

WRAP is a GUI creation & hosting platform, fully integrated with the Controls System configuration

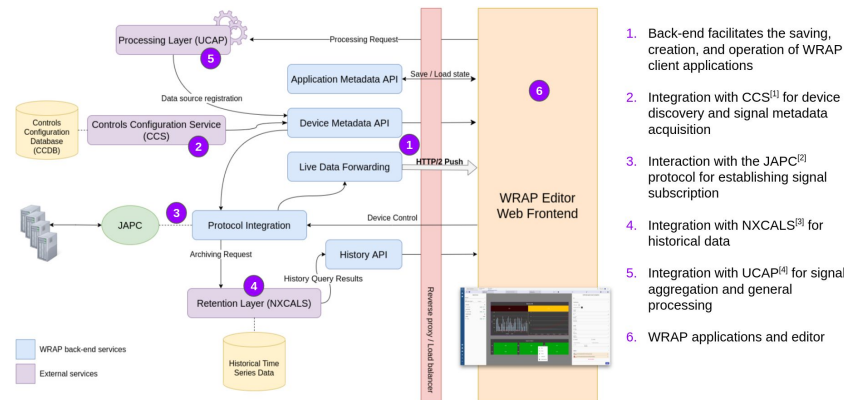
Key aims:

- Replace 100's standalone expert applications (Java SWING) with data-driven GUIs
- Shield equipment experts (*not software developers*) from inevitable GUI technology evolution

Status: transitioning from an advanced Proof-of-Concept to an Operational Platform

Targeted Qualities & Behaviour:

- **Ease of use:** Small learning curve, no-code, drag-n-drop configuration, minimal training
- **Performance:** Support for many parallel real time visualizations
- **Extensibility:** 3rd party component support, protocol-agnostic API
- **Maintainability:** High test coverage, minimal technical debt.
- **Stability:** No breaking changes for existing applications, forward compatibility, fault tolerance guarantees
- **Integrated Metadata:** Structured data source configurations, automatically adapting to underlying



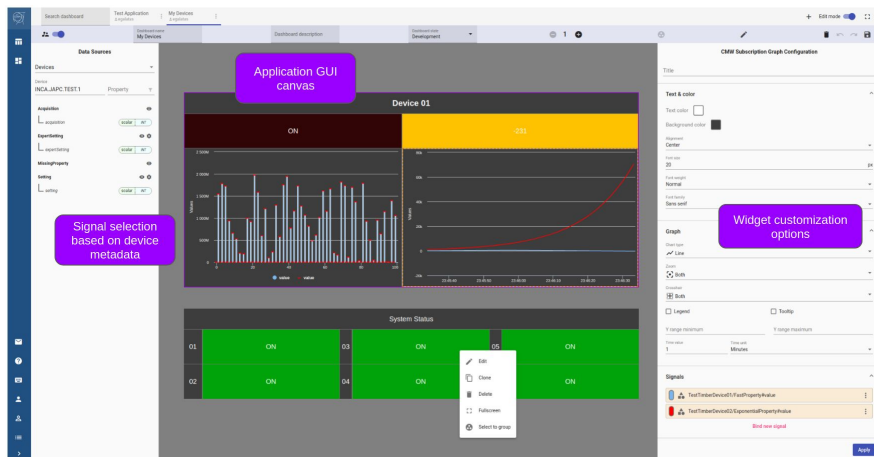
[3] L. Bartuszek et al., "CERN Controls Configuration Service - A challenge in usability", in proc. ICALEPCS27

[2] V. Bagdasarian et al., "JAPC - the Java API for Parameter Control", in proc. ICALEPCS16

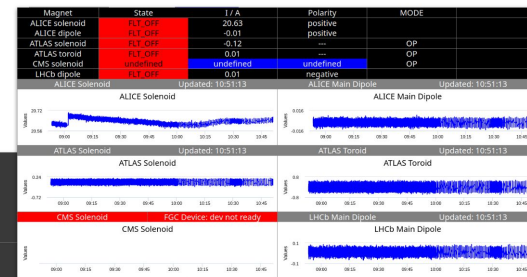
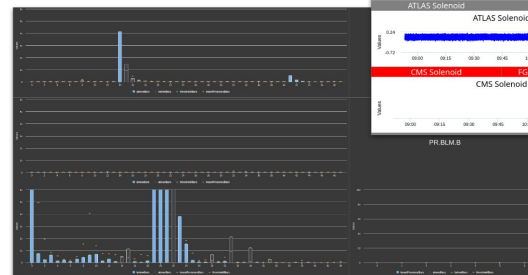
[8] J. Wozniak et al., "NXCALLS - Architecture and Challenges of the Next CERN Accelerator Logging Service", in proc. ICALEPCS19

