EIC CONTROLS SYSTEM ARCHITECTURE STATUS AND PLANS*

J. P. Jamilkowski†, S. Clark, M. Costanzo, T. D'Ottavio, M. Harvey, K. Mernick, S. Nemesure,
F. Severino, K. Shroff, K. Kulmatycski, C. Montag, V. H. Ranjbar, K. S. Smith
Brookhaven National Laboratory, Upton, USA
L. B. Dalesio, Osprey DCS LLC, Ocean City, USA

Abstract

Preparations are underway to build the Electron Ion Collider (EIC) once the Relativistic Heavy Ion Collider (RHIC) beam operations end in 2025, providing an enhanced probe into the building blocks of nuclear physics for decades into the future. With commissioning of the new facility in mind, Accelerator Controls will require modernization in order to keep up with recent improvements in the field as well as to match the fundamental requirements of the accelerators that will be constructed. We will describe the status of the Controls System architecture that has been developed and prototyped for EIC, as well as plans for future work. Major influences on the requirements will be discussed, including EIC Common Platform applications and our expectation that we'll need to support a hybrid environment covering both the proprietary RHIC Accelerator Device Object (ADO) environment as well as EPICS.

NEARING THE END OF THE RHIC ERA

Beam operations of the Relativistic Heavy Ion Collider (RHIC) are expected to cease after the FY25 run, having successfully supported several key detectors that have been used for nuclear physics experiments over 23 years. Colliding beams in use at the facility have been comprised of many hadron species including polarized protons and gold with beam energies between 3.85 GeV [1] and 255 GeV [2] to date.

The RHIC Controls System includes Front End Computers (FECs) that utilize either VME-standard components or a proprietary XILINX FPGA-based platform. The latter was primarily developed and utilized to support the LLRF system at RHIC and elsewhere in the hadron injector chain. VME boards in use at RHIC include a significant component of proprietary boards for multiple systems as well as commercial equipment. Additionally, the software infrastructure of the Controls System is based on a proprietary standard called Accelerator Device Objects (ADOs) [3]. The current version of the Controls System was developed in preparation for the commissioning of RHIC, though it has continued to develop and expand in scope and complexity as the needs of RHIC and other independent machines have continued to evolve.

WHAT IS THE EIC?

The US Department of Energy (DOE) approved the siting of the Electron Ion Collider (EIC) [4] at Brookhaven National Laboratory (BNL) in FY21, and a partnership has been established with Thomas Jefferson National Accelerator Facility (TJNAF) to construct the new machine after FY25 pending project approval. The RHIC tunnel will be reused for the new project, along with a subset of the RHIC magnets that comprise the existing Blue and Yellow Rings for a new Hadron Storage Ring (HSR) (Fig. 1). A new turnkey LINAC will be added to accelerate polarized electrons to 400 MeV, and the resulting beam is accelerated in a new Rapid Cycling Synchrotron (RCS) Ring. Bunches are then transferred from the RCS to a new Electron Storage Ring (ESR), which supports collisions with beam in the HSR. Center of mass collision energies between the electron and hadron beams will range between 20 and 140 GeV [5] at one or two Interaction Regions, the first of which will be serviced by the electron-Proton Ion Collider (ePIC) Detector. Beam from an additional Strong Hadron Cooling (SHC) electron accelerator will be used to reduce the size of the hadron bunches in both HSR injection and store conditions. Completion of the commissioning phase of the project is currently expected to occur in FY34.

