ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOAPP01

CONTROL SYSTEM OF SuperKEKB

H. Kaji*, A. Akiyama, T. Naito, T. T. Nakamura, J. Odagiri, S. Sasaki, H. Sugimura
High Energy Accelerator Research Organization, Ibaraki, Japan
Y. Iitsuka, N. Yoshifuji, East Japan Institute of Technology Co. Ltd., Ibaraki, Japan
K. Asano, M. Hirose, Kanto Information Service Co. Ltd., Ibaraki, Japan
T. Aoyama, M. Fujita, Y. Kuroda, T. Nakamura, K. Yoshii
Mitsubishi Electric System & Service Co. Ltd., Ibaraki, Japan

Abstract

The control system of the SuperKEKB collider covers its 3 km beamlines. It is the distributed control system based on EPICS. There are about 200 input/output controllers which are built at the Central Control Building and 26 Sub-Control Buildings. The Operation Interface is prepared with the Python and SAD scripts and CSS BOY. The SuperKEKB control system is configured on the dedicated network which is segmented from the KEK office network by the firewall. There are several middle-layer services which are developed and operated on the server computers. The monitoring system, alarm system, data archiver system, and electronic log system are developed with open-source software. Abort Trigger System, Beam Permission System, and the timing system have their own network and implement the fast and robust transferring of the operation-related information. Abort Trigger System collects about 260 abort request signals and delivers the enable signal to the abort kicker pulse. Beam Permission System ensures the beam operation. The timing system realizes the extremely complicated injection operation of SuperKEKB. Our control system is strongly enhancing the progress of the SuperKEKB project and its future prospect is promising.

INTRODUCTION

SuperKEKB [1] is an electron-positron collider that is built at KEK. It aims the world's largest luminosity record of $8 \times 10^{35}\ \mathrm{cm^{-2}s^{-1}}$. The operation is started in 2016. The first collision of two beams is realized in May 2018. The world's smallest beam size at the interaction point is achieved in June 2018. The physics run with the top-up filling operation is carried out in the 2019 spring run.

SuperKEKB has 10000 hardware components that are installed along the 3 km beamlines. Therefore the distributed control system is constructed in the Central Control Building (CCB) and 26 Sub-Control Buildings (SCBs) as shown in Fig. 1. Our system is based on EPICS [2]. The hardware components can be controlled and operated via the Channel Access (CA) protocol.

Even though the operation condition becomes much more complicated than that of the previous KEKB project [3, 4], the operation and commissioning of SuperKEKB are smooth and steadily because of the sophisticated control system. We develop a lot of new systems and services for the SuperKEKB control system.

In this report, we introduce the basic components of the control system. Then, the services and systems of the SuperKEKB control system are discussed.

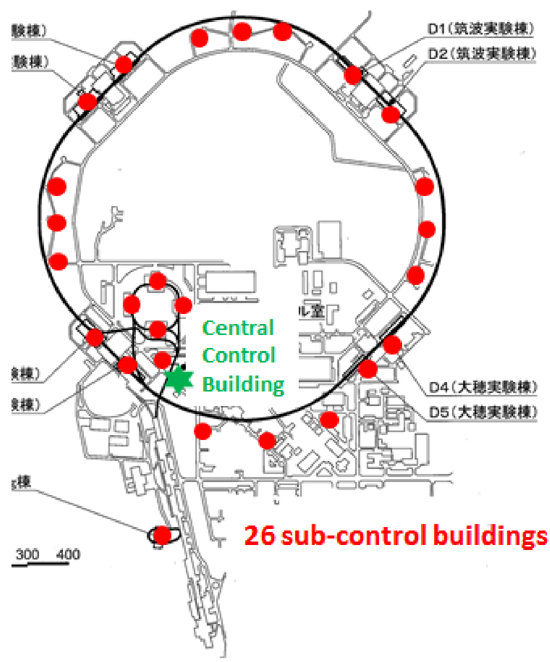


Figure 1: Location of Central Control Building, Sub-Control Buildings, and beamlines.

BASIC COMPONENTS

In this section, we introduce the basic components of the SuperKEKB control system.

Input/Output Controller

There are about 200 Input/Output Controllers (IOCs) on our EPICS control system. It is summarized in Table 1. To simplify the maintenance, most of the IOCs are constructed with the VME or PLC form-factors. In both form-factor cases, almost all IOCs are built with the SuperKEKB standard IOC, which is prepared by the control group.

* hiroshi.kaji@kek.jp

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOAPP02

THE SPIRAL2 CONTROL SYSTEM STATUS JUST BEFORE THE FIRST BEAM

C.H. Haquin, P. Anger, P.-E. Bernaudin, C. Berthe, F. Bucaille, P. Dolegieviez, C.H. Patard, D. Touchard, A.H. Trudel, Q. Tura, GANIL, Caen, France

Abstract

The SPIRAL2 Facility at GANIL is based on the construction of a superconducting LINAC (up to 5 mA - 40 MeV deuteron beams and up to 1 mA - 14.5 MeV/u heavy ion beams) with two experimental areas called S3 and NFS [1, 2]. At the end of this year, we will reach an important milestone with the first beam accelerated by the superconducting LINAC. The control system of the new facility relies on EPICS and PLC technologies. This paper will focus on the latest validated systems: machine protection system, the LINAC cryogenic system and the radio frequency system of the superconducting cavities. The validation requested a huge effort from all the teams but allow the project to be ready for this important moment.

INTRODUCTION

The SPIRAL2 facility will produce different beams (protons, deuterons and heavy ions) at very high intensity and will use actinide targets. Refer to Table 1 for beams specifications.

Table 1: Beam Specifications

Beam	Proton	Deuteron	Heavy Ions
Max Intensity	5 mA	5 mA	1 mA
Max. Energy	33 MeV	20 MeV/A	14.5 MeV/A
Max. Power	165 kW	200 kW	45 kW

SPIRAL2 is controlled by the French Nuclear Safety Authority (ASN) which means that the accelerator cannot be started without ASN authorization.

However, SPIRAL2 had a partial authorization allowing to produce protons to pre accelerate them with the Radio Frequency Quadrupole (RFQ) and study the beam in the medium energy beam line, refer to Fig. 1 for a view of the accelerator.

To obtain the authorization to start the whole accelerator (LINAC cavities and high energy beam lines), specific dispositions were taken to demonstrate the safety of the facility. Consequently, the validation of all the systems taking part in the safety was a prerequisite to obtain the authorization and start the commissioning of the ensemble.

MACHINE PROTECTION SYSTEM

Amongst the several systems involved in the safety surveillance, the machine protection system (MPS) [3, 4], a wide and central system in which the control system group was deeply involved, is responsible for the following functionality:

- Protect the beam pipes and insertion devices (slits, faraday cups, beam profile monitors, targets…) from thermal damages.
- Control the operating range of the facility.
- Control the accelerator device activation due to beam losses.
- Ensure a safety reinforced protection of the beam dumps and target, which all have their own protection system addressing beam off requests to the MPS
- Ensure a reliable and secure class protection of the safety class fast vacuum valves
- Provide an overview and interface of the system from the control rooms. The MPS transfers state-feedback of accelerator equipment and alarms to the EPICS interface, and receives instructions from it: handling insertion devices, acknowledgments, threshold management.

The MPS is made of the following subsystems.

The Thermal Protection Subsystem

This system consists in an interlock PLC communicating with fast electronics on one side and a GUI on the other side. It collects "beam off" requests from the beam diagnostics and, according to the machine state, gives a slow and a faster "beam stop" to reach the expected response time (< 10 ms). Slow beam stop is issued to insert beam dumps; faster beam stop order is issued to the Beam Time Structure Control Electronic. Though this system is not in the scope of the safety systems controlled by ASN, it can forbid the beam production, consequently it was validated progressively with the ions sources RFQ and the low and medium energy beam line commissioning.

Figure 1: View of the SPIRAL2 accelerator and beam lines.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOAPP03

CONTROL SYSTEM PLANS FOR SNS UPGRADE PROJECTS*

S. M. Hartman†, K. S. White‡
Oak Ridge National Laboratory, Oak Ridge, Tennessee, USA

Abstract

The Spallation Neutron Source at Oak Ridge National Laboratory is planning two major upgrades to the facility. The Proton Power Upgrade project, currently underway, will double the machine power from 1.4 to 2.8 MW by adding seven additional cryomodules and associated equipment. The Second Target Station project, currently in conceptual design, will construct a new target station effectively doubling the potential scientific output of the facility. This paper discusses the control system upgrades required to integrate these projects into the existing EPICS-based control systems used for the machine and neutron instrument beamlines. While much of the control system can be replicated from existing solutions, some systems require new hardware and software. Operating two target stations simultaneously will require a new run permit system to safely manage beam delivery.

SPALLATION NEUTRON SOURCE

The Spallation Neutron Source (SNS) at Oak Ridge National Laboratory (ORNL) is the world's most intense source of pulsed neutrons for research. The SNS consists of a 1 GeV superconducting hadron linac operating at 60 Hz, producing a 1.4 MW proton beam on target. An accumulator ring compresses the ~1 ms macropulse from the linac into a ~700 ns pulse. The pulse is extracted from the ring in a single turn and directed towards a liquid mercury target. Neutrons are spalled from the mercury nuclei, moderated to thermal or cold kinetic energies, and transported to experiment end-stations where they are used for a wide-range of science research.

Completed in 2006, the original SNS construction project was managed as a collaboration of six national laboratories with each of the major subsystems the responsibility of a partner laboratory. The control systems for the SNS project were managed as the "Integrated Control System" (ICS) at the same organizational level and reporting level as the primary facility components [1]. Experimental Physics and Industrial Control System (EPICS) [2] provided a common integration layer for subsystem controls delivered by partner laboratories. In addition to the use of EPICS for accelerator controls, EPICS was also used successfully for integration of industrial and process controls including utilities for the target systems, building automation systems for technical buildings, and process control for the cryogenics systems. EPICS, however, was not used initially for the neutron scattering beam lines and instrument data acquisition. A later multi-year upgrade migrated these systems to EPICS [3] with completion in early 2019.

SNS UPGRADE PROJECTS

Two projects are currently underway to substantively upgrade the SNS and expand the capabilities of the facility. The Proton Power Upgrade (PPU) Project will increase beam power. The subsequent Second Target Station (STS) Project will utilize this increased power to support a second experiment hall, effectively doubling the capacity for neutron scattering beam lines. The existing First Target Station (FTS) will remain optimized for thermal neutrons offering spatial resolution on the atomic scale delivered in short pulses optimized for fast dynamics studies of materials. The STS will be optimized as a facility to probe structure and dynamics of materials over extended length, time, and energy scales using longer wavelength cold neutrons pulsed at a lower repetition rate.

PROTON POWER UPGRADE

The Proton Power Upgrade (PPU) will double the SNS beam capability from 1.4 MW to 2.8 MW. This will be achieved through a 30 % increase in beam energy and a 50 % increase in beam current. 2 MW of power will be delivered to the FTS. The project includes installation of 7 additional superconducting cryomodules to the existing linac with supporting radio frequency (RF) systems, modifications to the ring injection region and extraction kickers for the higher beam energy, and improvements to the target systems for the higher beam power.

Accelerator and target controls upgrades are largely extensions of existing technologies. The control system will build on and leverage the existing EPICS-based control system.

PPU Controls

The PPU Project adds 7 new cryomodules to the linac. Controls for the existing cryomodules, which consist of Allen-Bradley ControlLogix programmable logic controllers (PLC) and EPICS VMEBus Input/Output Controllers (IOC), will mostly be replicated. However, to address obsolescence, some input/output (I/O) will be migrated from VMEBus to PLCs. Existing sequencers and automation routines will be updated to support the expanded linac. Beam line vacuum and cryomodule insulating vacuum will be replicated based on current PLC plus EPICS designs.

* This manuscript has been authored by UT-Battelle, LLC, under contract DE-AC05-00OR22725 with the US Department of Energy (DOE). The US government retains and the publisher, by accepting the article for publication, acknowledges that the US government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this manuscript, or allow others to do so, for US government purposes. DOE will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan (http://energy.gov/downloads/doe-public-access-plan)
† hartmansm@ornl.gov
‡ whiteks@ornl.gov

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOAPP04

STATUS OF THE NATIONAL IGNITION FACILITY (NIF) INTEGRATED COMPUTER CONTROL AND INFORMATION SYSTEMS

G. Brunton, A. Barnes, J. Castro-Morales, M. Christensen, J. Dixon, M. Fedorov, M. Flegel, R. Lacuata, D. Larson, A.P. Ludwigsen, D. Mathisen, V. Miller-Kamm, M. Paul, S. Townsend, S. Weaver, R. Wilson, B. Van Wonterghem
Lawrence Livermore National Laboratory, Livermore, USA

Abstract

The National Ignition Facility (NIF) is the world's most energetic laser experimental facility with 192 beams capable of delivering 2.1 megajoules of 500-terawatt ultraviolet laser light to a target. NIF experiments facilitate the study of extreme physical conditions at temperatures exceeding 100 million K and 100 billion times atmospheric pressure allowing scientists the ability to generate conditions similar to the center of the sun and explore the physics of planetary interiors, supernovae and thermonuclear burn. This year concludes a series of optimizations and enhancements to the control & information systems to sustain the quantity of experimental target shots while developing an enhanced precision diagnostic system to optimize and increase the power and energy capabilities of the facility. In addition, many new system control and diagnostic capabilities have been commissioned to increase the understanding of target performance. This year also concludes a multi-year sustainability project to migrate the control system software to Java. This talk will report on the current status of each of these areas in support of the wide variety of experiments being conducted.

INTRODUCTION

The National Ignition Facility (NIF) [1] provides a scientific center for the study of inertial confinement fusion (ICF) and matter at extreme energy densities and pressures [2]. Each NIF experiment, or shot cycle, is managed by the Integrated Computer Control System (ICCS) [3], which uses a scalable software architecture running code on more than 2300 front end processors, embedded controllers and supervisory servers. The NIF control system operates laser and industrial controls hardware interfacing with 66,000 control points (e.g. motors, calorimeters, sensors, etc) to ensure that all NIF's 192 laser pulses arrive at a target within 30 picoseconds of each other, are aligned to a pointing accuracy of less than 50 microns, and orchestrate a host of diagnostic equipment collecting experimental data in a few billionths of a second. Every NIF automated shot cycle [4] consists of approximately 2 million sequenced operations, such as beam path alignment, pulse shaping, and diagnostic configuration and each shot is typically conducted within 4-8 hours depending on the experiment complexity.

NIF has been a 24x7 operational facility since 2009 and has supported scientific advancement in various fields of physical studies such as High Energy Density (HED) experiments for Stockpile Stewardship, Inertial Confinement Fusion (ICF), National Security Applications and Discovery Science. The facility and control systems advancement has continued since becoming operational and many significant changes have occurred to increase its capabilities and efficiency since last reporting [5]. A summarization of the most recent enhancements is detailed in the following paper.

CONTROL SYSTEM STATUS

NIF Shot Rate Sustainment

As the NIF celebrates its 10th year of full-scale operations controls priorities continue to be sustaining a high system availability for maximizing the conduct of experimentation for all associated fields of research. With the deployment of new diagnostic capabilities, we continue to further advance our scientific capabilities and understanding however increased focus has been placed on modernizing the laser, controls and infrastructure to sustain many more years of valuable operation (Fig. 1).

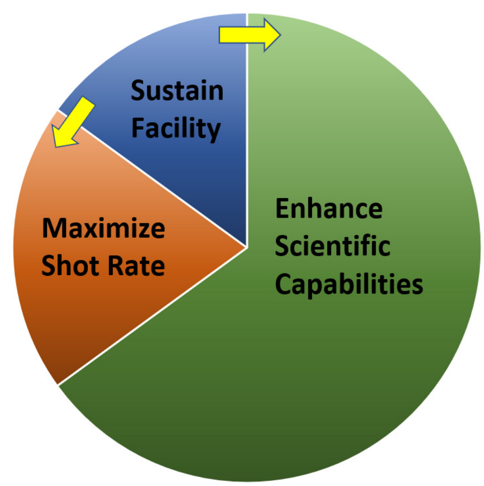


Figure 1: Balance of NIF primary priorities shifting to reduce risk to long term facility sustainment.

Although previous optimizations [6] identified major opportunities to improve the shot rate on NIF, the focus on efficiency has required to be continuous in order to offset the operational cost of increasingly complex capabilities and experimental configurations that are added annually (Fig. 2). Experimental configurations, such as the Advanced Radiographic Capability (ARC) [7] Petawatt laser is used with far greater frequency and these shot cycle configuration takes significantly longer than other target experiments (~12 hours per shot). Additionally, with greater

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOBPP01

PLCverif RE-ENGINEERED: AN OPEN PLATFORM FOR THE FORMAL ANALYSIS OF PLC PROGRAMS

D. Darvas, E. Blanco, CERN, Geneva, Switzerland
V. Molnár, Budapest University of Technology and Economics, Budapest, Hungary

Abstract

Programmable Logic Controllers (PLC) are widely used for industrial automation in industry and at CERN. The reliability of PLC software is crucial, but typically only testing is used to validate it. Our work targets the use of formal verification in practical ways for many years, which showed that it can be beneficial and practically applicable to various PLC programs. In this paper, we present PLCverif, our platform for formal analysis of PLC programs which has largely enhanced the quality of the deployed PLC software. By re-engineering the previous internal prototype tool, we built PLCverif to be an open, extensible platform that can be used not only for CERN's specific PLC programs. PLCverif is licensed under an open source license, allowing the interested parties to use and extend it.

INTRODUCTION AND MOTIVATION

Programmable Logic Controllers (PLCs) are widely used to implement process control systems and interlock systems. The incorrect behaviour of PLCs can cause service disruptions, consequently significant financial losses and injuries too in some cases, therefore ensuring their correct behaviour is essential.

Testing (mainly acceptance or system testing) represents the state of the art in PLC software quality assurance. While testing is effective in finding certain types of errors, it is often not sufficient as the sole verification method. Testing cannot be exhaustive, thus cannot prove the correctness of a system. In addition, it is very difficult to test the following types of requirements: *safety* (the system will never reach an unsafe state) or *invariant* (formulas which shall be true over all possible system run), as these errors may occur in very particular cases only.

Model checking can overcome some of the weaknesses of testing. This is a formal verification technique, which checks the satisfaction of a formalised requirement on a mathematical model of the system under analysis. It checks the requirement's satisfaction with every input combination, with every possible execution trace. In addition, if a violation is found, typically a trace leading to the violation (a counterexample) is provided. However, model checking is difficult to use by PLC developers who are not experts in this domain. The target of our work is to make model checking more practically applicable to the software of PLC-based systems by hiding the formal details, simplifying the user interaction and automating the process.

Our work is not the first that targets the formal verification of PLC programs. Among others, Arcade.PLC [1] and the toolset developed by VTT Technical Research Centre of Finland [2] both offer PLC program verification. However, none of the publicly available tools was applicable to the real-world PLC programs used at CERN, due for example, to the lack of support for the Siemens SCL language. In addition, we did not find possible to extend or adapt these tools for our use cases.

Previously, we have presented a methodology for practical model checking of PLC programs [3], a prototype tool that implements this workflow [4], as well as real-life case studies where model checking was proven to be beneficial [5, 6]. This paper reports about our re-engineering efforts done during the last two years and presents the final tool officially. This development made PLCverif richer in features, more robust and open to extensions. In addition, the paper discusses how *we* did benefit from PLCverif and how *users* can adapt it to their use cases.

DEVELOPMENT OF PLCverif

The first plans to evaluate the use of formal verification to PLC programs at CERN date back to 2012. After the initial experimentation phase, the design and development of the methodology used in PLCverif started in mid-2013. Within a year, a prototype version of PLCverif was developed. Already during the development, PLCverif was used to analyse parts of systems in production.

This prototype version was sufficient for our internal use cases. However, to make PLCverif more generally applicable, it had to be more robust, more generic and more extensible. To obtain the resources needed for this additional development, a CERN Knowledge Transfer Fund was requested and awarded in 2016. The proposed two-year-long re-engineering project was selected to be funded in mid-2016 [7]. The development project started in June 2017 and ended in May 2019. During that time, PLCverif was rebuilt from scratch, taking the previous experiences and knowledge into account.

The goal of this re-engineering work was to make PLCverif usable by any automation engineer at CERN and other interested parties outside the organisation.

PLCverif FOR USERS

This section discusses the principal use case of PLCverif from the users' point of view.

Verification Workflow

Out of the box, PLCverif offers a model checking workflow for the analysis of PLC programs. The verification workflow is shown in Figure 1 and it has the following main steps:

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOBPP02

DESIGNING A CONTROL SYSTEM FOR LARGE EXPERIMENTAL DEVICES USING WEB TECHNOLOGY

W. Zheng, Y. Wang, M. Zhang, F. Wu, N. Fu, S. Li
International Joint Research Laboratory of Magnetic Confinement Fusion and Plasma Physics, State Key Laboratory of Advanced Electromagnetic Engineering and Technology, School of Electrical and Electronic Engineering, Huazhong University of Science and Technology, Wuhan, China

Abstract

EPICS is mature in accelerator community. However, there are efforts to improve existing control system software like Tango and EPICS 7 mainly driven by the needs of flexibility of the control system and the development of computer technology. This paper presents a new way of building a large experimental device control system using web technology instead of EPICS toolkit. The goal is to improve the interoperability of the control system allowing different component in the control system to talk to each other effortlessly. An abstraction of the control system is made. The control system components are abstracted into resources. The accessing of the resources is done via standard HTTP RESTful web API. human machine interface is based on HTML and JavaScript in browsers. Web Socket is used for event distribution. The main feature of this design is that all interfaces in the system are based on open web standards, which are interoperable among almost all kinds of devices. The paper also presents a software toolkit to build this kind of control system. A control system for a diagnostic on J-TEXT tokamak built using this toolkit will be presented.

INTRODUCTIOIN

One key characteristic of as large experimental facility control system is its ability to adopt and integrate new systems. As experiment advances new diagnostics or detectors will be added to the machine. On J-TEXT currently there are more than one system been added of modified to the machine. Different types of subsystem are likely to be implemented with different technologies. New technologies are being integrated into those facilities frequently. Efficiently integrated those new technologies into the control system is very important. As the performance of the computer and network keep advancing performance of the supervisory control and data acquisition (SCADA) system is no long the focus of the development. Instead, interoperability became the priority of SCADA system. To achieve interoperability a common language for a control system is needed. There are various control system framework existing in big physics community, each with different characteristics. For fusion community, ITER chose the Experimental physics and industrial control system (EPICS) and Channel Access protocol (CA) as the common language. EPICS has been the common language to the accelerator control system for decades [1]. Now chosen by ITER, it is used by many tokamaks as well [2-5]. It is mature and well supported by the community. But the technologies used in tokamaks are different from those in accelerators. It is not a straight forward job to create EPICS support for equipment used in fusion experiment. EPICS CA was originally designed for performance not interoperability recently there are activities to improve the interoperability of EPICS [6]. Later emerged control system frameworks such as Tango uses object-oriented technique to improve interoperability and flexibility [7-9]. But still, it is hard to have all the equipment in a control system supporting the control system framework that you chose. Is it even possible to develop a control system protocol that everyone supports or it is necessary? There is a technology that is almost supported by all the devices, that's web. Countless web APIs have been published and consumed by all kinds of devices. If a control system is built on web, it could be supported by everyone effortlessly. This work was inspired by web technologies. We proposed an abstract model for control system and a framework that uses web technology to build a control system. It mainly addresses the interoperability issue of very large control systems in large experimental facilities. The proposed framework is based on simple and open web standard which has been used by the web industry for years. Therefore, anyone can implement a system that can be integrated into this control system with tools already available.

This paper first briefly talked about the web technologies and in section 3 we proposed the abstraction of the control system. Based on that abstraction the web technologies are introduced to make control system protocols. Then in section 4 software framework to implement the web based control system is described. Last an application example is presented.

WEB TECHNOLOGIES

Web technologies is an important part of our internet life. We keep using it every day. Web technologies not only power the web site. Today from mobile apps, online games, to smart sensor and IoT application, web plays a big role in them. Many devices have embedded web servers, and many client apps is running in browsers. They communicate using HTTP. So, what is web technologies exactly? There are different interpretations of web technologies. What is common is web is based on HTTP, HTML, and browser.

HTTP is an application protocol on top of TCP. It is HTTP to be specific HTTP/1.1 is a text-based request and response protocol. A client would send a request to a web server and get response. The request and response are fully in text. There are quite a lot of overhead here, but it boosts interoperability as text provide more redundant and easier

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOBPP03

FAULT TOLERANT, SCALABLE MIDDLEWARE SERVICES BASED ON SPRING BOOT, REST, H2 AND INFINISPAN

W. Sliwinski*, K. Kaczkowski†, W. Zadlo††, CERN, Geneva, Switzerland

Abstract

Control systems require several, core services for work coordination and everyday operation. One such example is Directory Service, which is a central registry of all access points and their physical location in the network. Another example is Authentication Service, which verifies caller's identity and issues a signed token, which represents the caller in the distributed communication. Both cases are real life examples of middleware services, which have to be always available and scalable. The paper discusses the design decisions and technical background behind these two central services used at CERN. Both services were designed using the latest technology standards, namely Spring Boot and REST. Moreover, they had to comply with demanding requirements for fault tolerance and scalability. Therefore, additional extensions were necessary, as distributed in-memory cache (Infinispan), or Oracle database mirroring using H2 database. Additionally, the paper will explain the tradeoffs of different approaches providing high-availability features and lessons learnt from operational usage.

INTRODUCTION

The potential for rapid growth of controls systems implies that every service has to be built to scale nearly instantly in response to growing requirements. CERN controls services are required to implement a high level of reliability, agility, and scale expected of modern computer systems.

High availability is a quality that aims to increase the time a service is available and it refers to systems that are durable and able to operate continuously without failure for a long time. It is generally achieved by scalability, failover and monitoring.

Two important CERN controls services, namely Authentication Service and Directory Service are examples of central, core services, which have to be always available, even during scheduled infrastructure upgrades or unexpected failures of dependent services. Both services are used in this paper to illustrate different architectural and design choices aiming at providing highly available, fault tolerant architecture, satisfying service requirements.

Authentication Service

Authentication Service (AS), at CERN part of the RBAC [1, 2] infrastructure, is a central authority, which verifies caller's identity, be it a human or an application, and issues a signed token, which represents the caller in the distributed communication. Users token holds several pieces of information, which are necessary to obtain access to protected resources, including: username, account type, list of roles, IP address and location name. AS provides several different types of authentication: explicit (username and password), location (trusted hosts by IP address), Kerberos [3] and SSO (SAML based Single-Sign On). This is made possible by aggregating different authentication mechanisms available at CERN and providing a common REST API [4] to all users. Figure 1 depicts the service architecture:

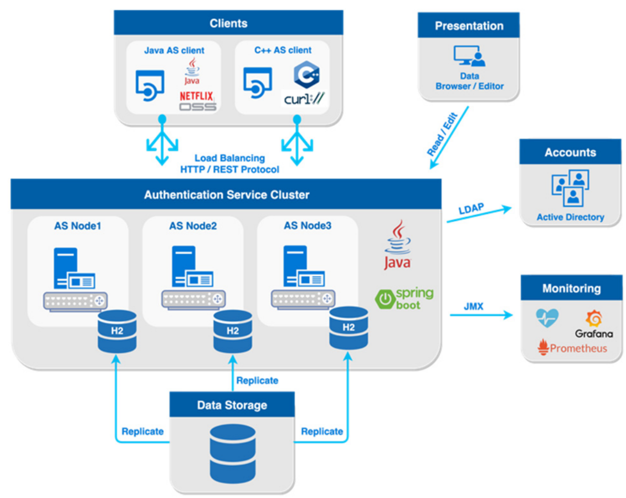


Figure 1: Authentication Service architecture.

