DATABROKER: AN INTERFACE FOR NSLS-II DATA MANAGEMENT SYSTEM
A. Arkilic, L. Dalesio, D. Allan, D. Chabot Brookhaven National Laboratory, NSLSII Upton, NY, USA

Abstract
A typical experiment involves not only the raw data from a detector, but also requires additional data from the beamline. This information is largely kept separated and manipulated individually, to date. A much more effective approach is to integrate these different data sources, and make these easily accessible to data analysis clients. NSLS-II data flow system contains multiple back-ends with varying data types. Leveraging the features of these (metadatastore, filestore, channel archiver, and Olog), this library provides users with the ability to access experimental data. The service acts as a single interface for time series, data attribute, frame data access and other experiment related information.

INTRODUCTION
The current and projected NSLS-II beamlines will support a wide range of disciplines from biology to physics with a wide range of techniques. Higher beam quality and new generations of area detectors are increasing data rates quickly. Increasingly all disciplines are using multi-element detectors to measure and record the results of the experiment. Current state-of-the-art detectors are capable of producing megapixel images with frame-rates in the kilohertz range which results in data rates of up to 800 Mb.s⁻¹. Such data-rates are fully compatible with current off-the-shelf RAID storage technology, effectively allowing the data to be written to a storage medium as it is produced. The challenge for NSLS-II, however is to be able to process and analyse the data at this rate, with these volumes to allow the experimenter to not only make the “first-cut” for decision making during the experiment, but to retrieve data and analyse it for publication. NSLS-II middle layer beamline applications (Fig. 1), powered by MongoDB, the General Parallel File System (GPFS), and EPICS provide such capabilities.

Given the modular architecture and complexity of these various applications, a need for an easy point of access for all experimental data has occurred. In order to simplify the access to critical experimental data, we have developed databroker.

ARCHITECTURE
databroker is the pathway for data analysis tools to access data without any knowledge of experimental data. The library in production is implemented in Python and provides users with broker objects that contain Pandas data frames, allowing semi and un-structured experimental data to be accessed within scientific Python framework with ease via its narrow interface.

Deployment Strategy and Access Environment
All python clients that are part of NSLS-II middle layer get deployed to each beamline using the Anaconda python package management toolkit. The sandbox approach allows us to deploy packages that are required but not natively supported by the Debian 8. IPython notebook is also part of this deployment and it is the primary platform to access databroker. In addition to local access to databroker, NSLS-II beamlines provide access to an identical databroker session remotely using jupyter hub.
Using Databroker

As described in Fig. 3, databroker provides different querying methods such as, unique identifier based, time based, and complex/aggregated. This sort of querying approach provides users with immense data mining capabilities that were not available as a part of most legacy systems. Once a query is submitted to databroker yields to a metadatastore query. The results of this query allow databroker to compose a header that is composed of a RunStart, RunStop, and a set of EventDescriptor documents [1]. Databroker lazily loads the Event documents, which hold the actual experiment metadata, using a generator object. This approach is performance savvy as in most cases users only want to access the header of the experimental data and request the data payload if needed. In addition to data in metadatastore, databroker is capable of locating images that are related to each data point within specific metadatastore events. As shown in the jupyter session in Fig. 4, querying and accessing complex experimental data requires a handful of comments, allowing users to get to analysis stage with very little effort.

CONCLUSION

Databroker provides access to complex, high-performance, and modular NSLS-II middle layer framework by providing a well-defined narrow interface that is accessible both locally and remotely. Its intuitive and simple API allows end-users to start analyzing
experimental data during their experiment. This allows them to discover interesting attributes of their samples while they still have the chance to investigate them further.

REFERENCES
