ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUDPP02

DATA ACQUISITION SYSTEM FOR THE APS UPGRADE*

S. Veseli†, N.D. Arnold, T. Berenc, J. Carwardine, G. Decker, T. Fors, T. Madden, G. Shen, S. Shoaf, Argonne National Laboratory, Lemont, USA

Abstract

The Advanced Photon Source (APS) Upgrade multi-bend achromat accelerator (MBA) uses state-of-the-art embedded controllers coupled to various technical subsystems. These controllers have the capability to collect large amounts of fast data for statistics, diagnostics, or fault recording. At times, continuous real-time acquisition of this data is preferred, which presents a number of challenges that must be considered early on in the design; such as network architecture, data management and storage, real-time processing, and impact on normal operations. The design goal is selectable acquisition of turn-by-turn beam position monitor (BPM) data, together with additional fast diagnostics data. In this paper we discuss engineering specifications and the design of the MBA Data Acquisition System (DAQ). This system will interface with several technical subsystems to provide time-correlated and synchronously sampled data acquisition for commissioning, troubleshooting, performance monitoring and fault detection. Since most of these subsystems will be new designs for the MBA, defining the functionality and interfaces to the DAQ early in the development will ensure the necessary components are included in a consistent and systematic way.

INTRODUCTION

State-of-the-art embedded controllers (microcontrollers, field-programmable gate arrays, digital signal processors, etc.) have a plethora of resources to implement a high-level functionality tightly coupled to the technical system equipment. A common use of these resources is to utilize large memory buffers to collect fast data for monitoring, statistics, diagnostics or fault recording. Each embedded controller may contain several gigabytes of memory for such purposes. This presents a number of challenges related to data acquisition and data management: one must collect, transfer, manage, and utilize a large amount of data from numerous controllers without affecting normal operations.

In this paper we describe engineering specifications and the design of the Data Acquisition System for the APS Upgrade multi-bend achromat accelerator. The DAQ system will interface with a number of technical subsystems to provide time-correlated and synchronously sampled data acquisition for commissioning, performance monitoring, troubleshooting and early fault detection. Most of these subsystems will be redesigned for the MBA, and therefore defining the functionality and interfaces to the DAQ early in their development process will make sure that the necessary hardware and software components are included in a consistent and systematic way across the board.

* Argonne National Laboratory's work was supported by the U.S. Department of Energy, Office of Science, Office of Basic Energy Sciences, under contract DE-AC02-06CH11357.
† sveseli@aps.anl.gov

REQUIREMENTS

The APS Upgrade MBA beam position will be regulated by a distributed network of 20 double sector (DS) controllers and one master controller [1]. Each double sector will contain hardware for several technical subsystems with interfaces to the DAQ system:

- Storage Ring RF Beam Position Monitors (SRRF BPM) providing turn-by-turn (TBT) data
- Storage Ring X-Ray Beam Position Monitors (SRXR BPM)
- Real-time Feedback (RTFB)
- Bipolar Power Supplies (BiPS)
- Unipolar Power Supplies (UniPS)

In addition to the above, the DAQ system will also interact with a number of other single-node subsystems, including the following:

- Storage Ring RF (SRRF)
- Injection Kicker Power Supplies (InjPS)
- Bunch Current Monitor (BunCM)
- Bunch Lengthening System (BunCM)

Table 1: Anticipated Data Rates for the APS MBA Technical Subsystems with Interfaces to the DAQ System

Subsystem	Sample Rate [kHz]	Data Rate [MB/s]	Number of Nodes
SRRF BPM	271	99.7	20
SRXR BPM	10	0.5	20
RTFB	22.6	25.8	20
BiPS	22.6	13.2	20
UniPS	22.6	32.7	20
SRRF	271	67.2	1
InjPS	0.2	1.3	1
BunCM	10	26.0	1
BunLS	2440	458.7	1

As shown in Table 1, the total double sector data output is about 172 MB/s, while the cumulative data rate for the entire DAQ system is about 4.0 GB/s. Note that the turn-by-turn rate indicated in Table 1 includes only the essential BPM signals (timestamp, x/y positions, sum). The generated TBT data could also include up to 8 additional signals per BPM (up to 240 signals per double sector), which could potentially result in as much as 260 MB/s of additional TBT data per double sector, or up to additional 5.2 GB/s of data for the DAQ system as a whole.