CERN's controls middleware framework RDA3 [5] integrates with AS to provide security facilities (authentication and authorization) for RDA3 clients and servers.

Directory Service

Directory Service (DS) is a central registry of all access points in the distributed control system (Fig. 2). It provides up-to-date information about the actual physical location of a device server in the network. This is possible, because each device server has to register its current location during the start-up phase. Additionally, DS resolves logical device names to actual device servers and returns the location information to the client. Thanks to this, high-level applications don't need to know any information related to a device server; only unique device name is sufficient to initiate communication.

The RDA3 communication stack depends on DS for server's binding registration, device to server resolution and server's location and device lookup queries.

** Wojciech.Sliwinski@cern.ch
† Konrad.Kaczkowski@cern.ch
†† Wojciech.Zadlo@cern.ch

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOBPP04

THE ELT M1 LOCAL CONTROL SOFTWARE: FROM REQUIREMENTS TO IMPLEMENTATION

L. Andolfato†, J. Argomedo, C. Diaz Cano, R. Frahm, T. R. Grudzien, N. Kornweibel, D. Ribeiro Gomes dos Santos, J. Sagatowski, European Organisation for Astronomical Research in the Southern Hemisphere (ESO), Garching bei Muenchen, Germany
C. M. Silva, Critical Software, Coimbra, Portugal

Abstract

This paper presents the ELT M1 Local Control Software. M1 is the 39m primary mirror of the Extremely Large Telescope composed of 798 hexagonal segments. Each segment can be controlled in piston, tip, and tilt, and provides several types of sensor data, totalling 24000 I/O points. The control algorithm, used to dynamically maintain the alignment and the shape of the mirror, is based on three pipelined stages dedicated to collect the sensors' measurements, compute new references, and apply them to the actuators. Each stage runs at 500Hz and the network traffic produced by devices and servers is close to 1.2 million UDP packets/s. The reliability of this large number of devices is improved by the introduction of a failure detection isolation and recovery SW component. The paper summarizes the main SW requirements, presents the architecture based on a variation of the estimator/controller/adapter design pattern, and provides details on the implementation technologies, including the SW platform and the application framework. The lessons learned from deploying the SW on CPUs with different NUMA architectures and from the adoption of different testing strategies are also described.

INTRODUCTION

The European Southern Observatory is building the Extremely Large Telescope (ELT): one of the largest optical/near-infrared telescope in the world that will gather 13 times more light than the largest optical telescopes existing today. The telescope is located on top of Cerro Armazones in the Atacama Desert of northern Chile.

One key component of the ELT is the concave 39m primary mirror (M1) made of 798 quasi-hexagonal mirror segments of approximately 1.45m in size. M1 segments are controlled by the M1 Local Control System (M1LCS).

M1LCS prototyping activities started in 2011 with the goal to validate and consolidate the system design [1]. Final design review was passed in October 2017 and one month later started the development of the control SW. After less than 2 years of development, the first version of M1LCS control SW is being released.

SYSTEM DESCRIPTION

A detailed description of the M1 local control system is given in [2]. The segmented primary mirror of the ELT (Fig. 1), is composed of six sectors with 133 segments each. Segments within a sector are organized in flowers. One flower groups up to seven segments which are connected to a segment concentrator cabinet (SegC). There are 132 SegC cabinets and each cabinet hosts:

- the controllers for the field electronic devices (FE): edge sensor (ES), position actuator (PACT), and warping harness (WH);
- a Programmable Logic Controller (PLC) and a power supply unit (PSU) for power distribution control and temperature monitoring;
- a network switch connecting the PLC and FE controllers to the sector distribution (SecD) network switch that is connected to the computer room (CR).

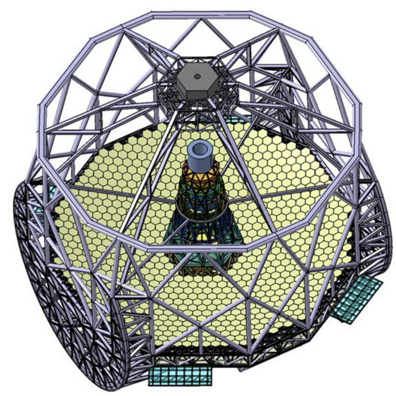


Figure 1: M1 primary mirror made of 798 segments within the ELT main structure.

Field Electronic Devices

The ES measure the relative out-of-plane (piston), and in-plane translation displacements (gap, shear) of a segment with respect to its neighbours. Each segment is equipped with six edge sensors with nm resolution.

The PACT are driving dynamically the segment in piston, tip and tilt in order to keep them aligned within the required accuracy under variable load conditions and disturbances. They are high accuracy linear actuators attaining nm resolution along a stroke of 10mm with internal feedback control.

The WHs are used to change the shape of the segment by applying different axial support forces to the mirror segment. This is achieved by a set of nine motors integrated in each segment support (Fig. 2).

† landolfa@eso.org

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-MOBPP05

DYNAMIC CONTROL SYSTEMS: ADVANTAGES AND CHALLENGES

S. Rubio-Manrique, G. Cuni, ALBA Synchrotron, Cerdanyola del Vallés, Spain

Abstract

The evolution of Software Control Systems introduced the usage of dynamically typed languages like Python or Ruby that helped Accelerator scientists to develop their own control procedures on top of the standard control system. This new high-level layer of scientist-developed code is prone to continuous change and no longer restricted to fixed types and data structures as low-level control systems used to be. This provides great advantages for scientists but also big challenges for the control engineers, that must integrate these dynamic developments into existing systems like user interfaces, archiving or alarms.

INTRODUCTION

ALBA [1], member of the TANGO Collaboration [2], is a third generation synchrotron light source in Barcelona, Europe. It provides synchrotron light since 2012 to users in its 8 beamlines, with 4 more in different stages of construction.

As a core member of the TANGO Control System Community [3], ALBA controls team has participated in the development of several tools and libraries shared across all institutes participating in the collaboration. Our main areas of development have been experiment control, GUI toolkits, alarm systems, archiving, simulation and dynamic user interfaces and device servers.

Control Systems in Modern Accelerators

Within the TANGO Community there are two clear different trends on developing accelerators control applications. Some of the institutes (ESRF, SOLEIL, Elettra, DESY) provide compact and uniform sets of applications to operators, applications mostly developed by control engineers. The newer institutes instead (ALBA, MaxIV, Solaris) have gradually taken a different path towards dynamically generated applications or user-developed interfaces on top of frameworks developed and deployed by the controls engineers [4].

This change has been gradual, as Java-based frameworks in TANGO already allowed some modularity and customizing of applications (ATKWidgets, JDraw), but the key factor has been the widespread usage of scripting languages (Matlab, Python) by accelerator scientists to write their own diagnostics and control software. An example of this practice is the Matlab Middle Layer [5] framework for accelerators control, that has become widely spread in the accelerators community, being used at ALBA to manage the Slow Orbit Feedback system [6].

Those scripting frameworks became a *de-facto* parallel control system and, although providing advantages to accelerator and beamlines scientists, but soon presented challenges in performance and unexpected behaviours in the control system, which required intervention from the Control team. These effects will be later explored in this paper, as well as the strategies used to cope with them.

Users as Developers

Despite the strategies that can be adopted from control and computing teams to keep up with control system changes, there's a fact that escapes from control teams and it's common for most scientific institutions: operators and scientists develop code on their own.

This fact has a practical explanation: most scientific careers include programming background, and the gap between programming and scientific languages is becoming smaller. Scientists feel enabled to translate to code their own ideas, and motivated scientists and long downtimes or operation shifts often lead to new toolkits or libraries developed by operators, scientists or users to enhance their daily work. These new tools often evolve from simple diagnostic tools to feedback control loops and GUI applications (Fig. 1), thus becoming at some point part of the control system. Sometimes it's a hard task for the Control engineers to adapt them to the integration/deployment workflows of the in-house controls team.

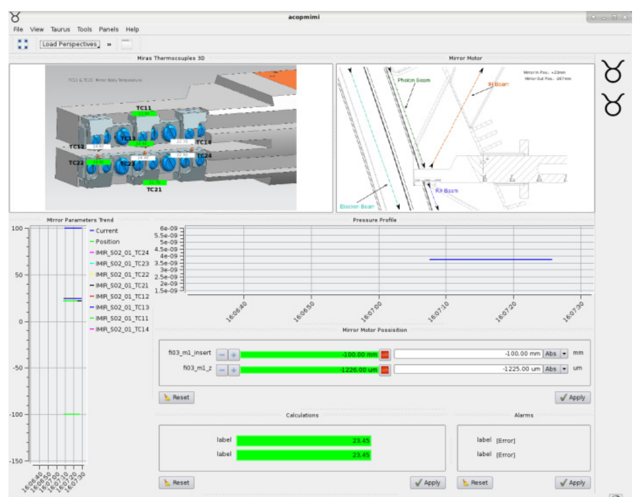


Figure 1: Taurus [4] GUI created by ALBA operators to manage the insertion/extraction of an infrared mirror.

To mitigate the cost of living with user-developed applications, three options can be adopted: forbid them to do it, give them total freedom, or adapt the Control System to their needs so they can continue developing but in a safer and more integrated way. Although the first option is the safest for the integrity of the Control System, it was not realistic nor acceptable for advanced users, so we started moving towards the third option, the design and development of a Dynamic Control System.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOBPP06

20 YEARS OF WORLD CLASS TELESCOPE CONTROL SYSTEMS EVOLUTION

T. D. Gaggstatter*, I. Arriagada, P. E. Gigoux, R. Rojas, Gemini Observatory, La Serena, Chile
J. Molgo, GMTO Corporation, Pasadena, California, USA
F. Ramos, Grantecan S.A., Breña Baja, La Palma, Spain

Abstract

This paper analyzes the evolution of control systems for astronomical telescopes. We look through the lens of three world class telescopes: Gemini, GTC and GMT. The first two are in operation for twenty and ten years respectively, whilst the latter is currently under construction. These facilities have a planned lifetime of 50+ years, therefore obsolescence management is a key issue to deal with. For the telescopes currently under operation, their real-time distributed control systems were engineered using state-of-the-art software and hardware available at the time of their design and construction. GMT and newer telescopes are no different in this regard, but are aiming to capitalize on the experiences of the previous generations so they can be better prepared to support their operations. We will compare and contrast software and hardware infrastructure choices including operating systems, middleware and user interfaces with a particular focus on obsolescence management.

INTRODUCTION

Every facility in the world, be it an industrial manufacturing plant or scientific installation, relies fundamentally on a control system to maintain optimal levels of performance. In addition to maintaining the control system as it evolves over time, support engineers must also remain informed of new technology as it becomes available to make careful adoption decisions balancing performance and stability. Telescopes are no different in this matter, and thus we present two decades of telescope control system evolution with examples from three telescopes at different stages in their life-cycle.

A telescope environment can be divided into three main control systems: the Telescope Control System, the Enclosure Control System and Support Systems. The Telescope Control System manages the main optics and its support structure. The Enclosure Control System controls the dome, bearing systems and safety infrastructure. Finally, the Support Systems manage climate control, wave front sensors and remote observations infrastructure.

The analysis is limited to three observatories which we consider a fair representation of the evolution of telescope control systems. We know we are not covering the full *observatory universe* and thus we try to compensate for this fact by presenting our conclusions in a technology-agnostic way.

In the next sections we will provide a short description of each observatory to provide a proper context. Later the most meaningful comparison points will be discussed with a closing paragraph in each section summarizing the lessons learned on that topic.

CONTROL SYSTEMS OVERVIEW

The next three sections show an introduction to each of the observatories under analysis.

Gemini Control System

The Gemini Observatory consists of twin 8.1-meter diameter optical/infrared telescopes located on two sites, Maunakea, Hawai'i, and Cerro Pachón, Chile. Having an installation on both hemispheres allows the observatory to cover the whole night sky, and the longitude separation between telescopes allows for longer tracking of events occurring in the shared zones of the sky. Gemini began its operations in Hawai'i in 1999 and in Chile in 2000. It operates mostly in queue observing mode [1] and started operating remotely from its base facilities in 2015 [2].

The Gemini telescopes are amongst the largest single mirror telescopes in the world. They were designed to be multi instrument telescopes, with a center located Cassegrain unit where the instruments are installed. [3]

Infrastructure and geographical location alone cannot guarantee performance, which is why the control system is a key part of the Gemini Observatory to ensure that the best data will be acquired and provided to scientists. With this objective in mind, Gemini uses the Experimental Physics and Industrial Control System framework (EPICS) [4]. This system was chosen because of its widespread adoption at large facilities such as particle accelerators, strong open source community support, and its high adaptability and ease of customization. It was adopted as the standard control framework in which to run the real-time telescope subsystems.

EPICS uses Client/Server and Publish/Subscribe techniques for the communication between the control computers (also called Input/Output Controllers, or IOCs). IOCs talk to each other, and other types of clients, using the EPICS Channel Access network protocol [5]. This protocol is also used by the high level software that sequences science observations. Channel Access is used at Gemini for soft real-time networking applications. Dedicated communication protocols and channels are used in places where faster communication rates are required.

The heart of an EPICS application is the database, a collection of function-block objects called records. Most EPICS records have a predefined functionality; others can be customized linking C code to them. Gemini developed a set of custom records to support the Action Command Model. In this model, actions are driven from changes to

* tgaggstatter@gemini.edu

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOCPL01

IBEX: BEAMLINE CONTROL AT ISIS PULSED NEUTRON AND MUON SOURCE

K. V. L. Baker, F. A. Akeroyd, D. Keymer, T. Löhnert, C. Moreton-Smith, D. Oram
ISIS Neutron and Muon Source, Didcot, UK
J. R. Holt, T. A. Willemsen, K. Woods, Tessella, Abingdon, UK

Abstract

For most of its over 30 years of operation the ISIS Neutron and Muon Source [1] has been using bespoke control software on its beamlines. In the last few years, we have been converting the beamline control software to IBEX [2], which is based on the Open Source EPICS toolkit [3]. More than half the instruments at ISIS are now converted. IBEX must be robust and flexible enough to allow instrument scientists to perform the many experiments that they can conceive of. Using EPICS as a base, we have built Python services and scripting support and are developing an Eclipse/RCP Graphical User Interface (GUI) based on Control System Studio [4]. We use an Agile based development methodology with heavy use of automated testing and device emulators. As we move to the final implementation stage, we are handling new instrument challenges (such as reflectometry) and providing new functionality (live neutron data view, script generator and server). This presentation will cover an overview of the IBEX architecture, our development practices, what is currently in progress, and our future plans.

INTRODUCTION

The ISIS Neutron and Muon Source is located at the Science and Technology Facility Council (STFC) Rutherford Appleton Laboratory in Oxfordshire, UK. The facility produces beams of both neutrons and muons to conduct experiments at this world leading centre for research in the physical and life sciences. To successfully deliver these experiments the beamlines have to be controllable, and this is the function that IBEX provides.

There are more than 30 beamlines at ISIS covering 7 main science techniques. The facility was opened in 1985 and 10 years ago the second target station at ISIS was completed, doubling the capacity for beamlines. With that length of history, and the established reputation of ISIS, changes to control systems have to be done carefully.

IBEX ARCHITECTURE

Hardware

A detailed description of our hardware architecture has been presented in the past [5]. The core instrument control computers are a Virtual Machine (VM) which hosts the Experimental Physics and Industrial Control System (EPICS) Input Output Controllers (IOCs) for the beamline devices. These VMs are run one per beamline on a dedicated server, and are accessible on the general ISIS network. They will also have a dedicated private network where this is required for the devices being controlled. An EPICS gateway is used to provide access control to IOCS.

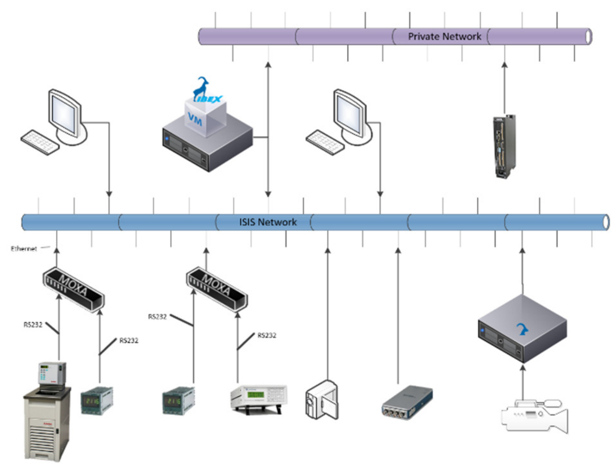


Figure 1: Hardware architecture.

More complicated devices typically have their own control computer, and will run a reduced version of IBEX to host the IOC more locally. An example of this are some of the imaging cameras used by the IMAT instrument. See Fig. 1 for an overview of the hardware architecture.

As the instrument control computer is a VM, there is usually an additional local computer provided to access the control system over the network by the user. IBEX's client-server architecture also enables users to view read-only information on remote computers without impacting local instrument control, which was not possible under the previous control system.

Software Architecture

A basic overview of the software architecture is shown in Fig. 2, and a more detailed description can be found in [2].

The IBEX Server controls individual devices via EPICS IOCs, which are wrapped within procServ [6] process harness instances. Each IOC has an associated config.xml file which provides information on configuration options (EPICS Macro names) for using within a beamline. The actual values for these macros (which might be used to set a communication port address) are maintained by the BlockServer process, which collates sets of IOC macros into "configurations" for selection by the user.

We run two Control System Studio [4] archive engines to store various Process Variables (PVs) of interest, one deals with scientifically useful data (which is placed into

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOCPL02

MODERNIZATION OF EXPERIMENTAL DATA TAKING AT BESSY II*

R. Müller†, A. Balzer, P. Baumgärtel, O.-P. Sauer, G. Hartmann, J. Viefhaus
Helmholtz-Zentrum Berlin für Materialien und Energie GmbH, Berlin, Germany

Abstract

The modernization approach for the automation of experimental data taking at BESSY II will be based on the data model of devices. Control of new components and re-factoring and reassembly of legacy software should fit into a device based framework. This approach guides the integration of motors, encoders, detectors and auxiliary subsystems. In addition modern software stacks are enabled to provide automation tools for beam line and experimental flow control and data acquisition (DAQ).

Strategic goal is the mapping of real beam line components into modelling software to provide the corresponding digital twin. First tests applying machine learning (ML) methods within this context for tuning are promising.

MOTIVATION

The soft x-ray light source BESSY II recently celebrated two decades of user operation and one decade of forming the Helmholtz-Zentrum Berlin (HZB) [1] by merging the Hahn Meitner Institute (HMI) with the BESSY II facility. Thus HZB could offer two complementary experimental options for material science: the neutron reactor BER II and the synchrotron. BER II is shutdown by the end of the year 2019 and for the end of the next decade HZB intensifies plans of a successor soft x-ray source BESSY III.

Todays controls environment of the BESSY II experimental floor has been very much determined by the early days rapid installation of an extraordinary number of beam lines and instruments. To allow for most efficient use of the light source the organizational structure of BESSY II has been set up exceptionally flexible: many beam lines provided open ports ready to welcome complete user instruments; external institutes and cooperation research groups (CRG) could contribute by co-financing and co-operating beam lines and instruments. Ubiquitous switching mirror units (SMU) allowed to exploit x-ray source points in a time sharing manner. As a result BESSY II with only 14 usable straight sections could ramp up to some 35 beam lines and about 50 end station instruments in a very short time.

Two multi-purpose packages emerged and helped to keep the phase of heroic experiments limited despite the large variety of installations: a data acquisition (DAQ) hard and software set-up could cover data handling needs. Monochromator functions of beamline control rapidly developed into a large monolithic, multi-protocol, *generic* control program (called MONO) that enabled maintainability for the small group and the fast roll out of new beam lines. Even today this VME program also handles the communication to instruments, experimental flow control and insertion devices; sophisticated energy fly mode variants for dipoles and ID beam lines are implemented too.

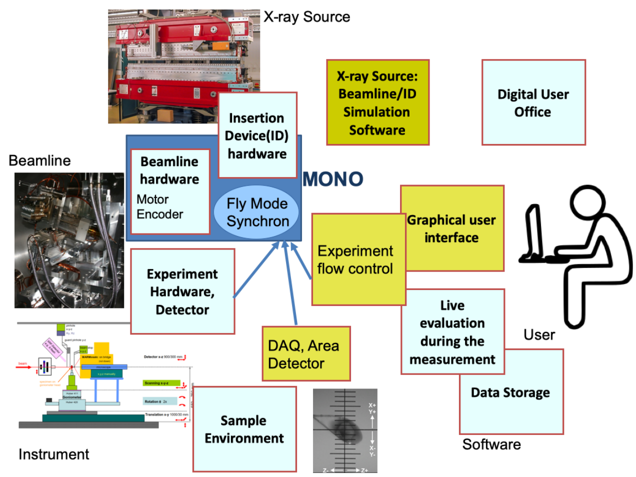


Figure 1: Sketch of the central role the multipurpose, generic software MONO (blue block) plays for the legacy experimental control infrastructure at BESSY

In consequence of a project oriented organizational structure of HZB new experiments are set up increasingly detector and method centric with instrument control, DAQ and experimental flow control planned and provided at a commissioning level with minimal resources. There is a free choice of the software tools needed (LabVIEW, IGOR, spec etc.), user friendliness and adaptability, failure tolerance and sustainability is not part of the project. In this low effort approach selection of the x-ray properties is again foreseen simply by remote control of the MONO software (Fig. 1). MONO has to integrate and adapt to new beam line components, e.g. hexapods, orchestrate with new insertion devices, handle obsolescence management of motors and encoders and provide handles to react on off-normal situation. As an VME program MONO is increasingly hard to maintain and develop on its generic level.

In general the control system structure in the experimental hall is by far too fragmented. Too many intertwined and interdependent connections need too many knowing hands to allow for easy and fast developments. At that point a clear strategic plan is inevitable to be able to over come the blocking constraints.

MODERNIZATION OPTIONS AND STRATEGY

Unlike other facilities BESSY II did not start with a holistic control system concept. Only for the accelerator a modern, consistent control system based on the EPICS framework was implemented. In the experimental hall no agree-

* Work funded by BMBF and Land Berlin
† roland.mueller@helmholtz-berlin.de

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOCPL03

BEAMLINE EXPERIMENTS AT ESRF WITH BLISS

M. Guijarro*, G. Berruyer, A. Beteva, L. Claustre, T. Coutinho, S. Debionne, M.C. Dominguez, P. Guillou, C. Guilloud, A. Homs, R. Homs, M.C. Lagier, A. Mauro, J. Meyer, V. Michel, C. Muzelle, S. Olhsson, P. Pancino, E. Papillon, M. Perez, S. Petitdemange, L. Pithan, F. Sever, V. Valls, H. Witsch
ESRF, The European Synchrotron, Grenoble, France

Abstract

BLISS is the new ESRF beamline experiments sequencer. BLISS is a Python library and a set of tools to empower scientists with the ability to write and to execute complex data acquisition sequences. Complementary with *TANGO*, the ESRF control system, and *silx*, the ESRF data visualization toolkit, BLISS ensure a smooth user experience from beamline configuration to online visualization. After a 4-year development period, the initial deployment phase is taking place today on half of ESRF beamlines, concomitantly with the ESRF Extremely Brilliant Source upgrade program. This document presents the BLISS project in large, focusing on feature highlights and technical information as well as more general software development considerations.

THE BLISS PROJECT

BLISS stands for BeamLine Instrumentation Support Software. The BLISS project started in December, 2015 inside the Beamline Control Unit (BCU, Software Group), and comes within the scope of the ESRF Extremely Brilliant Source upgrade program (ESRF-EBS) [1].

The ESRF-EBS is a global project, to put ESRF and all partners countries at the forefront of X-ray science and instrumentation. A major milestone will be reached in 2020 with the end of the construction of a new, revolutionary storage ring and the restart of ESRF user program. ESRF will then become the world's first high-energy, fourth-generation synchrotron light source. This exciting feat will offer unprecedented tools for the exploration of matter and for the understanding of life at the macromolecular level.

In particular, 4 new beamlines are being built and new instrumentation is currently under development ; the main objective of BLISS is to provide scientists with the more advanced experiments control and data acqusition software in order to take advantage of the new ESRF-EBS tools and equipments. BLISS has the ambition to fulfill the needs of the more demanding experiments.

Project Goals

- to empower scientists with the ability to write and to execute complex data acquisition sequences
- to offer an easy to use Command Line Interface (CLI) and an online data visualization application
- to provide generic building blocks to implement any kind of scan
- to get the most out of the capabilities of beamline hardware
- to provide frameworks to quicken integration of new hardware
- to facilitate online data analysis
- to enable data management

BLISS SOFTWARE

BLISS is primarily a Python 3.7 library. It is furnished with a set of tools to manage experimental setups using configured beamline devices, to write experiments control sequences and to execute them in an adapted environment, and to do online data visualization.

BLISS is an open-source software, licensed under *LGPL* [2]. BLISS is free to use by anyone ; however ESRF does not officially provide support to external users without a formal collaboration agreement.

BLISS Package

BLISS is packaged with [3]. Conda is an open source package management system and environment management system that runs on Windows, macOS and Linux. Although it was initially targetting Python software, it can be used to package any kind of software and offers:

- separated environments, different execution contexts
- OS-independent, community-packaged software (from the libc level)
- the ability to create custom packages easily

BLISS Releases

BLISS follows a one-month release cycle. The first BLISS stable version 1.0.0 is to be released at the beginning of next year (January, 2020). Starting with the first stable release, BLISS version numbers will follow *semantic versioning* [4]. Basically the first number (1) represents a major version, that can only increment in case of incompatible API changes. The second number (0) increases when backward-compatible changes are added. The last number (0) is a patch-level: this is mainly to indicate bug fix releases.

Source Code Management

As of today, all BLISS code is contained within a single *git repository* [5]. This simple approach helps to ensure coherency over the code base, in particular in case of refactoring or modifications with non-trivial consequences on existing parts of the project compared to a solution based on multiple git repositories linked with sub-modules for example.

* guijarro@esrf.fr

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOCPL04

SOFTWARE ARCHITECTURE FOR AUTOMATIC LHC COLLIMATOR ALIGNMENT USING MACHINE LEARNING

G. Azzopardi*,1,2, G. Valentino[1], B. Salvachua[2], S. Redaelli[2], A. Muscat[1]
[1] University of Malta, Msida, Malta
[2] CERN, Geneva, Switzerland

Abstract

The Large Hadron Collider at CERN relies on a collimation system to absorb unavoidable beam losses before they reach the superconducting magnets. The collimators are positioned close to the beam in a transverse setting hierarchy achieved by aligning each collimator with a precision of a few tens of micrometres. In previous years, collimator alignments were performed semi-automatically, requiring collimation experts to be present to oversee and control the entire process. In 2018, expert control of the alignment procedure was replaced by dedicated machine learning algorithms, and this new software was used for collimator alignments throughout the year. This paper gives an overview of the software re-design required to achieve fully automatic collimator alignments, describing in detail the software architecture and controls systems involved. Following this successful deployment, this software will be used in the future as the default alignment software for the LHC.

