Containerized EPICS-based framework for test setups toward the Silicon Tracking System in the Compressed Baryonic Matter experiment

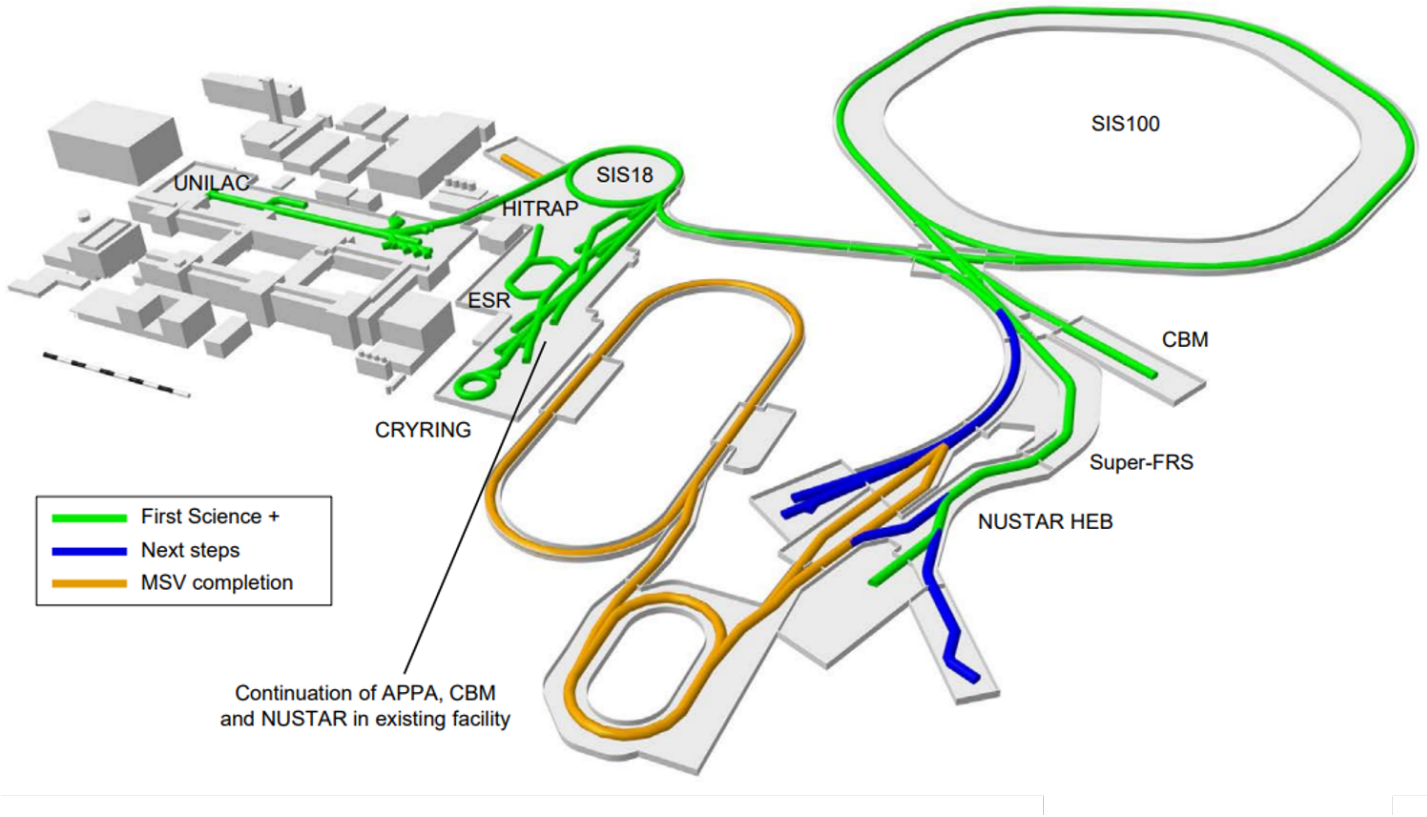
Marcel Bajdel[1,2], Peter Zumbruch[1], Philipp Klaus, Florian Feldbauer[3]

[1] GSI Helmholtzzentrum, Darmstadt, Germany
[2] Goethe University, Frankfurt, Germany
[3] Ruhr-Universität, Bochum, Germany

25.04.2023

Introduction to FAIR

In giant planets, stars, and during stellar explosions and collisions, the matter is subject to extreme conditions such as very high temperatures, pressures, and densities. FAIR will enable scientists to create such conditions in the laboratory.



The FAIR research is subdivided into the four experiment pillars:
NUSTAR, CBM, PANDA, APPA