17th Int. Conf. on Acc. and Large Exp. Physics Control Systems ICALEPCS2019, New York, NY, USA JACoW Publishing
ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUDPP02

SYSTEM DESIGN

Figure 1 illustrates the main components of the DAQ system. Wherever possible DAQ Input / Output Controllers (IOC) will directly interface with the hardware where the real-time acquisition is performed. In those cases where direct connection to hardware is not feasible, we have to utilize the system-specific "Front-end" (FE) software components. Each DAQ IOC will be somewhat customized for a given technical system, but will also utilize a common framework for capturing and transferring time-series data. This framework will support the transmission of data across dedicated subnets, either directly to services for continuous data collection, or to services responsible for distribution of data to storage, external services, or applications prepared to accept and analyze this data. The main advantage of this architecture is that it can handle those cases where real-time DAQ IOCs do not have sufficient network bandwidth or processing power to handle multiple applications interested in the same data.

The key aspects of the DAQ system will include:

- Capability to acquire data from several subsystems at various sample rates and correlate this data to within one beam revolution (3.6 µs) or better
- Scalability
- Support for continuous acquisition or triggered acquisition limited only by available storage
- The ability to route the data to any number of applications
- Separation from operational systems (networks, processors, servers) to allow trouble-shooting/enhancements during user operation

The primary components of the DAQ system are the DAQ FEs and IOCs, the communication medium and protocol between the IOCs and the services, the data collection and distributor services, as well as the client API and CLI interfaces for applications that consume the data. The following sections give additional details and suggested implementations of these components.

DAQ Front-ends

The DAQ FEs are needed in such cases where direct IOC connection to the technical system is not possible, or is highly impractical. A typical example for this use case is FPGA/DSP hardware that is running on a platform not supported by EPICS, from which data can be retrieved only via network connection. Although each FE will be implemented in a manner mostly specific to the underlying hardware, there may be some parts that come from a common DAQ software base (e.g., generic framework for TCP-based messaging).

DAQ IOCs

The DAQ IOCs will be built using the common Asyn/Area Detector (AD) [2] framework offering a flexible architecture that allows utilizing any number of real-time processing and communication plugins (Fig. 2). Each IOC driver will use a customized interface to collect the real-time data either directly from the technical system, or from its front-end. In addition, each IOC will need to define the mechanism by which it retrieves a precise timestamp that will be applied to the data. Many DAQ IOCs will contain event receivers; others may obtain timestamps from another system. The timestamp and event retrieval, together with the driver, represent the hard-real time components of the DAQ IOC.

Each DAQ IOC will also be customized with respect to a set of processing plugins appropriate for a given technical system. There may be plugins for statistics, filtering, fault detection, or pre-processing of the data. More generic plugins may be used in more than one IOCs, but some will have to be custom written for a specific hardware or analysis application.

