

MULTIMODAL DATA ACQUISITION SYSTEM AT MAX IV

V. Da Silva*, J. Just, M. Alcocer, K. Klementiev, A. Freitas, M. Lindberg
MAX IV Laboratory, Lund, Sweden

Abstract

The Balder beamline at MAX IV Laboratory, a state-of-the-art 4th generation synchrotron, is designed for X-ray absorption and emission spectroscopy. Delivering a high photon flux (10^{13} ph/s), it supports *in situ* experiments, which require fast, high-quality data acquisition and support for sequential multi-technique measurements. This work presents a data acquisition (DAQ) system that combines X-ray absorption spectroscopy (XAS) and X-ray diffraction (XRD) within a single, synchronized experiment. At the core of the system is a Double Crystal Monochromator, operated by ACS SPiiPlusEC motion controller. This controller enables stable and rapid energy scanning via programmable motion trajectories, allowing sequential acquisition of energy spectra and diffraction patterns. Experiment synchronization is achieved via FPGA-based PandABox, which generates TTL signals based on the real-time motor position, enabling technique-specific pulse trains to be sent to the respective XAS and XRD detectors, precisely gated to the energy scan. The entire experiment workflow is orchestrated using Sardana through dedicated macros and controllers. User interaction is streamlined through a Taurus GUI, providing an intuitive drag-and-drop interface for sequencing and configuring each technique. This contribution outlines the system architecture, integration challenges, and benchmarking results, highlighting the enhanced experimental capabilities made possible by this advanced DAQ system at MAX IV.

INTRODUCTION

The Balder beamline [1] is located at the 3 GeV storage ring of the Swedish 4th generation synchrotron, MAX IV, and is dedicated to X-ray absorption and emission spectroscopy in the energy range of 2.4–40 keV. With a photon flux of up to 1×10^{13} ph/s, Balder is optimized for experiments under *in situ* and *operando* conditions. These experiments increasingly demand fast energy scanning with reliable data quality, while also benefitting from the integration of complementary techniques for improved analysis of sample dynamics.

Following the MAX IV standard, the beamline control system is based on TANGO [2], with Sardana [3] used for scan configuration and execution. To address the demand for multimodal and time-efficient data acquisition (DAQ), a system was developed that integrates X-ray absorption spectroscopy (XAS) and X-ray diffraction (XRD) into a single synchronized workflow.

A major challenge for such experiments is synchronization across multiple detectors and techniques, especially when scans must run quickly while minimizing acquisition

overhead. Conventional approaches can lead to significant dead time, limiting experiment throughput and data quality.

By addressing synchronization challenges and reducing acquisition overhead, this multimodal DAQ system enables faster and more efficient XAS experiments at Balder, while establishing a framework for broader multimodal applications.

SYSTEM OVERVIEW

The combined XAS–XRD experiment at Balder [1] is executed as a sequential, repeated measurement program: one or more XAS energy scans are interleaved with one or more XRD acquisitions taken at fixed energies. The general setup for the multimodal XAS-XRD system is shown in Fig. 1.

The central element of the Balder beamline energy selection is a fixed-exit Double Crystal Monochromator (DCM) supplied by FMB Oxford. The monochromator hosts two crystal pairs, Si(111) and Si(311), covering the energy range from 2.4 to 40 keV. The crystals are indirectly cooled through liquid-nitrogen-cooled copper plates to maintain thermal stability under the high heat load from the wiggler source. Energy selection is achieved via the Bragg motor rotation axis combined with the perpendicular translation of the second crystal (vertical beam offset), ensuring a constant exit beam height. Both axes are controlled by an ACS Motion Control SPiiPlusEC [4](EtherCAT-based) motion controller, with high-resolution incremental encoders providing real-time positional feedback for precise trajectory execution.