[4] L. Cooper et al., "UCAP - A Framework for Accelerator Controls Data Processing at CERN", in proc. ICALEPCS23

1. Back-end facilitates the saving, creation, and operation of WRAP client applications
2. Integration with CCS^[1] for device discovery and signal metadata acquisition
3. Interaction with the JAPC^[2] protocol for establishing signal subscription
4. Integration with NXCALLS^[3] for historical data
5. Integration with UCAP^[4] for signal aggregation and general processing
6. WRAP applications and editor



Examples of first end-user applications created using the WRAP editor



LHC Experiment fixed display (top) and PS BLM fixed display (left)

WRAP is a GUI creation & hosting platform, fully integrated with the Controls System configuration

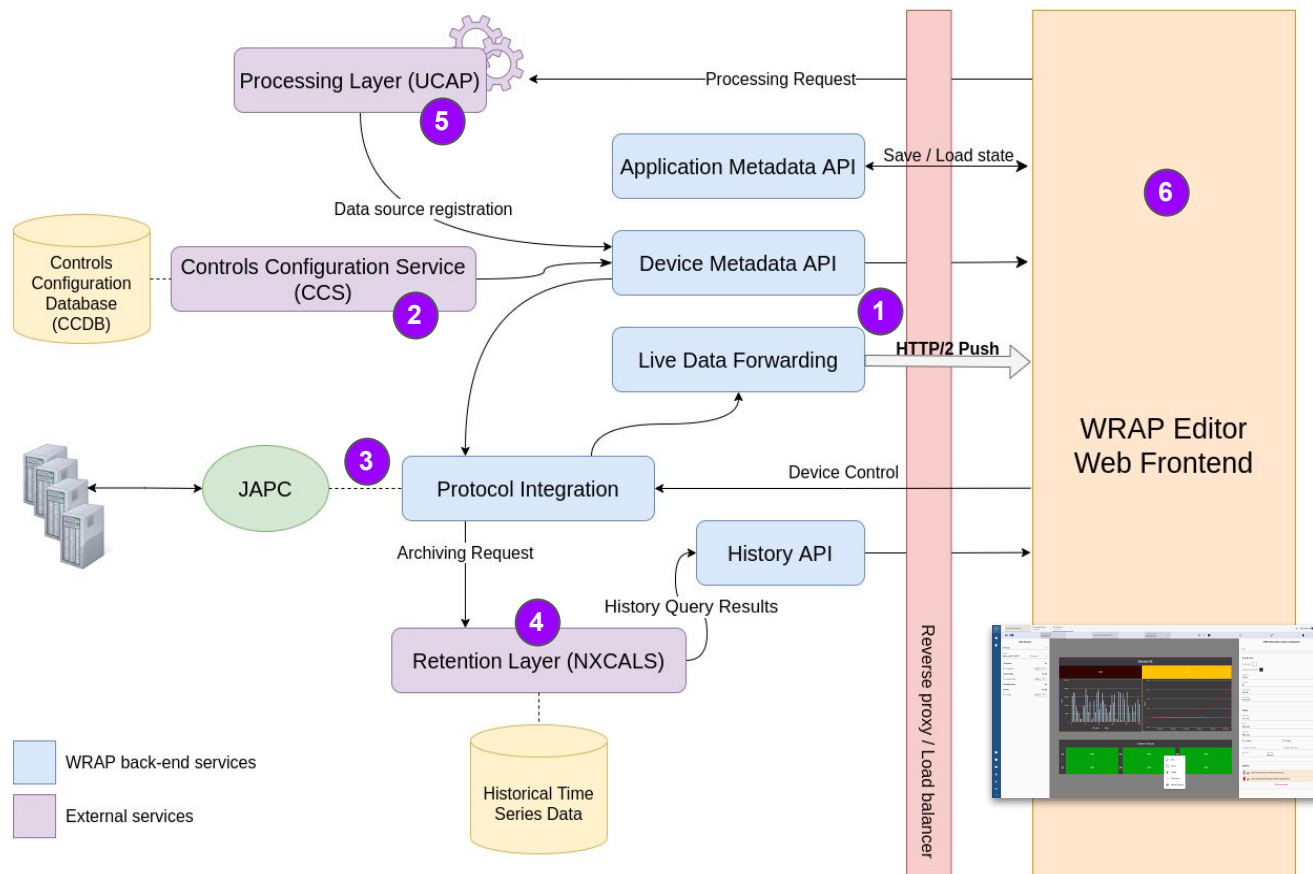
Key aims:

- Replace 100's standalone expert applications (Java SWING) with data-driven GUIs
- Shield equipment experts (*not software developers*) from inevitable GUI technology evolution