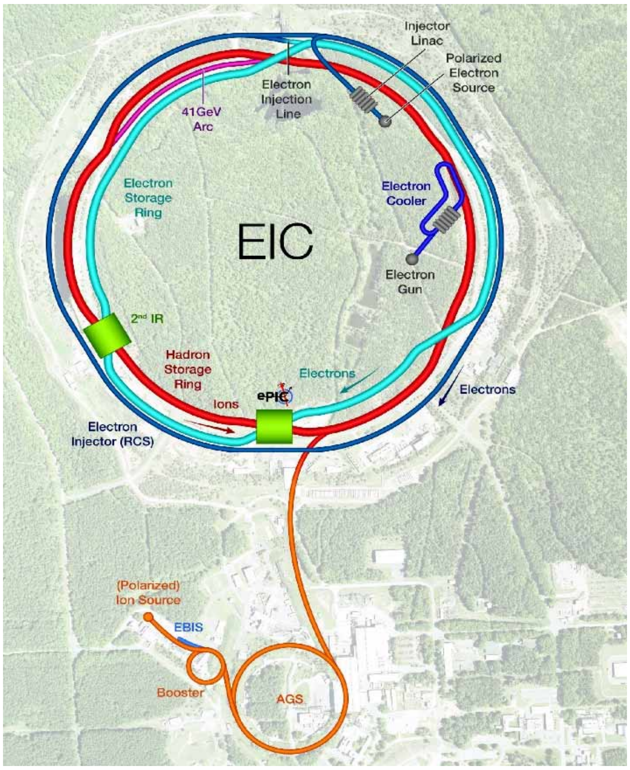

Figure 1: Cartoon Layout of the EIC.

The components required to achieve the functional and performance goals of the EIC Project will be typical of a large, modern particle accelerator. These include normal-conducting and super-conducting magnets, normal-

---

* Work supported by Contract Number DE-AC02-98CH10886 with the auspices of the US Department of Energy

† jpjamilk@bnl.gov

conducting and super-conducting RF cavities, a spectrum of power supplies and power amplifiers, a myriad of beam instrumentation equipment, and many types of support equipment that provide direct control and monitoring or establish the correct environmental conditions for the core systems. Noteworthy amongst the less common accelerator components, EIC will require the successful operation of multiple Crab RF cavities adjacent to the Experimental IR to optimize the transverse alignment of the electron and proton bunches as they enter and leave the IR [6]. These devices will help to maximize the luminosity of the collisions and will represent a significant engineering challenge for the Project.

EIC Controls Basics

EIC systems will require remote controls to many locations at the BNL site, which will most commonly be utilized from the existing Main Control Room (MCR) that serves RHIC and other machine operations within the Collider-Accelerator Department at BNL (Fig. 2).

Figure 2: Current view of the RHIC Main Control Room.

Field equipment will range from Commercial Off-The-Shelf (COTS) embedded devices from many different manufacturers to proprietary Front-End Computers (FECs) that help integrate the core functionality of many of the EIC systems that have been described. Since operations will require bunched beam circulating at high frequencies, a timing system is required that supports synchronized event and data distribution with high performance requirements. Standardized software tools will stitch together all hardware components of the EIC Controls System into a coherent set of interfaces as well as provide key services that are required to operate a distributed set of equipment in a modern environment. The underpinnings of the associated FECs and servers are the networking devices and the supporting fiber plant that provide the connectivity with sufficient performance for basic I/O, taking snapshots of activities on interesting events, data logging, data retrieval, as well as other common tools that operators of the EIC may require.

SOFTWARE STRATEGY

Since there is the potential for reuse of elements of the RHIC Controls System for the EIC era, we are reviewing the functional and performance requirements and the general utility of the associated codebase to determine if there will be a desire to adapt certain cases to our future EIC paradigm.

One example that has already been identified is the RHIC TAPE Sequencer application [7], which allows the user to easily create and run complex command processes in an on-demand fashion. It's anticipated that EIC will have very similar if not more demanding requirements for command sequencing, so we are considering improvements to the software design regarding separation of the User Interface from the sequencing processing back-end, in-process sequence management tools, as well as other new aspects of the implementation.