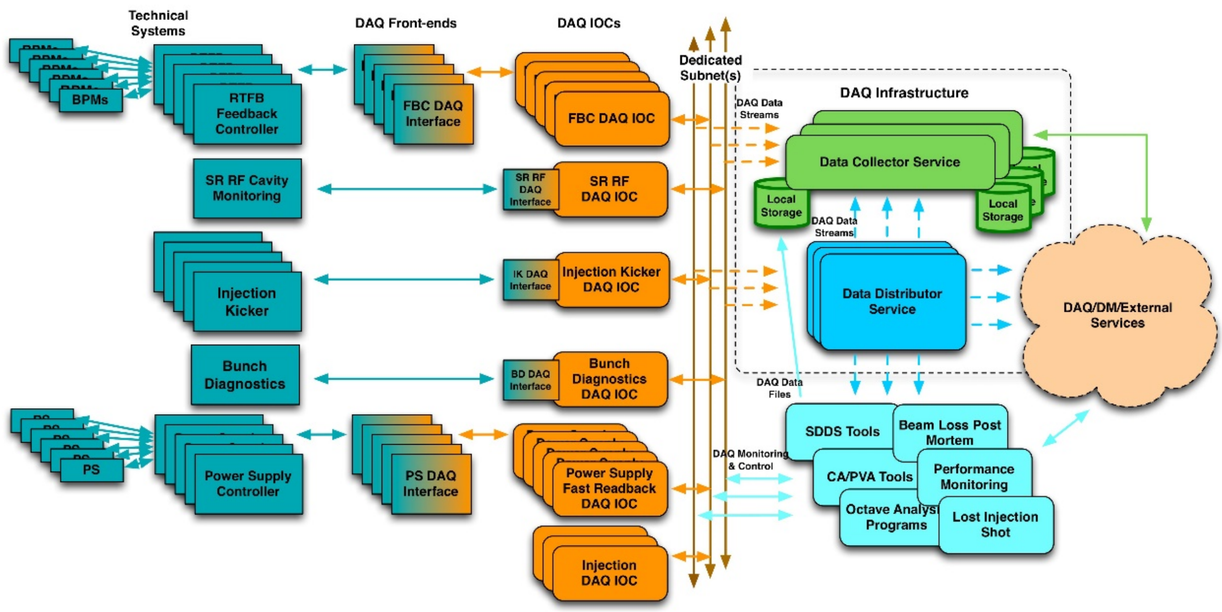

Figure 1: DAQ system architecture.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUDPP02

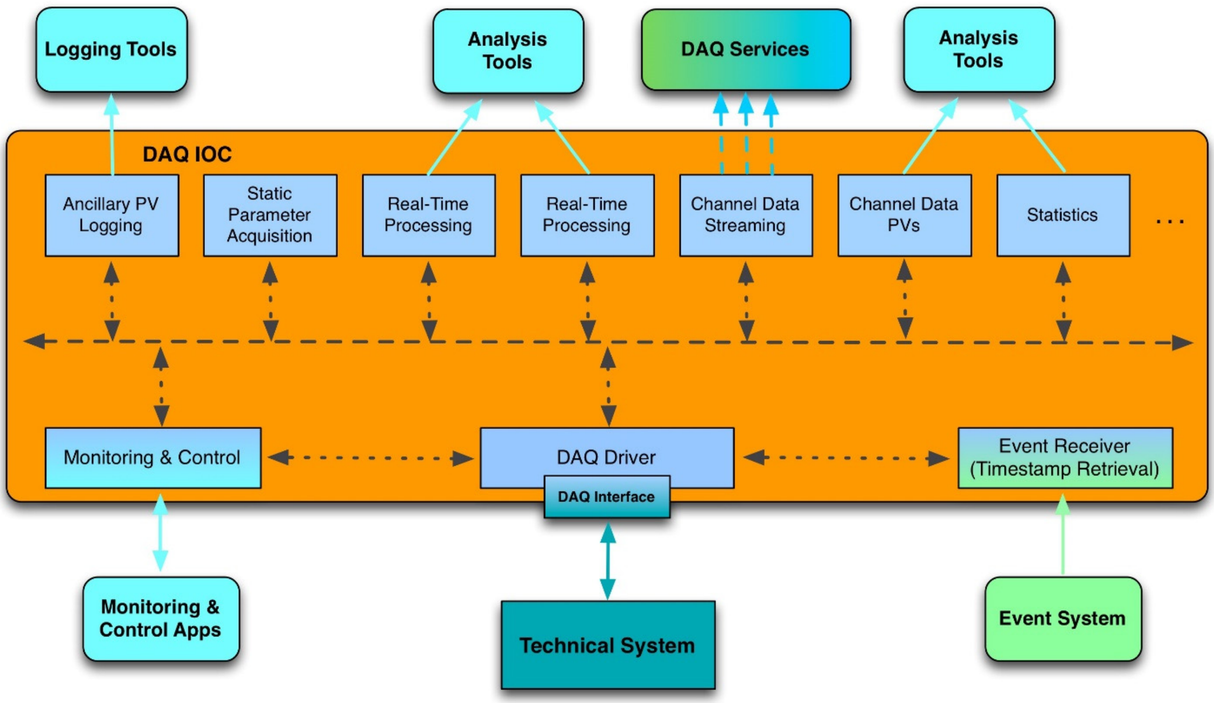

Figure 2: DAQ IOC.