Status: transitioning from an advanced Proof-of-Concept to an Operational Platform

Targeted Qualities & Behaviour:

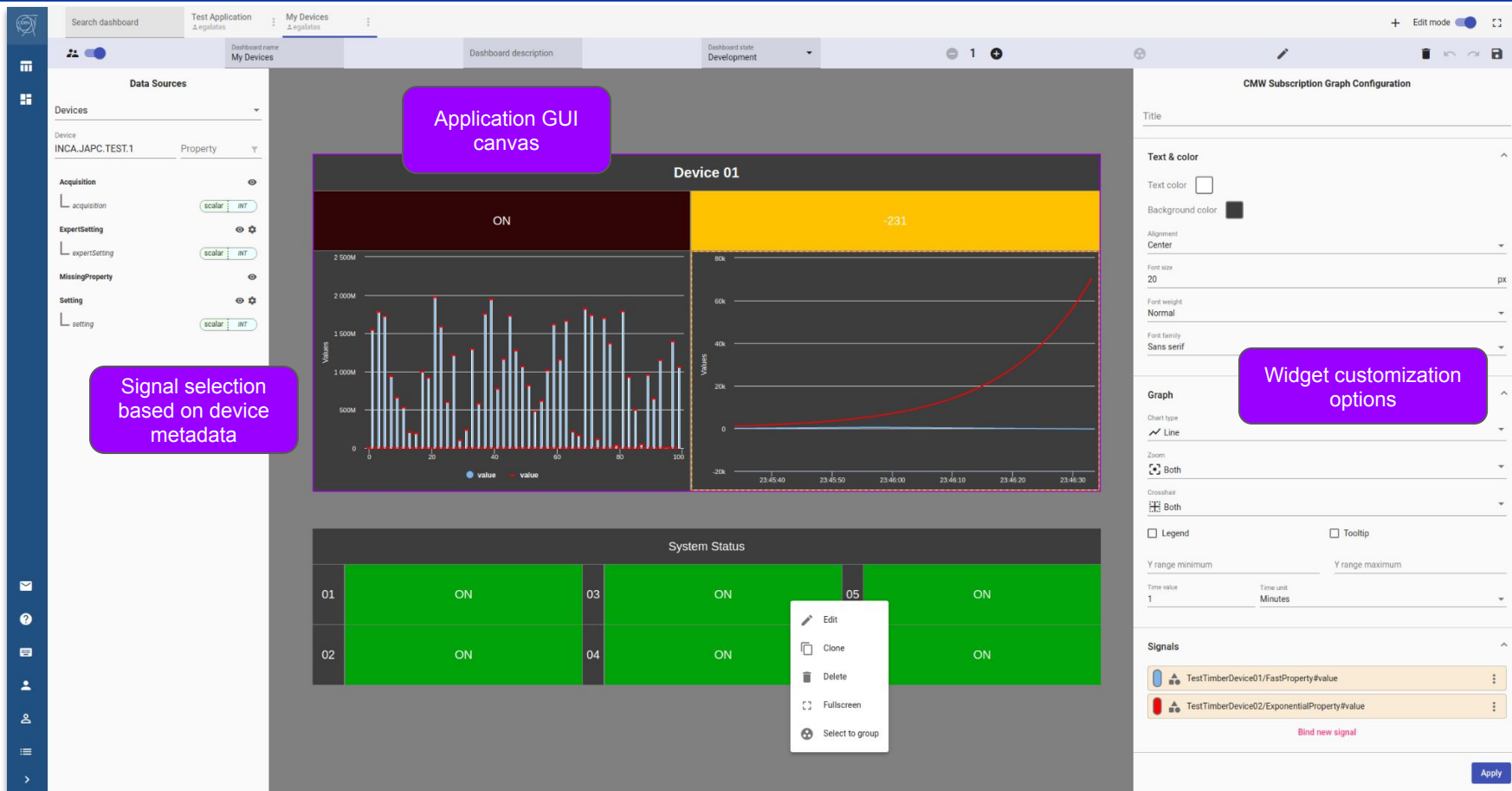
- **Ease of use:** Small learning curve, no-code, drag-n-drop configuration, minimal training
- **Performance:** Support for many parallel real time visualizations
- **Extensibility:** 3rd party component support, protocol-agnostic API
- **Maintainability:** High test coverage, minimal technical debt.
- **Stability:** No breaking changes for existing applications, forward compatibility, fault tolerance guarantees
- **Integrated Metadata:** Structured data source configurations, automatically adapting to underlying