We are evaluating the inclusion of EPICS tools into the services and UI tools that will be the backbone of the EIC Controls, presenting the opportunity to cooperate on software development with other facilities. While the RHIC Controls System does not make significant use of EPICS resources, we have consulted with the Controls Systems experts at the other major particle accelerator facilities including National Synchrotron Light Source II (NSLS-II) at BNL. The expertise that they have developed using EPICS since that machine was first constructed in 2014 has been a positive influence on our own planning efforts [8]. We elect to standardize on using EPICS version 7 [9] along with PV Access protocol and will likely incorporate Phoebus [10] as a primary UI tool in the EIC Controls System.

Given the mixture of RHIC and EPICS tools that are available, we are currently evaluating the options for each of our core software-based services. For example, the EPICS Archiver appears to be a robust and extensible tool to deliver the necessary EIC data logging and retrieval services. We will prototype this system as part of our planning for the construction phase of the Project, where the current expectations are that we will need to support several million control points. Name lookup services have multiple options that we are looking to evaluate; Controls Name Server (CNS) from the RHIC system, and Channel Finder from EPICS. Performance metric comparisons will be critical to making our ultimate choices in each services category.

Another area of consideration is the choice of codebases for our services and tools including ADOs and EPICS, which are both a mixture of C/C++, Java, and Python. While the C and Java dependencies are significant and are unlikely to be retired during the EIC Project, new developers who are unfamiliar with accelerator controls systems that have joined our organization have expressed a preference for using Python. Much progress has been made in the RHIC Controls System with development of the Python development environment for the purpose of standardizing and packaging code in ways that meet the performance expectations of the user community [11]. Python code is now used in both field device integration and UI applications in an increasing number of cases. We expect that this trend will increase, due to the ease of development of Python programs and the wide array of software libraries that are being actively developed and maintained.

While Artificial Intelligence and Machine Learning were not significant considerations when RHIC was being designed and constructed in the 1990's, the level of community interest for these areas of software development and data management has greatly increased over the last several years. AI/ML tools will need to be well-integrated with the EIC Controls System, and our intent is to leverage non-proprietary software that is dynamic enough to easily support the addition of new training data sets and new applications by the EIC Controls user community at any point. We are keenly interested in gaining knowledge about the AI/ML tools that are already well-established in the accelerator controls community, how they're being used, and how we might best integrate them into our data pipeline.

Software Challenges

Any significantly sized distributed controls system will require regular software maintenance due to the life cycle of communication, UI, and other libraries that underpin the framework. With the inception of a large development project, it's a critical decision point to consider existing choices in each area. Regarding device interfacing, we feel that including an avenue to leverage the existing RHIC ADO infrastructure is warranted. There may be situations where EPICS IOC interfaces do not exist for COTS or proprietary equipment that are already available in ADO form. Conversely, there will be situations where we could avoid novel ADO development efforts by leveraging existing EPICS IOCs for COTS equipment.

EIC design plans call for the use of a LINAC to accelerate electron beams between the source output and RCS injection energies, and we are preparing to procure the related systems from an outside source. All controls would be provided by the LINAC vendor, and the logical standard to leverage is EPICS. Additionally, the EIC Project has selected an existing Power Supply Controller (PSC) that is being developed for the Advanced Light Source Upgrade (ALS-U) Project at Berkeley Lab as the basis for power supply controls for the EIC RCS and ESR Rings. While we will have to adapt the digital interface to work with our timing system and different modes of operation, we would leverage the EPICS software interface in the EIC Controls System.

Due to our anticipated combination of ADO and EPICS device-level interfaces, we have considered several different options for hybridization of the two similar but incompatible software platforms. Figure 3 shows two of the leading options, where clients make use of the existing ADO or EPICS standards depending on the case. The choice that we are focusing on now is the EPICS-standard case that would require us to support EPICS PV Access protocol from ADOs, which was first demonstrated in a test environment for EIC in 2021. We continue to explore the related interface changes that would be required to complete the hybridization process, including mapping ADO property alarm traffic into the existing EPICS parameter value alarm distribution process.