DAQ Data Objects

Different technical systems gather different sets of fast time-series channel data, static parameters and slow data. Channel data also varies in type, such as signed and unsigned integers, floats, doubles, etc. In order to simplify handling of the DAQ data by the receiving services, as well as to enable use of generic EPICS client interfaces and tools, all data collected and provided by the DAQ software must be self-describing. To this end we propose utilizing EPICS PV Data (PVD) structures [3], which will allow creating system-specific and self-describing DAQ objects that can be retrieved via any standard EPICS PV Access (PVA) [4] command line tools or APIs, and that can be easily understood by any client application.

As an example, the power supply (PS) fast readback data structure might encompass the following array fields used for monitoring and diagnostics (N_c indicates number of correctors):

- Time (double array)
- Setpoint_i (i=1,..,N_c; N_c float arrays)
- CurrRdbk_i (i=1,..,N_c; N_c float arrays)
- StatusRdbk_i (i=1,..,N_c; N_c float arrays)
- FbcClockTimestamp (double array)
- UdpRxTimestamp (double array)
- EventNumber (slow data array)

In addition, the PS structure may contain a set of parameters (scalar fields) common to all DAQ data structures (e.g., object ID, DAQ timestamp, etc.), as well as a number of PS-specific fields describing the current state of the underlying power supply hardware. Note that some of the system-specific fields might represent slow data arrays. “Slow data” are values that do not need to be sampled at the high acquisition rate but may change during the acquisition interval. DAQ slow data objects are PVD structures, with timestamp and data fields, with the data field itself being a structure. This design supports the most generic use case where slow data might consist of several elements.

The IOC driver will fill in the PV Data object (or the “data buffer”) from the hardware data during the data acquisition cycle, and IOC will make this object available to client applications on the EPICS PV Access (PVA) channel via a custom Area Detector PVA server plugin.

DAQ Acquisition Modes

To accommodate high-capacity high-data-rate acquisitions, different techniques must be employed depending on the acquisition interval of interest and the size and rate of the DAQ Data Objects. Three acquisition modes are described in the following paragraphs.

Figure 3 illustrates the first case where an entire acquisition cycle (made up of possibly several waveforms at a high sampling rate) can effectively be transmitted as a single DAQ Data Object. The acquisition may be initiated periodically or by an aperiodic event. The resulting waveforms are analogous to EPICS V3 waveform records and can be easily interpreted by any client.

Figure 4 the desired acquisition interval results in a data packet (based on number of waveforms and sampling rate) too large to effectively transmit in a single DAQ Data Object. Therefore, the desired acquisition cycle is divided into a “burst” of DAQ Data Objects allowing for reasonable buffer sizes to be used in the IOC & clients. The client

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUDPP02

must have sufficient performance to receive these contiguous packets (without missing any) and reassemble them into the desired waveforms. In this case, the client must be aware that several DAQ Data Objects are included in a single acquisition. Similar to the first case, the acquisition may be initiated periodically or by an aperiodic event.

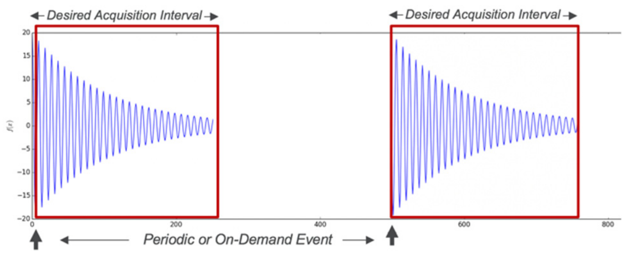

Figure 3: DAQ Acquisition Mode: Single DAQ Data Object (indicated by the red square).