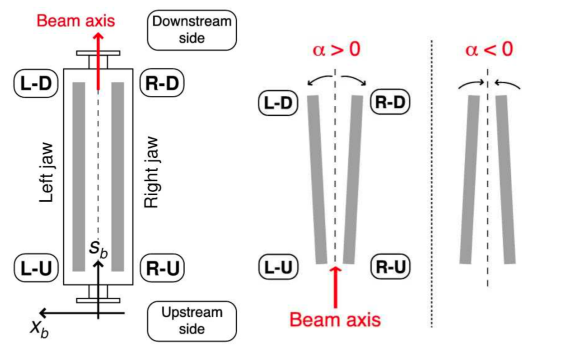
INTRODUCTION

The Large Hadron Collider (LHC) at CERN is the largest particle accelerator in the world, built to accelerate and collide two counter-rotating beams at an unprecendented center-of-mass energy of 13 TeV [1,2]. The LHC is susceptible to beam losses from normal and abnormal conditions, which can damage the state of superconductivity of its magnets. A robust collimation system handles beam losses of halo particles by safely concentrating them into room temperature collimation regions, with a 99.998% cleaning efficiency of all halo particles [3].

The LHC collimation system consists of around 100 collimators, each with two parallel absorbing blocks, referred to as jaws, inside a vacuum tank. The jaws are identified as *left* or *right*, depending on their position with respect to the incoming beam. The jaws must be positioned symmetrically around the beam and their coordinate system is displayed in Figure 1. Each jaw can be moved individually using two stepping motors at the jaw corners, allowing collimators to be positioned at different gaps and angles. The maximum possible operational angle in either direction is 1900 μrad [5]. The jaw corners are known as left-up (LU) and right-up (RU) when they are upstream of the beam and left-down (LD) and right-down (RD) when they are downstream of the beam.

Collimators provide halo cleaning using a multi-stage hierarchy, which is determined after aligning the collimators. Each year of LHC operation begins with a commissioning

* gabriella.azzopardi@cern.ch



(a) Jaw coordinate system (b) Jaw angular tilt convention

Figure 1: (a) The collimator coordinate system and (b) the jaw tilt angular convention as viewed from above, from [4].

phase which involves aligning all collimators and ensuring the correct operation to allow the LHC to achieve nominal operation [6]. Such alignments are performed to determine the beam orbit and beam size at each collimator location, which are otherwise not known sufficiently precisely as the actual beam orbit, collimator tank alignment and optics may deviate from the design orbit. This information is required to position the jaws within a certain number of standard deviations (beam σ) from the beam center [7].

Over the years various software and hardware upgrades were introduced to improve the alignment time and to simplify the alignment procedure. For six years collimators were aligned using a semi-automatic tool, however this reached its minimum alignment time of 3 hours at injection in 2017. This motivated the development of a fully-automatic tool, which was used for the first time in 2018 and has proved to be a beneficial advancement in view of the High Luminosity LHC (HL-LHC) upgrade [8].

LHC COLLIMATOR ALIGNMENTS

Collimator alignments are performed with a step precision of 5 μm. Each collimator has a dedicated Beam Loss Monitoring (BLM) device positioned outside the beam vacuum, immediately downstream, as shown in Figure 2. Such devices are used to detect beam losses generated when halo particles impact the collimator jaws. Recorded losses are proportional to the amount of beam intercepted by the collimator jaws and are measured in units of Gy/s. A collimator is considered aligned when a jaw movement towards the beam produces a clear loss spike in the BLM detector located further downstream [10].

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOCPL05

SOFTWARE FRAMEWORK QAClient FOR MEASUREMENT/AUTOMATION IN PROTON THERAPY CENTERS

A. Mayor, O. Actis, D. Meer, B. Rohrer, Paul Scherrer Institut, 5232 Villigen PSI, Switzerland

Abstract

Paul Scherrer Institute (PSI) operates a proton center for cancer treatments. For calibration measurements and quality assurance procedures which have to be executed on a frequent basis and involve different systems and software products, an in-house software framework (QAClient) was developed.

QAClient provides a configurable and extensible framework communicating with PSI control systems, measurement devices, databases and commercial products as LabVIEW and Matlab. It supports automation of test protocols with user interaction, data analysis and data storage as well as generating of reports.

INTRODUCTION

The Paul Scherrer Institute (PSI) was a pioneer in the field of proton therapy for cancer treatment by being the first center to implement spot scanning for dose delivery back in 1996 [1].

Different treatment areas as Gantry 2 (PSI in-house developed advanced scanning gantry [2]), Gantry 3 (a commercial gantry from Varian) and OPTIS2 (for ocular tumor treatments) are constantly being improved and extended for new irradiation techniques and have to be maintained permanently.

Especially for extensive and periodic quality assurance (QA [3]) test procedures but as well for long lasting calibration and measurement tasks there was an increasing need for an automation tool which could integrated all involved control systems, measurement devices and perform data analysis and storage.

It was therefore decided in 2011 to start the development of such a software tool. The tool had to be modular and extensible, suitable for a wide range of tasks and easy to use and adapt for different protocols and procedures without much programming efforts.

To be as platform independent as possible and to be able to integrate it easily into other PSI systems and environments, Java [4] was chosen as the programming language and XML [5] as configuration format. QAClient was developed as an extensible software framework and is still being extended and used in a variety of applications on daily basis.

Motivation

The complexity of all involved systems in proton therapy used for medical patient treatment (Fig. 1) requires extensive and periodic testing and QA. This testing has to be efficient, reproducible, easy to execute (also by less specialized personnel) and has to produce QA documentation and publish data to QA databases.

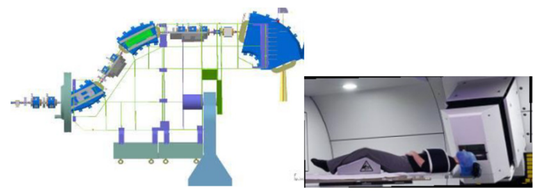


Figure 1: Beamline and treatment at Gantry 2.

Integrated Systems

In order to be able to automatize QA procedures QAClient has to integrate many different technical systems, actuators and detectors:

- Mechanical systems such as gantry, nozzle, patient couch and beam blockers
- Electromagnetic devices such as magnets, motors and valves
- Detectors such as ionization and strip chambers
- Control systems such as delivery system, verification system, couch robot and beam tune system
- Commercial systems such as Matlab [6], LabVIEW [7] and laser trackers

It has to be able to control these systems and to process data sent by these systems. Some measurement devices are controlled and read out via commercial tools such as LabVIEW which can be integrated in QAClient.

Data analysis and modelling tasks area often delegated to Matlab which also can be integrated in QAClient.

Requirements

To cover regulatory aspects of QA and to ensure that it can be executed reliably and efficiently also by non-specialized personnel the main requirements for QAClient were defined as follows:

- Full QA test procedure automation
- Integration of different systems and devices
- Interaction with human operators (data visualization)
- Integrated data analysis
- Comprehensible Reporting
- Long-term QA data storage and retrieval
- Easy to use
- Fully configurable (no programming required)

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOCPL06

2D-NANO-PTYCHOGRAPHY IMAGING RESULTS ON THE SWING BEAMLINE AT SYNCHROTRON SOLEIL

C. Engblom*, Y-M. Abiven, F. Alves, F. Berenguer, T. Bizien,
A. Gibert, F. Langlois, A. Lestrade, P. Montaville, J. Pérez
Synchrotron SOLEIL, St Aubin, France

Abstract

A new Nanoprobe system, which was originally developed in the scope of a collaboration with MAXIV (Sweden), has recently been tested and validated on the SWING beamline in Synchrotron SOLEIL. The aim of the project was to construct a Ptychography nano-imaging station. Initial steps were taken to provide a portable system capable of nanometric scans of samples with sizes ranging from the micrometer to fractions of a millimeter. Imaging was made possible by actuating a total of 16 Degrees Of Freedom (DOF) composed of a sample stage (3 DOF), a central stop stage (5 DOF), a Fresnel zone plate stage (5 DOF), as well as an order sorting aperture stage (3 DOF). These stages were actuated by an ensemble of piezo-driven and high-quality brushless motors, of which synchronized control (with kinematic modelling) was done using the Delta Tau platform. In addition, interferometry feedback was used for reconstruction purposes. Imaging results are promising: the system was able to resolve 40 nm measured with a Siemens star, the paper will describe the system and the achieved results.

INTRODUCTION

The *Nanoprobe Project* [1] was a 4-year joint collaboration between Synchrotron SOLEIL and MAXIV (Sweden), where a 3D scanning-nanoprobe prototype was produced. This project officially ended in december 2016 - after which a SOLEIL-based team dedicated to nano-positioning systems was formed (*Nanoprobe-SOLEIL*).

In part inspired by the cSAXS beamline (Swiss Light Source), the SWING beamline [2] at Synchrotron SOLEIL has decided to add high-resolution coherent diffractive imaging, ptychography, to its roster of experimental setups. The reason for this is two-fold; first to address a growing need amongst its users, and secondly to prepare for the upcoming SOLEIL synchrotron upgrade [3] which will utilize a more coherent light source. High-resolution imaging is the driving factor here - the overall aim is to achieve the *nanometer* scale of 20 nm (or better) over full-range sample scans of 10 µm up to several 100 µm. Another major constraint is system portability; to keep a level of flexibility between different type of experiments, any new system needs to be compact and capable of installing/uninstalling to/from the beamline within a few hours.

The scope of the project (and the subject of this paper) is therefore to: install a Nanoprobe system on the SWING beamline, and have it tested & validated through 2D nano-

* christer.engblom@synchrotron-soleil.fr

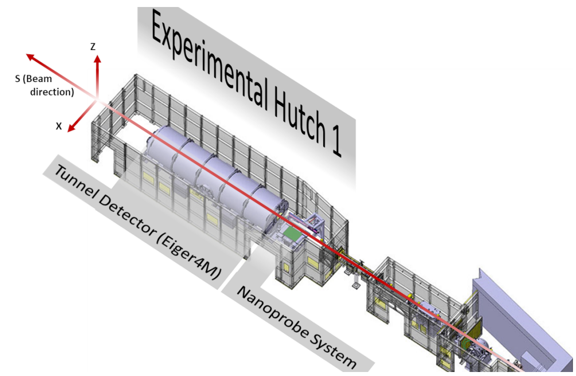


Figure 1: Overview of the SWING experimental hutch 1 (EH1), here marked out with the SOLEIL SXZ- orientation; the location of the Eiger 4M detector has been pointed out (inside a 7 meters long in-vacuum tunnel), as well as the location of the Nanoprobe System.

ptychography imaging. The *Nanoprobe Project* [1] will be used as an outline for the new SWING setup, where the previously produced prototype will be re-used & adapted to beamline specifications. As such, we will rely heavily on:

1. Support structure & system environment; rigidity and thermal stabilization via enclosure of some kind.
2. High- resolution (capable of resolving a nanometer), long-range (several millimetres) positioners.
3. Interferometry; a tool to qualify and measure motion errors - to either be used for correcting images post-process, or be implemented in closed-loop control.
4. Control Systems; for high-frequency control, & multi-axis synchronization via kinematic modeling if need be.

Figure 1 shows an overview of the EH1 experimental hutch where the Nanoprobe System will be installed and tested.

The term *Nanoprobe End-Station* will in this paper be referred to the support and environment that houses the active parts used in scans. The term *Nanoprobe System* will refer to the Nanoprobe End-Station *as well as* all its driving electronics & control systems.

NANOPROBE END-STATION OVERVIEW

The end-station can be divided up into five primary parts, four *stages* and their support structure. Figure 2 illustrates

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOCPR01

GRADUATE SOFTWARE ENGINEER DEVELOPMENT PROGRAM AT DIAMOND LIGHT SOURCE

A. A. Wilson*, T. M. Cobb, U. K. Pedersen, Diamond Light Source, Oxfordshire, United Kingdom

Abstract

Diamond Light Source is the UK's synchrotron facility. The support and development of the beamlines and accelerators at Diamond requires a significant quantity of specific knowledge and skills; the opportunity to acquire these beforehand is not available to many early in their career. This limits the field of candidates who can begin working independently at the level of software systems engineer. The graduate software engineer development program was started in 2015 to provide a route for engineers who are recent graduates or new to the field to develop the required skills and experience. Over the course of two years it comprises a series of projects in different groups, mentored on-the-job training and organized training courses. The program has recently been expanded to cover all groups in the Scientific Software, Controls and Computation department at Diamond, with an intake of four new engineers per year. This paper presents the structure and development of the program and invites discussion with other organizations to share knowledge and experience.

INTRODUCTION

The field of control systems for particle accelerators is highly specialised. To work independently in the field, a significant amount of specific knowledge and skills are required. It can therefore be difficult for organizations to find and recruit people with the requisite experience to start working independently in this function. Meanwhile there are few opportunities for early career professionals to gain the necessary experience in advance.

In 2015, the Controls Group at Diamond implemented a Graduate Training Program in order to create a route to hire talented but inexperienced people who are either recent university graduates or new to the field.

This program was later expanded to include the adjacent groups in the Scientific Software, Controls and Computation Department (SSCC). The SSCC department covers the work on the "full stack" of software and computing at Diamond, encompassing computing, systems administration, electronics, controls, data acquisition and analysis, and business applications.

This paper outlines the structure of the program, its current status and our plans for future developments. The purpose of the paper is to invite discussion: to share experience with the community and to gain some ideas to improve the program further.

* andrew.wilson@diamond.ac.uk

PROGRAM STRUCTURE

The program consists of two years of training. Participants begin in a graduate role, and move to the experienced engineer role following completion of the program.

Recruitment

The recruitment process takes place in November and December for a start date in the Autumn of the following year. This follows the pattern of many graduate training programs in industry, targeting students in their final year of university; in this way we aim to be competitive with such programs.

We aim to attract university leavers with a background in a STEM subject area (Science, Technology, Engineering and Mathematics), not only those with a background in Computer Science. We look particularly for an interest and some experience in software or computing, for example, a personal interest or an undergraduate project.

First Year

In the first year, participants undertake four projects, each lasting three months, in different groups within SSCC. The projects are generally stand-alone, although sometimes consecutive projects will build on each other, approaching different aspects of the same topic. As far as possible they are aligned to the interests of the individual and the needs of the business. Figure 1 shows the number of projects undertaken to date per group.

The purpose of these projects is for the participant to gain experience of working in different areas of the business; to sample the different technologies and working environments; and to build a network of connections which will be useful in their later role. The host groups benefit from the extra effort to complete a necessary piece of work, and the new ideas and different approaches that the participants bring. The projects are often interdisciplinary and therefore promote collaboration between the different groups.

In addition to the participant, the projects involve several people whose roles are as follows:

Supervisor Defines the scope and goals of the project; provides guidance and assistance to the participant throughout the project; involves the participant in some day-to-day work of the host team; arranges project meetings.

Customer As the target customer, defines the requirements for the project and reviews the results. This role is often performed by the Supervisor.

Technical support Additional people who may be involved if the supervisor does not have all the technical information needed to support the project. For example, the technical support helps plan the architecture and review the results.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOCPR02

THE EPICS COLLABORATION TURNS 30

L. R. Dalesio, Osprey DCS LLC, Ocean City, USA
A. N. Johnson, Argonne National Laboratory, Lemont, USA
K.-U. Kasemir, Oak Ridge National Laboratory, Oak Ridge, USA

Abstract

At a time when virtually all accelerator control systems were custom developments for each individual laboratory, an idea emerged from a meeting between the Los Alamos National Laboratory developers of the Ground Test Accelerator Control System and those tasked to design the control system for the Advanced Photon Source at Argonne National Laboratory. In a joint effort, the GTACS toolkit concept morphed into the beginnings of a powerful toolkit for building control systems for scientific facilities. From this humble beginning the Experimental Physics and Industrial Control System (EPICS) Collaboration quickly grew. EPICS is now used as a framework for control systems for scientific facilities on seven continents. The EPICS Collaboration started from a dedicated group of developers with very different ideas. This software continues to meet the increasingly challenging requirements for new facilities. This paper is a retrospective look at the creation and evolution of a collaboration that has grown for thirty years, with a look ahead to the future.

HISTORY

This year is noted as the 30th anniversary of the Experimental Physics and Industrial Control System (EPICS). We consider 1989 to have been the start of the collaboration between Los Alamos National Laboratory (LANL) and Argonne National Laboratory (ANL), formed by Mike Thuot, LANL, and Marty Knott, ANL, after discussions at the 1989 ICALEPCS and a subsequent technical APS evaluation of toolkits that had been developed at Los Alamos.

The precursors to EPICS were the Ground Test Accelerator Control System (GTACS) started in 1985 [1-4] and the Los Alamos Accelerator Control System (LAACS) presented in 1989 [5, 6]. The name Experimental Physics and Industrial Control System (EPICS) was adopted to reflect the collaborative nature of the toolkit.

Most of the core ideas were presented and evolved at two meetings that were the predecessors of the ICALEPCS series. The first was held in Los Alamos and included a number of other international laboratories. A second meeting was held at CERN in 1987. In 1989 these meetings started the ICALEPCS conference series. In 2013 the ICALEPCS International Scientific Advisory Committee presented the early organizers Peter Clout, Axel Daneels, Shin-Ichi Kurokawa, Dave Gurd, Daniele Bulfone and Ryotaro Tanake with a lifetime achievement award.

Channel Access [7] was the key network protocol that connected the distributed EPICS components, and initially the version number of the communication protocol also determined the major EPICS version number. EPICS version 3 started with the third iteration of the Channel Access protocol, but in 1991 a modification to the Channel Access code introduced a minor protocol version number field and started a long history of network compatibility between released versions of the toolkit.

From the start of this collaboration, all source code was meant to be shared. Code was co-developed, inherited and developed serially or competitively. This sharing of software was done openly from 1989 on, prior to the now familiar notion of Open Source Software (OSS) development. Lacking the commonly adopted OSS legal structures, joining the EPICS collaboration initially required signing an agreement to gain access to the source code. Until 2004, over 100 organizations had signed this agreement. Finally, the legal departments of APS and LANL allowed EPICS to become OSS, with all source code and documentation made freely available from web sites hosted by collaboration members. Since that time, there has been no record kept of the number of facilities using EPICS.

TECHNOLOGY ENABLES SCIENCE

The EPICS collaboration has been supported by international science facilities. The science that is produced in these facilities has been at the forefront of discovery including: high brightness X-Rays, Neutrons, Gravitational Wave Sensors, Radioactive Ion Beams and Nuclear Fusion, as well as Optical, Radio, and Infrared Telescopes. Innovation in the capacity, precision and speed of electronics, networks, sensors, and storage have all advanced the ability to gather data for analysis. The expectations of a control system have grown with these capabilities and the collaboration model allows facilities to build on the work of others. EPICS is now being used to control large and small scientific projects on every continent (including Antarctica).

MOTIVATION BEHIND THE DEVELOPMENT OF EPICS

The EPICS toolkit approach had these goals [8]:

- Allow independent subsystem development
- Reduce programming costs on projects by 85%
- Provide the best possible performance
- Allow extensions at every level of the toolkit.

The seamless integration of new Input-Output Controllers (IOCs) with self-contained application directories satisfied the independent development goal. For every newly developed component, performance measurements were performed. Designs were iterated to improve throughput and reduce latency. Narrow, well defined interfaces were provided at many levels of the system to support extensibility. EPICS dramatically reduced the time to produce control system applications. But the overall cost of control

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOCPR03

PLANNING OF INTERVENTIONS WITH THE ATLAS EXPERT SYSTEM

I. Asensi*,CERN and University of Valencia, Valencia, Spain
C. Solans†,CERN, Geneva, Switzerland
A. Rummler‡,CERN, Geneva, Switzerland
J. Torres§,University of Valencia, Valencia, Spain

Abstract

The ATLAS Technical Coordination Expert System is a tool for the simulation of the ATLAS experiment infrastructure that combines information from diverse areas such as detector control (DCS) and safety systems (DSS), gas, water, cooling, ventilation, cryogenics, and electricity distribution. It allows the planning of an intervention during technical stops and maintenance periods, and it is being used during the LS2 to provide an additional source of information for the planning of interventions. This contribution will describe the status of the Expert System and how it is used to provide information on the impact of an intervention.

INTRODUCTION

The ATLAS [1] Expert System [2] is a diagnostic tool created by ATLAS Technical Coordination to increase the knowledge base of the ATLAS experiment, allow easier turn over of knowledge between experts and foresee complications before the interventions take place. It describes different systems like sub-detectors, gas, cooling, ventilation, electricity distribution and detector safety systems which result in an extremely complex tree of relations between them. It consist of a friendly user interface in the form of a graphical simulator which allows the user to simulate an intervention and to foresee its consequences on all the other systems of the experiment.

The requirements of the ATLAS Expert System are the following:

- Provide a description of ATLAS and its elements in a way that is understandable to a multidisciplinary team of experts.
- Emulate the behaviour of the sub-systems to foresee possible consequences of interventions.
- Help to understand what would remain operative in case of events like fire or electrical perturbations.
- The simulator has to accept input from the user and quickly answer how ATLAS would behave with the given input.
- Use standard technologies as possible to simplify maintenance.

* ignacio.asensi@cern.ch
† carlos.solans@cern.ch
‡ andre.rummler@cern.ch
§ jose.torres@uv.es

Goals

The goal of the ATLAS Expert System is to become an important tool in the Technical Coordination of ATLAS as part of the standard procedure prior to an intervention and to be a helpful tool in the ATLAS Control Room in many situations.

It can help in the diagnostic of a complicated problem. For example when a system is out of power and the reason is unknown it can help to find what are the possible causes. It also can help to take a quick and well educated action when time is critical and experts are not available.

Brief Description

The ATLAS Expert System contains a virtual representation of the ATLAS experiment which is presented to the user in the form of visual diagrams. This representation is also a simulation that imitates the behaviour of the infrastructure of ATLAS. In this simulation the user can take actions like switching off a system or triggering an alarm. Once an action is taken, the simulation is executed and the user can immediately see the consequences. The units of the structure are the systems, which normally can accept one input, either to be switched on or off. Different ways of obtaining information are available to the user such as list-oriented interfaces and different deduction explanation levels.

The construction of the ATLAS experiment was completed in 2008 and more than ten years after the completion, the reliability of certain systems is decreasing over time. The ATLAS Expert System was started in 2016 and the following sections of this document summarises the current state of the Expert System description of ATLAS and outlines the last improvements in descriptions and tools.

ARCHITECTURE

From a technical perspective the system is divided in three well separated components: database, python server and a web application.

Backend

The backend is composed of the database and the python server. The database stores all the information about the elements that describe ATLAS and the relations between them. The database used is the ATLAS TDAQ object oriented configuration database, so called Object Kernel Support (OKS) [3], which holds the description of objects, classes, relationships, inheritance and it is expected to be maintained during the life of the experiment. To simulate the ATLAS

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOCPR04

MOVING BEYOND BIAS*

K. S. White†, Oak Ridge National Laboratory, Oak Ridge, USA

Abstract

The benefits of diverse work groups have been well documented. Creativity, innovativeness and productivity are all improved by creating a team with a variety of backgrounds and perspectives. [1] While scientific laboratories strive for more diversity and inclusion, the field of accelerator controls remains strikingly homogeneous. This trend continues despite many long-standing programs to attract females and people of color to STEM (Science, Technology, Engineering and Math) careers and the explicit desire of leadership to create more inclusive organizations. Research consistently points to the strong role implicit bias plays in preventing organizations from truly providing equal opportunities. In reality, the desire of leadership to improve diversity must be coupled with a strong culture, cultivated to change deeply rooted practices which influence recruiting, hiring, development and promotion decisions based on stereotypes rather than accomplishments, skill and potential. Real change in this arena requires intentional action across the board, not just from human resources and less represented groups. This paper discusses practical approaches to changing organizational culture to enable diverse work groups to grow and thrive.

THE VALUE OF DIVERSITY

Numerous studies analyze ways diverse work groups produce better results in terms of productivity, creativity and innovation; all important qualities in a scientific enterprise. Bringing a wide range of perspectives to the table means a greater variety of ideas and methods are considered in getting to the best solution. We easily appreciate the variety of hardware and software and diverse set of skills needed to successfully build and operate control systems. If we did not have the option to use many types of commercial hardware and software, build custom cards, leverage open source software and write our customer software, it would be considerably more challenging if not impossible to meet the requirements of our customers. Similarly, having a variety to people from different backgrounds provides more perspective and enhances our ability to build the best solutions.

Taking a different point of view, it may be easier to appreciate the value of building a diverse and inclusive organization by considering the potential risk and consequences of failing to do so. There are abundant examples detailing how the shortage of diversity amongst working groups has led to notably bad results including products that are less than satisfactory to half the population and those that fail to perform their intended safety function for people who fall outside the standard model.

Rebuilding

In 2001, as the result of a powerful earthquake in Gujarat in western India, nearly 400,000 housing units were destroyed. [2] Thousands of people died, and an exclusively male team was formed to design and build replacement homes. Similarly, a devastating tsunami struck along the coast of the Indian Ocean in 2004, killing over 250,000 people across fourteen countries. In response, Sri Lanka formed an all-male rebuilding team. While both teams succeeded in building housing, neither included kitchens in the new homes. Surely a detail that would not have been missed if even a small number of women had been included in the planning or, conversely, if men were involved in the cooking.

Similar mishaps occurred even in the United States following hurricanes Andrew (1992) and Katrina (2005) when rebuilding efforts included few women and failed to account for the needs of low-income people, many people of color and single mothers, who were disproportionally impacted by these storms. In Miami, following Andrew, rebuilding focused on larger businesses and neglected child-care facilities, accessible health care and needs of small businesses. In post-Katrina New Orleans, repairable low-income housing was demolished and mostly replaced with more expensive units. Due to the costs of the new units, former residents were forced to relocate to less central locations that without important basics like public transportation and access to food markets and childcare. Without affordable, reliable transportation and childcare, people lost their jobs as well as their homes.

Such non-inclusive efforts to rebuild demonstrate the blinders in place when workgroups do not adequately represent the populations they serve. These examples show not only the need to include women, but also people of color, people of different income levels and those with various family situations when planning community resources. The design teams likely had no idea what was missing from their solutions until the inevitable backlash began as the omitted features were not central to their daily lives.

Safety

While the absence of an in-house kitchen is certainly an inconvenience that disproportionally impacts lower status women in developing countries, it is absolutely tragic when products and processes meant to address safety fail to protect the majority of the population due to testing on narrow, non-diverse models or groups.