Software Architecture

Based on our internal review of the available RHIC and EPICS software tools and external input from experts in the EPICS community, we have developed an architecture that continues to evolve as we gain more prototyping experience and improve our understanding of the functional and performance requirements from all stakeholders on the Project (Fig. 4).

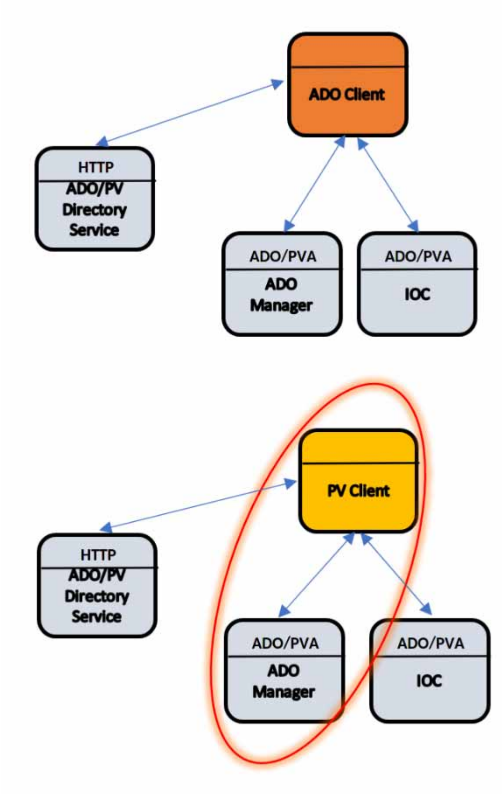

Figure 3: Two of the ADO/EPICS hybridization design options that have been considered.

As has been described previously, elements of both RHIC ADO and EPICS software are expected to play key roles in the overall EIC Controls System. The core solidifying design choice is standardization around the PV Access protocol from EPICS as the binding mechanism between the devices/services layers and the UI applications layer.

HARDWARE STRATEGY

Whereas RHIC relies heavily on VME-based interfaces using the VxWorks RTOS, the desire is to reduce the reliance on the aging VME standard in as many of the EIC systems as possible. Other bus-driven Front End Computer standards have been considered as a replacement, including MTCA and VPX. However, another alternative scheme has become the core design of the future EIC FEC platform after years of experience developing a previous generation. RHIC required a higher performance platform for RF systems control, and a modular XILINX FPGA-based system was first created in 2009 to meet those needs. Using a pizza-box form factor and Carrier/Daughter Card layout, it allowed the RF controls to flexibly adapt to many individual deployments and provide a reliable service [12]. Our

