ISBN: 978-3-95450-221-9 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2021-FRBR02

AN INTEGRATED DATA PROCESSING AND MANAGEMENT PLATFORM FOR X-RAY LIGHT SOURCE OPERATIONS *

N.M. Cook†, E. Carlin, P. Moeller, R. Nagler, B. Nash, RadiaSoft LLC Boulder, CO 80301, USA
A. Barbour, M. Rakitin, L. Wiegart, National Synchrotron Light Source II,
Brookhaven National Laboratory, NY, 11973, USA

Abstract

The design, execution, and analysis of light source experiments requires the use of increasingly complex simulation, controls and data management tools. Existing workflows require significant specialization to account for beamline-specific operations and pre-processing steps in order to collect and prepare data for more sophisticated analysis. Recent efforts to address these needs at the National Synchrotron Light Source II (NSLS-II) have resulted in the creation of the Bluesky data collection framework, an open-source library providing for experimental control and scientific data collection via high level abstraction of experimental procedures, instrument readouts, and data analysis. We present a prototype data management interface that couples with Bluesky to support guided simulation, measurement, and rapid processing operations. Initial demonstrations illustrate application to coherent X-ray scattering beamlines at the NSLS-II. We then discuss extensions of this interface to permit analysis operations across distributed computing resources, including the use of the Sirepo scientific framework, as well as Jupyter notebooks running on remote computing clusters.

INTRODUCTION

X-ray light sources are prominent drivers of scientific discovery across a range of disciplines. These facilities serve a diverse user community, often providing concurrent beam time and user support to tens of domain scientists with unique backgrounds. Increasing demand for beam time, coupled with the increasing sophistication of experiments, places constraints on the infrastructure required to successfully carry out experiments within time and resource constraints. Recently, significant development efforts have been made towards improving experimental planning and execution; however, significant challenges remain to integrating real-time analysis tools within the experimental workflow. In this proceedings, we discuss a strategy for incorporating analysis pipelines within common experimental workflows, focusing on applications at the NSLS-II light source. We present a schematic workflow for orchestrating analysis in concert with experimental execution. We then demonstrate this workflow via an open source, browser-based interface furnishing beamline agnostic analysis pipelines.

* This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research under Award Number DE-SC00215553.
† ncook@radiasoft.net

AN INTEGRATED FRAMEWORK FOR EXPERIMENT AND ANALYSIS

Our proposed strategy is to integrate a flexible analysis platform with a mature controls framework, leveraging pre-existing workflows and data schemes wherever possible. To this end, we have adopted the Bluesky Data Collection Framework, which is in active use across many beamlines at the NSLS-II [1]. BlueSky aims to provide end-to-end experimental planning, execution, and data acquisition tools through a set of interoperable Python libraries. We highlight a few of the critical libraries for the application discussed below. First, the eponymous `bluesky` library implements a run engine and event model to permit experimental control and data collection through the execution of high level plans. The `ophyd` library provides hardware abstraction to communicate plans to devices along the beamline. The `databroker` library implements an API for structured access to experimental data and metadata generated during an experiment executed by Bluesky.

For the analysis component, we chose to use the Sirepo platform to orchestrate execution of analysis pipelines. Sirepo is an open-source scientific computing gateway that provides access to community codes through custom, browser-based interfaces and an embedded JupyterHub instance. Sirepo is designed to be hardware agnostic; simulation environments are deployed via Docker containers, and can be executed across a range of computing systems, ranging from a laptop to a GPU cluster an high performance computing facility. Sirepo provides support for numerous accelerator modeling and related tracking codes; existing applications have been employed to provide customized simulations of X-ray beamlines at the NSLS-II using the Synchrotron Radiation Workshop code [2]. Sirepo has also been integrated with Bluesky to enable the asynchronous execution of long-running SRW simulations to support multi-parametric optimizations of beamlines [3].