Synchronization between the beamline energy position (Bragg motor motion) and detector acquisition is performed by PandABox [5], an FPGA-based timing and control platform. At Balder, PandABox is configured to generate gating signals derived from the Bragg axis encoder. These gates define acquisition windows during which detectors integrate data. This hardware-based gating eliminates software latency, minimizes dead time, and ensures deterministic alignment of detector readout with the monochromator position.

For intensity monitoring of ionization chambers, ALBA Electrometer [6] is used as a current amplifier and to record the incident (I_0) and transmitted (I_T) beam intensity, enabling acquisition of XAS data. For diffraction measurements, Balder employs Eiger 1M (2 kHz readout), a two-dimensional hybrid pixel X-ray detector, which combines high frame rates with large dynamic range, making them suitable for *operando* XRD acquisition and as complementary technique during multimodal experiments.

HARDWARE INTEGRATION

The beamline control system is based on TANGO Controls [2] framework, which provides a distributed architec-

* vanessa.silva@maxiv.lu.se

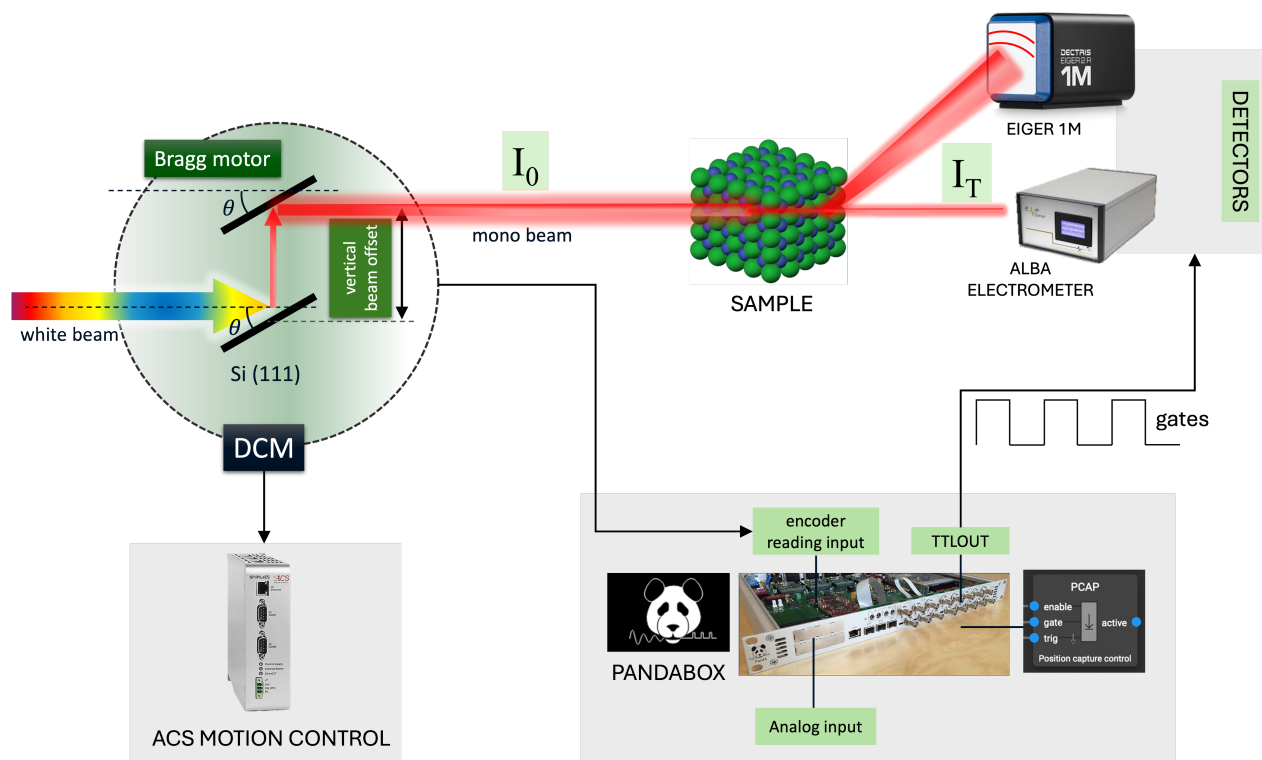


Figure 1: XAS-XRD experiment general setup overview.

ture for seamless communication between various hardware components and software modules. The experiment and scan orchestration is handled by Sardana, which is built on top of TANGO and provides an advanced scan framework.