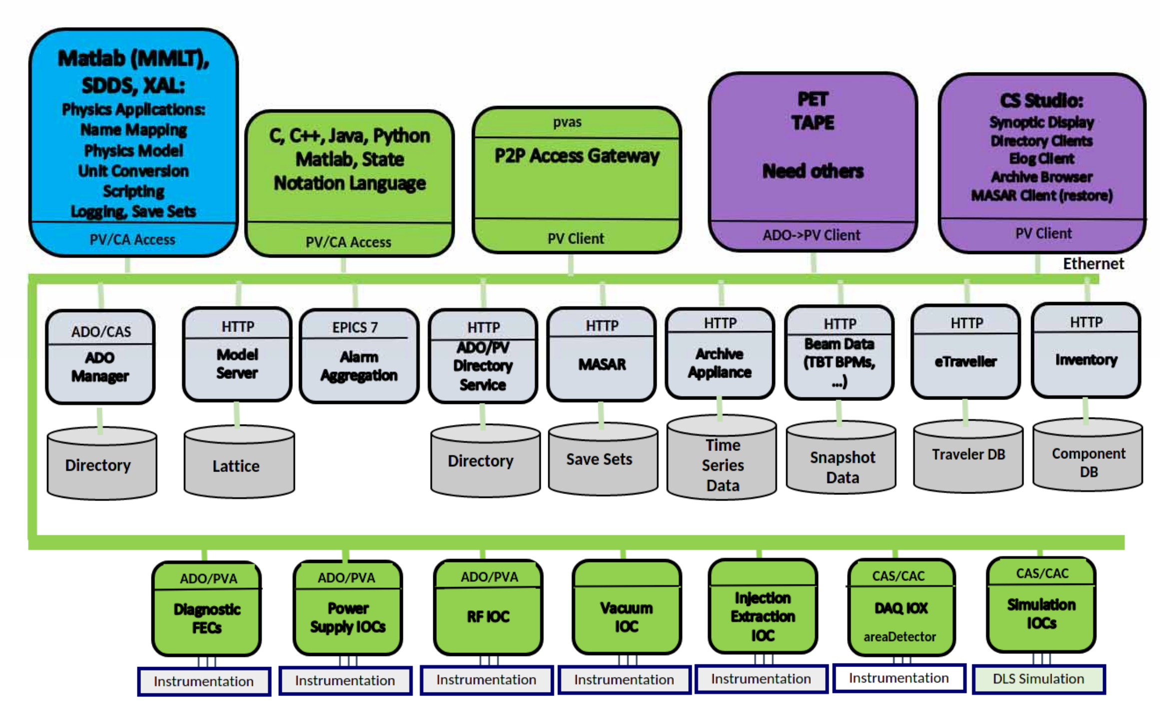

Figure 4: Diagram of proposed EIC Software Architecture.

EIC design borrows the main concepts of this existing platform including a single serial link for distribution of data and timing information, with modern enhancements. The new platform carefully considers reusability of the design in more than one generation of FPGA to minimize recurring engineering costs as the life cycle of those components churns onwards.

Another concept from the earlier RF platform that has influenced the new design is that timing distribution at the FEC level should be supported with a maximum of flexibility in the performance requirements. For example, delay outputs of a particular chassis could support both delays with coarse synchronization requirements as well as delays with much tighter synchronization and jitter requirements for RF or instrumentation applications. The goal is to achieve this flexibility without increasing the cost of the platform in a significant manner.

The experience operating the RHIC Controls System has highlighted concerns with the increasing number of client applications and services that represent a load to each FEC, with certain popular systems providing more extreme examples. Excessive client load can interfere with delivery of data published by the FEC, contribute to memory overutilization, and has required manual intervention to recover from various forms of system crashes over decades of machine operations. Additionally, administratively burdensome schemes had to be developed to mitigate the client load concerns, with varying levels of improvement to the reliability of the systems and maintainability of the configurations. These situations have influenced our considerations for integrating FECs into the EIC Controls environment. It is our preference that FECs will be designed to interface with middleware running on external servers to manage all client I/O traffic [13], rather than rely on internally hosted server processes that may not be able to scale well to meet increasing demands. The basic concept is that the FEC hardware is more expensive to upgrade to handle upward trends in client load than typical server hardware, which can be purchased off-the-shelf and has continued to improve in terms of cost and performance over the last two decades.

Hardware Challenges

One uncommon challenge presented by the constraints of the RHIC Tunnel on the EIC Controls design planning are the use of the existing Tunnel Alcoves for hosting a subset of power supplies for magnets as well as certain control electronics for beam instrumentation. Each alcove is situated close to the RHIC Blue and Yellow Rings, providing short cable runs and thereby keeping performance requirements in check and reducing costs. The impact on the related controls equipment is that the proximity to the Blue and Yellow beamlines leave any FECs in those locations in a radiation environment during beam operations.

Single Event Upsets (SEUs) are not uncommon and tend to have an impact on operations by delaying tasks or triggering software and hardware failures. The mitigation plan for SEUs involves minimizing the number and scope of the control interfaces that need to be installed in the Alcove locations. Certain equipment will be wholly removed from the Tunnel, which is the preferred option where feasible. In cases where the field devices cannot themselves be moved from the Alcoves, we are developing a strategy to deploy our platform to external locations due to concerns about increased sensitivity to SEUs and utilize fiber optic connections to the equipment in the Alcoves.