Our approach is to provide support for analysis pipelines that complements Bluesky's support for experimental execution. Figure 1 depicts a relational diagram between the different components of the envisioned platform. The Sirepo API and user interface will support the design, templating, and execution of analysis software, to be run in tandem with experimental execution. Sirepo templates simulations via JSON schemas, providing descriptive metadata, as well as mechanisms for sharing simulations or downloading and reproducing them elsewhere. This approach is akin to Bluesky's event model for describing documents generated by experimental plans. Sirepo enables hardware-independent descriptions

ISBN: 978-3-95450-221-9 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2021-FRBR02

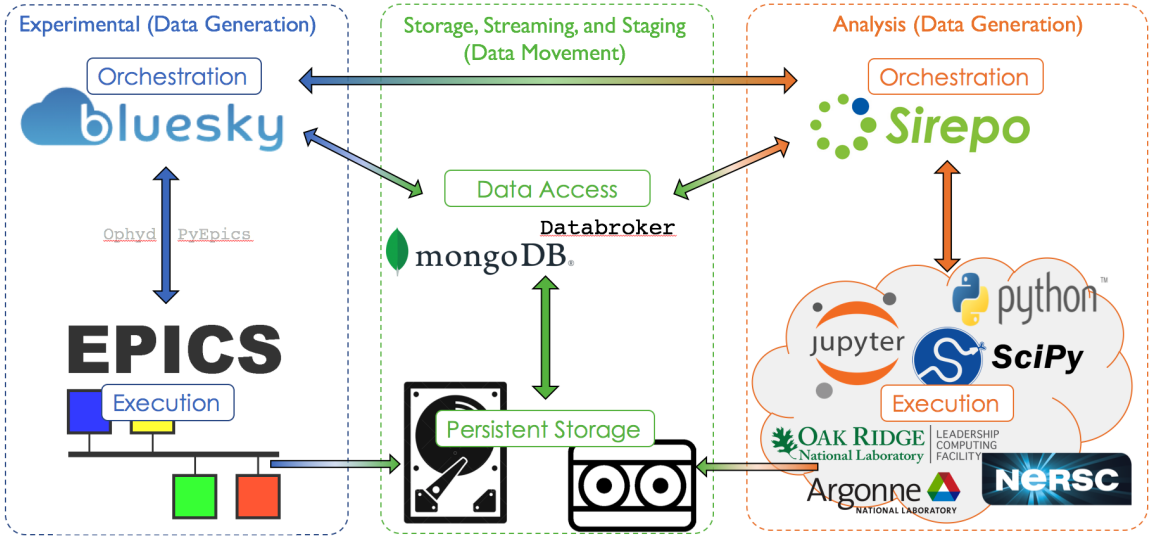

Figure 1: A high level schematic reflecting parallel implementations of Bluesky for experimental orchestration and Sirepo for analysis orchestration. While Bluesky provides schemas to define an experimental plan, along with high-level abstraction of the experimental hardware, Sirepo will provide schemas for analysis pipelines, along with access to requested computing resources, ranging from local endstations to remote high performance computing facilities. Both tools will leverage the `databroker` library to handle data selection, staging, and storage, relying on document schemas and searchable dictionaries to streamline access.

of simulations. Analyses can be executed across varied computational resources, ranging from local machines to high performance computing facilities, via the deployment of Docker images. Again, this approach has a parallel in Bluesky, which enables high-level abstraction of hardware, with low-level details handled by device-specific representations in the `Ophyd` library.

A COMMON ANALYSIS WORKFLOW

To design an analysis platform that is beamline agnostic, we have extrapolated a high-level analysis workflow predicated on typical use cases at the Coherent Soft X-Ray (CSX) and Coherent Hard X-Ray (CHX) beamlines at NSLS-II. While this workflow does not uniquely specify all possible configurations, it aims to span a representative sample of workflows, and therefore serve as a rubric for guiding interface feature design. This workflow distinguishes five steps in the analysis procedure: staging, pre-processing, analysis, documentation, and post-processing. Figure 2 describes the resulting pipeline.