* This manuscript has been authored by UT-Battelle, LLC, under contract DE-AC05-00OR22725 with the US Department of Energy (DOE). The US government retains and the publisher, by accepting the article for publication, acknowledges that the US government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this manuscript, or allow others to do so, for US government purposes. DOE will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan (http://energy.gov/downloads/doe-public-access-plan)
† whiteks@ornl.gov

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOCPR05

CI-CD PRACTICES WITH THE TANGO-CONTROLS FRAMEWORK IN THE CONTEXT OF THE SQUARE KILOMETRE ARRAY (SKA) TELESCOPE PROJECT*

M. Di Carlo†, INAF Osservatorio Astronomico d'Abruzzo, Teramo, Italy
M. Bartolini, SKA Organisation, Macclesfield, UK
S. Williams, UK Astronomy Technology Centre, Edinburgh, UK
K. Madisa, A. J. Venter, M.J.A. de Beer, SKA South Africa, Cape Town, South Africa
J. B. Morgado, D. Bartashevich, D.F. Nunes, Instituto de Telecomunicações (GRIT), Universidade de Aveiro, Portugal

Abstract

The Square Kilometre Array (SKA) project is an international effort to build two radio interferometers in South Africa and Australia to form one Observatory monitored and controlled from the global headquarters (GHQ) based in the United Kingdom at Jodrell Bank. The project is now approaching the end of its design phase and gearing up for the beginning of formal construction. The period between the end of the design phase and the start of the construction phase, has been called bridging and, one of its main goals is to promote some CI-CD practices among the software development teams. CI-CD is an acronym that stands for continuous integration and continuous delivery and/or continuous deployment. Continuous integration (CI) is the practice to merge all developers local (working) copies into the mainline very often (many times per day). Continuous delivery is the approach of developing software in short cycle ensuring that it can be released anytime and continuous deployment is the approach of delivering the software frequently and automatically. The present paper wants to analyse the decision taken by the system team (a specialized agile team devoted to developing and maintaining the tools that allows continuous practises) in order to promote the CI-CD practices with TANGO controls framework.

INTRODUCTION

When creating releases for the end-users, every big software industry faces the problem of integrating the different parts of the software and bring them to the production environment, that is where users work. The problem arises when many parts of the project are developed independently for a period of time and when merging them into the same branch, the process takes more than what was planned. In a classical waterfall software development process this is usual, but the same happens also following the classical git flow, also known as feature-based branching, that is when a branch is created for a particular feature. Considering, for example, one hundred developers working in the same repository each of them creating one or two branches, then it is easy to understand what is called the "merge hell" that is when every merge has to deal with conflict parts and it is impossible, for a single developer, to solve all the conflicts thus creating delay in publishing any release. This problem was analyses at the Square Kilometre Array (SKA) project, an international effort to build two radio interferometers in South Africa and Australia to form one Observatory monitored and controlled from the global headquarters (GHQ) based in the United Kingdom at Jodrell Bank. The selected development process is Agile (Scaled Agile framework) that is basically incremental and iterative with a specialized team (known as system team) devoted to support the continuous Integration, test automation and continuous Deployment.

TANGO-CONTROLS OVERVIEW

One of the most important decisions taken by the SKA project is the adoption of the TANGO-controls framework [1] which is a middleware for connecting software processes together mainly based on the CORBA standard (Common Object Request Broker Architecture). The standard defines how to exposes the procedures of an object within a software process with the RPC protocol (Remote Procedure Call). The TANGO framework extends the definition of object with the concept of Device which represents a real or virtual device to control that expose commands (that are procedures), attributes (like the state) and allows both synchronous and asynchronous communication with events generated from the attributes (for instance a change in the value can generate an event). Figure 1 shows a module view of the framework.

CONTINUOUS INTEGRATION (CI)

CI refers to a set of development practices that requires developers to integrate code into a shared repository several times a day. Each check-in is then verified by an automated build, allowing teams to detect problems early. According to Martin Fowler [2], there are a number of best practices to implement to reach CI:

- Maintain a single source repository (for each component of the system) and try to minimize the use of branching, in favor of a single branch of the project currently under development.

* Work supported by Italian Government (MEF - Ministero dell'Economia e delle Finanze, MIUR - Ministero dell'Istruzione, dell'Università e della Ricerca)
† matteo.dicarlo@inaf.it

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOMPL001

QUALITY ASSURANCE PLAN FOR THE SCADA SYSTEM OF THE CHERENKOV TELESCOPE ARRAY OBSERVATORY

E. Antolini†, I. Oya, CTAO gGmbH, Heidelberg, Germany
D. Melkumyan, K. Mosshammer, DESY, Zeuthen, Germany for the CTA Observatory

Abstract

The Array Control and Data Acquisition (ACADA) software is a crucial part of the Cherenkov Telescope Array (CTA) Observatory and requires the definition of an appropriate Software Quality Assurance (SQA) process. This activity is necessary to ensure the development and maintenance of a high-quality product, throughout the construction stage and the whole lifetime of the Observatory. Software Development, Software Management, and Software Verification and Validation Plans are the mainstays of the SQA activities documented in the SQA Plan (SQAP). The Scope of this paper is to describe the SQAP proposed for the ACADA work package, which includes all the necessary actions planned to guarantee process and product conformance to the ISO/IEC 25010:2011 standard.

INTRODUCTION

The Cherenkov Telescope Array (CTA) is the next-generation atmospheric Cherenkov gamma-ray observatory. It will be the world's largest ground-based facility for gamma-ray astronomy at very-high energies. The observatory will be composed of more than 100 telescopes and different types of calibration devices that need to be centrally managed and synchronized to perform the required scientific and technical activities.

The operation of the array requires the presence of a complex Supervisory Control and Data Acquisition (SCADA) system, named Array Control and Data Acquisition (ACADA) [1]. The quality level of the ACADA work package is crucial for maximizing the efficiency of the CTA operations.

An appropriate SQA activity is fundamental to ensure the development and maintenance of a high-quality ACADA product. Furthermore, it will help to improve reliability, maintainability, and cost-saving through the prompt discovery of problems.

In the next sections, we will describe the SQA organization and planned activities. We will show how the defined roles and tasks are coherently scheduled and organized with the ACADA Management Plan, Software Development Life Cycle (SDLC) Plan [2]. We will present the quality models and the related metrics defined to comply with the required quality standards. Finally, we will describe the procedures and methods applied to guarantee that, for each phase of the project, the required level of quality in the design, implementation, testing, integration, configuration, usage and maintenance of the ACADA product are met.

QUALITY ASSURANCE MANAGEMENT ORGANIZATION

This section describes relevant roles and responsibilities applicable to the ACADA-SQAP, together with the software quality tasks to be performed.

Roles and Responsibilities

The ACADA SDLC follows an iterative and incremental model which is based on the ISO/IEC 12207:2017 standard [3]. The roles and responsibilities of the personnel assigned to the ACADA quality assurance activity are listed in Table 1. They are individuated based on the roles and responsibilities defined in the ACADA Management Plan and ACADA SDLC [4]. Depending on the level of responsibility of the QA activity to be performed, four main bodies have been defined[1].

ACADA SQA Management Body is responsible for the organization and implementation of the SQA activities related to the definition of the program and associated budget and personnel needs. This body is also responsible for the supervision and control of the planned activities for the product and maintenance quality assurance.

The quality aspects of the ACADA system in the global quality framework of the whole CTA project involve the presence of the SQA Top Management Body, responsible for the top-level tasks of the quality process assurance.

The SQA Professional Body is responsible for the technical implementation of the measurable quality quantities (metrics) and the execution of the tests. This body is mainly composed of designers, developers, and testers.

The validation and verification activities are performed by the SQA Validation and Verification Body, which can be composed of external and internal members of the project (e.g. ACADA Coordinator, stakeholders). This body is in charge to supervise the quality tests of the product and to organize audits to evaluate the related quality reports.

PRODUCT ASSURANCE

The quality models adopted for ACADA are two, as recommended by [5]: Product Quality Model and Quality in Use Model. They are related to the definition of the quality

† elisa.antolini@cta-observatory.org

[1] These bodies are organization roles covered by already existing or planned personnel from the CTA project organization chart.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOMPL006

AUTOMATIC DEPLOYMENT IN A CONTROL SYSTEM ENVIRONMENT*

M. Konrad†, S. Beher, A. Lathrop, D. Maxwell, J. Ryan
Facility for Rare Isotope Beams, Michigan State University, East Lansing, USA

Abstract

Development of many software projects at the Facility of Rare Isotope Beams (FRIB) follows an agile development approach. An important part of this practice is to make new software versions available to users frequently to meet their changing needs during commissioning and to get feedback from them in a timely manner. However, building, testing, packaging, and deploying software manually can be a time-consuming and error-prone process. We will present processes and tools used at FRIB to standardize and automate the required steps. We will also describe our experience upgrading control system computers to a new operating system version as well as to a new EPICS release.

INTRODUCTION

FRIB [1] is a project under cooperative agreement between the US Department of Energy and Michigan State University (MSU). It is under construction on the campus of MSU and will be a new national user facility for nuclear physics. Its driver accelerator is designed to accelerate all stable ions to energies >200 MeV/u with beam power on the target up to 400 kW [2]. Commissioning of the second linac segment is currently underway and the accelerator is planned to support routine user operations in 2022 [3].

FRIB's controls group strives to support commissioning and operation by rolling out bug fixes and new features as fast as possible. To make this happen controls engineers are following principles of agile software development which include iterative, incremental and evolutionary development and a short feedback and adaption cycle. Unfortunately, this approach can be slowed down significantly by the fact that building and deploying control-system software can be a complex and error-prone process that often requires considerable manual work by experts. In the following we will describe how we speed up the build and deployment process for FRIB's controls software by following continuous integration (CI) and continuous delivery (CD) principles.

CONTROL SYSTEM ENVIRONMENT

The vast majority of computers on FRIB's control-system network are based on the x86-64 architecture. This includes workstations, servers, as well as industrial computers in cPCI and MicroTCA form factor. FRIB has standardized on Debian GNU/Linux as operating system for control-system computers. Even hard real-time applications are running on Debian GNU/Linux with a real-time kernel rather than on special real-time operating systems like VxWorks or RTEMS. This heavily standardized environment helps to keep the test matrix small, simplifies the CI infrastructure and allows sophisticated tools developed in the IT industry to be leveraged for software deployment, configuration management as well as for monitoring.

FRIB's controls network is independent from its office network. For security reasons the two networks use separate hardware and a firewall allows only predefined connections between them. In addition to the production control system used to operate the accelerator, FRIB's controls group operates a development control system on a separate network to support development and testing. The development network mimics the architecture of the production network. In particular both networks share the same network topology, run the same IT infrastructure services (DHCP, DNS, storage servers, hypervisors,...) as well as the same control-system services (Alarm Server, Archiver Appliance, Channel Finder,...). A limited number of “control-room” workstations as well as control-system devices like programmable-logic controllers, motor controllers, LLRF controllers etc. are available to support development and testing. In some cases simulation applications are mimicking the behavior of hundreds of devices based on a simplified model [4]. Virtualization is used extensively on both networks to increase availability and flexibility as well as to reduce hardware and maintenance cost. The similarity between both networks allows many bugs to be discovered on the development network before software is deployed to the production network.

CONTINUOUS INTEGRATION

All source code required to build software for FRIB's control system is stored on a central Git [5] repository server on the office network. The revision-control workflow largely follows the Gitflow [6] approach which requires developers to implement new features and bug fixes on feature branches allowing them to work on their feature without the risk of breaking other developer's build. Each repository contains branches for “unstable” and “release” targets. Completed feature branches are merged into an “unstable” branch. Tested software is released by merging from an “unstable” branch into a “release” branch. For both branches software is automatically built, tested and packaged on a Jenkins CI cluster [7]. The resulting Debian packages are pushed into separate Aptly [8] repositories for the development and production environments. See [9] for an in-depth description of FRIB's CI approach.

AUTOMATIC DEPLOYMENT

The Debian package repository as well as the Git repositories are mirrored into the corresponding controls network

* Work supported by the U.S. Department of Energy Office of Science under Cooperative Agreement DE-SC0000661
† konrad@frib.msu.edu

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOMPL007

THE DESIGN OF INTELLIGENT INTEGRATED CONTROL SOFTWARE FRAMEWORK OF FACILITIES FOR SCIENTIFIC EXPERIMENTS

Z. G. Ni†, L Li, J. Luo, J. Liu, X. W. Zhou
Institute of Computer Application, China Academy of Engineering Physics, Mianyang City, China
Y. Gao, Stony Brook University, New York, USA

Abstract

The control system of the scientific experimental facility requires heterogeneous equipment access, domain algorithm, sequence control, monitoring, log, alarm and archiving. We must extract common requirements such as monitoring, control, data acquisition. Based on the TANGO framework, we design typical device components, algorithms, sequence engines, graphical models and data models for scientific experimental facility control systems developed to meet common needs, and are named the intelligent integrated Control Software Framework of Facility for Scientific Experiments (iCOFFEE). As a development platform for integrated control system software, iCOFFEE provides a highly flexible architecture, standardized templates, basic functional components and services for control systems that increase flexibility, robustness, scalability and maintainability. This article focuses on the design of the framework, especially the monitoring configuration and control flow design.

INTRODUCTON

Large scientific facilities [1] generally have tens of thousands to hundreds of thousands of irregular control points, and the types and quantities of controlled devices are huge. The control system must be highly automated and robust, requiring continuous operation for many days. The construction period of the project is long, and the demand in many fields will constantly change. The control system must accept these changes and adapt to the changing and expanding needs.

We urgently need to consolidate the common requirements of monitoring, control, data acquisition and storage of the control system to meet the functional performance requirements of the facility's control system. iCOFFEE is a distributed, hierarchical, object-oriented control software framework through which many similar application software systems are built.

This control software framework [2] is devised to address the general problem of providing distributed control for large scientific facilities that do not require real-time capability within the supervisory software. Sometimes real-time control is also necessary, which is partly solved by an integrated time system or an industrial control system, which is not discussed in this article.

† email address: drops.ni@caep.cn

SYSTEM ARCHITECTURE

The control system architecture of a typical large scientific facilities is a two-tier architecture consisting of a monitoring layer of the network structure and a control layer of the fieldbus structure. The monitoring layer is deployed on the virtual server and the console computer to provide centralized operations for control, status, and data storage. The control layer is deployed on the virtual server or embedded controller to provide real-time collection and control of the device.

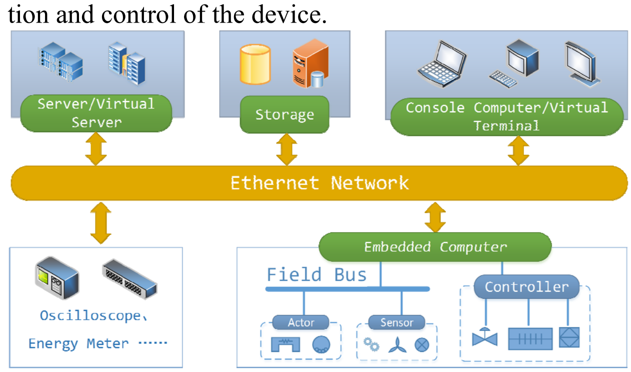


Figure 1: The control system architecture of the typical large scientific facilities [3].

As provided in Fig. 1, the monitoring layer is a software system based on Ethernet structure. It consists of a network switching system, a server system, and a console computer. It provides system services and human-machine interfaces for the facility control system, including control, monitoring, and data management.

The control layer is a fieldbus-based data acquisition and control software system consisting of a network switching system, a server system and an embedded controller. The device service software is used to collect and control the device.

The control point consists of sensors and actuators that are connected to the control layer via a fieldbus or network interface. A large number of IO devices are accessed through the PLC controller, and several intelligent controllers directly access the aggregation switching system through a serial port server or a network interface.

SOFTWARE ARCHITECTURE

In order to realize the special requirements of the facility control system, the software architecture will adopt a hierarchical SOA model, which is a typical pyramid model, which is aggregated layer by layer, and the granularity is larger. The software is divided into three layers: device

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOMPL008

NEW NEUTRON SENSITIVE BEAM LOSS MONITOR (nBLM)

Y. Mariette †, V. Nadot ††, Q. Bertrand, F. Gougnaud, T. Joannem, T. Papaevangelou, L. Segui,
IRFU, CEA, Université Paris-Saclay, F-91191, Gif-sur-Yvette, France
I. Dolenc Kittelmann, F. Alves, ESS ERIC, Lund, Sweden
G. Jabłoński, W. Cichalewski, W. Jałmużna, R. Kiełbik, TUL, DMCS, Łódź, Poland

Abstract

The beam loss detection is of the utmost importance for accelerator equipment safety. At CEA, we are closely collaborating with ESS and DMCS on development of ESS nBLM. The system is based on Micromegas gaseous detector sensitives to fast neutrons produced when beam particles hit the accelerator materials. This detector has powerful features: reliable neutron detection and fast time response.

The nBLM control system provides slow monitoring, fast security based on neutron counting and post mortem data. It is fully handled by EPICS, which drives 3 different subsystems: a Siemens PLC regulates the gas line, a CAEN crate controls low and high voltages, and a MTCA system based on IOxOS boards is in charge of the fast data processing for 16 detectors. The detector signal is digitized by the 250Ms/s ADC, which is further processed by the firmware developed by DMCS and finally retrieved and sent to EPICS network.

For other accelerator projects, we are designing nBLM system close to ESS nBLM one. In order to be able to sustain the full control system, we are developing the firmware and the driver.

This paper summarizes CEA's work on the nBLM control system for the ESS and other accelerators.

INTRODUCTION

The CEA Institute of Research on Fundamental Universe law (IRFU) works on astrophysics, particle physics, nuclear physics and instrumentation for projects like accelerator, magnetic resonance imaging, ITER (International Thermonuclear Experimental Reactor), etc. The institute has a lot of experience in contributing in accelerator facilities and especially for RFQ construction, and instrumentation.

In the context of an in-kind collaboration between IRFU and ESS (European Spallation Source), a new type of beam loss monitor (BLM) based on detection of fast neutrons has been designed, manufactured and characterized. The BLM is crucial in any accelerator; it quickly detects, measures and locates the beam loss. The new neutron sensitive Beam Loss Monitoring (nBLM) system has been designed for fast and accurate measurement of number of neutrons produced when beam particles hit the accelerator material.

The nBLM detectors are powered with high and low voltages and filled with gas, while the acquisition continuously makes analysis to detect and count neutrons. All electronics devices are remotely controlled and monitored by EPICS.

At IRFU, the System Engineering Department (DIS) is in charge of the nBLM control system development for the ESS-nBLM project and it is also involved in other projects based on the same detector technology. The Department of Detector Electronics and Software for the Physics (DEDIP) designs, constructs and qualifies the nBLM detectors.

THE NBLM DETECTOR

In low energy region of hadron accelerators, x-rays, gammas and neutrons are present. Neutrons are produced by the beam loss whereas gammas and x-rays result from the RF. As the x-rays levels can exceed the neutron levels, it is difficult to detect a beam loss in such region [1].

The new nBLM detector is able to detect the produced fast neutrons while having a low sensitivity to gammas and x-rays.

nBLM detectors are based on the use of Micromegas detectors [2]. Micromegas (MicroMesh Gaseous Structures) are gaseous amplification structures. The detectors are bulk Micromegas [3] and are equipped with the fast front-end electronics cards (FEE) placed on-board of the detector, outside the gas volume protected by a Faraday cage (see Fig. 1). The FEE is based on the FAMAS current amplifiers (Fast Amplifier Module for Micromegas ApplicationS) [4]. The advantage to amplify the signal before transmitting it over the long cable is that we are sensitive to smaller signals.

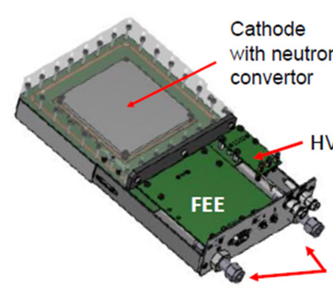


Figure 1: Detector schematic with its FEE.

Two types of nBLM detectors with complementary functionality have been developed. Fast detectors aim to detect serious losses in case of an accident. On the other hand, slow detectors primarily aim to accurately monitor small losses when low particle fluxes are expected but it introduces a reaction delay with around a hundred microseconds. Details of the final detector module design and performance are available in [5].

† yannick.mariette@cea.fr
†† victor.nadot@cea.fr

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOMPL009

CONTROL SYSTEM VIRTUALIZATION AT KARLSRUHE RESEARCH ACCELERATOR

W. Mexner*, B. Aydt, D. Hoffmann, E. Bründermann,
E. Blomley, M. Schuh, A.-S. Müller, Karlsruhe Institute of Technology, Karlsruhe, Germany
S. Marsching, aquenos GmbH, Baden-Baden, Germany

Abstract

With the deployment of a Storage Spaces Direct hyperconverged cluster in 2018, the whole control system server and network infrastructure of the Karlsruhe Research Accelerator have been virtualized to improve the control system availability. The cluster with 6 Dell PowerEdge R740Xd servers with 1,152 GB RAM, 72 cores and 40 TB hyperconverged storage operates 120 virtual machines in total. We will report on our experiences running EPICS IOCs and the industrial control system WinCC OA in this virtual environment.

INTRODUCTION

The Karlsruhe Research Accelerator (KARA) at KIT [1] is a 110m 2.5 GeV electron storage ring with a 53 MeV microtron and 500 MeV booster. The design of the KARA control system is based on EPICS 7.0 with Control System Studio (CSS) as the main operators interface to the control system. Due to historical reasons, the beam lines are controlled by TANGO and the commercial SCADA System WinCC OA 3.15 [2].

Already in 2014, first steps in control system virtualization were successfully taken based on KVM at FLUTE [3] and also at KARA with a small Hyper-V Cluster with a network attached storage for the virtual machines (VM). As most of the control system servers had to be replaced in 2017, the decision was taken to move towards a complete virtualization of all servers. The reasons have been:

- Higher control system availability due to the ability to have zero-downtime hardware and base-OS updates through live-migration of VMs
- The ability for a VM to restart on another host in the event of sudden hardware failure
- Automatic deployment of new servers without any hardware depencies and without buying extra servers.
- Easy resource management of storage and RAM.
- Simple hyper-visor based backup routines

On the other hand, virtualization introduces also new pitfalls:

- As it is easy to create new VMs, you can also easily keep preliminary versions of control servers. These old dormant VMs can deviate far from the actual control system baseline causing a lot of trouble, if inadvertently started.
- Automatic migration: It is tempting to configure automatic migration of control servers in case of hardware failures of a virtualization cluster, but if you have not really a high availability cluster, then this means restart of the servers on a new node causing short interruptions of several servers.
- Control over virtual networks: It is very comfortable having also the network virtualized. But changing from one network to the other means only changing a number for the network adapter, shutting down the network, if inadvertently changed.
- Network stability: Your cluster backbone has to be really stable. From experience, some control hardware has a corrupt network stack implemented, working badly with a virtualized environment
- Resource over provisioning: Virtual machines are splitting the maximum I/O of the virtualiziation hosts. Therefore you need to take care for the I/O load of all virtual machines.

HYPER-V CLUSTER CONCEPT

For KARA we started with the traditional concept of a network attached storage (RAID 5 with 16 hard disks, connected with 10 GBit Ethernet) for the VM images and a pool of Hyper-V compute hosts. Quickly we found that in this simple setup with around 60 virtual machines the limiting factor was the I/O capacity of the NAS storage slowing down all virtual machines.

In 2016, Microsoft introduced the concept of Storage Spaces Direct with Windows Server 2016, so we decided to test for the next productive cluster this promising concept. Storage Spaces Direct [4] uses industry-standard servers with local-attached drives to create highly available, highly scalable software-defined storage at a fraction of the cost of traditional SAN or NAS arrays. Its converged or hyperconverged architecture radically simplifies procurement and deployment, while features such as caching, storage tiers, and erasure coding, come together with the latest hardware innovations such as RDMA networking and NVMe drives.

Hardware Concept

The cluster consists of 6 Dell PowerEdge R740Xd servers each having 3.2 TB NVME for cache, 3.2 TB SSD for Hot-Data Tier and 16 TB HDD for Cold Data and a Intel Xeon Gold CPU with 12 cores, resulting in a final cluster capacity of 1,152 GB RAM, 72 cores and 40 TB storage (see Fig. 1).

In 2019, the control group faced the problem, that the used version of WinCC OA (3.11) was discontinued and not anymore running under modern operating systems. In addition, the newest version 3.15 was incompatible to the

* wolfgang.mexner@kit.edu

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOMPL010

DATA STREAMING WITH APACHE KAFKA FOR CERN SUPERVISION, CONTROL AND DATA ACQUISITION SYSTEM FOR RADIATION AND ENVIRONMENTAL PROTECTION

A. Ledeul, G. Segura Millan, A. Savulescu, B. Styczen, CERN, Geneva, Switzerland

Abstract

The CERN HSE - occupational Health & Safety and Environmental protection - Unit develops and operates REMUS - Radiation and Environmental Unified Supervision - , a Radiation and Environmental Supervision, Control and Data Acquisition system, covering CERN accelerators, experiments and their surrounding environment.

REMUS is now making use of modern data streaming technologies in order to provide a secure, reliable, scalable and loosely coupled solution for streaming near real-time data in and out of the system.

Integrating the open-source streaming platform Apache Kafka allows the system to stream near real-time data to Data Visualization Tools and Web Interfaces. It also permits full-duplex communication with external Control Systems and IIoT - Industrial Internet Of Things - devices, without compromising the security of the system and using a widely adopted technology.

This paper describes the architecture of the system put in place, and the numerous applications it opens up for REMUS and Control Systems in general.

INTRODUCTION

CERN Radiological and Environmental Monitoring System

CERN HSE Unit is in charge of the implementation of CERN Safety Policy, which includes Radiation Protection and Environmental Impact Monitoring.

This CERN-wide monitoring program measures the radiological and environmental impact of CERN, and ensures workplace safety. It also provides the necessary data for reporting to the public and host states authorities.

In order to achieve these objectives, the HSE Unit sets up and operates heterogeneous instrumentation, able to measure the nature and quantity of ionizing radiations produced by the accelerators, possible contaminations and conventional environmental parameters.

In addition, CERN HSE Unit provides a SCADA - Supervision, Control And Data Acquisition - system, called REMUS [1], based on WinCC Open Architecture - WinCC OA [2], interfacing this instrumentation and allowing its operation from various control rooms scattered across the Organization.

At the time of writing, REMUS is interfacing 80 device types. It contains 650,000 tags, manages 84,000 alarms and handles a throughput of 3,700 changes per second. REMUS archives roughly 38 billion measurements per year.

Need for a Stream Processing Platform

One of the main challenges of SCADA systems is to exchange data to and from other systems, in particular data visualization tools, web applications, data storage solutions and other SCADA systems.

Furthermore, the expansion of Industrial IoT devices and their respective monitoring solutions, created a need for a fast and standard intercommunication solution that would allow the SCADA to interface with these devices.

Recent developments in feature-rich data streaming platforms, such as Amazon Kinesis [3] or Apache Kafka [4] have made this achievable with minimal investment.

In the context of Safety SCADA Systems, certain requirements must be met for such data streaming solutions:

- Security: data shall be streamed and accessed in a secure manner; Access Control Lists - ACLs - must be put in place.
- Reliability: given the important nature of data, the streaming solution has to be fault tolerant and highly available.
- Scalability: the platform shall be scalable horizontally in order to face the steady growth of data volumes and exchanges.
- Performance: the platform shall be able to handle an important throughput, and provide the data with a low latency.
- Loose coupling: the SCADA system must be able to stream data in and out with low to no impact on the system itself. In the context of near-real time data streaming, an asynchronous transmission of data is preferable, in order to reduce latency, diminish producer / consumer dependency and accommodate simultaneously multiple types of data consumers.