As we are planning a modular platform with a proprietary design that must interface with many varieties of field equipment, it will be important to streamline the component design of our FECs as much as possible. This will encourage our development processes to emphasize additional modularity in the firmware and software interfaces, such that we can easily reuse core components in multiple designs. The concept is that it can be used by many groups within the EIC Project to meet disparate goals without significant divergence of the individual component and interface designs.

Figure 5: A mockup version of a Common Platform chassis.

Common Platform Hardware Design

Our previous experience has influenced the new proprietary Common Platform FEC design: within a 2U chassis, resides a Zynq FPGA-based Carrier module and up to two Daughter Cards to meet the application requirements (see Fig. 5). Each Daughter Card can optionally interface with signal conditioning modules to achieve physical interface standardization.

Since certain applications will require a higher density of I/O but not require high performance data distribution or processing, we have a concept under development called the Extended Carrier Interface (ECI). Each ECI unit is proposed to support up to two Daughter Cards, and up to three ECIs can be connected to a single Carrier board via an SFP+ interface (see Fig. 6). The resulting multi-chassis setup is expected to reduce costs in the situations where it may fit the performance requirements, which have yet to be determined. Applications under consideration include RF, instrumentation, and controls I/O interfaces.

Several Daughter Cards are currently at different stages of development: SFP+ Breakout module [14], Digital I/O module, Baseband ADC module, BPM module [15], RF ADC, and an RF DAC module. We expect to start testing the first spin of the Carrier board and SFP+ Breakout module in the Fall of CY23. One concept that is planned for specific Common Platform Daughter Cards is to include standard FMC compatible interfaces to leverage readily available COTS hardware rather than replicating all related functionality within our proprietary designs.

Figure 6: A rendering of the Common Platform Carrier board.

Timing System Architecture

The EIC timing system will be widely distributed around the Complex, where different systems will have varying sensitivity to synchronization and jitter. Another paper will discuss the application of the EIC Common Platform and the SFP+ Breakout Daughter Card to meet the related timing and data distribution requirements. The Timing and Data Link (TDL) is based on the existing serial Update Link in use with the RF platform at RHIC, though it will operate at higher speeds.

NETWORK AND SERVER STRATEGY

With the new Common Platform generation of FEC hardware and larger number of modern embedded devices planned to be integrated into the EIC Controls System over an Ethernet network, efforts are underway to revise the topology of the network and consider improvements to the fiber plant. Higher capacity links will be needed to meet the performance requirements, with higher bandwidth needs pushing further into the branches than under the RHIC design. Key concerns are the load from FEC data publishing, Gigabit Ethernet camera imagery distribution, and the peak bandwidth expected when snapshot-worthy events occur in any of the major machines that make up the EIC.

To meet user demands, machine operating requirements, and minimize costs, we are planning to utilize server virtualization as much as possible. While we have not selected a distribution yet, we anticipate using a form of Linux on all servers and workstations.

Cyber security requirements will continue to increase, and they must be factored into our network, server, FEC, and workstation deployment planning at the current stage of the Project. Continuous background monitoring and per-device vulnerability analysis prior to connecting equipment will be cornerstones of our efforts to mitigate security concerns, as will use of authentication tools in support of limiting remote access to the accelerator controls.

While the RHIC network is largely based on IPv4 traffic, we will be expected to select or build networked devices that use the IPv6 standard by our EIC stakeholders. We anticipate challenges in obtaining COTS equipment in certain areas that will comply with IPv6, and there will also be a need to contribute to IPv6 support and testing within the software community to achieve a good result.