The XAS-XRD experiment hardware components – ACS Motion Control, Alba Electrometer, Eiger 1M and PandABox – are integrated into the control system as Tango Devices. Each Tango Device has an associated Tango Device Class, which provides a list of attributes, commands and properties available for each device. A Tango Device Class is responsible for translating the communication protocol used by each equipment into Tango Controls communication.

ACS Motion Control SPiiPlusEC

The monochromator motion at Balder is driven by an ACS Motion Control SPiiPlusEC [4] multi-axis controller. The system manages the Bragg rotation and second crystal translation axes, ensuring constant exit beam height and precise synchronization.

A key feature used for fast scanning is the real-time AC-SPL+ programming language and runtime environment, which is a firmware feature that allows pre-programmed motion sequences to be stored and executed directly on the controller. Instead of sending real-time commands from a software layer while scanning (which introduces latency and jitter), motion instructions are loaded into a buffer inside the controller's memory.

The buffer mechanism supports motion trajectories to be preloaded into the controller, which can execute polynomial or piecewise-linear functions with deterministic timing. Buffers can be configured to repeat indefinitely or for a defined number of cycles, which can be useful for repeated multi-edge or sequential XAS-XRD scans. By offloading the trajectory execution to the controller, overheads from software communication are eliminated, enabling smooth and repeatable motion for energy scans.

The controller is integrated into the beamline control system as a TANGO device server, which provides full access to buffer configuration, execution, and monitoring through high-level tools. In addition, it exposes read/write attributes for the Bragg and vertical beam offset motors, allowing users to monitor and control positions, offsets, and motion. Key motion parameters such as velocity and acceleration for each motor axis are also available as TANGO attributes, ensuring comprehensive access for both operation and diagnostics.

Alba Electrometer

The Alba Electrometer [6] is a high-sensitivity current measurement device originally developed at the ALBA synchrotron to monitor beam intensity using ionization chambers. It is designed to provide stable, low-noise readout over a wide dynamic range, making it suitable for transmission XAS experiments where incident (I_0) and transmitted (I_T) beam intensities must be recorded with high accuracy. The device can operate in fast acquisition modes, allowing continuous measurements during energy scans.

The Alba Electrometer is integrated into the control system as a TANGO device server. Within the experiment acquisition chain, the key TANGO attributes are $nTriggers$, which defines the number of gates expected during an energy scan, and $triggerMode$, which configures the electrometer to operate in GATE mode. To decouple data acquisition from data recording, since different detectors may produce different numbers of frames, the device server includes both an HDF5 file writer and a data republisher. This design allows the electrometer to generate its own HDF5 files based on the received gates while simultaneously streaming data to a live plotting interface for visualization and basic online analysis.

PandABox

PandABox (Position and Acquisition Box) [5] is a FPGA based platform for position readout and experimental synchronization. It is the main experimental synchronization platform at MAX IV beamlines. PandABox provides a single interface to multiple position and synchronization signals (encoders, electrical triggers, optical triggers). The main supported I/Os used for the combined XAS-XRD experiment are multi-channel TTL for synchronous, high resolution triggering/clocking, analog-digital conversion of input analog signals, and LVDS (Low-Voltage Differential Signaling) input signals, accepting binary inputs from external hardware via LVDS connectors.

PandABox configuration can be performed via a web GUI, which provides visualization and interaction with the FPGA functional blocks, as illustrated in Fig. 2. The main functional blocks used for the XAS-XRD experiment are *INENC*, *SEQ*, *LUT*, *LVDSIN*, and *PCAP* blocks.