Open-source Data Streaming

With regards to previous requirements, REMUS opted for Apache Kafka, a scalable, open-source and widely used distributed streaming platform, originally developed by LinkedIn for log processing [5]. It is today used by more than 13,000 companies, including Netflix, Uber, Spotify, Cisco, Yahoo, Twitter, Square, and overall a third of the Fortune 500 companies.

Apache Kafka provides the following key functionalities:

- Publisher / Subscriber mechanism for streams of data (Asynchronous message queuing system)
- Built in Kerberos / TLS - Transport Layer Security - based authentication, authorization and encryption mechanisms
- Distributed and fault-tolerant data retention

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOMPR002

IMPROVING USER INFORMATION BY INTERFACING THE SLOW CONTROL'S LOG AND ALARM SYSTEMS TO A FLEXIBLE CHAT PLATFORM

M. Ritzert*, Heidelberg University, Heidelberg, Germany
on behalf of the Belle II PXD Collaboration

Abstract

Research groups operating large experiments are often spread out around the globe, so that it can be a challenge to stay informed about current operations. We have therefore developed a solution to integrate a slow control system's alarm and logging systems with the chat system used for communication between experimenters. This integration is not intended to replace a control screen containing the same information, but offers additional possibilities:

- Instead of having to open the control system's displays, which might involve setup work (VPN, remote desktop connections, . . .), a web interface or an app can be used to track important events in the system.
- Messages can easily be filtered and routed to different recipients (individual persons or chat rooms).
- Messages can be annotated and commented on.

The system presented uses Apache Camel to forward messages received via JMS to Rocket.Chat. Since no binding to Rocket.Chat was available, this interface has been implemented. On the sending side, a C++ logging library that integrates with EPICS IOCs and interfaces with JMS has been designed.

IMPLEMENTATION

The gateway combines data from the BEAST alarm system [1] and the message log. Both systems are configured to publish their messages via the ActiveMQ message broker, that is central to the distribution of all messages in the system. The gateway is registered as a listener on the respective channels. It receives all messages, filters them and forwards to the Rocket.Chat server.

All messages are also archived in an Elasticsearch database. Archived log messages from this database are combined with live messages received via JMS to provide a fast and comprehensive overview in the CSS GUI, that is also used to control the alarm system.

The outline of the system is shown in Fig. 1. The gateway between ActiveMQ and Rocket.Chat is implemented as an Apache Camel application. Camel is a message routing engine implemented in Java [2]. Its modular design means that it is easy to provide new functionality on all levels interesting for this work: message reception, transformation, routing and output. Messages received in Camel applications are passed through several different modules connected together

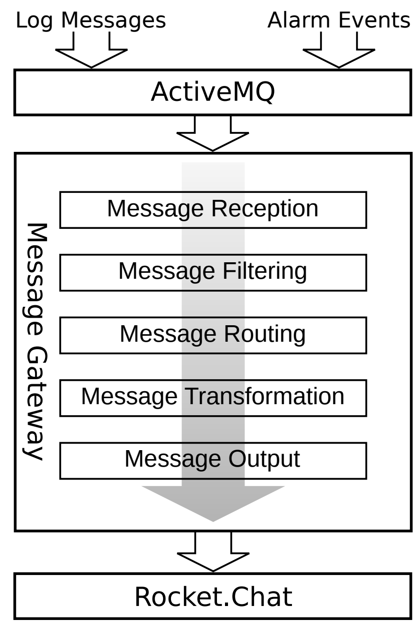


Figure 1: Processing Steps for Log and Alarm Messages.

```
from("activemq:topic:LOG")
.filter().method("LogFilter")
.bean("LogConverter")
.to("rocketchat:https://my.chat:↩
#channel?accessToken=token&userId=user");
```

Figure 2: Example Camel Route for Log Messages.

to form a route. From the framework's point of view, messages are passed around as anonymous Java objects. It is the responsibility of the application to ensure that the output and input expectations of connected modules match. Modules can discard messages, or modify them to the extend that an object of a different class is passed on.

In a typical configuration, there are at least two routes, one for alarm messages, and another one for log messages. Figure 2 shows the Java code to create a route for log messages from the `LOG` topic to `#channel` on a Rocket.Chat server. The modules described below are implemented in the `LogFilter` and `LogConverter` classes, and made available as the `rocketchat:` output module.

Message Logging

In order to log messages from an IOC, a C++ library, Logfile, has been implemented. The most important functionality is logging to an ActiveMQ server via the STOMP protocol. The messages will be converted to the MapMes-

* michael.ritzert@ziti.uni-heidelberg.de

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOMPR003

DATA VISUALIZATION WITH DATA BROWSER SOFTWARE

K.Saintin, IRFU, CEA, Université Paris-Saclay, Gif-sur-Yvette, France R.Girardot
Synchrotron SOLEIL, Saint-Aubin, Gif-Sur-Yvette, France

Abstract

Scientific facilities need to visualize a large amount of data through several dedicated applications. These applications can monitor variables from a PLC, visualize data acquisition or browse them offline. Thus, an intuitive GUI (**G**raphical **U**ser **I**nterface) tool is necessary to handle multiple data sources.

In 2012, the computing team of SOLEIL[1] started the development of the Data browser tool. This software uses modular and extendable frameworks on which several institutes collaborated:

- CDMA (**C**ommon **D**ata **M**odel **A**ccess), initially developed by ANSTO, unifies the data access regardless of its physical container (files, database …), and its logical organization.

- COMETE (**COM**munity of **E**xtendable **T**oolkit for **E**xperiment) graphical framework, initially developed by SOLEIL, which provides a set of data visualization widgets and unifies the way there are connected to the data they represent regardless of the source of this data.

Since then, SOLEIL developed several plugins for Data browser: HDF/Nexus, TANGO. In addition, a collaboration with IRFU [2] control software team which is using EPICS has been created. To fit Irfu needs, Data browser tool is integrating new plugins: EPICS Channel Access, EPICS Archiver Appliance.

CONTEXT

Today in scientific institutes, our end users require that GUI applications display data coming from various sources. Live data from control system as EPICS [3] process variables, TANGO [4]'s attributes or archived data stored in a database as TANGO archiving system and scientific measurement data stored in files, as NeXus [5]/HDF5 [6] files or EPICS archiver appliance [7] Google protocol buffers files[8]. Users would like to use the same collection of widgets for displaying the different data sources (see Fig.1). Furthermore, it is convenient for them to have a single tool to explore those data.

To address this issue, the computing team of SOLEIL developed the Data browser tool. This software is based on two main frameworks explained in the following chapters, CDMA [9] and COMETE [10].

Figure 1: Project context.

COMMON DATA MODEL ACCESS

Working independently, SOLEIL and ANSTO [11] software development focused on the design of frameworks for data processing, operating on top of a HDF data format. ANSTO worked on a Gumtree Data Model [12] abstracting data file access of the underlying HDF files (the standard data format at ANSTO) by designing a data model with a set of Java interfaces. This seemed to be a very promising development to share. SOLEIL became interested in the concept as it was coincidentally looking for a unified data access layer based on NeXus (the standard data format used at SOLEIL). The collaboration started between ANSTO and SOLEIL in January 2010, after a meeting between the authors at ICALEPCS 2009. The work started from the data access layer of ANSTO's GumTree project, written in Java.

Common Data Model Access (CDMA) framework proposes a solution and allows separation of responsibilities between data reduction developers and the institute. Data developers are responsible for data reduction code; the institute provides a plug-in to access the data.

The CDMA is a core API that accesses data through a data format plug-in mechanism and scientific application definitions (sets of keywords) coming from a consensus between scientists and institutes. Using an innovative "mapping" system between application definitions and physical data organizations, the CDMA allows application development independently of the data file container and schema (see Fig. 2). Each institute develops a data access plug-in for its own data file formats along with the map-ping between application definitions and its data files.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOMPR004

CONTROL AND ANALYSIS SOFTWARE DEVELOPMENT AT THE EUROPEAN XFEL

S. Brockhauser†[1], M. Beg, M. Bergemann, V. Bondar, C. Carinan, R. Costa, F. Dall'Antonia, C. Danilevski, W. Ehsan, S.G. Esenov, R. Fabbri, G. Flucke, D. Fulla Marsa, G. Giovanetti, D. Goeries, S. Hauf, D.G. Hickin, T. Jarosiewicz, E. Kamil, Y. Kirienko, A. Klimovskaia, T.A. Kluyver, D. Mamchyk, T. Michelat, I. Mohacsi, A. Parenti, R. Rosca, D.B. Rück, R. Schaffer, A. Silenzi, M. Spirzewski, S. Trojanowski, C. Youngman, J. Zhu, H. Santos, H. Fangohr[2]
European XFEL, Schenefeld, Germany

[1]also at University of Szeged, Szeged, Hungary, and
Biological Research Center, Szeged, Hungary
[2]also at University of Southampton, Southampton, U.K.

Abstract

Agile Project Management (Agile PM), coupled with the DevOps concept, has been worked out as a fundamental approach in a highly uncertain and unpredictable environment to achieve mature software development and to efficiently support concurrent operation [1]. At the European XFEL [2], Agile PM and DevOps have been applied to provide adaptability and efficiency in the development and operation of its control system: Karabo [3,4]. In this context, the Control and Analysis Software Group (CAS) has developed in-house a management platform composed of the following macro-artefacts: (1) Agile Process; (2) Release Planning; (3) Testing Infrastructure; (4) Roll-out and Deployment Strategy; (5) Automated tools for Monitoring Control Points (i.e. Configuration Items [5]) and; (6) Incident Management [6]. The software engineering management platform is also integrated with User Relationship Management to establish and maintain a proper feedback loop with our scientists who set up the requirements. This article aims to briefly describe the above points and show how agile project management has guided the software strategy, development and operation of the Karabo control system at the European XFEL.

INTRODUCTION

European XFEL is a new facility with six scientific instruments as experimental stations at the end of its three SASE beamlines [2] producing trains of very short X-ray pulses. While the electrons are accelerated along a 2.1 km long tunnel, the SASE photon beams are created and transported in an additional tunnel system with the total length of 3.6 km. The accelerator machine is operated by DESY [7] and is controlled by the DOOCS [8] control system which is also supporting the FLASH facility. In contrast, EuXFEL's photon transport and its experiments at the scientific instruments are controlled by a novel in-house developed control system Karabo [3,4].

Karabo version 2.1.5 has been released in March 2017 and has been used to start the commissioning of the first beamline SASE1. Although this version could already support basic demonstration setups properly, it was lacking a huge list of essential control system features and was still suffering from reliability and stability issues. Hence, control software framework development had to be continued parallel to providing 24/7 on-call support for the commissioning and later even user operation activities (see the applied priority pyramid on Fig. 1). Next to on-call support, the parallel commissioning of continuously upcoming instrumentation set an additional challenge because of the conflicting and frequently changing priorities which was heavily influencing the development roadmap.

Another challenge was that only a few developers knew the Karabo system and even less the code itself. Progress required the introduction of new personnel while keeping and spreading the know-how, as well as avoiding single-point of failures and technical debts.

With the support of our management, we have introduced an efficient group structure, and took appropriate management, software engineering and technology choices for addressing these issues and have successfully driven the facility through its commissioning phase and started its operation.

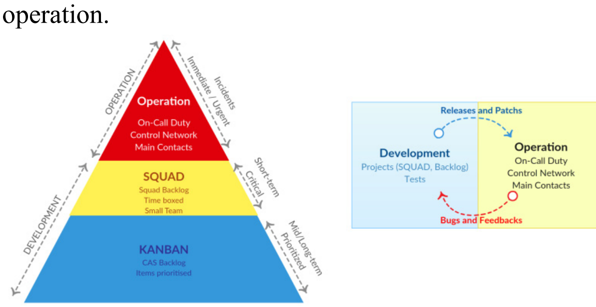


Figure 1: Priority of Operation vs. Development.

SOFTWARE ENGINEERING PLATFORM

The software engineering management platform has changed according to the different phases it had to support:

- *software framework development before commissioning*. In a short period before commissioning, we had the chance to implement major (and incompatible) changes in the framework which would have been very difficult during operation. In this limited period, we

† sandor.brockhauser@xfel.eu

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOMPR005

DEVELOPMENT OF A NEW DATA ACQUISITION SYSTEM FOR A PHOTON COUNTING DETECTOR PROTOTYPE AT SOLEIL SYNCHROTRON

G. Thibaux†, Y.M. Abiven, D. Bachiller-Perea, J. Bisou, A. Dawiec, A. Jarnac, B. Kanoute, F. Langlois, C. Laulhe, C. Menneglier, A. Noureddine, F. Orsini, Y. Sergent
Synchrotron SOLEIL, Saint-Aubin, France
P. Grybos, A. Koziol, P. Maj, AGH University of Science and Technology, Krakow, Poland
C. Laulhe, Université Paris-Saclay, Saint-Aubin, France

Abstract

Time-resolved pump-probe experiments at SOLEIL Synchrotron (France) have motivated the development of a new and fast photon counting camera prototype. The core of the camera is a hybrid pixel detector, based on the UFXC32k readout chips bump-bonded to a silicon sensor. This detector exhibits promising performances with very fast readout time, high dynamic range, extended count rate linearity and optimized X-ray detection in the energy range 5-15 keV. In close collaboration with CRISTAL beamline, SOLEIL's Detector, Electronics and Software Groups carried out a common R&D project to design and realize a 2-chips camera prototype with a high-speed data acquisition system. The system has been fully integrated into Tango and Lima data acquisition framework used at SOLEIL. The development and first experimental results will be presented in this paper.

CONTEXT

The goal of the project was to develop a complete acquisition system for a detector prototype based on the UFXC32k readout chip [1]. It consisted of design and realization of the specific electronics hardware, firmware and software that allows performing pump-probe experiments at SOLEIL as depicted in Figure 1 and described in [2].

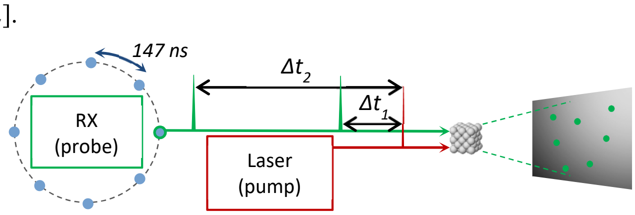


Figure 1: Pump and probe-probe experiment.

The principle of the experiment consists of having laser pulses, at half the frequency of X-ray pulses, to excite a sample (Pump). X-ray pulses, coming from synchrotron bunch, lighting the sample (Probe). Some preliminary results are given in last section, after description of the detector architecture with main focus on the data acquisition system.

HARDWARE DESIGN

In order to optimize development time of the detector acquisition system, several elements of hardware and open firmware/software framework from the PandABox [3,4,5] platform were chosen, reused and adapted for this project.

We have developed hardware architecture in a compact format for the need of the 2chips detector prototype.

The 3 stages electronics, in Fig. 2, consist of:

- The DETECTOR prototype to host the 2-chips hybrid pixel module.
- The DAQ (Data AcQuisition) Box as readout electronics, designed to be a compact hardware variation of PandABox.
- A server to accommodate the slow control and the storage of the fast throughputs of experimental data.

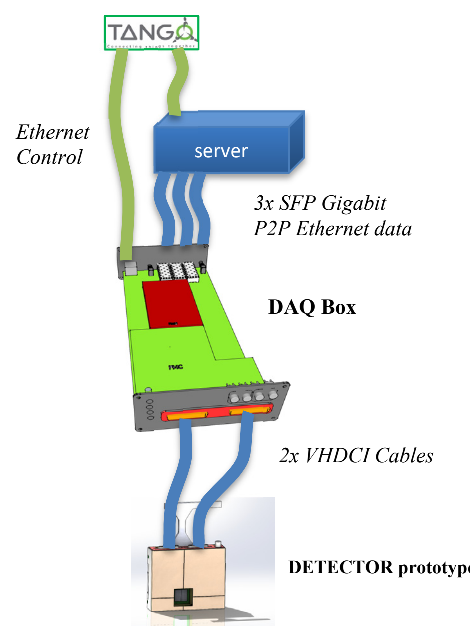


Figure 2: Hardware architecture.

* Work supported by Synchrotron SOLEIL, Paris, France
† gauthier.thibaux@synchrotron-soleil.fr

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOMPR006

PERFORMANCE OF THE ALICE LUMINOSITY LEVELING SOFTWARE ARCHITECTURE IN THE Pb-Pb PHYSICS RUN

G. Valentino*, University of Malta, Msida, Malta
G. De Cataldo[1], A. Franco[1], M. Hostettler, CERN, Geneva, Switzerland
O. Visnyei, Hungarian Academy of Sciences, Budapest, Hungary
[1]also at INFN, Bari, Italy

Abstract

Luminosity leveling is performed in the ALICE experiment of the Large Hadron Collider (LHC) in order to limit the event pile-up probability, and ensure a safe operation for the detectors. It will be even more important during Run 3 when 50 KHz Pb ion-Pb ion (Pb-Pb) collisions will be delivered in IP2. On the ALICE side, it is handled by the ALICE-LHC Interface project, which also ensures an online data exchange between ALICE and the LHC. An automated luminosity leveling algorithm was developed for the proton-proton physics run, and was also deployed for the Pb-Pb run with some minor changes following experience gained. The algorithm is implemented in the SIMATIC WinCC SCADA environment, and determines the leveling step from measured beam parameters received from the LHC, and the luminosity recorded by ALICE. In this paper, the software architecture of the luminosity leveling software is presented, and the performance achieved during the Pb-Pb run and Van der Meer scans is discussed.

INTRODUCTION

A Large Ion Collider Experiment (ALICE) [1] is optimized to study the collisions of nuclei at ultra-relativistic energies, which provide a phase of matter known as the quark-gluon plasma. It is one of the major experiments of the CERN Large Hadron Collider (LHC) [2], which is a 27-km long particle accelerator designed to collide protons and heavy ions with center-of-mass energies of $\sqrt{s} = 14$ TeV and 2.759 TeV/nucleon, respectively. The LHC Interface project [3] (LHC_IF) ensures the online data exchange between the LHC and ALICE, as well as the safe operation of the complex experiment-collider interface.

One of its main tasks is to coordinate the luminosity levelling, which is necessary in order to limit the event pile up probability to a few percent. In the 2018 heavy-ion run, the LHC could provide up to 6×10^{27} cm^{-2}s^{-1} of instantaneous luminosity, however in the ALICE experiment this value had to be reduced through luminosity levelling to a value of 1×10^{27} cm^{-2}s^{-1} by applying a beam-beam separation in the horizontal plane of up to several σ (beam size units). When collisions between the beams are established and the LHC is in stable beams mode, a controlled and automatic luminosity ramp up kicks in to reach the target luminosity defined by ALICE. During the remainder of the fill, minor corrections are applied to the beam separation in order to maintain the luminosity at a constant value. Luminosity levelling was also required during special very low luminosity runs in order to measure and map the distortions in the ALICE Time Projection Chamber with the two B-field polarities.

In this paper, following an description of the luminosity levelling procedure, an overview of the software architecture is provided, and the operational experience with levelling in the Pb-Pb physics run in 2018 is presented and discussed.

LUMINOSITY LEVELLING PROCEDURE

Luminosity levelling via beam separation was first carried out in ALICE with proton-proton collisions in May 2011 [4], and for the first few years of operation made use of a fixed step size in σ to displace the beams until the target luminosity was reached. In 2015, due to the higher beam intensity and the running in main-main beam mode, the fixed step size was deemed no longer viable due to the long and not always convergent levelling procedure. As a result, the LHC machine operator was required to manually steer the beams until the target luminosity was reached, resulting in a loss of data taking time and luminosity overshooting which can damage ALICE's detectors. Therefore, the luminosity levelling procedure was upgraded in 2016 to calculate a dynamic levelling step size based on an inversion of the luminosity formula and taking into account measured beam and machine parameters such as the bunch intensities, number of colliding bunches and β^* (which defines the beam size at the ALICE experiment) [5].

The following intervals of validity for $\Delta\delta$ with the corresponding values of Δd were defined:

$$
\begin{aligned}
\text{if } \Delta\delta \geq 0.52; &\quad \Delta d = 0.5 \\
\text{if } \Delta\delta \geq 0.27 \text{ and } \Delta\delta < 0.52; &\quad \Delta d = 0.25 \\
\text{if } \Delta\delta \geq 0.07 \text{ and } \Delta\delta < 0.27; &\quad \Delta d = 0.06 \\
\text{if } \Delta\delta < 0.07; &\quad \Delta d = 0.025
\end{aligned}
$$

SOFTWARE ARCHITECTURE

The LHC_IF control software is implemented in the SIMATIC WinCC SCADA environment [6] and comprises over 200 control processes, called managers. LHC_IF subscribes to the beam conditions published by the LHC via the data interchange protocol (DIP) [7]. Depending on the particle type (protons or heavy ions), the geometric beam

* gianluca.valentino@um.edu.mt

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOMPR007

SCALABLE HIGH DEMAND ANALYTICS ENVIRONMENTS WITH HETEROGENEOUS CLOUDS

K. Woods, R. Clegg, R. Millward, Tessella Ltd, Abingdon, UK
F. Barnsley, C. Jones, STFC/RAL, Didcot, UK

Abstract

The Ada Lovelace Centre (ALC) at the Science and Technology Facilities Council (STFC) provides on-demand, data analysis, interpretation and analytics services to scientists using UK research facilities. ALC and Tessella have built software systems to scale analysis environments to handle peaks and troughs in demand as well as to reduce latency by provision environments closer to scientists around the world. The systems can automatically provision infrastructure and supporting systems within compute resources around the world and in different cloud types (including commercial providers). The system then uses analytics to dynamically provision and configure virtual machines in various locations ahead of demand so that users experience as little delay as possible. In this poster, we report on the architecture and complex software engineering used to automatically scale analysis environments to heterogeneous clouds, make them secure and easy to use. We then discuss how analytics was used to create intelligent systems in order to allow a relatively small team to focus on innovation rather than operations.

INTRODUCTION

The ALC at the STFC in the United Kingdom (UK) has been established as an integrated, cross-disciplinary data intensive science centre, for better exploitation of research carried out at large scale UK National Facilities including the Diamond Light Source (DLS), the ISIS Neutron and Muon Facility, the Central Laser Facility (CLF) and the Culham Centre for Fusion Energy (CCFE).

The ALC has the potential to transform research at the facilities through a multidisciplinary approach to data processing, computer simulation and data analytics. The impact will be felt across the many science disciplines and communities these facilities support, including industry and academia. However, for many National Facilities user communities, the computing infrastructure can be very difficult to use. Many scientists lack the specialist expertise to fully exploit the advanced computing infrastructure and services available to them. Scientists need to focus on analysing and interpreting the growing volume of data they obtain during their research; the mechanics of managing datasets and configuring environments can be an obstacle to their work. The Ada Lovelace Centre and Tessella have developed a set of tools to facilitate the scaling of computing infrastructure in order to respond flexibly to variations in demand.

PROBLEM

In supporting users of UK National Facilities, ALC faces several challenges:

- the very large and growing volumes of data generated by scientific instruments at facilities such as DLS, ISIS and CLF.
- the variety of scientific techniques (e.g. neutron scattering, x-ray scattering, laser scattering) employed by researchers requires different data analysis techniques.
- each data analysis technique must be made available to researchers in a consistent, repeatable manner.
- analysis of experimental data is typically performed at a scientist's home institute.
- demand for computing resources is variable and unpredictable.

Large datasets generated by experiments are most efficiently stored in data archives located at the National Facilities. It is frequently impractical to transfer such large datasets (often several hundred GB or more in size) to a scientist's home institute, either on physical media or via the internet. Therefore, there is a pressing need to ensure data can be moved quickly and efficiently between archives and compute clusters. Furthermore, scientists need to select the best tools to process and analyse their data, which requires that their experimental data is available on the most appropriate compute resources.

SOLUTION ARCHITECTURE

To address these problems, ALC has created the Data Analysis as a Service (DAaaS) platform. ALC and Tessella have created three key infrastructure components which underpin the operation of DAaaS.

External Cloud Provisioning

STFC operates a high-performance computing system, which runs the OpenStack [1] software platform. Within the STFC network this resource appears to researchers as a private cloud. DAaaS is a collection of cooperating utilities which run on top of this OpenStack instance.

ALC has access to additional compute clouds (for example, via STFC's IRIS eInfrastructure initiative [2]). Commercial cloud offerings also represent a potential source of compute resource that ALC can exploit, if necessary.

To support the operation of the DAaaS across multiple, heterogeneous clouds, we created a provisioning service. Provisioning DAaaS infrastructure on an external cloud is a multi-step process.

1. We use Ansible [3] playbooks and AWX [4] (running on the core DAaaS system in the ALC cloud) to provision the network infrastructure needed for the external DAaaS infrastructure with:
   - a Virtual Private Cloud (VPC) [5].
   - a public and a private subnet within that VPC.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOMPR008

SharePOINT FOR HEPS TECHNICAL SYSTEMS AND PROJECT MANAGEMENT

X.H. Wu†, C.P. Chu, L. Bai, Institute of High Energy Physics, Chinese Academy of Sciences, Beijing, China

Abstract

High Energy Photon Source is the latest planned synchrotron light source in China which is designed for ultra-low emittance and high brightness. The accelerator and beamlines contains tens of thousands of devices which require systematic management during their construction and operation. It is also necessary to capture project management information systematically. HEPS chooses the Microsoft SharePoint as the document tool for the project office and all technical systems. Additionally, Microsoft Project Server on top of SharePoint is used for the project management. Utilizing the SharePoint and Project software can facilitate a lot of daily work for the HEPS project. This paper describes the SharePoint and Project server/client structure and various applications been developed so far.

INTRODUCTION

In Recent years, the Institute of High Energy Physics (IHEP) has launched several "big science" projects, which require high-efficient management and project control in order for on-time and within budget goals. However, the use of modern collaborative work platform is not accustomed in these projects. Most IHEP employees are used to traditional ways such as e-mail and standalone project management tools to perform daily work. The old working ways may be suitable for small or medium research projects but definitely not good for big projects. On the other hand, in the past twenty years, many IT tools have developed to ease our daily work. It can be very beneficial to look into these solutions and pick up suitable tools for our projects. SharePoint [1] based office and project tools are good candidates for IHEP initial IT upgrade work.

A 4th generation synchrotron light source features ultra-low emittance and high brightness, the High Energy Photon Source (HEPS) designed by the IHEP, has started its construction since June 2019. There are about 500 members in HEPS project which includes 5 technology divisions and 50 engineering systems. During the construction phase of HEPS project, more than 20TB size documents will be generated, about tens of thousands professional project tasks will be tracked. Each HEPS's member needs to visit or update documents and project progress frequently. In order to achieve these works accurately and efficiently, it is necessary to build a modern Technical Systems and Project Management Platform.

The purpose of this work is to provide a modern service platform for HEPS project. This paper present the system design and platform development. Members within the project whom are authorized with appropriate permissions in a HEPS resource pool database. With this right permission, one can share and manage his/her project data, documents, photo and other resources to improve work efficiency. Meanwhile, this platform also provides many powerful features such as advanced search and analysis functions through which users can find useful information quickly and then make reasonable decisions accordingly. Additionally, this platform provides Project Web App (PWA) application which allows multiple users to work collaboratively and coordinating effectively on planning, tracking and updating the status of HEPS project.