Our initial prototype aims to address the first three steps, without compromising existing documentation and post-processing capabilities in place for user's at NSLS-II. Specific emphasis has been placed on accommodating the analysis step, because it presents unique demands on software flexibility and resource management. At CHX and CSX, many users leverage Jupyter notebooks running IPython kernels [4] are customized to provide near real-time analysis for a subset of the data generated during runs. Executing these notebooks requires dedicated support for required dependencies, as well as sufficient computing resources to run notebooks for each of the hundreds to thousands of datasets generated during a given experiment. Notebook execution may take 1 – 10 minutes depending upon the analysis, while data capture may only require a few seconds, meaning that the analysis step constitutes a bottleneck in completing the pipeline in real-time.

A PROTOTYPE SIREPO INTERFACE FOR INTEGRATED ANALYSIS

With this pipeline in mind, we demonstrate a prototype browser-based interface for carrying out beamline agnostic analysis at X-Ray light sources, hosted by Sirepo, and to be used in conjunction with experimental execution carried out via Bluesky. The initial prototype consists of three tabs , addressing the first three stages of the previously discussed analysis pipeline.

Figure 3 depicts the data selection tab, which permits users to browse data generated by Bluesky runs at a particular beamline. By leveraging the `databroker` library's catalog structures, runs can be searched and filtered according to primary metadata such as start and stop time, unique identifier (UID), user, and other relevant flags. The search results can be subsequently sorted, and users may select a subset of remaining runs for further processing.

Following this selection, users are guided to the metadata tab, which enables additional inspection of high level metadata regarding the run, alongside specific device and analysis details provided by the Bluesky run documents. Currently, a fixed subset of metadata is presented to the viewer. However, future implementations will expand the ability to search metadata entries using queries permitted by `databroker`'s

ISBN: 978-3-95450-221-9 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2021-FRBR02

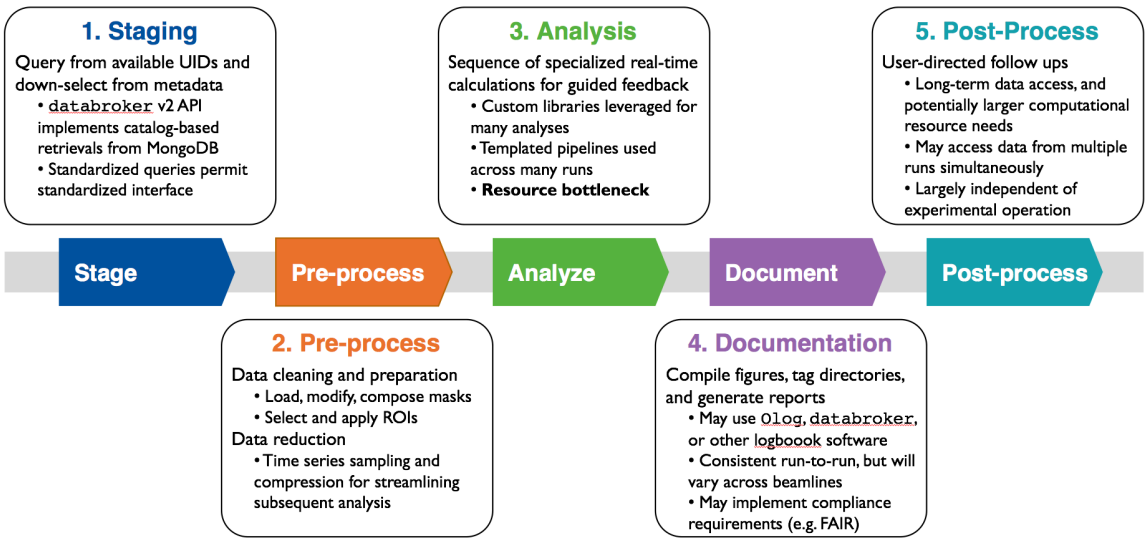

Figure 2: A high-level workflow characterizing an analysis pipeline which meets the needs for beamline experiments. During steps (1) and (2), Experimental schemas, in this case provided via `databroker`. are leveraged to provide metadata to support downselection and preparation of data for analysis. During step (3), an external codebase (e.g. Jupyter) may be employed to carry out specialized calculations. For experiments performed using Bluesky, documentation can be handled via `databroker` or `Olog`, while post-processing steps are often carried out off-site.