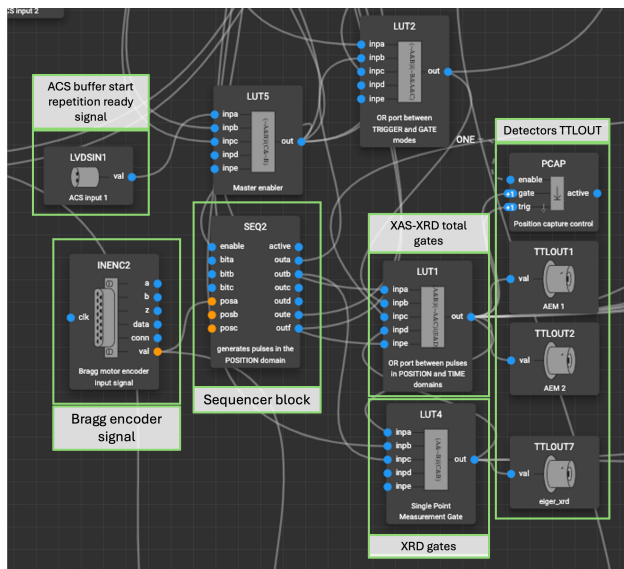


Figure 2: PandABox web interface layout with main blocks used to compose the XAS-XRD experiment synchronization.

The *INENC* block is used to read the encoder value of the Bragg motor, making the instantaneous position available

inside PandABox. A *SEQ* (Sequencer) block then compares this live encoder value with a predefined table of positions that specify where gates should be generated. Whenever the encoder position matches an entry in the table, the *SEQ* block outputs a gate signal.

The *LUT* (Look-Up Table) block, based on configurable Boolean expressions, is used to shape and distribute these gates. Gates generated from the XAS trajectory are routed to the Alba Electrometer for intensity measurements, while gates defined at fixed energy positions are assigned to the Eiger 1M detector for XRD acquisitions.

The *LVDSIN* block receives an external signal from the ACS controller at the start of each buffer repetition. This re-arms the sequencer logic and ensures that gate generation is synchronized across repeated scans.

Finally, The *PCAP* (Position Capture) block records internal PandABox values synchronously with gate events. It can capture encoder positions, sequencer states, or other internal signals with deterministic timing. During each acquisition window, the *PCAP* samples the encoder signal while the gate is high and stores the average position value. These captured positions are written alongside the experimental data, allowing direct correlation between the actual monochromator position and the XAS-XRD measurements during later analysis.

With this configuration, PandABox provides position-based synchronization, delivering dedicated gates to the XAS and XRD detectors as shown in Fig. 3. This ensures that intensity and diffraction data are consistently aligned with the monochromator energy trajectory, with minimal dead time and reproducibility over multiple scan cycles.

The gate signals (Fig. 3) are generated as follows: every time the Bragg motor encoder value matches a position defined in the *SEQ* block table, the gate output transitions low for a fixed duration (typically about 0.1 ms). After this period, the gate returns high until the next encoder match occurs. During a constant-velocity energy scan, the non-linear relation between Bragg angle and photon energy (Bragg's law [7]) results in gates of varying widths. Despite this variation, the scheme ensures highly efficient beam usage, with detectors acquiring data for more than 99% of the scan time while dead time can be reduced to as little as 1%.

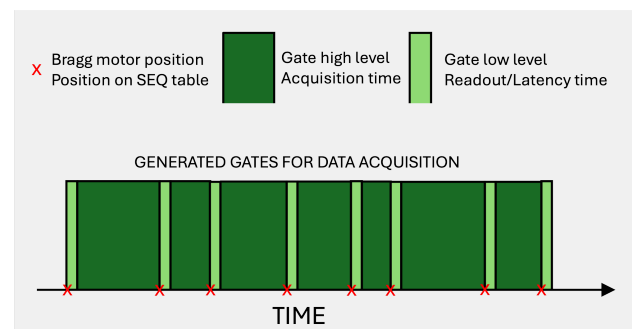


Figure 3: Generated gates based on Bragg motor position.