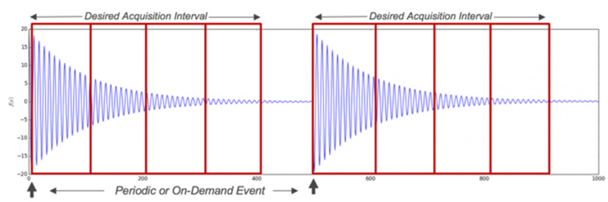

Figure 4: DAQ Acquisition Mode: Multiple Contiguous DAQ Data Objects.

Figure 5 illustrates continuously streaming DAQ data that may be used to capture very long acquisitions or to look for random or hard to detect anomalies in the data. Again, the client must be aware of this mode and have sufficient performance to capture contiguous DAQ Data Objects. Properly configured, a client could continuously capture, process, and store this data potentially limited only by available disk space.

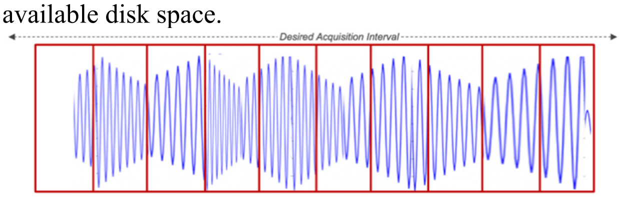

Figure 5: DAQ Acquisition Mode: Continuous "Streaming" DAQ Data Objects.

Storing DAQ data into files during long-term acquisition is one of the most important use cases of the continuously streaming acquisition mode. This will be accomplished via custom AD streaming plugins. These plugins will utilize PV Access protocol to transmit sequential data bursts to the Data Collector service, which will store each burst into its own file and possibly trigger a pre-configured processing workflow for that file. The number of DAQ Data Objects that form a single data burst determines the file size, and the stream duration determines the number of files. Both of those parameters will be configurable at runtime for each DAQ streaming plugin. This effectively enables continuous saving of the DAQ data, limited only by the available storage size.

It is also worth mentioning that a subset of DAQ streaming plugins will keep circular arrays of DAQ PV Data objects, and will be able to start streaming data from some point in the past, and until some point in the future. This may be used for triggered data acquisition and may also aid diagnostics and troubleshooting efforts after certain events (e.g. post-mortem analysis of unscheduled beam dumps).

DAQ Services

The Data Distributor service will be responsible for accepting DAQ PVD objects that are part of a data burst, and forward those objects to any applications or external services subscribed to the data generated by a particular technical system. If the burst should be stored as an SDDS file, the DAQ PVD objects will also be sent to the Data Collector service.

The Data Collector service will accept DAQ PVD objects, process them as needed, and save them into SDDS [5] files, so that each SDDS file corresponds to one data burst. This service will also keep track of files that belong to a stream. After completion of each burst, as well as upon completion of each stream, the Data Collector will notify any external services responsible for processing this data. Note that the Data Collector service will be able to operate and accept data independently of the Data Distributor. This provides DAQ IOCs with an option to stream data directly to the Data Collector.

Both Data Distributor and Data Collector services will be able to accept simultaneous connections from multiple clients such as streaming plugins and various client applications. They will also be able to handle data generated by all DAQ technical systems and by multiple DAQ IOCs. These features will go far in ensuring system scalability and robustness.

DAQ PVD objects will be exposed to client applications via EPICS PV Access protocol [4], which enables usage of standard EPICS tools and user interfaces. However, if need arises, we will also keep in mind a possibility of using additional protocols that might be more efficient than PVA for those use cases where large number of client applications may be interested in the same data. One particularly appealing option for this is using ZeroMQ Message Transport Protocol and ZeroMQ APIs [6], which provide generalized sockets that represent many-to-many connection between endpoints. Note that this would require providing clients with DAQ ZeroMQ-based APIs for accessing the data (i.e., for converting raw ZeroMQ messages to PVD objects).