* Work supported by the National Development and Reform Commission, City of Beijing, and the Chinese Academy of Sciences.
† wuxh@ihep.ac.cn

ARCHITECTURE

System Structure

As shown in Fig. 1, the architecture of HEPS SharePoint includes three layers: network layer, application layer and data layer. The servers in the network layer interacts directly with users, is responsible for publishing the proxy site, processes and forwards all requests from the users, and isolates users and data. Users need to access SharePoint Web Front End (WFE) runs through this layer to achieve security control, load balancing, etc. The network layer cannot directly access data layer and can only be accessed through the application layer. The WFE and APP servers of SharePoint are built at the application layer. The WFE servers are responsible for handling users' requests, and the APP server are used for all running background services. The date layer is used to store most of the system data and user data. There are firewalls between the layers to ensure the independence and security of each layer.

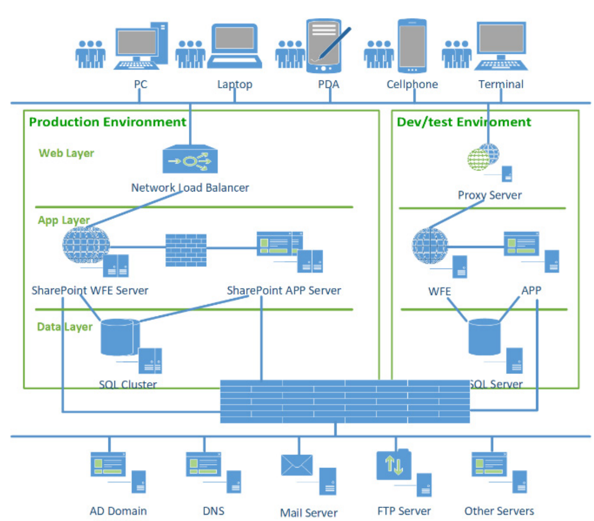


Figure 1: Structure of HEPS technical systems and project management platform.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOMPR009

PROTOTYPE DESIGN FOR UPGRADING EAST SAFETY AND INTERLOCK SYSTEM

Z.C. Zhang†, B.J. Xiao, Z.S. Ji, Y. Wang, Institute of Plasma Physics, CAS, Hefei, PR. China
F. Xia, Southwestern Institute of Physics, Chengdu, PR. China

Abstract

The national project of experimental advanced superconducting tokamak (EAST) is an important part of the fusion development stratagem of China, which is the first fully superconducting tokamak with a non-circle cross-section of the vacuum vessel in the world. The safety and interlock system (SIS) is in charge of the supervision and control of all the EAST components involved in the protection of human and tokamak from potential accidents. A prototype for upgrading EAST SIS has been designed. This paper presents EAST machine and human protection mechanism and the architecture of the upgrading safety and interlock system.

INTRODUCTION

Nuclear fusion which is a controllability reaction has unquestionably been recognized to be one of the best ways to solve the future energy crisis, and the tokamak is a favourable equipment to realize nuclear fusion [1]. The Experimental Advanced Superconducting Tokamak (EAST) in Hefei, a central region city of China, has been a key role for the research of peaceful utilizations of fusion energy. The huge tokamak project is complicated and high cost, needs to be monitored to eliminate the probability of potential hazard during the physical campaign [2]. The role of safety and interlock system (SIS) is focuses on protecting the machine and human from accidents and preventing the propagation of the risk from an accident during the operating campaign.

MOTIVATIONS FOR AN UPGRADE

The SIS is constituted by two horizontal layers, one for the central safety and interlock system (CSIS), and another for the different plant safety and interlock systems, they are connected through optical network. With the development of physical experiment, the SIS had come close to reaching its limits for expandability. For instance, the former central safety and interlock system based on PLC just offers digital I/O channels with 1ms scan time, and the response time of event is around 4ms. What more, the primitive GREEN and RED circles dashboard, and intermit-tent spurious monitor problem make the central control team determines to update the safety and interlock system.

* Work supported by the National Key R&D Program of China No.2017YFE0300504 and No.2018YFE0302104.
† zzc@ipp.ac.cn

The future needs of the EAST SIS research program are: (a) more stable supervisory and control operation; (b) high-precision execute and respond actions; (c) better user-friendly interface [3].

Fulfilling all of these needs will require certain improvements to the previous capabilities including: (1) redesign the structure of SIS, (2) introduction of new equipment, (3) further development of the HMI, and (4) enhancements to the redundancy mechanism capability.

SYSTEM STRUCTURE

The new SIS keeps the former two horizontal layers, and redistributes three vertical architectures according to timing requirements. The structure of the prototype SIS is illustrated in detail in Fig. 1.

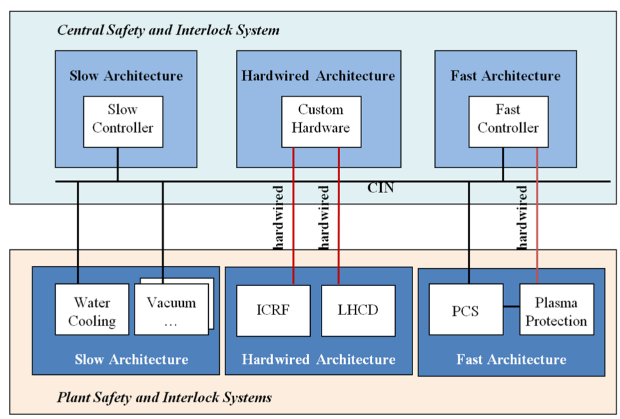


Figure 1: Structure of the prototype SIS.

The CIS is in charge of implementing the central protection functions via the plant safety and interlock systems (PSIS) through the central interlock network (CIN). It also provides access to the local interlock data of the different plant interlock systems. Each PSIS provides local protection implementing the local interlock functions of the corresponding plant system. The CIN provides connection between the CSIS and PSIS for inter-plant systems investment protection functions. Communication within one plant system is carried out through the plant interlock network (PIN) and/or hardwired interconnections.

The systems such as water cooling, vacuum, etc. have slow response time requirements are categorized into slow architecture. System statuses are inspected through digital channels. For the functions of plasma safe stop and/or Poloidal Field (PF) coil current discharge in case an interlock event arises, plasma protection module has to be implemented on a fast architecture. Not only digital

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA001

ROBOTIZING SOLEIL BEAMLINES TO IMPROVE EXPERIMENTS AUTOMATION*

Y.-M. Abiven†, T. Bucaille, L. Chavas, E. Elkaim, P. Gourhant, Y. Liatimi, K. Medjoubi, S. Pierre-Joseph Zephir, V. Pinty, A. Somogyi, F. Thiam, Synchrotron SOLEIL, Paris, France
S. Bouvel‡, EFOR, Levallois Perret, France
B. Pilliaud, Mediane Systeme, Le Pecq, France

Abstract

Synchrotron beamlines can benefit from the implementation of industrial robots in several ways: minimization of dead-time, maximization of experimental throughput, and limiting human presence during experiments. Furthermore, the robots add flexibility in task management. The challenge for SOLEIL is to define a robotic standard on both hardware and software which would be versatile enough to cover beamlines requirements, while being: easy to implement, easy to use, and to maintain in operation.

This paper will present the process of finding the standard definition at SOLEIL, using 6-axis industrial robot arms. It will detail all aspects of this development, from market studies up to technical constraints. The specifications of the robots are aimed at addressing the most common technical constraints of beamlines, with a special care for mechanical properties. The robotic systems will be integrated into the TANGO [1] control system using a feature-based approach. This standard implementation is driven by two applications: picking and placing samples for powder diffraction on the CRISTAL beamline and positioning of a detector for x-rays coherent diffraction experiments on the NANOSCOPIUM beamline.

INTRODUCTION

Beamlines in synchrotron facilities can benefit from the automation of tasks that do not require a high level of expertise. This is especially true when scientists or users have to perform repetitive tasks over long periods of time, e.g. continuously switching between measurements and sample replacements. The industrial robot is commonly used for repetitive tasks such as these. In this paper, the term “robot” refers to six degrees-of-freedom serial-link robotic arms. Robots have been extensively used in the industry, as they can perform a variety of tasks while being very robust.

At SOLEIL, a survey among beamlines has been organized in order to poll for the potential use of industrial robots. Among the beamlines which responded to the survey: 40% expressed a need for robots, 30% answered they would need to study whether a robot would be useful or not, and 30% answered they would not benefit from a robot.

Robots in synchrotron facilities have several potential use-cases:

- Increased beam-availability: beam-exploitation could increase, as they can operate in harsher environments and inconvenient time schedules.
- Increased experimental efficiency: Less time lost at opening hutches, changing samples, etc.
- Limit human intervention to increase overall safety: Less tasks and time in hazardous environments directly diminishes risk – not only for personnel but also for delicate materials, such as samples or detectors.
- Improve the user experience by providing additional automation and therefore lowering the amount of dull tasks.

In order to deploy robots in an easy and efficient way, a standard has been defined: setting up robot criteria, and on the interfaces between the robot and the other equipment. This standardization process is essential for maintaining equipment operation in a complex facility such as a synchrotron.

For a successful implementation of the robots, they have to be fully integrated into the SOLEIL TANGO-based control architecture. This development is user-centered to make sure it is effectively beneficial to the users.

This paper presents the definition of the SOLEIL robotic standard as well as its implementation in two separate applications: a pick-and-place robot for powder diffraction on the CRISTAL [2] beamline, and a detector-positioning robot on the NANOSCOPIUM [3] beamline.

ROBOT INTEGRATION STANDARDIZATION

Several aspects of the robots have to be standardized to ensure a proper implementation. Any new robotic system must be compatible with maintenance in large facilities that holds a limited amount of personnel. This requires: a hard limitation on the assortment of deployed equipment, easy deployments for non-expert roboticists, as well as efficient interventions of technical personnel.

Standardization of the Robot Brand

There are several industrial robot brands that all produce robust and efficient products. Yet, synchrotron applications are quite different from common industrial applications. Each robot brand has its own perks over its competitors.

The criteria’s for choosing industrial robots are generally: price, mechanical characteristics, and velocity. For synchrotron applications, up until now, velocity is not the

* Work supported by Synchrotron SOLEIL, Paris, France
† yves-marie.abiven@synchrotron-soleil.fr
‡ simon.bouvel@synchrotron-soleil.fr

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA002

A MODEL-DRIVEN SERVICE-ORIENTED WIZARD-BASED MULTI-TARGET DEVELOPMENT KIT FOR SUPERVISION SYSTEMS *

C. F. Afonso[1], C. Larizza, Department of Electrical, Computer and Biomedical Engineering
University of Pavia, Pavia, Italy
S. Foglio, S. Gioia, M. Necchi, M. Pullia, Centro Nazionale di Adroterapia Oncologica, Pavia, Italy
S. Toncelli, Consultant on behalf of Centro Nazionale di Adroterapia Oncologica, Novara, Italy
L. Casalegno, Consultant on behalf of Centro Nazionale di Adroterapia Oncologica, Como, Italy
[1] also at Centro Nazionale di Adroterapia Oncologica, Pavia, Italy

Abstract

The operation of particle therapy facilities requires complex control and supervision systems, often spanning several software development environments, each containing a large number of applications. While the component-based development approach brings many benefits, the integration of the parts is still left to the initiative of each developer without a repeatable path that can be demonstrated when the artefacts have to undergo the certification process.

Besides the development of such control and supervision systems is not any more limited to a network based on fixed workstations, but mobile devices have to be taken into account introducing an extra need for enhanced security.

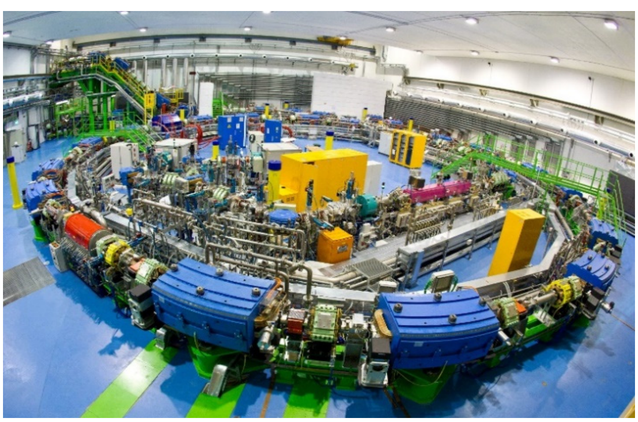
As part of the technological update of one of its development environments, the Centro Nazionale di Adroterapia Oncologica (CNAO) chose to build a development kit to face these new challenges.

This development kit is based on models of applications and services that are managed by wizards that configure the general layout and create the connections among the components so that the integration of the parts is performed in a substantially unique way.

The developer is responsible to choose among the available models that already include the integration to the mandatory services and use the wizards to create the application and the project that is able to build the executable.

Developers are still able to add the specific business logic and the required interactions; nevertheless, they will be directed in doing so by a set of 'hooks' present in each model that shall guarantee repeatability of behaviors also in this area of the work.

In this document, we examine the development kit, highlighting the most valuable aspects that enable to build easily certifiable applications running in a multi-target system. The services that support the actions performed by each application are described in a companion paper [1].



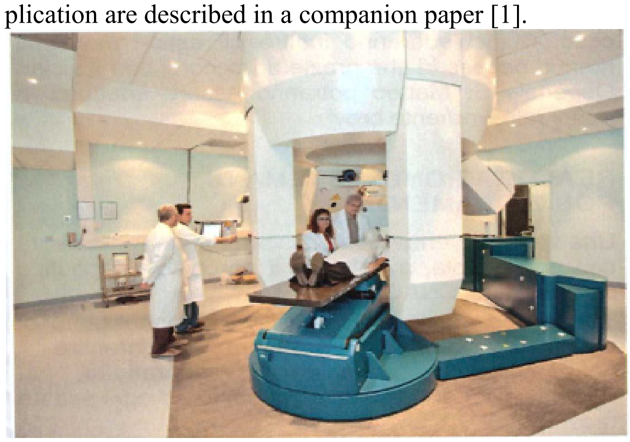


Figure 1: The CNAO accelerators and treatment complex.

MOTIVATION

CNAO (National Centre for Oncological Hadrontherapy) is a clinical facility, established and funded by the Italian Ministry of Health and the Lombardy Region, that uses hadrontherapy for cancer treatments. To date, about 2500 patients completed successfully the treatment.

Hadrons (basically protons and carbon ions) are characterized by a maximum of energy deposition at the end of their range and a sharp penumbra that allows to achieve a precise coverage of the target and an enhanced sparing of the surrounding healthy tissues. In addition, carbon ions are more suitable for the treatment of radio-resistant tumors.

The treatment has the aim to transfer a given amount of energy to a tumor. In doing so, the tumor is subdivided into several slices orthogonal to the beam direction. Each slice can be treated with several beams. Depending on the energy, beams have different characteristics. Thus, during

* This project has received funding from the European Union's Horizon 2020 research and innovation program under the Marie Sklodowska-Curie grant agreement 675265

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA003

INTEGRATING MOBILE DEVICES INTO CNAO'S CONTROL SYSTEM, A WEB SERVICE APPROACH TO DEVICE COMMUNICATION*

C. Afonso[1], C. Larizza, University of Pavia, Pavia, Italy
S. Foglio, S. Gioia, M. Necchi, Centro Nazionale di Adroterapia Oncologica, Pavia, Italy
S. Toncelli, Consultant on behalf of Centro Nazionale di Adroterapia Oncologica, Novara, Italy
L. Casalegno, Consultant on behalf of Centro Nazionale di Adroterapia Oncologica, Como, Italy
[1] also at Centro Nazionale di Adroterapia Oncologica, Pavia, Italy

Abstract

The Italian National Hadrontherapy Center (CNAO) is a cancer treatment facility that employs a synchrotron to accelerate charged particle beams.

The configuration and support environment of CNAO's control system is responsible for managing the repository, configuring the control system, as well as performing non-real time support operations. Applications in this environment interface with the relational repository, remote file systems, as well as lower level control system components. As part of the technological upgrade of the configuration and support environment, CNAO plans to integrate mobile applications into the control system.

In order to lay the groundwork for the new generation of applications, new communication interfaces had to be designed. To achieve this, a web services approach was taken, with the objective of standardizing access to these resources. In this paper we describe in detail the update of the communication channels. Additionally, the solutions to challenges encountered, such as access management, logging, and interoperability, are presented.

CURRENT CONFIGURATION AND SUPPORT ENVIRONMENT

The current physical architecture of the control system is presented in Figure 1. This figure displays the physical levels of the control system, specifying the hardware equipment and role of each level [1]. Additionally, the diagram presents the data transfer periodicity of the communications between components.

The first layer contains the several types of applications. The largest component present in this level is the collection of WinCC SCADA (Supervisory Control and Data Acquisition) [2] applications. Also present are Virtual Instruments, written in LabVIEW, which provide a graphical interface used by operators to manage subsystems that are currently not integrated into the SCADA. Additionally, the first layer also contains the configuration and support applications with a graphical user interface.

In the second layer, the WinCC SCADA system applications are responsible for obtaining data and alarms from the third layer, as well as delivering these to the first level's SCADA terminals. Operational data from the accelerator is archived at this level in a SCADA database. This layer is also where the repository and repository services reside. The repository is an Oracle DB cluster containing data for running the facility. This data includes the physical characteristics of the accelerator, configuration settings for software applications, accelerator settings for all currently available charged beams, and information to link patient identifiers to their scheduled treatments.

The third layer is responsible for performing data transfer between the second and forth layers through the OPC and OPC-UA protocol. Finally, the fourth layer, the closest to the accelerator equipment, possesses the strictest real-time requirements, and contains hardware and software to manage each subsystem.

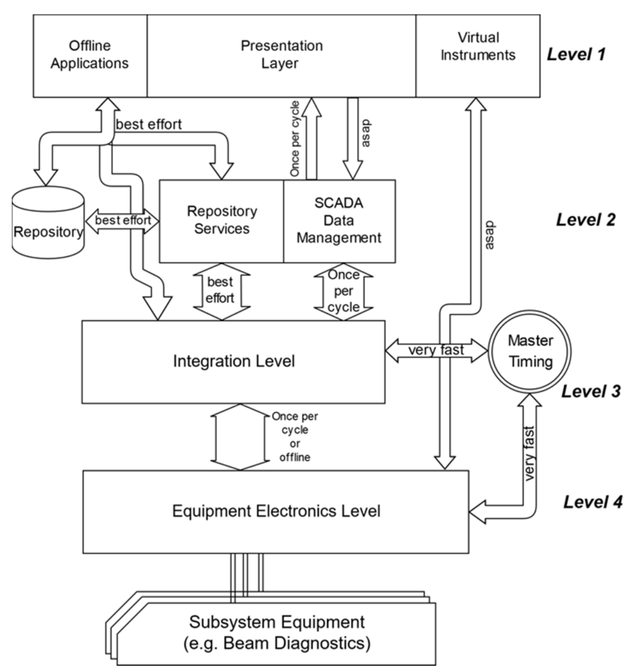


Figure 1: CNAO's Control system, adapted from [1].

The Data transfer frequency between the fourth and third layer is synchronized with events generated by the Timing System. Namely, once per acceleration cycle, the data in the fourth layer is refreshed, and can safely be read during the time period between the two synchronization events. From the third layer upwards, the periodicity

* This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 675265.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA006

SwissFEL UNDULATOR CONTROL SYSTEM

A. Alarcon†, Paul Scherrer Institut, Villigen, Switzerland

Abstract

SwissFEL has successfully commissioned the Aramis beamline, hard x-rays (2 – 12.4 keV), and the Athos line, soft x-rays (200 eV to 2 keV), will start commissioning in 2020. The Aramis undulator line is currently composed of 13 variable-gap in-vacuum undulators. The Athos line will be made of 16 APPLE II type undulators (Advanced Planar Polarized Light Emitter). Both beamlines have each undulator segment on a 5D mover system; they both also have phase shifters and movable quadrupole tables in between segments. PLCs and DeltaTau motor controllers are used to control motion, for I/O interface, and interlocks. EPICS IOCs communicate with the controllers and provide additional logic and some high level functionality. Further higher level functions are provided through python scripts and other high level languages.

INTRODUCTION

Free Electron Lasers, FELs, represent the forefront in high brilliance x-ray light source technology. They are able to produce light twelve orders of magnitude brighter than a synchrotron [1]. Also, as opposed to synchrotrons, FELs provide very short coherent pulses, a short pulse x-ray laser. FELs are used to probe the structures and properties of organic and inorganic materials, but with much better resolution and speed than synchrotrons. SwissFEL is such a machine, built at the Paul Scherrer Institute, PSI, near the town of Villigen, in the canton of Aargau in Switzerland. Construction started in 2013 and commissioning in 2016.

All current FELs are composed of three fundamental parts, a short pulse electron accelerator, a coherent x-ray generating section, and experimental hutches. The x-ray generating section consists of undulators and supporting devices. Undulators are repeating magnet structures built in such a way as to modulate the electron beam in order to generate coherent and amplified X-ray pulses. This process is known as self-amplified spontaneous emission, or SASE. SwissFEL will have two parallel X-ray generating sections, Aramis for hard x-rays (2 – 12.4 KeV), and Athos for soft x-rays (200 eV to 2 keV) [2]. The Aramis line is operational and commissioning for Athos will begin in 2020.

SwissFEL Aramis Undulator Components

The Aramis undulator line is built from two repeating structures, the undulator assembly, and an inter-undulator section. There are currently thirteen undulator assemblies, and 20 inter-undulator sections. The undulator assembly and part of the inter-undulator section are in movers, as alignment of the undulators and optical elements within a few microns is critical for x-ray laser generation [2].

At the heart of the undulator assembly is a variable gap, in vacuum, permanent magnet undulator. The movable gap is actuated using a wedge system mounted on slides with motors outside the vacuum assembly. A large cylindrical vacuum chamber surrounds this part, which is then mounted on a large frame for mobility and stability. Supporting devices on this assembly are temperature sensors, and emergency off switches. The undulator assembly is then mounted onto a 5D, (x, y, pitch, roll, and yaw) cam mover system.

The inter-undulator assembly also has a 2D, x and y, wedge mover system, on which a focusing quadrupole, corrector dipoles, and a beam position monitor are mounted to. Another section of the inter-undulator contains a phase shifter, which is a small variable gap, permanent magnet device used to correct the natural phase delay between the electrons and the x-rays. The last element in this section is a retractable, compressed air actuated, alignment quadrupole

SwissFEL Aramis Basic Undulator Controls

The undulator assembly has two independent Beckhoff PLC controllers mounted onto its frame. One is for the variable gap undulator, the phase shifter, and alignment quadrupole, and the second one is for the 5D mover system (Fig. 1). Each has a touch panel interface for local controls. The PLCs provide motor controls, kinematics, digital and analog I/O, and interlocks.

The mover system for the inter-undulator assembly is controlled by a custom DeltaTau motor controller installed outside of the accelerator tunnel. This controller supports eight motors and encoders, and runs Linux. DC power supplies for the quadrupoles and dipoles are also installed outside the accelerator tunnel, along with vacuum support equipment.

SwissFEL Aramis Undulator High Level Controls

The SwissFEL project decided on using EPICS, Experimental Physics and Industrial Control System, due to experience at the Swiss Light Source, SLS, also at PSI, and for being in used in other FELs like the Linac Coherent Light Source, LCLS, at the SLAC National Accelerator Laboratory in Menlo Park, in the state of California, USA [3]. This implies that all controllers ultimately talk to an IOC, Input Output Controller, running EPICS, in order to interface to the SwissFEL control system.

The undulator-assembly connects to two master IOCs, one for all PLCs which support the variable gap undulator, the phase shifter, and alignment quadrupole, and the other one for all PLCs for the 5D mover system. These

† arturo.alarcon@psi.ch

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA008

LIPAc RFQ CONTROL SYSTEM LESSONS LEARNED

L. Antoniazzi, A. Baldo, M. Giacchini, M. Montis, INFN-LNL, Legnaro, Italy
A. Jokinen, A. Marqueta[1], F4E, Garching, Germany

Abstract

The Linear IFMIF Prototype Accelerator (LIPAc) [1] Radio Frequency Quadrupole (RFQ) will accelerate a 130 mA deuteron beam up to 5 MeV in continuous wave. Proton beam commissioning of RFQ cavity, together with Medium Energy Beam Transport Line (MEBT) and Diagnostics Plate, is now ongoing to characterize the accelerator behavior [2]. The RFQ Local Control System (LCS) was designed following the project guideline. It was partially assembled and verified during the RFQ power test in Italy [3]. The final system configuration was pre-assembled and tested in Europe, after that it was transferred to Japan, where it was installed, commissioned and integrated into LIPAc Central Control System (CCS) between November 2016 and July 2017, when the RFQ Radio Frequency (RF) conditioning started [4]. Now the RFQ LCS has been running for 2 years. During this time, especially in the initial period, the system required several adjustments and modifications to its functionality and interface, together with assistance and instructions to the operation team. This paper will try to collects useful lessons learned coming from this experience.

THE LIPAC RFQ LCS

The RFQ Local Control System (LCS) architecture is designed to optimize the reliability, robustness, availability, safety and performance, minimizing the costs related to its purchase and maintenance. Following this philosophy and the IFMIF EVEDA guidelines, the control sys-tem network is composed by two different kinds of hosts:

- physical hosts for critical control system tasks;
- virtual hosts where no particular functional tasks or hardware is required.

The architecture realizes the 3 layer structure described by the IFMIF-EVEDA guidelines, each layer defines a proper hosts group (equipment directly connected to the apparatus, control devices and Human Machine Interface), while the EPICS framework provides the interface between them (Figure 1). In the final stage LCS is integrated in the LIPAc Central Control System (CCS). The upper layer of the CCS implements all the general services required by a control system architecture and provides them to all the LCSs; EPICS Channel Access protocol is the bridge between the CSS and LCSs for control parameter exchange [4]. The architecture is designed to work either as a standalone environment, as used during the RFQ power test in Legnaro National Laboratories [3,5,6] and as part of the LIPAc control system environment.

The RFQ system is a complex apparatus composed by many kinds of subsystems (radio frequency, vacuum, water cooling, etc.) developed using different hardware solutions. As a consequence every part of this structure must be properly integrated to obtain the desired degree of control.

Following these criteria, the system is de-signed and realized using these assumptions:

- PLC hardware is chosen in tasks where security is the most critical feature;
- VME system is used where the acquisition speed rate is crucial.

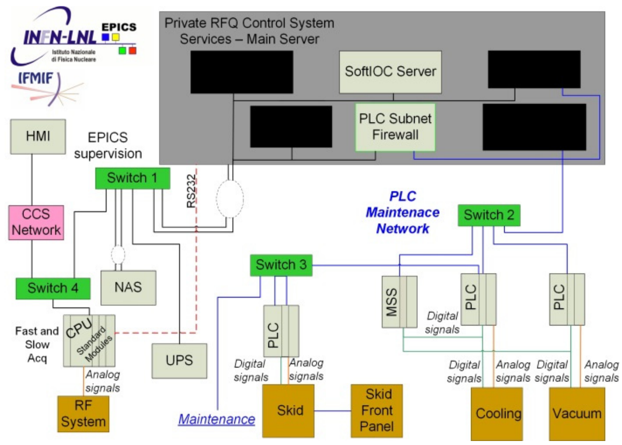


Figure 1: LIPAc RFQ LCS architecture.

The main functionalities of LCS are:

- Fast acquisition system for the RFQ cavity power;
- Vacuum system;
- Cooling system;
- Machine Protection System (MPS).