Integration of PandABox into the beamline control system is achieved through the PandABlocks-server [8], which exposes the FPGA functional blocks over TCP/IP. On top of this, the PandABlocks-client [9] provides a pure Python API, enabling direct interaction with the functional blocks from high-level applications. This client library is used to implement the dedicated TANGO device server, which present PandABox functionality to the control system. From the TANGO device, it is possible to configure the sequencer table position array, define gate assignments for different detectors, and control the start and stop of gate generation and acquisition. All parameters are automatically set up by the device server according to the currently configured PandABox layout, ensuring consistent and reproducible operation.

EXPERIMENT ORCHESTRATION

Integration in Sardana

The XAS–XRD experiment is performed and orchestrated by Sardana [3]. For integration into the scan framework, each hardware component must provide a Sardana controller capable of translating the TANGO Device Class API into a Sardana-specific controller API. Sardana offers a variety of controller classes, many of them tailored to specific types of hardware. The Eiger 1M is integrated using the *TwoD-Controller* class, designed for experimental channels that produce two-dimensional data. The Alba Electrometer is accessed through the *Referable* class, since only the HDF5 reference generated by its TANGO device is read by Sardana. The PandABox controller inherits from the *TriggerGateController* class, which supports hardware used to synchronize experimental channels through digital gates. In Sardana terminology, these controlled elements axes are registered as experimental channels.

An exception is the ACS Motion Control SPiPlusEC motor controller. Sardana does not currently support motor controllers that execute pre-programmed trajectories loaded into buffers using the ACSPL+ language. For this reason, trajectories are generated, loaded, and executed directly via the ACS TANGO device, which manages motion independently of Sardana.

Energy Scan Orchestration

Experiment orchestration is implemented through a Sardana macro called `hmoscan` (Hardware Motion Orchestrated Scan). Unlike Sardana's built-in continuous scan macros, in this case the motion is orchestrated by the motor controller hardware itself rather than by Sardana. The overall workflow is illustrated in Fig. 4.

The macro syntax is given by:

```
hmoscan configurationFile repeats latencyTime
```

Where *configurationFile* is a JSON document describing the sequence of XAS and XRD measurements. For XAS, the file specifies the trajectory type, absorption edge

positions, pre- and post-edge ranges, and the total acquisition time during which the monochromator is in motion. For XRD, it defines the fixed energy positions and gate times used for diffraction acquisitions. Multiple scan dictionaries can be listed to construct an experiment sequence.