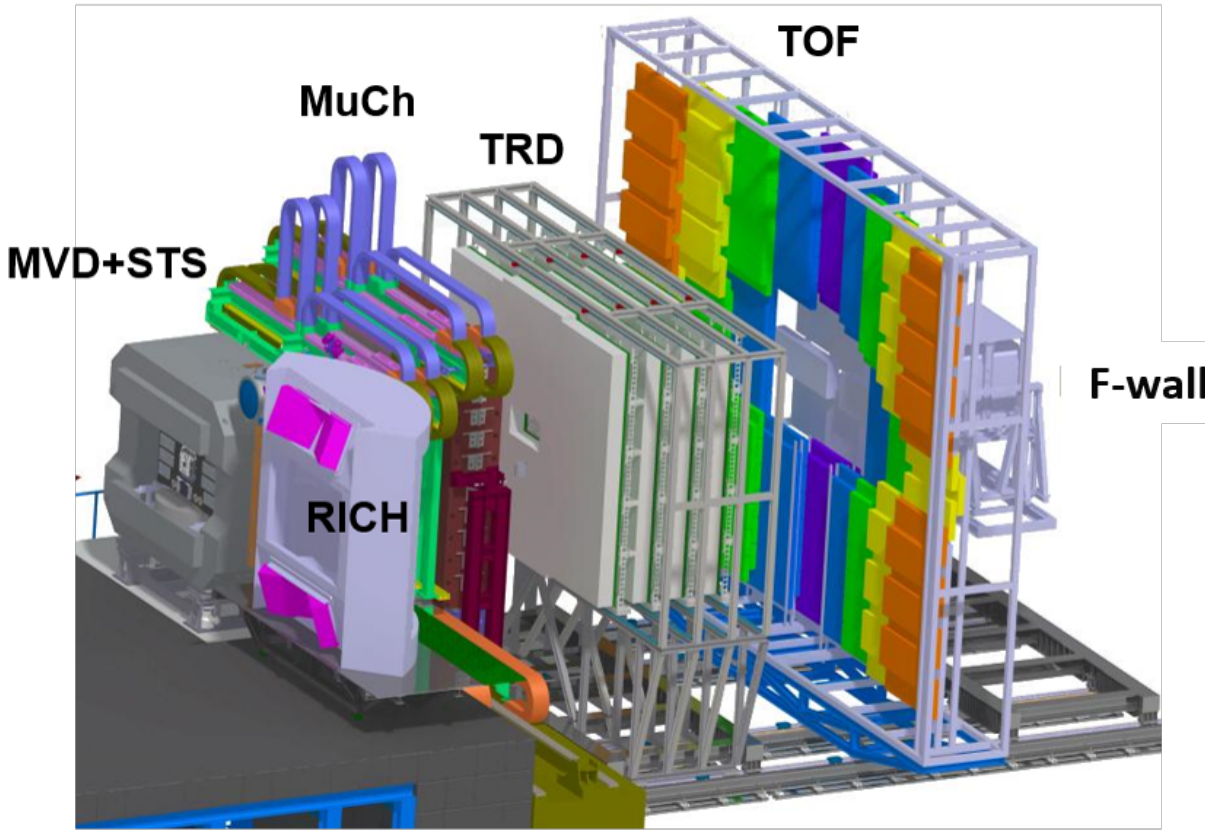
Introduction to Compressed Baryonic Matter (CBM) Experiment

Goals

To explore the QCD phase diagram at high net-baryon densities and moderate temperatures. High interaction rates ($10^5 - 10^7$Hz); ideal for systematic, high precision studies of high-density nuclear matter.

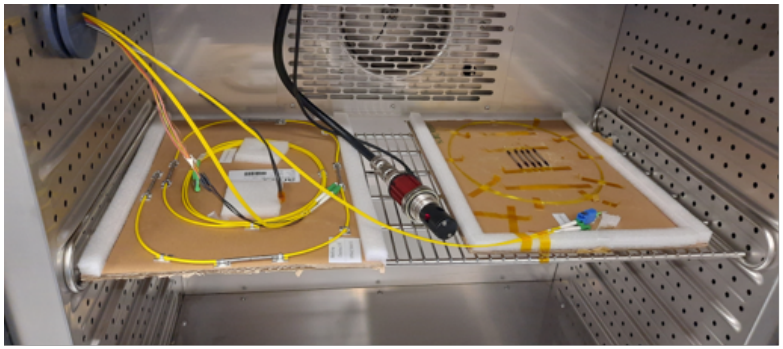
The main features of the experiment:

- fixed target experiment,
- free-streaming triggerless Data Acquisition (DAQ),
- online event reconstruction and selection,
- fast and radiation hard detectors,
- tracking acceptance: $2.5° < \theta_{lab} < 25°$,

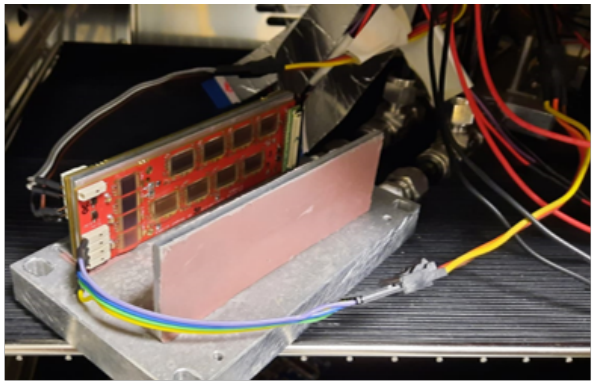


Control framework requirements

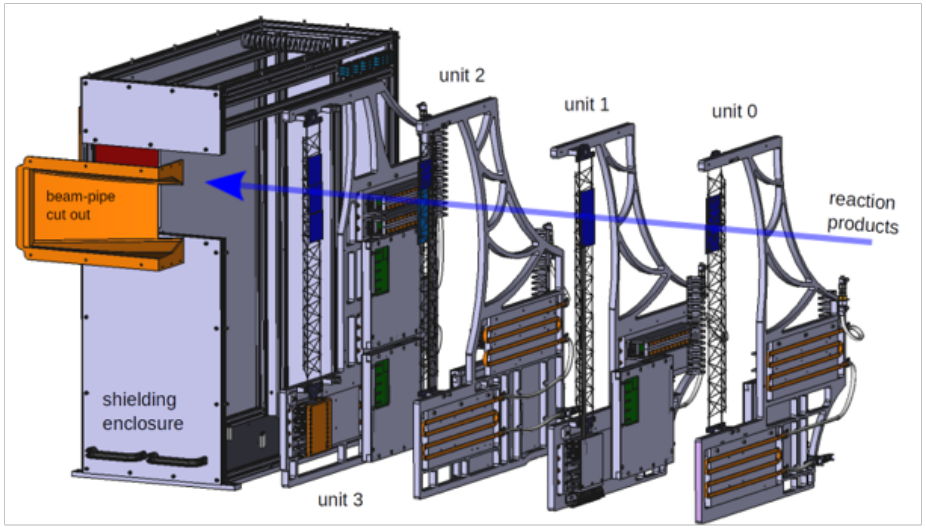
Requirements: Easy and fast to implement and maintain, cross-platform compatible (mostly Unix based machines), easily scalable EPICS/Phoebus related applications stack



RH + T Fiber Optic Sensors testing (100 PVs)