In addition to the Data Distributor and Data Collector, the DAQ system will provide a number of other services, including the following:

- SDDS to PVD Object Service
- Area Detector Image Service
- Data Correlation and Alignment Service
- Orbit Service
- Injected Bunch Diagnostics Service

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUDPP02

All of those services will be deployed with a goal of aiding machine commissioning, troubleshooting, diagnostics, user analysis and monitoring.

DAQ Clients

As mentioned earlier, all DAQ PVD objects will be accessible to client applications and users via standard EPICS PVA command line (CLI) and graphical (GUI) tools, as well as via PVA APIs in various supported programming languages (C++, Java, Python, etc.). We also anticipate the need to provide real-time access to DAQ data in SDDS and Matlab/Octave. In addition to this, the system will provide DAQ-specific APIs and GUI/CLI tools that will allow configuring data bursts in terms of size and channels of interest, retrieving data bursts, monitoring and troubleshooting IOCs and services, etc. One such tool being developed is APS C2 Data Viewer [7], which is capable of displaying streams of multiple scalar array fields of DAQ data objects simultaneously in real-time (Fig. 6).

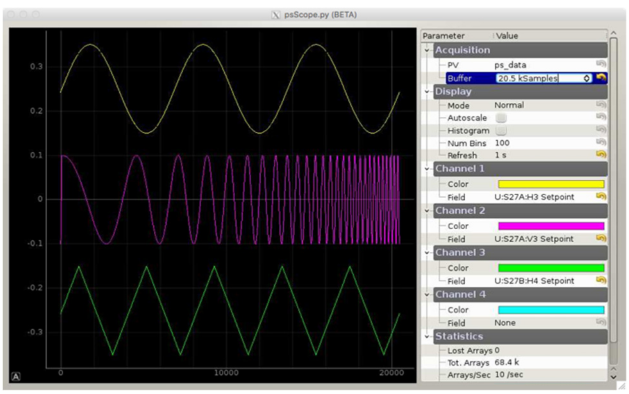

Figure 6: APS C2 Data Viewer application displays streaming DAQ data.

DATA MANAGEMENT

It is likely that most of the data generated by the DAQ system will not reach any client application or be saved in storage. Nevertheless, the overall design of the DAQ system must consider various data management needs:

- Storage area management
  Handle data movement from local (short-term) storage to a more permanent location (medium/long-term storage)
  Data archival
- Enable users and applications to easily find and access data
  Metadata catalog
  Integration with Component Database [8, 9, 10]
- Facilitate data processing and analysis
  Real-time (or near real-time) processing of the DAQ data using automated workflows
  User-initiated processing and analysis of data

The need for data management is not unique to APS Upgrade. Similar functionality is typically required for any system that deals with large volume of continuously collected data (e.g., any APS beamline). Rather than designing and developing APSU data management system from scratch, we propose to utilize and adapt as much as possible the existing Data Management (DM) solution [11] that is being developed and maintained for APS beamlines. This promotes software reuse and also minimizes data management software development effort required by the APS Upgrade.

Figure 7 shows the DAQ/DM system integration diagram. On every Data Collector node DM Data Acquisition Service will be responsible for live disk monitoring, data uploads, and Metadata Catalog updates. Depending on its configuration, Data Collector may notify the DM Data Processing Service every time data burst is saved into an SDDS file, or after an entire data stream is completed, which may trigger automated workflow-based processing. Users may request uploading of certain datasets to storage, interact with the Metadata Catalog (or Component Database) to find files they are interested in, or submit analysis jobs to the DM Data Processing Service. Data in medium/long-term storage will be managed by the DM Data Storage Management Service.