The fast acquisition is based on VxWorks real time OS which run over a VME architecture. The most important signals about RF power are sampled with a maximum rate of 1MEvents/s (250kEvents/s on 8 channels).

Vacuum, Cooling and MPS functionalities are entirely realized by SIEMENS® S7-300 PLCs and modular safety system (MSS). The integration of these PLCs into the EPICS control network is accomplished by S7plc EPICS driver, developed by SLS (Swiss Light Source) [7].

THE LCS DEVELOPMENT

Power Test

As anticipated, the first version of the RFQ LCS was assembled in 2014 to perform the power test of the RFQ power couplers and cavity [3, 5, 6].

The RFQ power test was executed to validate the most critical RF components of the cavity and, on the other hand, it allowed to test the performances of the main ancillaries as vacuum, cooling/tuning and control system.

[1] now at ITER Organization, St.Paul lez Durance, France

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA009

COMMISSIONING THE CONTROL SYSTEM FOR CRYOMODULE CRYOGENICS DISTRIBUTION SYSTEM IN TEST STAND 2

E. Asensi*, W. Hees, N. Elias, M. Boros, P. Van Velze, J. Fydrych
European Spallation Source ERIC, Lund, Sweden
W. Gaj, IFJ PAN, PL-31342 Krakow, Poland

Abstract

The linear accelerator for the European Spallation Source (ESS) contains 13 cryomodules with 26 double spoke cavities and 30 cryomodules with 120 elliptical cavities [1]. Before installation, these cryomodules will be tested in two dedicated test facilities: Test Stand 2 (TS2) at the ESS site in Lund and the FREIA Laboratory at Uppsala University for the elliptical and spoke cryomodules respectively [2].

In this paper, the authors present the commissioning of the Programmable Logic Controller (PLC) based control system for the cryomodule cryogenic distribution system (CDS) in TS2. Once the cryomodule is connected to the CDS, these circuits will allow the circulation of gas Helium at 4.5 K and liquid Helium at 2 K to cool down the niobium cavities and reach the material superconducting state, as well as to keep a thermal shield with gas Helium at 50 K. Cryogenic valves, temperature and pressure sensors are controlled and monitored to operate this system successfully from a Control Room using dedicated Operator Interfaces (OPI) developed in CS-Studio and following the Experimental Physics and Industrial Control System (EPICS) architecture.

INTRODUCTION

The CDS for TS2 is dedicated to transferring cooling power from the Test and Instruments Cryogenic Plant (TICP) to the ESS elliptical cryomodules under their site acceptance tests in the test stand bunker. The system includes a cryogenic transfer line (CTL), one valve box and four auxiliary process lines (Fig. 1).

The CTL runs from the TICP cold box in the cold box building to the test stand bunker placed in the klystron gallery. The line is a vacuum insulated multichannel line and its vacuum jacket houses four cold process lines (so-called headers), thermal shield, supports and thermal compensation system. The cryoline ends in the test stand valve box, in which four branch process lines connect the headers with the cryomodule cold circuits. Thus the whole system consists of four main and four branch cold process lines [3].

All the CTL cold main process lines at the interface to the cold box are equipped with temperature sensors. These sensors are mainly dedicated for the measurements of the thermal performance of the whole CDS. Other instrumentation required for the commissioning tests, such as flow and pressure transmitters, are located in the cold box and are contracted out separately. The valve box is dedicated for the direct connecting of the tested cryomodules and controlling

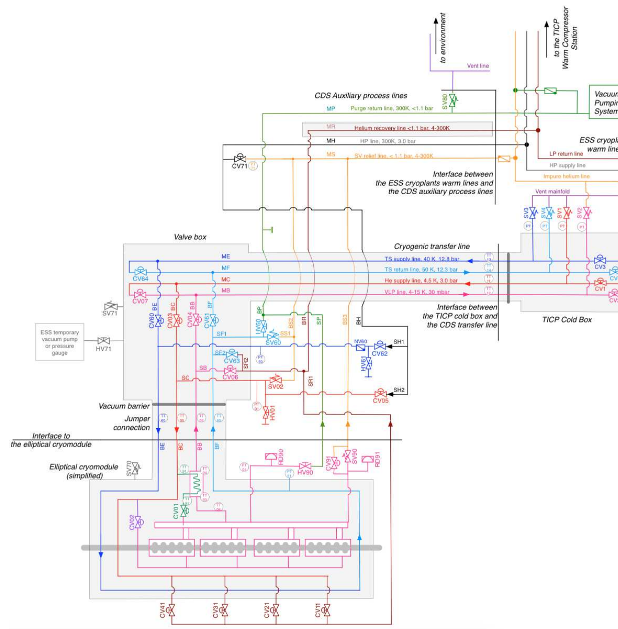


Figure 1: CDS Process and Instrumentation diagram.

the helium flows in different operation modes. For these purposes it is equipped with a branch cryoline (so-called jumper connection) and a set of control valves. The branch cold process lines are equipped with temperature sensors as well as with pressure transmitters. These sensors and transmitters are used to measure the thermodynamic states of the helium flowing in and out of the cryomodule. For this purpose they shall be supported by pressure transmitters installed on the cryomodule circuits. The mentioned temperature sensors are located in the jumper connection close to the interface to the cryomodule [3].

The Control System in TS2 is developed to control the processes in the cryogenic distribution system, valve box and cryomodule in TS2, in order to cool down, maintain and recover the elliptical cryomodule under test from cryogenic conditions, needed to allow the superconductive Radio-Frequency (SRF) to test and operate the cavities in the cryomodule.

CONTROL SYSTEM ARCHITECTURE

The architecture of the control system for TS2 includes sensors and actuators for cryogenics, cooling water, vacuum, motion control and RF (Fig. 2). These last two disciplines are not PLC controlled and therefore out of this scope, all the previous disciplines are connected to signal conditioner elements or power supplies to be controlled by a PLC controller. OPI, archiver and alarm services are developed and

* emilio.asensi@esss.se

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA010

AUTOMATIC BEAM LOSS THRESHOLD SELECTION FOR LHC COLLIMATOR ALIGNMENT

G. Azzopardi*[1,2], G. Valentino[1], B. Salvachua[2], S. Redaelli[2], A. Muscat[1]
[1] University of Malta, Msida, Malta
[2] CERN, Geneva, Switzerland

Abstract

The collimation system used in the Large Hadron Collider at CERN is positioned around the beam with a hierarchy that protects sensitive equipment from unavoidable beam losses. The collimator settings are determined using a beam-based alignment technique, where collimator jaws are moved towards the beam until the beam losses exceed a predefined threshold. This threshold needs to be updated dynamically, corresponding to the changes in the beam losses. The current method for aligning collimators is semi-automated requiring a collimation expert to monitor the loss signals and continuously select and update the threshold accordingly. The human element in this procedure is a major bottleneck for speeding up the alignment. This paper therefore proposes a method to fully automate this threshold selection. A data set was formed from previous alignment campaigns and analysed to define an algorithm that produced results consistent with the user selections. In over 90% of the cases the difference between the two was negligible and the algorithm presented in this study was used for collimator alignments throughout 2018.

INTRODUCTION

The CERN Large Hadron Collider (LHC) is the world's largest particle accelerator. It accelerates and collides two counter-rotating beams, each having a nominal energy of 6.5 TeV during Run 2 [1]. The LHC is susceptible to beam losses from normal and abnormal conditions [2, 3]. Such beam losses are handled by a robust collimation system to safely dispose of the losses in the collimation regions.

The collimation system makes use of 100 collimators in the LHC, able to provide a cleaning efficiency of 99.998% of all halo particles [4]. A collimator is made up of two parallel absorbing blocks, referred to as *left* and *right* jaws, which are positioned symmetrically around the beam.

The position of each collimator's left and right jaw respect a hierarchy, with the settings determined following a beam-based alignment (BBA). This procedure moves collimator jaws separately towards the beam halo, whilst monitoring the measured beam loss signal. A collimator is said to be aligned when both jaws are centred around the beam after touching the beam halo. At present, collimator jaws automatically move towards the beam until the losses exceed a threshold selected by the collimation expert. Once the jaws stop moving, the expert must determine whether the collimator is aligned or not, and update the threshold accordingly. This provides a semi-automatic approach which requires collimation experts to oversee and control the entire alignment campaign.

Collimators are aligned each year during commissioning, to ensure the correct setup for the LHC to achieve nominal operation. They are aligned for different machine states; at injection (450 GeV) 79 collimators are aligned, and at flat top (6.5 TeV) 75 collimators are aligned. The collimator settings are monitored along the year as the beam orbit may shift over time [5], thus potentially requiring the collimators to be realigned. Moreover, different collimator setups are required when machine parameters are changed.

The frequency of collimator alignment campaigns motivated the development of an automatic method, to allow for collimator alignments to be performed more efficiently and at regular intervals. Automating the alignment procedure requires replacing each of the user tasks with dedicated algorithms. This paper proposes to automate one of these tasks by automatically selecting the threshold for stopping the movement of the jaws based on real-time beam loss data. This paper is structured by first presenting research on threshold usage in time-series data and looking at the initial attempt at automatic threshold selection in LHC collimation. This is followed by introducing the newly designed algorithm and finally comparing the results.

BACKGROUND

A Beam Loss Monitoring (BLM) device is associated with each collimator to detect the beam losses generated when halo particles impact the collimator jaws. This BLM detector is positioned outside the beam vacuum, immediately downstream, as shown in Figure 1. Such particle losses are proportional to the amount of beam intercepted by the collimator jaws, which are in units of Gy/s.

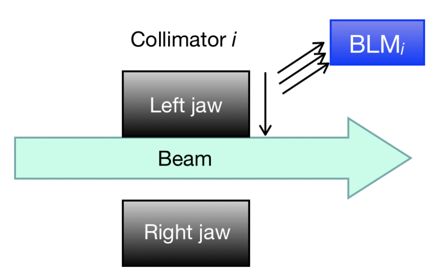


Figure 1: The jaws of collimator *i* around the beam, with its left jaw scraping the beam halo and the showers are detected by the corresponding BLM detector downstream, from [6].

* gabriella.azzopardi@cern.ch

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA011

IMPROVING GESTURE RECOGNITION WITH MACHINE LEARNING: A COMPARISON OF TRADITIONAL MACHINE LEARNING AND DEEP LEARNING

R. Bacher, Deutsches Elektronen-Synchrotron DESY, Hamburg, Germany

Abstract

Meaningful gesturing is important for an intuitive human-machine communication. This paper deals with methods suitable for identifying different finger, hand and head movements using supervised machine learning algorithms. On the one hand it discusses an implementation based on the k-nearest neighbour classification algorithm (traditional machine learning approach). On the other hand it demonstrates the classification potential of a convolutional neural network (deep learning approach). Both methods are capable of distinguishing between fast and slow, short and long, up and down, or right and left linear as well as clockwise and counter-clockwise circular movements. The details of the different methods with respect to recognition accuracy and performance will be presented.

INTRODUCTION

Controlling video games through a gaming console or acting in a virtual or mixed reality environment recognizing arm, head and body motions all lead to popular and intuitive interface features currently in common use. Even in the case of industrial applications, novel interaction technologies are gaining in importance, e.g. to simplify quality assurance of manufacturing processes.

In the field of accelerators, hardware commissioning and maintenance use cases might profit from such novel interaction capabilities. For instance, wearing rough and dirty working gloves during cooling-water maintenance work is not adequate for touch sensitive devices. Interacting via hand or arm gestures might be a better choice.

Today's users of accelerator control applications have developed intuitions based on click-like interactions through a mouse or a touch-sensitive device. Both interfaces provide high reliability, a very accurate pointing capability and standardized user actions normally associated with graphical widgets. Therefore, any new interaction capability such as gesture recognition will only be accepted by the users if it provides comparable or even better handiness, ease-of-use, and reliability to click-like interactions.

This paper discusses methods and their implementations aiming at improving the quality and reliability in recognizing gestures based on different finger, hand and head movements using machine learning algorithms. It compares a traditional machine learning (TML) and a deep learning approach (CNN) including a non-linear regression for extracting the features of the movement and a k-nearest neighbour method for movement classification using memorized training data (traditional machine learning) and a trained convolutional neural network for classification (deep learning).

GESTURE TYPES

The consumer market provides various devices capable of recognizing 2D/3D spatial gestures including hand-gestures, hand- or arm-gestures and 3-axis (yaw, pitch, and roll) head movements (smart glasses). The native or device-specific gestures such as "Closed-Hand", "Fingers-Spread" or "Nodding" can be combined with preceding or following linear movements or rotations including

- Horizontal: left, right
- Vertical: upward, downward
- Diagonal: upward-left, upward-right, downward-left, downward-right
- Circular: clockwise, counter clockwise

In addition, each of these enriched gestures can be performed as a long-or-short and slow-or-fast linear movement or rotation which is projected to a virtual plane in front of the user.

GESTURE RECOGNITION WORKFLOW

The workflow to predict a movement associated with an enriched gesture consists of two distinct phases: a training phase and a real-time prediction phase. It is important that both the training and the real-time prediction phase use the same algorithms.

The input for both phases is continuously recorded position data of the user's input device in Cartesian (X, Y) coordinates resulting from finger, hand, arm or head linear movements or rotations. The continuous stream of sensor data is pre-processed to determine the position of the signal within a certain time window. Optionally the noise floor of the measurement can be reduced or obvious outliers can be removed. If properly centred within the time window, the cleansed data array is fed into the machine learning algorithms.

Traditional Machine Learning

The basis for the traditional approach is a well-engineered mathematical model (Eq. (1)) describing the movements by a set of representative parameters (features).

$$
\begin{aligned}
&\mathbf{t \le t_{start}}: && \mathbf{f(t) = s_{start}} \\
&\mathbf{t_{start} < t < t_{end}}: && \mathbf{f(t) = s_{start} + \left(\left(\frac{s_{end} - s_{start}}{t_{end} - t_{start}}\right) * (t - t_{start})\right)} \quad (1) \\
&\mathbf{t \ge t_{end}}: && \mathbf{f(t) = s_{end}}
\end{aligned}
$$

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA012

INTERRUPTING A STATE MACHINE

K. V. L. Baker, ISIS Neutron and Muon Source, Didcot, UK

Abstract

At the ISIS Pulsed Neutron and Muon Source [1] we interact with a variety of types of beamline systems for controlling the environment of samples under investigation. A state machine is an excellent way of controlling a system which has a finite number of states, a predetermined set of transitions, and known events for initiating a transition. But what happens when you want to interrupt that flow? An excellent example of this kind of system could be a field ramp for a magnet, this will start in a "stable" state, the "ramp to target field" event will occur, and it will transition into a state of "ramping". When the field is at the target value, it returns to a "stable" state. Depending on the ramp rate and difference between the current field and the target field this process could take a long time. If you put the wrong field value in, or something else happens external to the state machine, you may want to pause or abort the system whilst it is running you will want to interrupt the flow through the states. This paper will detail a solution for such an interruptible system within the EPICS [2] framework.

WHAT IS A STATE MACHINE?

A state machine can be defined in an electronics context as "a device which can be in one of a set number of stable conditions depending on its previous condition and on the present values of its inputs" [3]. An event is something that happens, it may be a timeout or value is reached, or it may be that a button is pressed. This event will in turn trigger a transition from one state to another – there is a change in the inputs for the state machine. Most practicable state machines are finite state machines – that is they have a finite number of states and transitions. Infinite state machines are plausible, but impractical, as such the use of the term state machine will typically refer to a finite state machine.

One of the simplest examples of a state machine is something which can be on or off, such as a light. Most light circuits will have a switch of some variety, typically one that is open or closed. If the switch is open the light is off. Pressing the switch to closed is an event which will trigger the turn light on transition, and the light will then settle into the on state. Figure 1 shows a simple state diagram for the on/off state of the light.

The complexity is already apparent, for example: if there is no power then the light cannot be on, if the lamp has broken then it can be off in the on state, if the switch develops a fault then the state might not be alterable from the present one, and so on. The more complex the system, the longer the list of states that can occur.

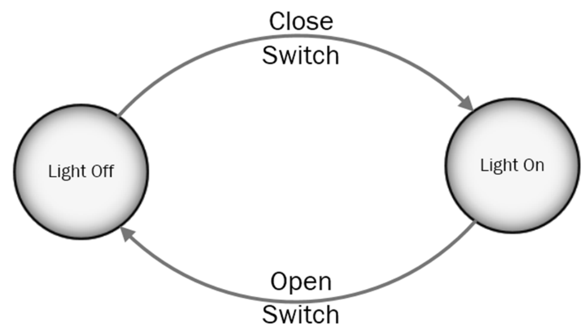


Figure 1: Simple light state diagram.

This electronics concept is also applicable to programming, especially of systems which interact. If, rather than a light, the button was to switch on a large scale lighting system that can only be run at night which requires water cooling. To allow the on state to be achieved the switch must now be closed, the water flow must be of a suitable rate and we need to know that the present time is between sunset and dawn. Whilst this set of criteria can be fed into a circuit and considered at an electronics level, this example is well suited to a software solution. It is straightforward to determine sunset and dawn by looking it up from a trusted online source, and this can then be defined as an appropriate action to consider in the state machine.

WHAT IS AN INTERRUPT ?

Typically an interrupt can be defined as a break or stop of something continuous [4]. This is true in an electronics or computing context as well, where an interrupt is often a signal used to break the flow of code through a processor. This can then be thought of as a specific type of event, one which might alter the flow more rapidly than waiting for a state to complete. An interrupt is in essence an input which is treated in a prioritised fashion.

If we take the complex light situation above, then the clock ticking over so that dawn has occurred will immediately mean a need to switch the light off. But, where is that time being checked?

Each loop of the on state could go to that external source to get the time of dawn and sunset for the current date, and check the present time to see if it is night or not, then trigger a transition to the off state.

A better programming solution would be for the on state to go and get the dawn and sunset times each day and store them in variables to compare, so it doesn't need to keep spending time getting that data as well as checking on everything else.

An ideal programming solution would be to have a separate process collect the dawn and sunset times once a day,

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA014

BUILDING AND PACKAGING EPICS MODULES WITH CONDA

B. Bertrand*, A. Harrisson, ESS, Lund, Sweden

Abstract

Conda is an open source package, dependency and environment management system. It runs on Windows, macOS and Linux and can package and distribute software for any language (Python, R, Ruby, C/C++...). It allows one to build a software in a clean and repeatable way. EPICS is made of many different modules that need to be compiled together. Conda makes it easy to define and track dependencies between EPICS base and the different modules (and their versions). Anaconda's new compilers allow conda to build binaries that can run on any modern linux distribution (x86_64). Not relying on any specific OS packages removes issues that can arise when upgrading the OS. At ESS, conda packages are built using gitlab-ci and pushed to a local channel on our Artifactory server. Using conda makes it easy for the users to install the EPICS modules they want, where they want (locally on a machine, in a docker container for testing...). All dependencies and requirements are handled by conda. Conda environments make it possible to work on different versions on the same machine without any conflict.

INTRODUCTION

Distributing binaries of C/C++ programs is still not an easy task today. Compiling locally and making modules available via an NFS share is a common solution in the EPICS community. Operating system package managers like yum or apt are not new of course but they require one to build different packages for each Linux distribution. In recent years, language specific package managers became more and more popular. There is no new programming language without its own package manager. All modern languages have their own: Python (pip), JavaScript (npm), Rust (cargo)... It's even coming to C/C++ with Conan [1] for example. Conda [2] is another popular solution that is not linked to a specific OS or language. Conda makes it possible to build, distribute and install binary packages with all their dependencies allowing users to concentrate on their task and be more productive.

EPICS AT ESS

e3

The EPICS Environment at ESS named e3 [3] is based on the concept developed by Dirk Zimoch at PSI that allows dynamically loading of EPICS module resources. All the IOCs use the same executable from EPICS base (softIoc) that is not linked with any libraries. Shared libraries and resources are all loaded at runtime by the require [4] module that also performs dependency resolution. An IOC is started by running the *iocsh.bash* script and a startup command script that describes the modules to load. The *iocsh.bash* script is a wrapper that starts a softIoc. For *require* to load the required modules, they have to be in a defined file structure as depicted in Fig. 1.

```
/epics/base-3.15.5/require/3.0.0/siteMods/
├── asyn
│   └── 4.33.0
│       ├── db
│       ├── dbd
│       ├── include
│       └── lib
├── autosave
│   └── 5.9.0
│       ├── bin
│       ├── db
│       ├── dbd
│       ├── include
│       └── lib
```

Figure 1: e3 tree structure.

The current e3 implementation includes some scripts making it easy for developers to build locally the modules they want and has proven to work very well for development.

Development vs Production

With an NFS share for production use, some extra complexity appear compared to local development:

- keeping multiple versions of modules
- OS compatibility

For production use, the tree structure must allow adding new versions of modules while keeping the existing ones. This is easy for modules that nothing depend on, but becomes quickly very difficult to manage when rebuilding dependencies is required. When a new module version is added, all modules that depend on it have to be recompiled to use this new release (due to ABI compatibility). But their own version might not have changed. The notion of build number is thus needed to have different builds of a same version. This is exactly one of the problem that package managers are designed to handle.

Another issue is that the modules can only be used on the OS they were built on. Having modules built for different Linux distributions, in the same tree structure, requires one to modify the EPICS_HOST_ARCH, changing it from linux-x86_64 to something specific to the distribution, and compiling each module on different OS. This is not only true between Linux distribution like Debian and CentOS but even between major versions of a distribution like CentOS 7 and CentOS 8. As we'll see, this is also something conda can solve.

CONDA

Conda is an open-source, cross-platform, binary package manager first released in 2012. It comes from the Python

* benjamin.bertrand@esss.se

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA015

REVERSE ENGINEERING THE AMPLIFIER SLAB TOOL AT THE NATIONAL IGNITION FACILITY

A. Bhasker, R. D. Clark, J. E. Dorham
Lawrence Livermore National Laboratory, Livermore, USA

Abstract

This paper discusses the challenges and steps required to convert a stand-alone legacy Microsoft Access-based application, in the absence of original requirements, to a web-based application with an Oracle back-end and Oracle Application Express/JavaScript/JQuery front-end. The Amplifier Slab Selection Tool (ASL) provides a means to manage and track the amplifier slabs on National Ignition Facility (NIF) beamlines. ASL generates simulations and parameter visualization charts of seated Amplifier Slabs as well as available replacement candidates to help optics designers make beamline configuration decisions. The migration process, undertaken by the NIF Shot Data Systems (SDS) team at Lawrence Livermore National Laboratory (LLNL), included reverse-engineering requirements from an end-user perspective, identifying obsolete requirements, and identifying new requirements due to evolving processes and changing NIF usage patterns.

INTRODUCTION

The National Ignition Facility (NIF) at Lawrence Livermore National Laboratory (LLNL) is composed of 192 laser beamlines, that converge on a hydrogen-filled target the size of a peppercorn to cause nuclear fusion. Each of these 192 beamlines contains two large amplifier sections, the main amplifier and the power amplifier. These amplifiers consist of precisely positioned neodymium-doped phosphate glass slabs and provide 99.99% of NIF's power and energy [1]. There are 3072 slabs in use across all 192 beamlines. These slabs are expected to last through years of daily use but do suffer damage over time. After a decade of operations, and over 2700 shots fired, amplifier slabs are being replaced due to wear and tear at the greatest rate since NIF experimental operations began.

Replacing an amplifier slab on a NIF beamline requires a number of criteria to be satisfied, which calls for interplay between optics designers and component engineers. The Amplifier Slab Selection Tool serves as a single place for all of this – ranging from identifying available candidates to verifying the candidacy of an available amplifier slab for a given position by performing complex optical calculations for a given beam path.

THE LEGACY APPLICATION

The original ASL was developed when NIF was under construction, over a decade ago. The original tool was built using MS Access. Its front-end was comprised of forms and reports for component engineers and optics designers-serving as an optics tracking tool for the former, and a simulation environment for amplifier slab configurations on NIF beamlines for the latter. It saved the beamline configuration of amplifier slabs in an MS Access Database back-end. This standalone MS Access tool was important for getting NIF built. However, it did not evolve with NIF. This was largely because amplifier slabs did not demand significant maintenance for many years, so the standalone tool did not require any enhancements during that time.

THE TARGET APPLICATION

One of the major applications used by NIF Engineers and Scientists is a tool suite called Production Optics Reporting and Tracking (PORT), which has been developed using Oracle Application Express (APEX). Oracle APEX is a web-application development tool for the Oracle databases [2], that supports utilization of JavaScript and JQuery components as well.

The PORT tool covers a wide domain of capabilities – ranging from reporting and analysis to task-scheduling and optics installation, as well as optics and targets tracking. For the project described in this paper, the legacy MS Access Amplifier Slab tool was reverse engineered, revamped and given a new home as a component of PORT.

PORT was implemented in APEX version 5.0 at the time of this undertaking.

METHODOLOGY

The basic methodology used to convert this legacy application was to reverse-engineer, re-design and re-implement the application. The aim was to have a new application that was functionally equivalent to the legacy application but redesigned in a way to allow the team to fully leverage the power and features of Oracle APEX.

Since the low-level requirements needed to be extracted, all the exploratory reverse engineering work was performed in a development environment that was a replica of the production environment. The process began with extraction of the low-level design details from the source code, and then extracting the high-level design information from that. This information collected drove the design and implementation of the new system using the target technology. The conversion of this tool from a legacy MS Access application to an Oracle APEX application involved the following steps (Fig. 1):

Identifying All the Legacy Functionality

The initial goal was to gain a broad understanding of what features this tool provided to users. In the absence of the original requirements, key application users were asked for their input to understand usage patterns for the legacy tool. Steps were also taken to gain a preliminary understanding of how data from other NIF systems, pertaining to

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA017

pyAT, Pytac AND pythonSoftIoc: A PURE PYTHON VIRTUAL ACCELERATOR

W. Rogers*, T. J. R. Nicholls, A. A. Wilson, Diamond Light Source, Oxfordshire, UK

Abstract

Virtual accelerators are used for testing control system software against realistic accelerator simulations. Previous virtual accelerators for synchrotron light sources have used Tracy and Elegant as the simulator, but without Python bindings for accelerator simulations it has been difficult to create a virtual accelerator using Python. With the development of Python Accelerator Toolbox (pyAT), that is now possible. This paper describes the combination of pyAT, Python Toolkit for Accelerator Controls (Pytac) and pythonSoftIoc to create an EPICS-based virtual accelerator for Diamond Light Source.

MOTIVATION

High-level control system software is designed to interact with the control system that is connected to the real hardware. Testing this software can be inconvenient for two main reasons: during design and commissioning the hardware might not yet exist, and during its operational lifetime the hardware is in use most of the time.

A *virtual accelerator* is an application that allows testing a control system by providing the same interface as a subset of the control system that is required for the operation of high-level applications. Although it is possible to provide dummy values for different control system parameters, it is much more useful to combine those parameters with a simulation so that they respond in a physically accurate way to any changes. The software under test then requires no changes in order to run against a virtual accelerator.

Python has a number of advantages for developing a virtual accelerator: it is free and open source, is very widely used in both science and industry, has many useful third-party libraries available, is simple to start using and is capable of building large applications that scale well.