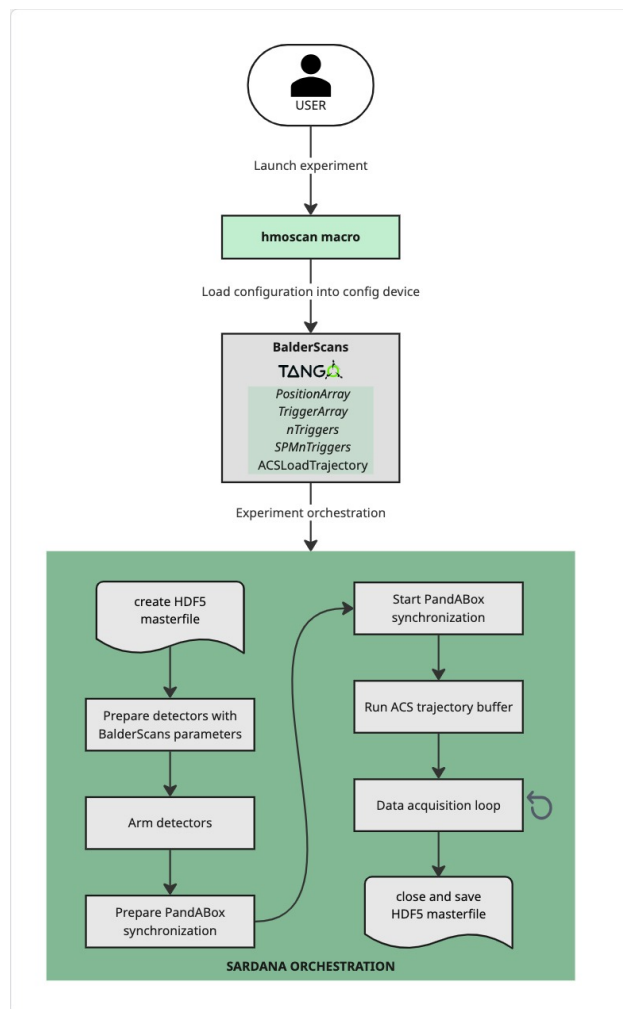


Figure 4: Orchestration workflow diagram.

The parameter *repeats* sets the number of times the sequence is executed. This repetition is managed directly within the ACS buffer, using ACSPL+ functionality that allows the buffer to cycle for a defined number of repetitions. The parameter *latencyTime* defines the delay between gate signals. Synchronization is performed in GATE mode as described in the PandABox subsection.

Because Sardana does not yet provide native support for multimodal experiments, a dedicated TANGO device class called *BalderScans* acts as an additional configuration layer. This device receives the JSON file from the scan macro and translates it into TANGO attributes used by the hardware. It prepares the trajectory for the ACS motor controller, calculates the positions where gates should be generated, and writes them to the *PositionArray* and *TriggerArray* attributes. It also sets detector frame counts through the attributes *nTriggers* and *SPMnTriggers*, which define the number of gates

for XAS scans and XRD fixed-position acquisitions, respectively, including repetitions.

Once *BalderScans* is configured, the orchestration phase begins. A Sardana recorder creates a HDF5 masterfile where detectors references will be linked and metadata will be recorded. Through the `PrepareOne` API, all detectors (Eiger, Alba Electrometer, and PCAP) are set to GATE mode and armed with the appropriate number of frames. The parameters passed to this method are provided by the *BalderScans* TANGO device configuration layer rather than by the usual Sardana parameters. Importantly, this preparation is performed only once for the entire experiment, ensuring that detectors remain armed across all repetitions and, thereby, minimizing overhead.

PandABox is configured through the `SynchOne` API method, which loads the position table into the SEQ block and applies synchronization parameters such as latency time, again using values from *BalderScans*. Finally, the ACS motor controller trajectory buffer is started via the `TriggerGate-Controller` API. Starting motion from this API guarantees that scanning begins only after all components have been configured and armed, since motion orchestration is handled by the controller hardware and not by Sardana. With this approach, both the detector configurations and the motor trajectory are loaded a single time, while the experiment cycles run entirely in hardware.

In the Fig. 5 is shown a typical configuration of the XAS–XRD experiment. The monochromator energy axis is scanned across several absorption edges (red line), moving at maximum velocity between them. Each edge can be configured independently with user-defined parameters such as acquisition time and scan range. XRD data are typically collected at the end of each XAS scan at a fixed energy position. During XAS acquisition, multiple position-based gates are generated, while a single gate is used for the XRD measurement. The sequence can be repeated at rates up to 2 Hz.

At the end of the experiment, the master HDF5 file is finalized, linking detector outputs with the corresponding metadata. This configuration enables efficient multimodal acquisition with minimal overhead, providing synchronized spectral and structural information within a single experiment cycle.