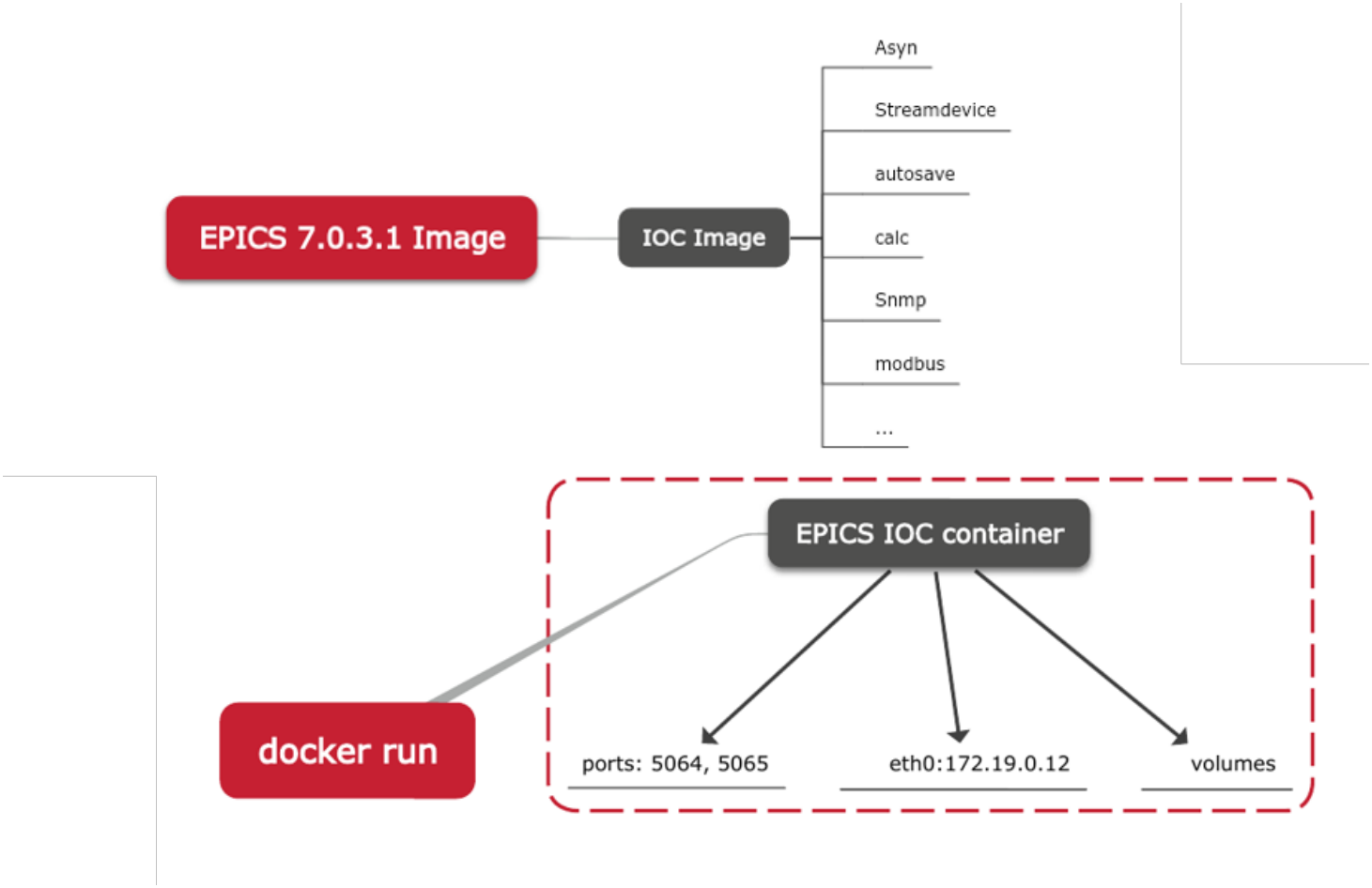
Readout and Thermal cycling of the Front-End Boards and detector modules (up to 1000 PVs)



The mSTS (up to 5000 PVs)

Architecture

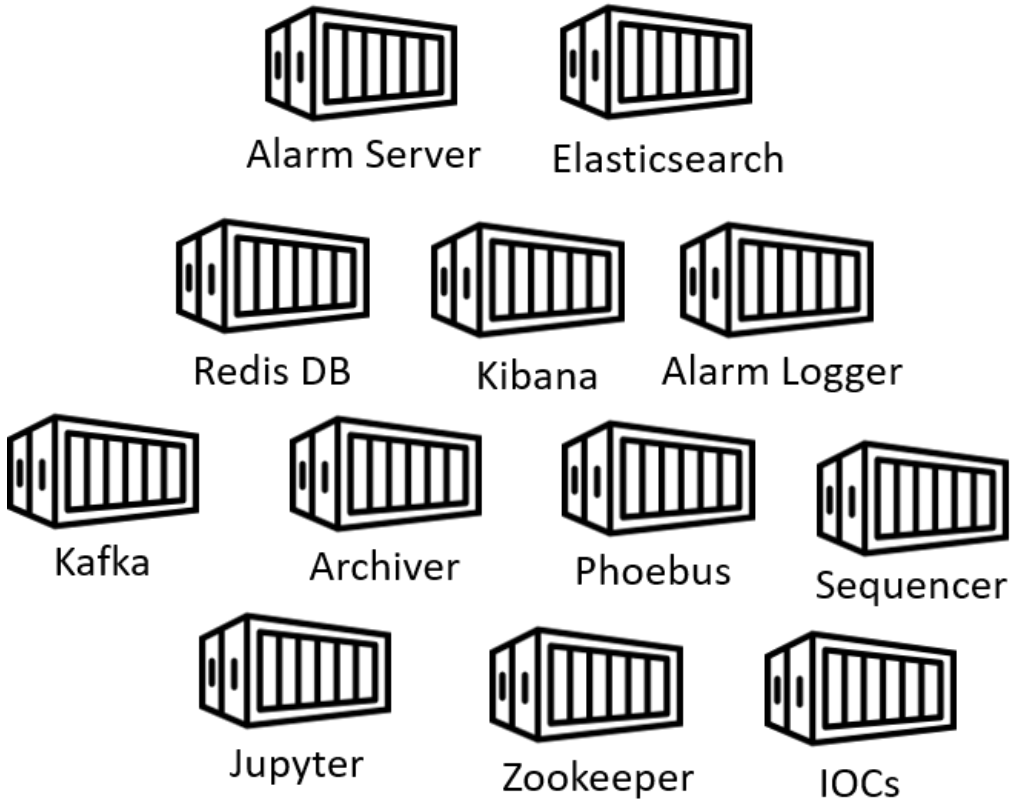
Idea Create images of the required applications
Requirements lightweight, containing only necessary dependencies and packages