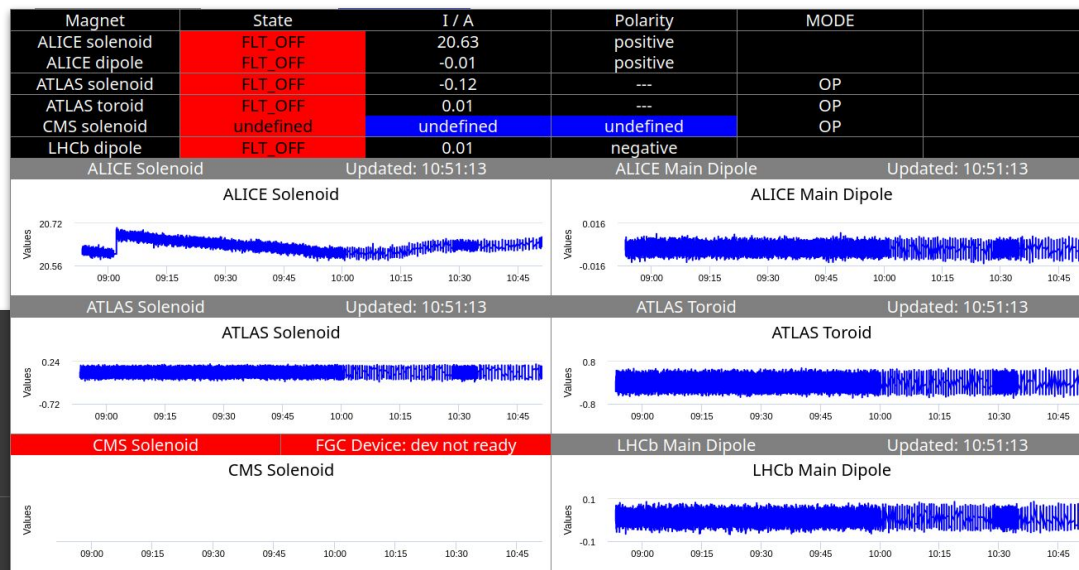
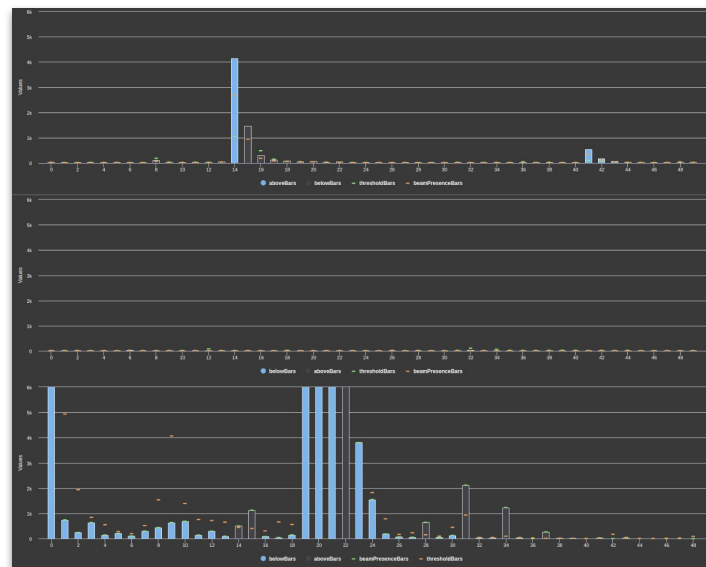
1. Back-end facilitates the saving, creation, and operation of WRAP client applications
2. Integration with CCS^[1] for device discovery and signal metadata acquisition
3. Interaction with the JAPC^[2] protocol for establishing signal subscription
4. Integration with NXCALs^[3] for historical data
5. Integration with UCAP^[4] for signal aggregation and general processing
6. WRAP applications and editor

[1] L. Burdzanowski et al., "CERN Controls Configuration Service – A challenge in usability", in proc. ICALEPCS'17
 [2] V. Baggiolini et al., "JAPC - the Java API for Parameter Control", in proc. ICALEPCS'05

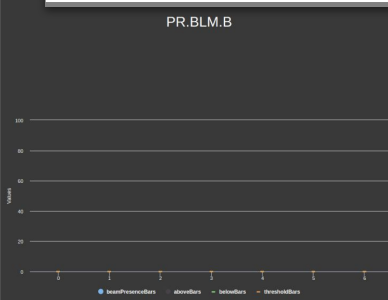
[3] J. Wozniak et al., "NXCALs - Architecture and Challenges of the Next CERN Accelerator Logging Service", in proc. ICALEPCS'19
 [4] L. Cseppentő et al., "UCAP: A Framework for Accelerator Controls Data Processing at CERN", in proc. ICALEPCS'21



Examples of first end-user applications created using the WRAP editor



PR.BLM.B



LHC Experiment fixed display (top)
and PS BLM fixed display (left)