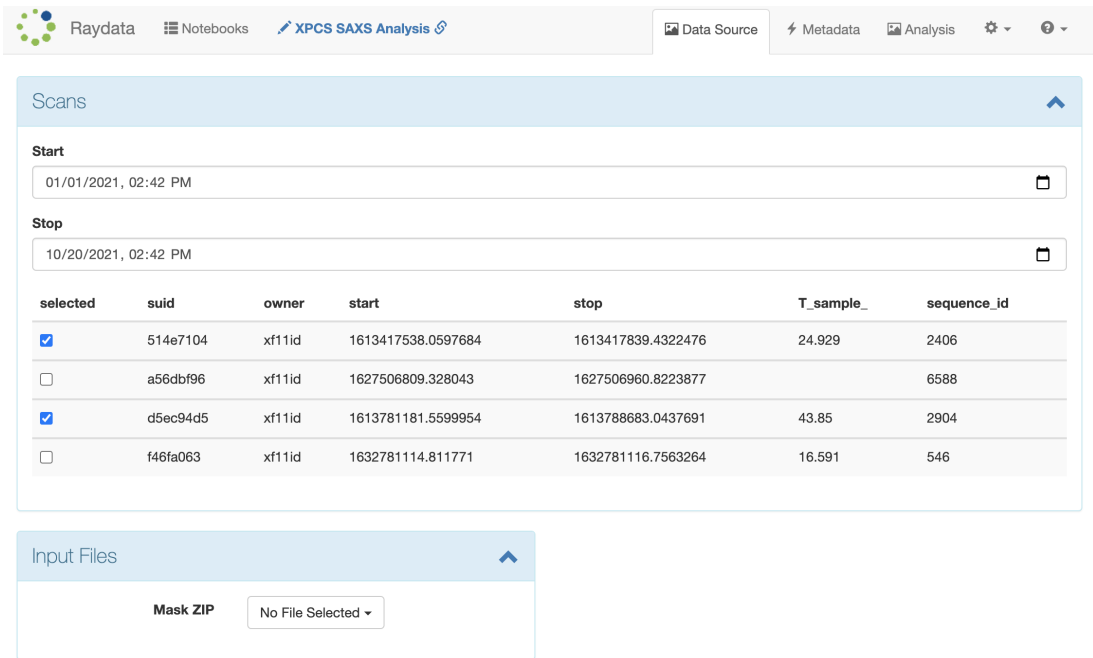

Figure 3: The data selection tab permits users to search, sort, and select from available runs in a prescribed catalog, using high level metadata to guide decisions about selections.

mongoDB-based implementation. Figure 4 illustrates the use of this interface for inspecting two selected datasets. Further improvements will permit quick snapshots of raw data files as guided by the user.

Lastly, the analysis tab addresses the analysis portion of the pipeline. For a given dataset, a user-specified Jupyter notebook is executed to carry out any specialized analysis of the initial datasets. The notebooks are executed via Docker images designed to include all relevant dependencies for the calculations, permitting their deployment across distributed computational resources. Products of the analysis notebook, such as figures, are inspected by the interface and presented to the user immediately, enabling real-time feedback during execution. Figure 5 depicts a snapshot of the analysis tab, illustrating the generation of figure panels reflecting the products of the notebook.

ISBN: 978-3-95450-221-9 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2021-FRBR02