Docker-compose

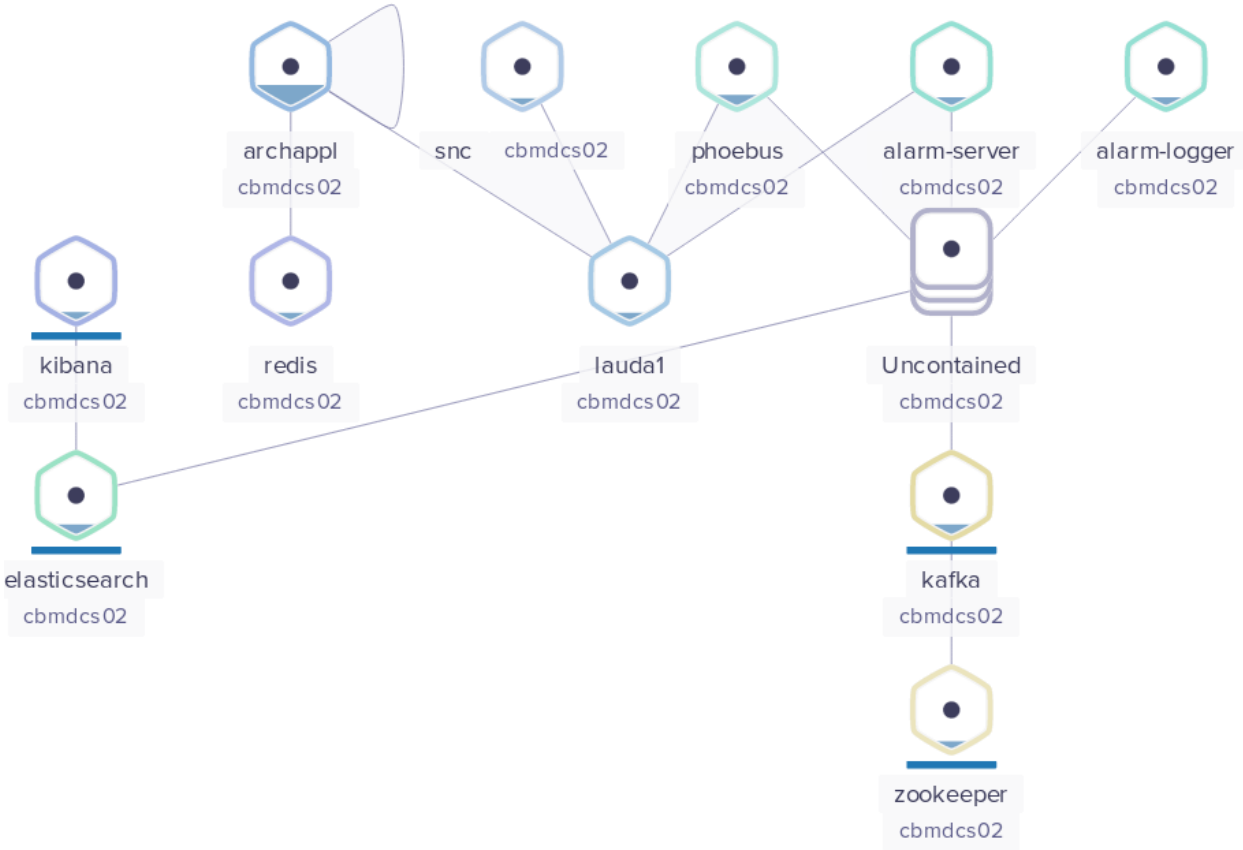
Compose your services
a tool for defining and running multi-container Docker applications

- Quick and easy configuration - proper YAML file to configure and connect containers
- Single host deployment
- User-defined volumes to store data and ensure persistency
- Restart mechanisms



Monitoring

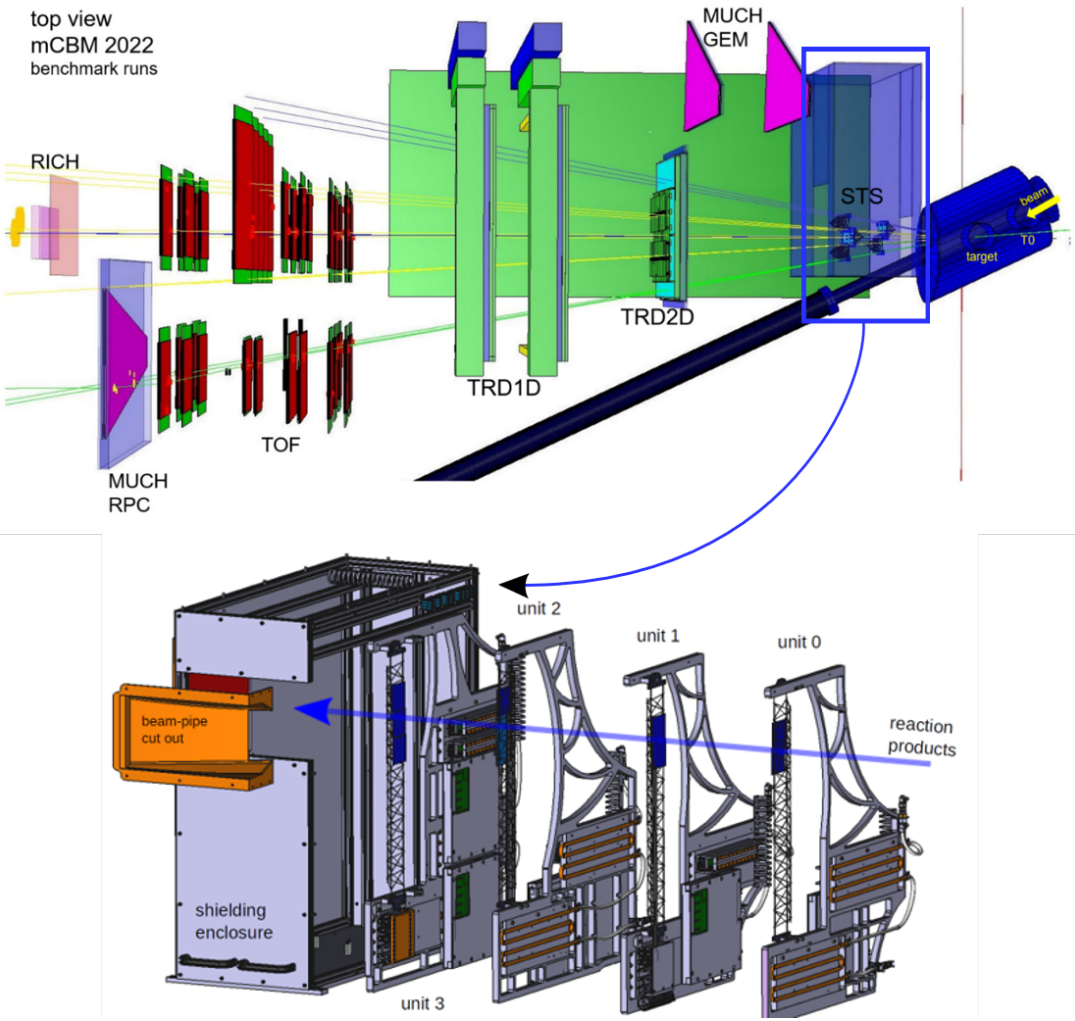
- Weave Scope - Access, control, Metrics, Metadata, Automatic topologies and intelligent grouping
- Pimo IOC (ping monitor IOC) - network monitoring using the ping
- IOC heartbeat