XAS-XRD User Interface

User-friendliness of the XAS–XRD experiment is provided through an Energy Sequencer Qt GUI shown in Fig. 6. The panel allows users to define one or more X-ray Absorption Spectroscopy (XAS) scans, optionally complemented by X-ray Diffraction (XRD) acquisitions, and to arrange them in the desired order. The entire sequence can then be executed with a specified number of repetitions. The interface includes tooltips that assist the user in setting scan ranges and provide estimates of timing, making it more efficient to prepare and load new configurations.

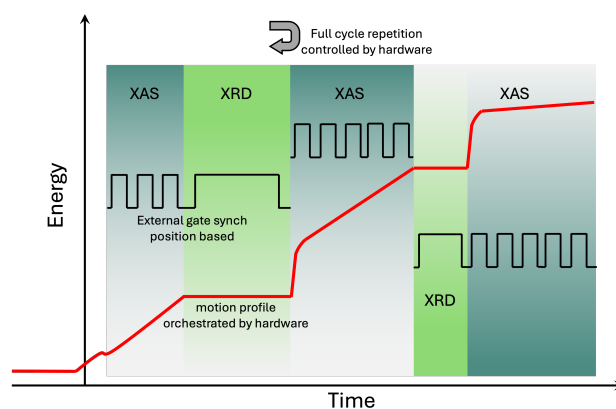


Figure 5: Hardware Motion Orchestrated scan correlating motion trajectory, external output gate and technique type.

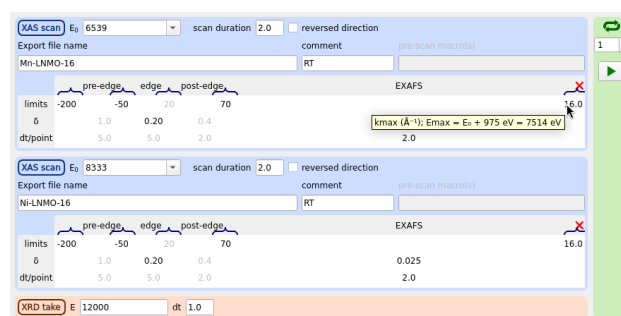


Figure 6: Energy Sequencer Qt GUI panel.

INITIAL RESULTS

Sardana's built-in continuous scans already represent a major improvement over the traditional step-scan approach by significantly reducing overhead. However, when experiments require multiple repetitions, each new scan still incurs preparation overhead: detector configuration, motion setup, and HDF5 file creation are repeated for every cycle.

The objective of this project was to minimize these repeated overheads by shifting motion orchestration into the motor controller and performing the preparation phase only once. In this scheme, configuration is performed once and detectors are armed at the beginning, while repetitions are executed directly in hardware.

Figure 7 presents a comparison between Sardana continuous scans and the proposed *hmoscan*, for identical scan parameters (energy edge, range, and acquisition time) with 1, 2, and 5 repetitions. In the continuous case, repetitions are handled in software by re-launching the scan each time, including a full preparation phase and new HDF5 file creation. In *hmoscan*, repetitions are executed in hardware using the ACSPL+ buffer mechanism, which cycles the trajectory for the requested number of repetitions without reinitialization.

Each bar in the figure represents the total elapsed time from macro launch until the data file is closed, normalized to 100%. Within each bar, the relative proportions of active time (time when data acquisition is being performed) and overhead time are highlighted. With a single repetition,

hmoscan shows higher overhead time compared to Sardana continuous scans. This is because preparation of detectors and HDF5 master file initialization takes a similar amount of time in both cases, while the motion orchestration itself is faster when executed in hardware.

The main improvement is observed when multiple repetitions are executed (2 and 5 in this test). Since hmoscan does not repeat the preparation phase between cycles, the relative overhead decreases compared to Sardana continuous scans, reaching about 20% at 5 repetitions. The larger the number of repetitions, the smaller the overhead fraction becomes, highlighting the advantage of hmoscan for experiments involving multiple techniques or energy edges that require repeated scans.