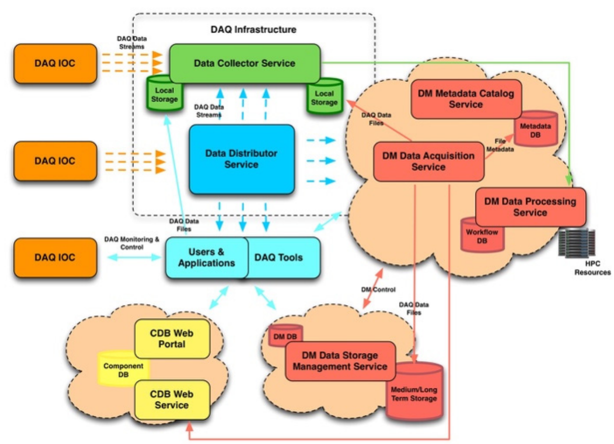

Figure 7: DAQ/DM System Integration.

IMPLEMENTATION STATUS AND FUTURE DEVELOPMENT

During the MBA R&D phase we have implemented DAQ IOCs for five technical subsystems for the APS Upgrade, two additional IOCs for the APS Injector Particle Accumulator Ring, along with the prototype Data Collector and Data Distributor,[1] as well as with SDDS to PVD Object Service. The prototype DAQ system has been used in numerous machine studies, and has also aided in eliminating the 147Hz vibration source in the APS storage ring [12].

We have successfully completed initial integration of the DAQ system with the DM software, and used it for prototypes of real-time processing applications. The DM system

[1] We have used EPICS version 7 P2P gateway [3] as our initial Data Distributor service.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUDPP02

is also used for automated management of the APS Upgrade Magnetic Measurement data.

Plans for the future work include adding several missing features to the DAQ IOC framework, development of additional DAQ IOCs, development of new services to aid data analysis, enhancing the existing functionality, as well as the system production deployment.

CONCLUSIONS

In this paper we discussed requirements and design of the Data Acquisition System for the APS Upgrade multi-bend achromat accelerator. The system will interact with a number of technical subsystems in order to provide time-correlated and synchronously sampled data acquisition for the new machine commissioning, troubleshooting, fault detection and performance monitoring.

REFERENCES

[1] APS-U Project Team, "Advanced Photon Source Upgrade Project Final Design Report," *APS Technical Report*, APSU-2.01-RPT-003, 2019.

[2] areaDetector documentation, https://cars.uchicago.edu/software/epics/areaDetectorDoc.html

[3] Experimental Physics and Industrial Control System (EPICS) documentation, https://epics.anl.gov

[4] PVA specification, https://github.com/epics-base/pvAccessCPP/wiki

[5] Advanced Photon Source, SDDS documentation, https://ops.aps.anl.gov/SDDSInfo.shtml

[6] ZeroMQ website, https://zeromq.org/

[7] APS C2 Data Viewer, https://git.aps.anl.gov/C2/C2DataViewer

[8] S. Veseli, N. D. Arnold, J. Carwardine, G. Decker, D. P. Jarosz, and N. Schwarz, "Component Database for APS Upgrade", in Proc. 15th Int. Conf. on Accelerator and Large Experimental Control Systems (ICALEPCS'15), Melbourne, Australia, Oct. 2015, pp. 1127-1130. doi:10.18429/JACoW-ICALEPCS2015-THHC2O02

[9] D. P. Jarosz, N. D. Arnold, J. Carwardine, G. Decker, N. Schwarz, and S. Veseli, "The Evolution of Component Database for APS Upgrade", in Proc. 16th Int. Conf. on Accelerator and Large Experimental Physics Control Systems (ICALEPCS'17), Barcelona, Spain, Oct. 2017, pp. 192-194. doi:10.18429/JACoW-ICALEPCS2017-TUBPA01

[10] D. P. Jarosz et al., "Tracking APS-U Production Components with the Component Database and eTraveler Applications", presented at the 17th Int. Conf. on Accelerator and Large Experimental Control Systems (ICALEPCS'19), New York, NY, USA, Oct. 2019, paper WEMPL009.

[11] S. Veseli, N. Schwarz, and C. Schmitz, "APS Data Management System," *J. Synchrotron Rad.*, vol. 25, no. 5, p. 1574, Sep. 2018. doi:10.1107/S1600577518010056

[12] L. Emery, Argonne National Laboratory, Advanced Photon Source, unpublished information.