Applications of the framework

- Testing of different thermal interfaces, glues → 200 PVs, Archiver, DB
- Readout of the front-end electronics → 200 PVs, Archiver, DB
- Thermal stress tests of the front-end electronics → 300 PVs, Phoebus, Archiver, DB
- Fiber optic sensors design and characterization → 100 PVs, Phoebus, Archiver, DB
- Prototype of the final detector → 5000 PVs, **complete application stack**

FAIR Phase - 0 - the mCBM experiment



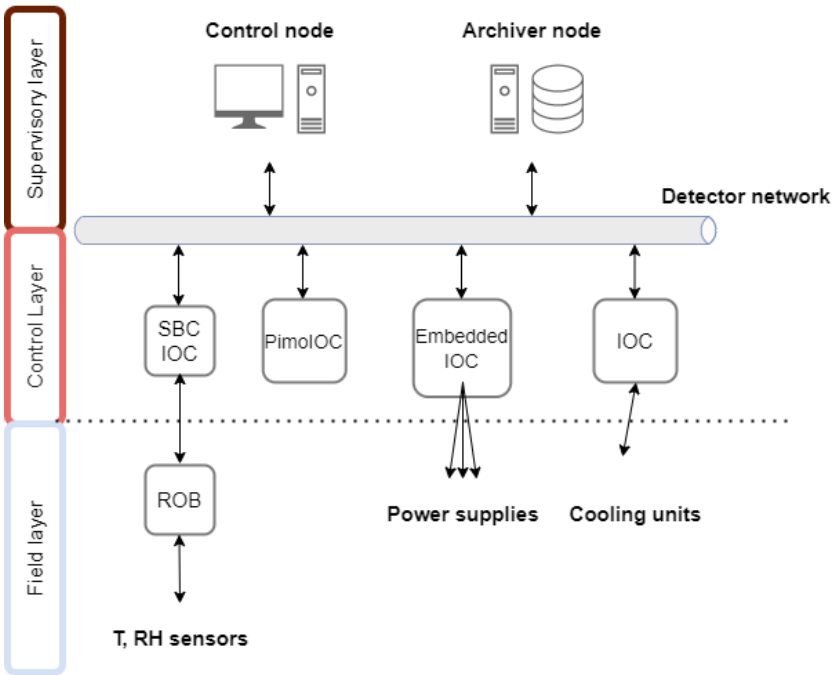
Motivation

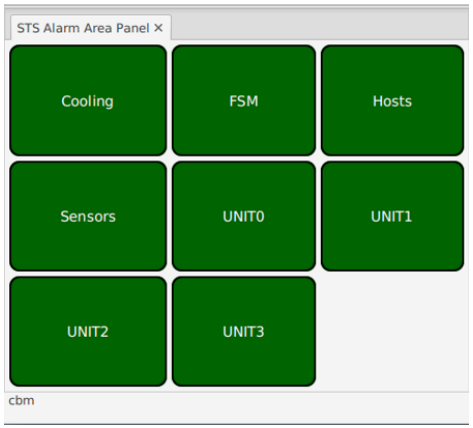
- hit/track reconstruction
- online events reconstruction
- physics analysis