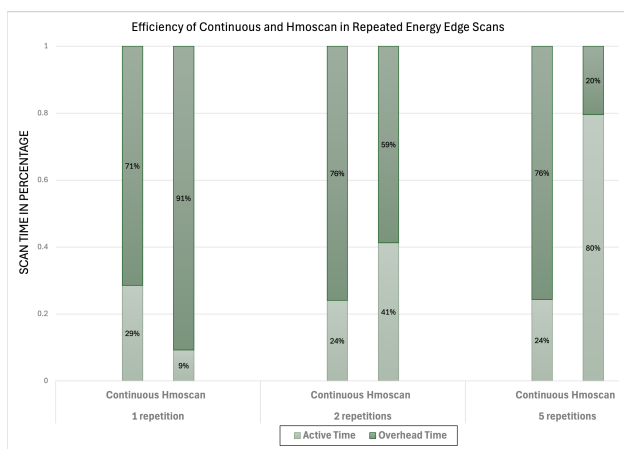


Figure 7: Scan efficiency comparison between Sardana continuous scan and hmoscan.

FUTURE WORK

Future developments will focus on further integrating multimodal experiments natively into Sardana. Recent additions such as the MultiSynchDescription [10] feature provide a foundation to migrate most of the multimodal configuration into Sardana itself, reducing the need for an external configuration layer. Work should be done to extend Sardana support for trajectory motors, enabling pre-programmed motion profiles to be managed directly by Sardana rather than through device-specific interfaces.

These improvements will allow XAS–XRD experiments to be combined more seamlessly, for example by embedding them into two-dimensional mapping scans. Such functionality will expand the range of supported experiment types and simplify user operation while preserving the performance benefits of hardware-based orchestration.

REFERENCES

- [1] K. Klementiev *et al.*, “The BALDER Beamline at the MAX IV Laboratory”, *J. Phys.: Conf. Ser.*, vol. 712, p. 012023, May 2016. doi:10.1088/1742-6596/712/1/012023
- [2] J. M. Chaize *et al.*, “TANGO - An Object Oriented Control System Based on CORBA”, in *Proc. ICALEPCS'99*, Trieste, Italy, Oct. 1999, paper WA2I01
- [3] T. M. Coutinho *et al.*, “Sardana: The Software for Building SCADAS in Scientific Environments”, in *Proc. ICALEPCS'11*, Grenoble, France, Oct. 2011, paper WEAAUST01, pp. 607–609.
- [4] *SPiiPlusEC Installation Guide, Version 3.14a*, ACS Motion Control, Israel, 2022. <https://www.acsmotioncontrol.com/KC/Content/KnowledgeCenter/Products/SPiiPlusEC.htm>
- [5] S. Zhang *et al.*, “PandABox: A Multipurpose Platform for Multi-technique Scanning and Feedback Applications”, in *Proc. ICALEPCS'17*, Barcelona, Spain, Oct. 2017, pp. 143–150. doi:10.18429/JACoW-ICALEPCS2017-TUAPL05
- [6] J. A. Avila-Abellan *et al.*, “Em Electrometer Comes to Light”, in *Proc. ICALEPCS'17*, Barcelona, Spain, Oct. 2017, pp. 137–142. doi:10.18429/JACoW-ICALEPCS2017-TUAPL04
- [7] W. H. Bragg and W. L. Bragg, “The Reflection of X-rays by Crystals,” *Proc. R. Soc. Lond. A*, vol. 88, no. 605, pp. 428–438, 1913. doi:10.1098/rspa.1913.0040
- [8] PandABlocks-Server, <https://github.com/PandABlocks/PandABlocks-server>
- [9] PandABlocks-Client, <https://github.com/PandABlocks/PandABlocks-client>
- [10] O. Vallcorba *et al.*, “Modular scientific SCADA suite with Sardana and Taurus – latest developments”, presented at ICALEPCS'25, Chicago, IL, USA, Sep. 2025, paper WEPD026, this conference.