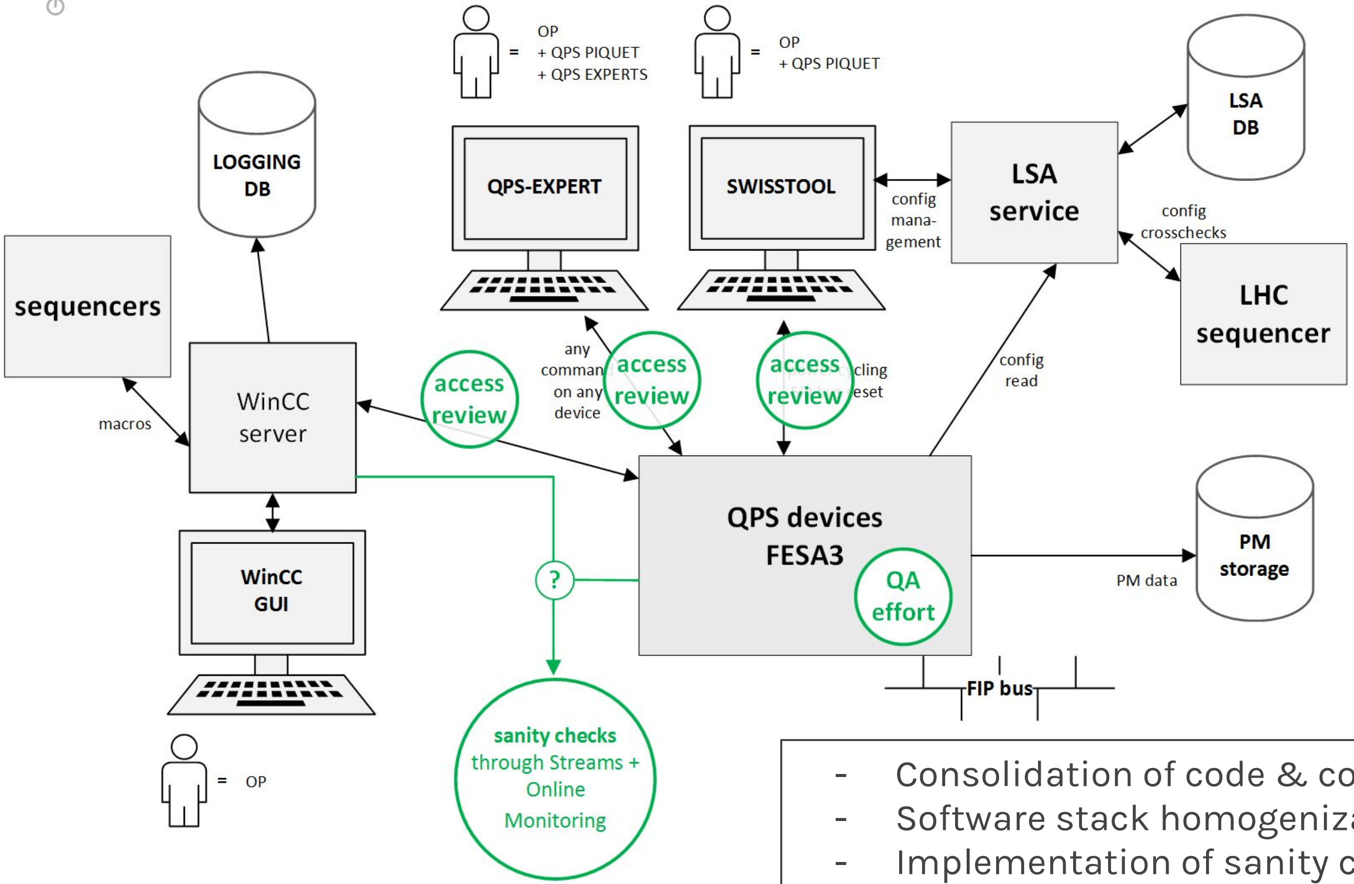
Example - Commissioning / Operation transition



Low-level control,
from **high-level** applications

- Crosses all abstraction layers
- *Desirable* during commissioning
- *Risky* during operation
- Requires cautious handling when developing, configuring and delivering software
- Difficult to get around once in place

Example: QPS planned evolutions



- Consolidation of code & configuration
- Software stack homogenization
- Implementation of sanity checks

Thank you

Questions?