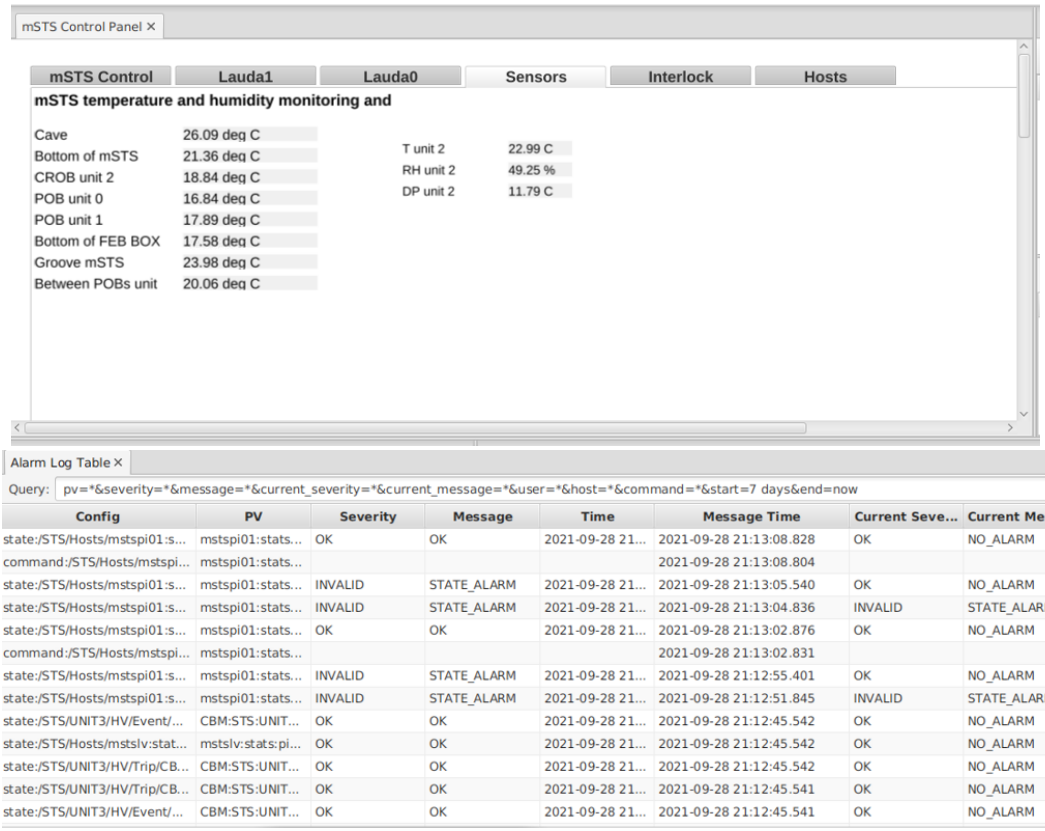
mSTS

- fully integrated detector prototype with close to final building blocks: the silicon sensors, ASIC and LDOs
- 11 detector modules ($\approx$ 1% of the final system) forming 2 tracking stations
- tests of the DAQ chain ($>$ 20000 readout channels) and Detector Control System

mSTS - the pathfinder for the STS







Alarm Log Table ×

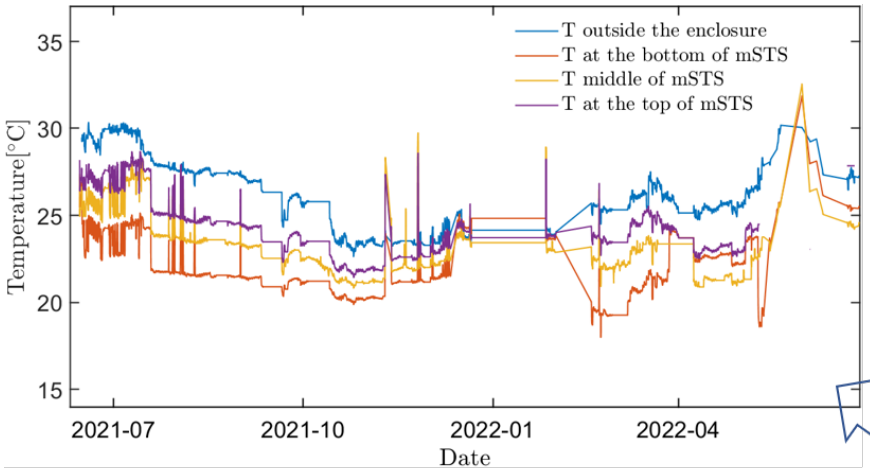
Query: pv=*&severity=*&message=*&current_severity=*&current_message=*&user=*&host=*&command=*&start=7 days&end=now

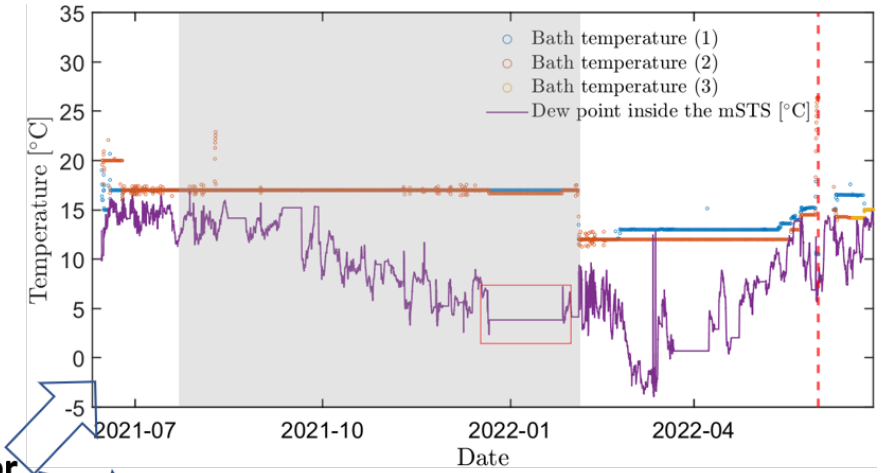
Config	PV	Severity	Message	Time	Message Time	Current Seve...	Current Mess...
state:/STS/Hosts/mstspi01:s...	mstspi01:stats...	OK	OK	2021-09-28 21...	2021-09-28 21:13:08.828	OK	NO_ALARM
command:/STS/Hosts/mstspi...	mstspi01:stats...				2021-09-28 21:13:08.804		
state:/STS/Hosts/mstspi01:s...	mstspi01:stats...	INVALID	STATE_ALARM	2021-09-28 21...	2021-09-28 21:13:05.540	OK	NO_ALARM
state:/STS/Hosts/mstspi01:s...	mstspi01:stats...	INVALID	STATE_ALARM	2021-09-28 21...	2021-09-28 21:13:04.836	INVALID	STATE_ALARM
state:/STS/Hosts/mstspi01:s...	mstspi01:stats...	OK	OK	2021-09-28 21...	2021-09-28 21:13:02.876	OK	NO_ALARM
command:/STS/Hosts/mstspi...	mstspi01:stats...				2021-09-28 21:13:02.831		
state:/STS/Hosts/mstspi01:s...	mstspi01:stats...	INVALID	STATE_ALARM	2021-09-28 21...	2021-09-28 21:12:55.401	OK	NO_ALARM
state:/STS/Hosts/mstspi01:s...	mstspi01:stats...	INVALID	STATE_ALARM	2021-09-28 21...	2021-09-28 21:12:51.845	INVALID	STATE_ALARM
state:/STS/UNIT3/HV/Event/...	CBM:STS:UNIT...	OK	OK	2021-09-28 21...	2021-09-28 21:12:45.542	OK	NO_ALARM
state:/STS/Hosts/mstslv:stat...	mstslv:stats:pi...	OK	OK	2021-09-28 21...	2021-09-28 21:12:45.542	OK	NO_ALARM
state:/STS/UNIT3/HV/Trip/CB...	CBM:STS:UNIT...	OK	OK	2021-09-28 21...	2021-09-28 21:12:45.542	OK	NO_ALARM
state:/STS/UNIT3/HV/Trip/CB...	CBM:STS:UNIT...	OK	OK	2021-09-28 21...	2021-09-28 21:12:45.541	OK	NO_ALARM
state:/STS/UNIT3/HV/Event/...	CBM:STS:UNIT...	OK	OK	2021-09-28 21...	2021-09-28 21:12:45.541	OK	NO_ALARM