CONCLUSION

With the end of RHIC beam operations approaching in FY25, design activities for the EIC Controls System are ramping up. EPICS will be used for core services and at least a portion of the device integration that will be required, though we also anticipate supporting certain software tools and interfaces from the existing RHIC Controls System. A new Common Platform design has been developed as a replacement for the RHIC FEC platforms, and we are preparing to prototype the initial components and complete the design work for the additional equipment that will be required to meet the requirements for the EIC Project. Timing System and networking infrastructure planning are underway, including considering changes to the fiber plant and network topology.

REFERENCES

[1] C. Liu *et al.*, “Accelerator Performance During the Beam Energy Scan II at RHIC in 2019”, in *Proc. NAPAC'19*, Lansing, MI, USA, Sep. 2019, pp. 26-30. `doi:10.18429/JACoW-NAPAC2019-MOYBA6`

[2] V V. H. Ranjbar *et al.*, “RHIC Polarized Proton Operation for 2013”, in *Proc. IPAC'13*, Shanghai, China, May 2013, paper TUPFI084, pp. 1544-1546.

[3] K. A. Brown *et al.*, “The RHIC and RHIC Pre-Injectors Controls Systems: Status and Plans”, in *Proc. ICALEPCS'11*, Grenoble, France, Oct. 2011, paper MOBAUST04, pp. 13-16.

[5] D. Xu *et al.*, “EIC Beam Dynamics Challenges”, in *Proc. IPAC'22*, Bangkok, Thailand, Jun. 2022, pp. 1576-1582. `doi:10.18429/JACoW-IPAC2022-WEIXGD1`

[4] BNL EIC, `https://www.bnl.gov/ec/files/eic_cdr _final.pdf`

[6] Q. Wu, S. U. De Silva, Y. Luo, J. A. Mitchell, and B. P. Xiao, “EIC Crab Cavity Multipole Analysis”, in *Proc. IPAC'21*, Campinas, Brazil, May 2021, pp. 2589-2591. `doi:10.18429/JACoW-IPAC2021-WEPAB006`

[7] T. D’Ottavio, J. vanZeijts, T. Satogata, J. T. Morris, and B. Frak, “Description of the RHIC Sequencer System”, in *Proc. ICALEPCS'01*, San Jose, CA, USA, Nov. 2001, paper THAP062, pp. 597-599.

[8] F. J. Willeke, “Commissioning of NSLS-II”, in *Proc. IPAC'15*, Richmond, VA, USA, May 2015, pp. 11-16. `doi:10.18429/JACoW-IPAC2015-MOYGB3`

[9] EPICS, `https://epics-controls.org/`

[10] CS-Studio Pheobus Documentation, `https://control-system-studio.readthedocs.io/en/latest/`

[11] S. L. Clark, P. S. Dyer, and S. Nemesure, “Standardizing a Python Development Environment for Large Controls Systems”, in *Proc. ICALEPCS'21*, Shanghai, China, Oct. 2021, pp. 277-280. `doi:10.18429/JACoW-ICALEPCS2021-MOPV049`

[12] F. Severino *et al.*, “Embedded System Architecture and Capabilities of the RHIC LLRF Platform”, in *Proc. PAC'11*, New York, NY, USA, Mar.-Apr. 2011, paper MOP296, pp. 672-674.

[13] S. Clark *et al.*, “Reflective Servers: Seamless Offloading of Resource Intensive Data Delivery”, presented at the ICALEPCS’23, Cape Town, South Africa, Oct 2023, paper THMBCMO07, this conference.

[14] P. Bachek *et al.*, “Development of a Timing and Data Link for EIC Common Platform”, presented at the ICALEPCS’23, Cape Town, South Africa, Oct 2023, paper MO4O05, this conference.

[15] R. Michnoff *et al.*, “Beam Instrumentation Hardware Architecture for Upgrades at the BNL Collider-Accelerator Complex and the Future Electron Ion Collider”, presented at the IBIC’23, Saskatoon, Canada, Sep. 2023, unpublished.