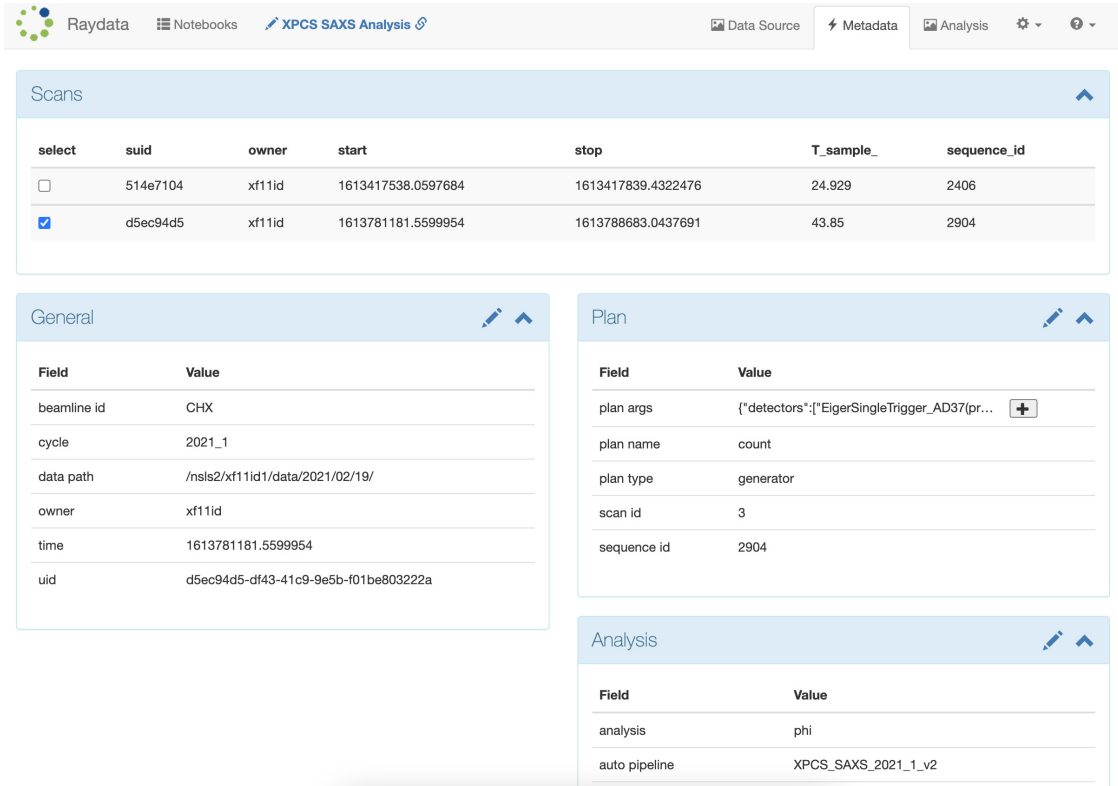

Figure 4: The metadata inspection tab provides additional details from the run metadata and start documents to permit inspection of selected runs.

Figure 5: The analysis tab of the interface, displaying previews of several of the figures generated by running the analysis notebook for this particular run.

ISBN: 978-3-95450-221-9 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2021-FRBR02

CONCLUSION

We report on progress in the development of an interactive user interface for real-time analysis of X-ray light source experiments, built atop the Bluesky framework and Sirepo platforms. The interface leverages shared metadata structures provided by the Bluesky, making use of the `databroker` library to enable active searching, sorting, and retrieval of runs from catalog structures generated during experimental execution. Custom analysis routines are supported by deploying templated Jupyter notebooks via Docker images containing the necessary dependencies. Job management, resource allocation, and queueing are provided by Sirepo, along with real-time feedback through automated figure previews and reports. Future developments will enhance queueing capabilities to support user-directed screening and prioritization of data, with the goal of providing an integrated, automated, and customizable analysis workflow to complement Blueksy's experimental procedures on active beamlines at NSLS-II and elsewhere.

REFERENCES

[1] Bluesky Project, `https://blueskyproject.io`

[2] Rakitin M S *et al.*, "Sirepo: an open-source cloud-based software interface for X-ray source and optics simulations", *J. Synchrotron Radiat.*, vol. 25, pp. 1877–1892, 2018.

[3] Rakitin M S *et al.*, "Introduction of the Sirepo-Bluesky interface and its application to the optimization problems", *Proc. SPIE*, vol. 11493, 2020. `doi:10.1117/12.2569000`

[4] Pérez, F. and Granger B., "IPython: A System for Interactive Scientific Computing", *Comput Sci. Eng.*, vol. 9 pp. 21–29, 2007.