GUIs, Alarm handler, Alarm logging

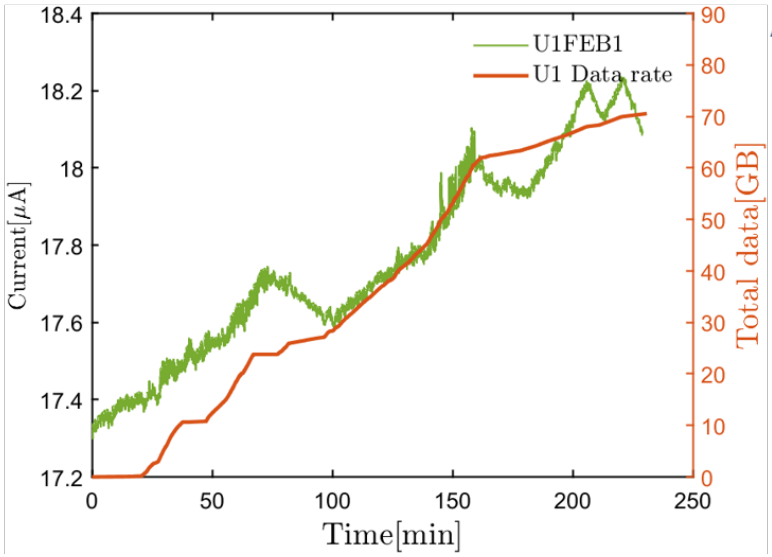
mSTS - the pathfinder for the STS

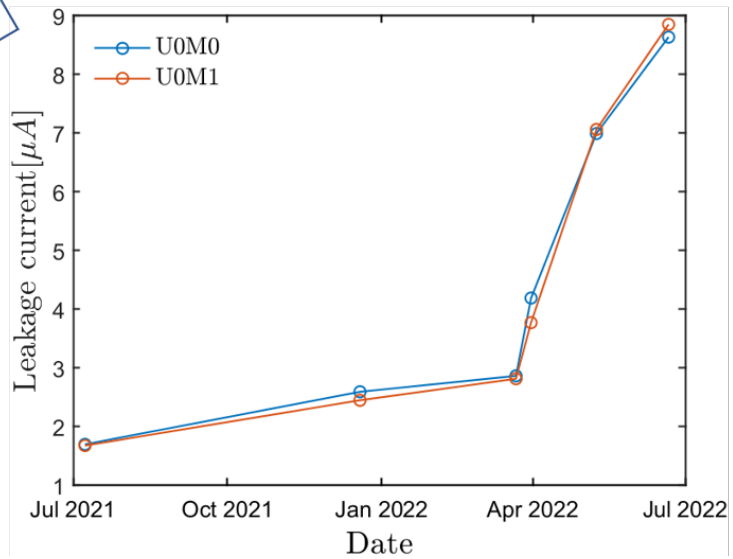
About two years of operation..





Detector





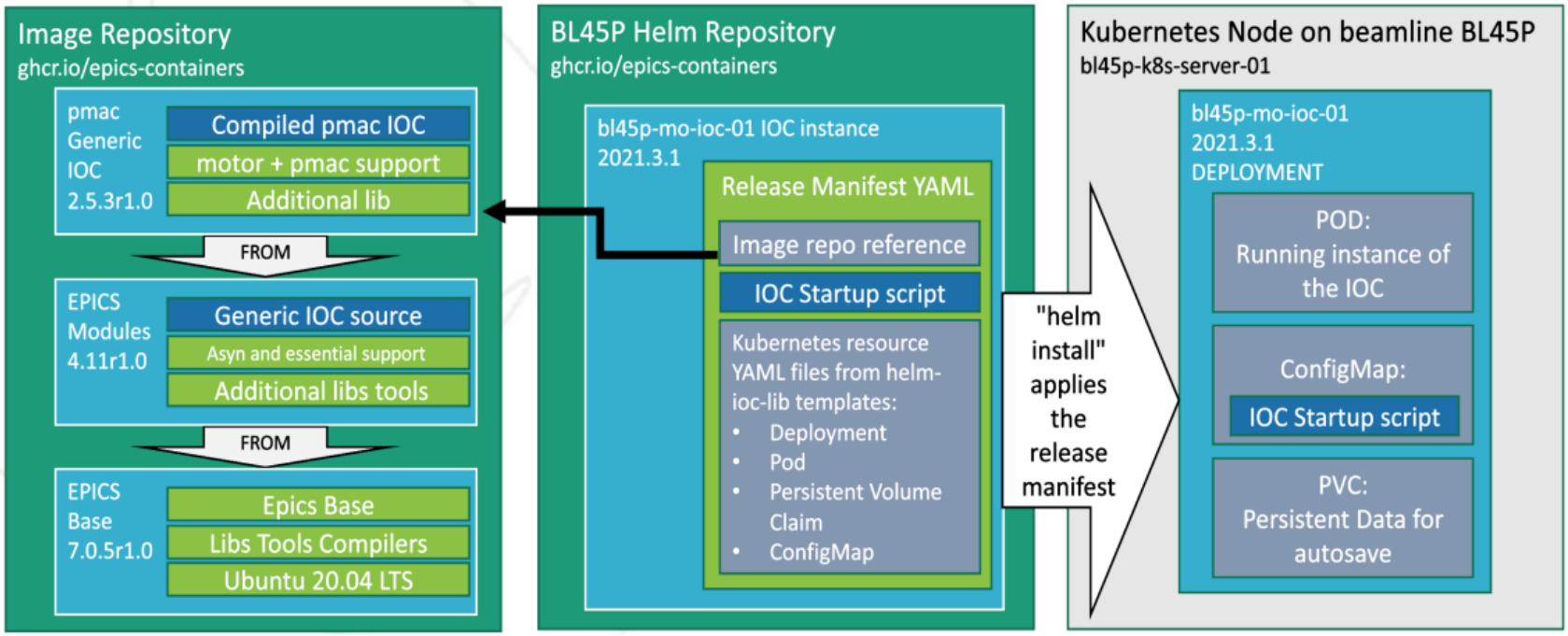
mSTS - the pathfinder for the STS

Takeaways

- ❇ no issues with the system, apart from the power cuts → On reboot, every service that is marked for restart will be immediately started without waiting for their dependency services to be healthy first → fixed with cronjob
- ❇ perfect for small setups, easily customizable, once deployed almost no maintenance efforts required
- ❇ initial idea - explore the containerized environment, prototype the control system → fully orchestrated system for all the subsystems similar to for example DLS (https://github.com/epics-containers)
- ❇ another layer could be for example **bluesky**

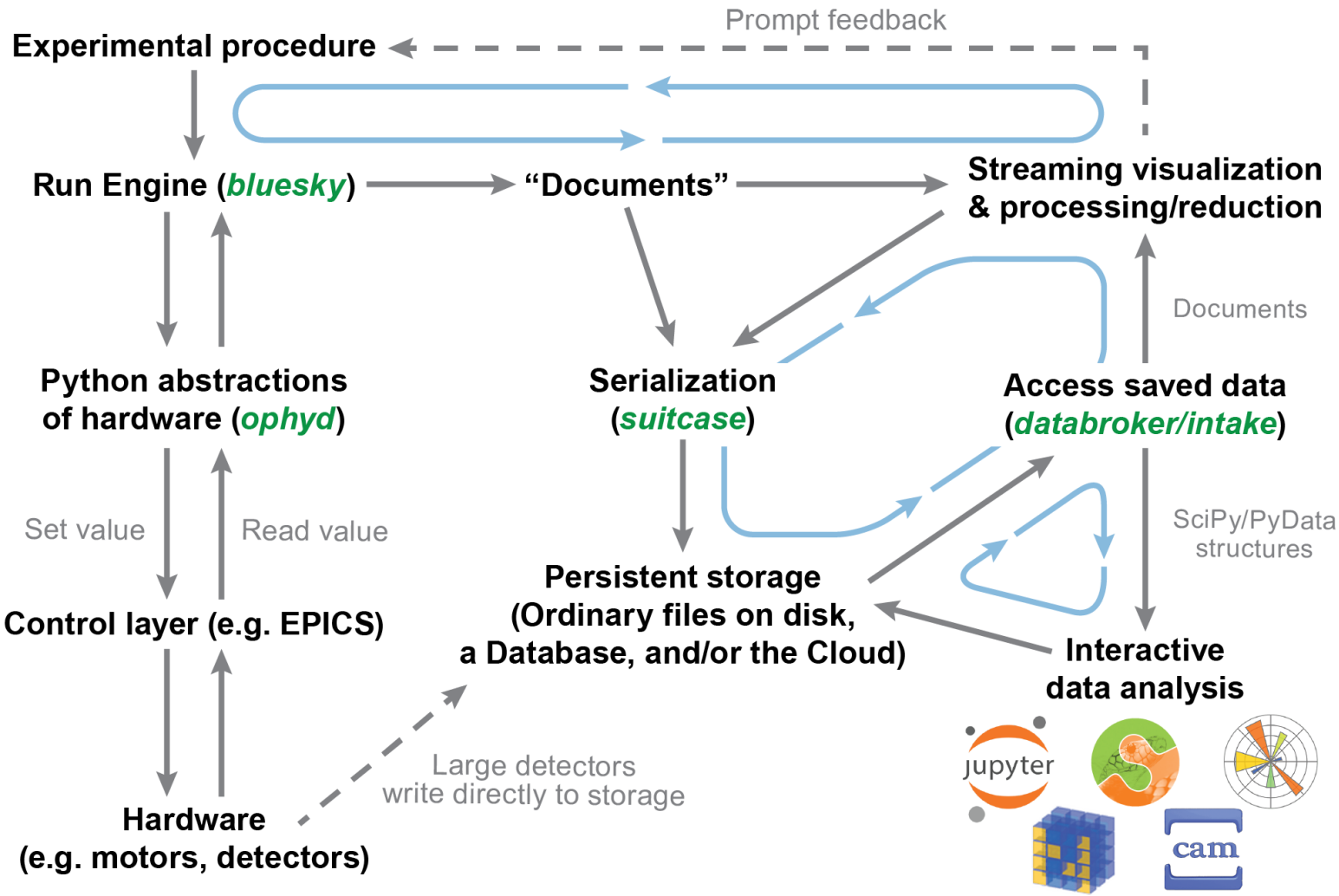
Full orchestration - Diamond Light Source

- Containers: package each IOC with its dependencies and execute it in a lightweight virtual environment.
- Kubernetes: centrally orchestrates all IOCs at the facility.
- Helm Charts: deploy IOCs into Kubernetes and provide version management.
- Repositories: Source, container and Helm repositories hold all the assets required to define a beamline's IOCs.
- Continuous Integration: source repositories automatically build containers, Helm charts and deliver them to package repositories



Bluesky project

The software libraries in the Bluesky Project interoperate through carefully-defined software interfaces. Any given piece may be separately used, extended, or replaced.



Examples of python tools

GUI tools

- PyDM - Python Display Manager is a PyQt-based framework for building user interfaces for control systems (https://slaclab.github.io/pydm/)
- phoebusGEN - Python to CS-Studio (Phoebus) XML (https://github.com/als-epics/phoebusgen)

PV access tools

- caproto - purely pythonic channel access library (https://nsls-ii.github.io/caproto/)
- pyEPICS - PyEpics is an interface for the Channel Access (CA) library

IOC

- pythonIOC - EPICS IOC with Python Device Support to be run from within the Python interpreter (https://dls-controls.github.io/pythonSoftIOC/3.0b2/index.html)

Thank you! Questions?