To build this virtual accelerator we needed a number of components:

- a simulation code for synchrotron light sources that can be called from Python
- a Python framework that understands the elements of a particle accelerator
- the ability to convert between engineering units (used in the control system) and physics units (used in simulation codes)
- a server that can hook the Python code into the control system

The following sections describe these components.

* will.rogers@diamond.ac.uk

pyAT

Accelerator Toolbox (AT) [1] is a simulation code for synchrotron light sources developed for use in Matlab. Its numerical engine is based around 'integrators' that calculate the effect of a particle in 6D phase space. For efficiency purposes these integrators were written in C and compiled for use in Matlab.

This design allowed for the same integrators to be compiled for use in Python code. pyAT uses the same numerical engine as AT, with Python classes and functions written to derive accelerator parameters from the engine. It uses the libraries NumPy and SciPy for several numerical utilities. Previous virtual accelerators for synchrotron light sources have used Tracy [2] [3] and Elegant [4] [5] as the simulator, but these were missing Python bindings.

pyAT provides a number of features. Lattice and element types are defined and may be loaded and saved to different file formats. It performs particle tracking and allows calculation of transfer matrices and closed orbit with radiation included or excluded. Other derived parameters that are calculated include linear optics, radiation integrals and detail of the beam envelope. It includes a plotting package that uses Matplotlib to provide various plotting functions.

Testing has shown pyAT to give exactly the same numerical results as AT with a speed comparable to other accelerator simulations [6].

Pytac

Python Toolkit for Accelerator Controls (Pytac) is a Python library designed to enable working with the different parts of accelerators. Each element in an accelerator is represented by an object that may have one or more 'fields' corresponding to physical parameters. The ability to address the different elements of an accelerator by position in the accelerator and element family is often useful in high-level applications and Python scripts.

```
>>> bpm1 = lattice.get_elements('BPM')[0]
>>> bpm1.fields()[pytac.LIVE]
['enabled', 'x', 'y']
>>> bpm1.get_value('x', data_source=pytac.LIVE)
-2.1e-5
```

Pytac allows requesting the live values of the parameters from the accelerator control system. It also allows efficiently requesting the values for an entire family.

```
>>> lattice.get_element_values('BPM', 'x')
[-4.6e-05, 8.2e-05, 7e-05, ...]
```

Many of the ideas in Pytac were inspired by a similar application Matlab Middle Layer (MML) [7].

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-MOPHA019

UPGRADE OF THE CONTROL SYSTEM FOR THE LHC HIGH LEVEL RF

Y. Brischetto, L. Arnaudon, V. Costa, D. C. Glenat, D. Landré, CERN, Geneva, Switzerland

Abstract

The acceleration of particles in CERN's Large Hadron Collider (LHC) is carried out by sixteen superconducting radiofrequency (RF) cavities. Their remote control is taken care of by a complex system which involves heterogeneous equipment and interfaces with a number of different subsystems, such as high voltage power converters, cryogenics, vacuum and access control interlocks. In view of the renovations of the CERN control system planned for the Long Shutdown 2 (LS2), the control software for the RF system recently underwent a complete bottom-up refactoring, in order to dispose of obsolete software and ensure the operation of the system in the long term. The upgraded software has been deployed one year before LS2, and allowed successful operation of the machine. This paper describes the strategy followed in order to commission the system and to guarantee LHC nominal operation after LS2.

INTRODUCTION

Any kind of software needs recurring updates, and the control software for accelerator equipment at CERN does not differ. Taking advantage of the downtime of the accelerators during the Long Shutdown 2 (which started in January 2019 and will last until end of December 2020), it was decided to perform a major update to FESA (Front-End Software Architecture) [1]. Most of the CERN control software applications rely on this framework, therefore they will have to be renovated before the end of LS2.

One year before the start of LS2, our team started the renovations of the software controlling the LHC 400 MHz RF system (ACS), which is based on FESA, and as a result we were able to validate the new software during one year of LHC operation.

OVERVIEW OF THE ACS RF SYSTEM

Sixteen superconducting radiofrequency cavities guarantee the acceleration of particles in LHC [2]. The 400 MHz cavities are installed in groups of four inside cryomodules, and two cryomodules are installed on each beam line. Each cavity is powered by a 300 kW klystron and shares a 58 kV voltage power converter and a high voltage equipment bunker with the other cavities installed in the same cryomodule.

From an RF point of view, each cavity is driven by one klystron amplification chain, thus the concept of *line* was defined to identify the full chain of devices controlling one specific klystron and the cavity connected to it. Moreover, since a group of four klystrons shares a common power converter and high-voltage equipment bunker, and drives the cavities installed inside one cryomodule, the concept of *module* was also introduced, to address the control of the power converter and of the HV equipment related to each cryomodule (as illustrated in Fig. 1). Each *line* and each *module* is controlled by one dedicated programmable logic controller (PLC), therefore the overall control system is composed of sixteen *line* PLCs and four *module* PLCs.

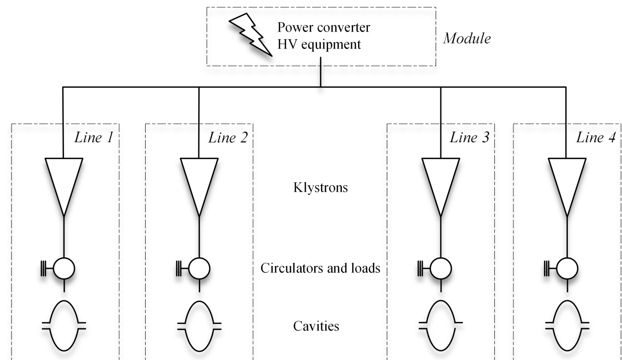


Figure 1: Logical view of one ACS module and four lines.

Along with them, one additional PLC is charged with the control of *services* as the fast interlock module for the beam dump and environmental sensors monitoring the LHC Faraday cages, which are installed underground and host some of the low level RF controls.

CONTROL BEFORE RENOVATIONS

The remote control of equipment is generally integrated into CERN's control system by using FESA, a CERN-made C++ framework that lets developers create control software in a common manner, regardless of the equipment to be controlled. Until some years ago, FESA allowed the integration of PLCs, offering developers the opportunity of only having to define communication interfaces, relieving them from the development of the code necessary to handle data transfer with the controllers.

The control software for the ACS RF system that was in use before the renovations presented in this paper was composed of six different FESA classes, three dedicated to the communication with the PLCs, and three others handling the exchange of status and control data between the lines, modules and the additional systems. High-level control interfaces were developed using standard tools offered by CERN's BE-CO (Controls group of the Beams department), whereas most of the expert interfaces were implemented through LabVIEW panels. The variety of the control software was obviously a limiting factor, which implied that users had to use different applications to control different parts of the system, something that could definitely be improved.

SOFTWARE UPGRADES

In light of this situation, along with the advent of the major update of the FESA framework, and important changes

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA022

IMPLEMENTATION OF ISO 50001 ENERGY MANAGEMENT SYSTEM WITH THE ADVANTAGE OF ARCHIVE VIEWER IN NSRRC

C. S. Chen†, W. S. Chan, Y. Y. Cheng, Y. F. Chiu, Y. C. Chung, K. C. Kuo, M. T. Lee, Y. C. Lin, C. Y. Liu, Z. D. Tsai, NSRRC, Hsinchu, Taiwan

Abstract

Due to the limited energy resources in Taiwan, energy conservation is always a big issue for everyone who lives in this country. According to the data from the related departments, nearly 98% of energy is imported from abroad for more than a decade. Despite the strong dependency on foreign fuel imports, the energy subsidy policy leads to a relatively low cost of energy for end users, while it is not reasonable. In order to resolve the energy resource shortage and pursue a more efficient energy use, the implementation of ISO 50001 energy management system (EnMS) is activated with the advantage of the Archive Viewer in NSRRC this year. The energy management system will build up an overall energy usage model and a certain number of energy performance indicators to help us achieve efficient energy usage.

INTRODUCTION

According to the data released from ISO in 2017, more than 22,870 enterprises have been certified to meet the requirements of ISO 50001 globally. Comparing 2017 to 2016, the increment of certificated enterprises is nearly 13%. There are 290 enterprises certified in 2018 in Taiwan. It shows the determination of Taiwan in energy conservation. Since 2011, the Taiwan Green Productivity Foundation has been entrusted to execute the consulting project in deployment of ISO 50001. This project has successfully assisted 125 enterprises in developing the energy management systems and earned the third-party certifications from 2011 to 2018 in Taiwan. This project sums up to carry out 1,131 energy conservation treatments and saves 120 million kWh of electrical energy in total over the past eight years.

Figure 1 shows the trends of power usage and gross domestic product per capita in Taiwan from 1995 to 2018. Taiwan's factories are an indispensable part of the global supply chain for high-tech industry. Over the last two decades, the total energy consumption grew up obviously along with the GDP in Taiwan except for the drop in 2009 due to the great earthquake of September 21. It is always a real dilemma when it comes to economic development and environmental protection together. Due to the limited natural resources in Taiwan, the development plan in this country must be sustainable and environment-friendly. It is the main intention to deploy the energy management system to NSRRC.

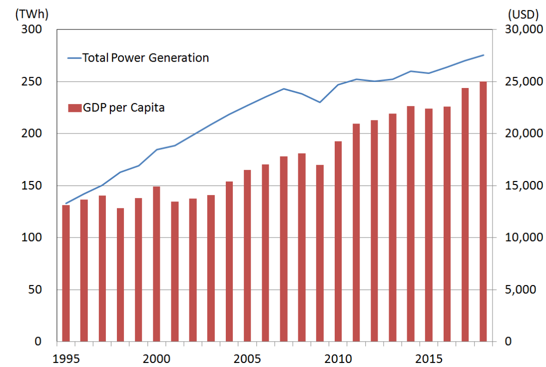


Figure 1: Total power generation and GDP in Taiwan from 1995 to 2018.

ISO 50001:2018

ISO 50001 is an international standard which integrate the energy performance system and the institutionalized system. It provides enterprises a completed energy management system to fulfil energy saving affairs. Its first edition was released by International Organization for Standards (ISO) in 2011, and revised to the latest version in 2018. It combines the viewpoint from stockholder and the concept of risk management to operate the PDCA management cycles, which stands for Plan-Do-Check-Action, a repetitive four-stage model for continuous improvements shown as Fig. 2.

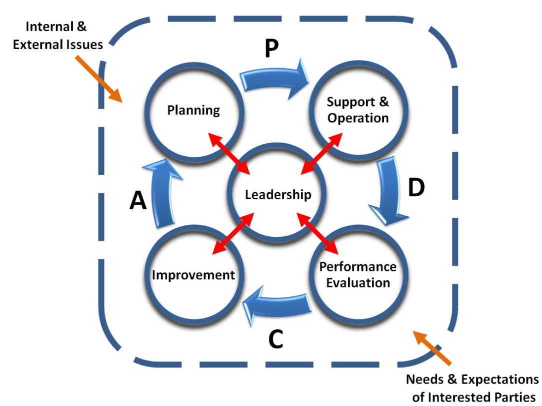


Figure 2: PDCA cycles in ISO 50001:2018 energy management system.

The latest version of standards labelled as ISO 50001:2018 are mostly the same with the former. The biggest difference between ISO 50001:2018 and the former version is laying a greater emphasis on the role of high level structure, which implies the stronger importance of leadership in the operation of PDCA cycles. Furthermore, the clarifications on the content of energy

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA023

APPLICATIONS OF AN EPICS EMBEDDED AND CREDIT-CARD SIZED WAVEFORM ACQUISITION

Y. S. Cheng†, D. Lee, C. Y. Liao, C. Y. Wu, K. H. Hu, K. T. Hsu
National Synchrotron Radiation Research Center, Hsinchu, Taiwan

Abstract

To eliminate long distance cabling for improving signal quality, the remote waveform access supports have been developed for the TPS (Taiwan Photon Source) and TLS (Taiwan Light Source) control systems for routine operation. The previous mechanism was that a dedicated EPICS IOC has been used to communicate with the present Ethernet-based oscilloscopes to acquire each waveform data. To obtain higher reliability operation and low power consumption, the FPGA and SoC (System-on-Chip) based waveform acquisition which embedded an EPICS IOC has been adopted to capture the waveform signals and process to the EPICS PVs (Process Variables). According to specific purposes use, the different graphical applications have been designed and integrated into the existing operation interfaces. These are convenient to observe waveform status and to analyse the caught data on the control consoles. The efforts are described at this paper.

INTRODUCTION

TPS is a new and highly bright synchrotron light source constructed at National Synchrotron Radiation Research Center (NSRRC) at Taiwan. It consists of a 150 MeV electron linear accelerator, a booster synchrotron, a 3 GeV storage ring and experimental beam lines. Civil construction began in the first quarter of 2010 and was completed in the first half of 2013. Installation and integration of the accelerator system began later in 2013. The control system environment was ready by mid of 2014 to support commissioning and final subsystem integration without the beam. Commissioning with the beam was successful in December 2014. User service with 400 mA top-up routine operation was kicked off since 2016.

TLS is a third generation of synchrotron light source which built at the NSRRC site, and it has been operated since 1993. The TLS consists of a 50 MeV electron linear accelerator, a booster synchrotron, and a 1.5 GeV storage ring with 360 mA top-up injection mode. The TLS Control system is a proprietary design [1]. It consists of console level workstations and VME based intelligent local controller (ILC) to interface with subsystems. Hardware and software on console level workstation change several times due to evolution of fast evolution of computer technology.

EPICS (Experimental Physics and Industrial Control System) were chosen as control system framework for the TPS and had been integrated and commissioned [2]. On the other hand, in order to adopt newer technology and re-use expertise of manpower, the upgrade and maintenance for TLS control system have been decided to employ the EPICS as its framework. Thus new installed and rejuvenated sub-systems have operated at the EPICS control environment. Mixed existing TLS control system and EPICS framework were proofed without difficult problems.

Many waveforms in synchrotron light sources are necessary to monitor during routine operation, and especially include current waveforms of pulsed magnet power supplies (septa/kickers), waveforms of klystron current and voltage, waveforms of LINAC RF power, etc. Moreover remote waveform access supports have been widely utilized to eliminate long distance cabling for improving signal quality. Preliminary solution of remote waveform access support is that a dedicated EPICS IOC (Input Output Controller) is built to interface with Ethernet-compliant oscilloscopes for capturing and publishing real-time waveform data [3].

Recently, an advanced solution of remote waveform access support is adopting a credit-card sized waveform acquisition. The major benefits of this credit-card sized waveform acquisition are EPICS embedded, better resolution, better stability, and lower power consumption. The efforts of applying a new credit-card sized waveform acquisition on the TPS and TLS control systems are summarized in the following paragraphs.

CREDIT-CARD SIZED WAVEFORM ACQUISITION

A new credit-card sized waveform acquisition has been developed to be as remote waveform access support for observing necessary waveforms during routine operation. This waveform acquisition module is a FPGA-based (Field Programmable Gate Array) hardware architecture, named "Red Pitaya" [4], which is an open-source hardware formed into a credit-card sized layout. It equips one dual-core processor, one editable FPGA-based SoC, ADC (Analog-to-Digital Converter) of two-channel-input, external trigger functionality, DI/DO (Digital Input Output) pins, one micro-SD storage, and one gigabit Ethernet port. The main specification of hardware components is shown as Table 1. This acquisition module has 7.5 Watt power consumption, and is much lower than power consumption of traditional oscilloscope. This acquisition module internally supports Linux operation system which installed into micro-SD card for setting up related software packages, compiled FPGA codes and application programs. According to the test of real signal inputs, this waveform acquisition module can be equivalently employed to replace several traditional oscilloscopes for being long-term waveform observation.

† E-mail: cheng.ys@nsrrc.org.tw

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA026

DEVELOPMENT OF AN ONLINE DIAGNOSTIC TOOLKIT FOR THE UPC CONTROL SYSTEM

H. Z. Chen, Y. S. Cheng, K. H. Hu, K. T. Hsu, C. Y. Liao, C. Y. Wu
National Synchrotron Radiation Research Center, Hsinchu, Taiwan

Abstract

Most IOC (Input Output Controller) platforms and servers at the TPS control system have been connected to uninterruptible power supplies (UPS) to prevent short downtime of the mains electricity. To accomplish higher availability, it is necessary to maintain batteries and circuits for the UPS system periodically. Thus, an online diagnostic toolkit had to be developed to monitor the status of the UPS system and to notify which abnormal components should be replaced. One dedicated EPICS IOC has been implemented to communicate with each UPS device via SNMP. The PV states of the UPS system are published and archived and specific graphical applications are designed to show the existing control environment via EPICS CA (Channel Access). This paper reports the development of an online diagnostic toolkit for the UPS System.

INTRODUCTION

The Taiwan photon source [1] (TPS) is a highly bright synchrotron light source constructed at the National Synchrotron Radiation Research Center (NSRRC). It consists of a 150-MeV electron linear accelerator, a booster synchrotron, a 3-GeV storage ring and experimental beam lines. The TPS control system [2] is based on the Experimental Physics and Industrial Control System (EPICS) [3] framework, which is a set of open source software tools, libraries and applications developed collaboratively and used to create distributed soft real-time control systems for scientific instruments. There are many devices in the TPS control system that are compatible with SNMP, such as: CompactPCI (cPCI) crates, network switches, UPSs etc.

Establishing a tool to automatically diagnose the device condition is desirable because the device number is large and they are distributed in different CIAs. We already developed diagnostic tools for Compact PCI creates. Next we chose to monitor the UPSs and create tools to diagnose their condition.

Most of the IOC platforms and servers at the TPS control system have been connected to the UPS system to prevent short downtime of mains electricity. The UPSs are installed inside the Control Instrumentation Areas (CIA) which are distributed along the outside wall of the machine tunnel.

To accomplish higher availability, it is necessary to maintain batteries and circuits for the UPS system. Thus, an online diagnostic toolkit had to be developed to monitor the status of the UPS system.

The EPCIS IOC can read the UPS status and use the diagnostic toolkit to monitor their condition. When abnormal conditions occur, such as a UPS working abnormally, overheating, too high output load in percent of rated capacity, battery working abnormal and UPS battery charge too low. The diagnostic toolkit will show an alarm message on the GUI (Graphical User Interface) and send an alarm notification to the responsible personnel via E-mail.

SNMP DEVICE SUPPORT FOR EPICS

SNMP

The SNMP (Simple Network Management Protocol) is a component of the Internet Protocol Suite as defined by the Internet Engineering Task Force (IETF) and is a standard Internet protocol for managing and organizing information of Ethernet-based devices. It consists of the following three components, a managed device, an agent and a Network Management System (NMS). The NMS is a software installed on the management side to query the agent for information about the managed device, which is a node in the network and implements a SNMP interface to collect and store management information and make this information available to NMS using SNMP. The agent is the software installed on the managed device and allows it to collect management information from the managed device database and makes it available to the NMS.

EPICS SNMP Device Support Module

The devSNMP [4] uses the net-snmp library for access to SNMP-based devices. Net-snmp also includes a number of useful utilities such as snmpget and snmpset for reading and writing SNMP variables, and snmpwalk for listing which variables a host makes available. The devSNMP provides EPICS device support for hardware devices that communicate via SNMP. By using devSNMP, the EPICS IOC can query data from devices via SNMP, then store data in the EPICS database for PV channel access.

Figure 1 shows the system structure of an EPICS integrating an SNMP with a UPS in the TPS. The EPICS IOC retrieves information via SNMP from the UPS and the information will be stored in the EPICS database for PV channel access. The GUI can show UPS information via channel access and the diagnostic toolkit can use the data to diagnose the condition of the UPS.

DIAGNOSTIC TOOLKIT FOR UPS SYSTEM

The TPS UPS include a Network Management Card (NMC) that can receive the status information of the UPS and can send commands to control the UPS. The user can manage the UPS with the NMC via web browser or via a network management software which supports the SNMP protocol.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA028

HIGH ENERGY PHOTON SOURCE CONTROL SYSTEM DESIGN*

P. Chu†, D. Jin, G. Lei, G. Li, C.H. Wang, G. Xu, L.X. Zhu, Institute of High Energy Physics, Chinese Academy of Sciences, Beijing, China

Abstract

A 6-GeV high energy synchrotron radiation light source is being built near Beijing, China. The accelerator part contains a linac, a booster and a 1360-m circumference storage ring, and fourteen production beamlines for phase one. The control systems are EPICS based with integrated application and data platforms for the accelerators and beamlines. The number of devices and the complexity level of operation for such a machine is extremely high, therefore, a modern system design is vital for efficient operation of the machine. This paper reports the design, preliminary development and planned near-future work, especially the databases for quality assurance and application software platforms for high level applications.

INTRODUCTION

An ultra-low emittance and high brightness 4th generation synchrotron light source, the High Energy Photon Source (HEPS) designed by the Institute of High Energy Physics (IHEP), has started its construction since June 2019. The main parameters for HEPS are listed in Table 1 which contains many challenging goals. It is necessary to have accurate installation, state-of-art equipment and high precision controls with high reliability. The control systems and related computing facilities are extremely important for the HEPS which includes not only traditional control architecture design but also quality control for the project. Also, the HEPS control systems support not only the accelerator but also 14 beamlines which will be constructed at the same time. To build such a complex accelerator-based user facility, it is necessary to have an overall complete design for the control systems.

Table 1: HEPS Main Parameters

Main Param.	Value	Unit
Top beam energy	6	GeV
Main Ring circumference	1360.4	m
Emittance	<60 (<40 with anti-bend)	pm-rad
Beam current	200	mA
Brightness	$>10^{22}$	$Phs/s/mm^2/mrad/0.1\%BW$
Injection	Top-up	
Bunch structure	680 (high-brightness mode), 63 (timing mode)	

* Work supported by the National Development and Reform Commission, City of Beijing, and the Chinese Academy of Sciences.
† chuzm@ihep.ac.cn

The HEPS control system which considers of data for future machine learning (ML) capability is designed. Accelerator and beamline controls are coordinated together for the design. Quality control tools are under development at this early stage of the project. Also, due to tight schedule, a test bench is essential for mimicking online environment and perform parallel test work to save overall construction time.

The basic design principles for HEPS Control Systems are listed below:

- Top-down architecture design: understanding the big picture
- Distributed control systems
- Integrated development tools (GUI code editors, repository management…) for higher software quality
- Choosing advanced yet matured technologies
- Using industrial standards, choosing commercially available products first for lowering costs
- Considering expandability at design, balancing the price and performance while satisfying physics requirements
- Collaborating with other accelerator projects
- Possible commercialization for R&D results

DATABASES

Accelerator generated data, from design to operation, should be captured and saved systematically as much as possible. Furthermore, applications to utilize the saved data should be developed as well. However, due to the large data-base scale and tremendous amount of work, it is necessary to divide the entire database into many nearly independent sub-database modules and connect them via API (Application Programming Interface) or services. Optionally, one can join some sub-databases together with minor modifications in the schemas. This way the database module development work can be shared independently by many institutes and also avoid overwhelming complexity for a single monolithic database.

Based on IRMIS [1] which is a good overall database system for accelerators, as listed in Table 2, there are 17 sub-database modules identified. At this stage of the project, i.e. the design and early implementation phase, databases such as Parameter List, Naming Convention, and Magnet have been developed to suit the project's current needs. In addition, colleagues from another IHEP facility, the China Spallation Neutron Source (CSNS) is collaborating with the HEPS team to develop a Log-book and Issue Tracking database and application for CSNS's early operation need. Besides the four database modules currently under development, a few others like Accelerator Model/Lattice, Physics Data and Machine State, and Work

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA029

FORS-UP: AN UPGRADE OF THE FORS2 INSTRUMENT @ ESO VLT

R. Cirami †, V. Baldini, I. Coretti, P. Di Marcantonio, INAF-OATs, Trieste, Italy
H. M. J. Boffin, F. Derie, A. Manescau, R. Siebenmorgen, ESO, Garching bei München, Germany

Abstract

The FORS Upgrade project (FORS-Up), financed by the European Southern Observatory (ESO), aims at upgrading the FORS2 instrument currently installed on the UT1 telescope of the ESO Very Large Telescope in Chile. FORS2 is an optical instrument that can be operated in different modes (imaging, polarimetry, long-slit and multi-object spectroscopy). Due to its versatility, the ESO Scientific Technical Committee has identified FORS2 as a highly demanded workhorse among the VLT instruments that shall remain operative for the next 15 years. The main goals of the FORS-Up project are the replacement of the FORS2 scientific detector and the upgrade of the instrument control software and electronics. The project is conceived as "fast track" so that FORS2 is upgraded to the VLT for 2022. This paper focuses on the outcomes of the FORS-Up Phase A, ended in February 2019, and carried out as a collaboration between ESO and INAF – Astronomical Observatory of Trieste (INAF – OATs), this latter in charge of the feasibility study of the upgrade of the control software and electronics with the latest VLT standard technologies (among them the use of the PLCs and of the latest features of the VLT Control Software).

INTRODUCTION AND UPGRADE MOTIVATION

FORS2, acronym for FOcal Reducer/low dispersion Spectrograph, is a multimode (imaging, polarimetry, long slit and multi-object spectroscopy) optical instrument mounted on the Antu Unit Telescope (UT) Cassegrain focus of ESO Very Large Telescope (VLT) located on Cerro Paranal in the Atacama desert in Chile. Originally, two twin instruments have been built and installed, named FORS1 and FORS2, but FORS1, after roughly ten years of successful operations, has been decommissioned to make way for the second generation of VLT instrumentation.

FORS2 entered regular science operations on April 2000 and has been observing ever since without significant interruptions. It is one of the most successful instruments in Paranal and led, with its twin FORS1 now decommissioned, to more than 2600 refereed publications (as of 01.09.2019), of which almost 1600 are from FORS2 alone. As a multi-mode instrument it has contributed to a broad range of science topics like spectroscopic study of the GOODS-South field and the Chandra Deep Field-South, Ly-alpha emitters in the early universe, spectro-polarimetry of massive stars, photometry studies of young stellar regions, astrometric studies of brown dwarfs and transmission spectroscopy of exoplanets and many others [1].

FORS2 covers a wide wavelength range spanning from 330 nm to 1100 nm, with a very great sensitivity: the transmission is above 60% over 360-1100 nm and reaches almost 80% around 440 nm. It is equipped with a mosaic of two blue-optimized 2k x 4k detectors, which can be exchanged with a red-optimized detector mosaic. Due to its versatility, the ESO Scientific Technical Committee has identified FORS2 as a highly demanded workhorse among the VLT instruments that shall remain operative for the next 15 years. However, the current science done with FORS2 differs from what was initially foreseen. Many current observing programmes, photon-noise limited, use relatively short exposure times, and therefore, beside efficiency, the read-out noise of the CCD become an issue. The VLT Instrument Operation Team, since years, is requesting therefore the upgrade of FORS2 with a 4k x 4k broad band detector that shall improve the operations of the instrument, eliminating also the need for the exchange of the red or blue detector systems on the instrument (currently only one can be installed at a time).

Moreover, both software and electronics controlling the instrument have been developed at the end of the '90s and several control parts are obsolete, not supported anymore by vendors and not "VLTSW compliant", i.e. they do not follow the standards imposed by ESO for the instruments currently installed at the ESO VLT.

These considerations led to the FORS-Up project, financed by the European Southern Observatory. The main goals of the project, beside the upgrade of the FORS2 scientific detector (and of some instrument optics), is the upgrade of the instrument control software and electronics [2], which is the main topic of this paper.

INSTRUMENT DESCRIPTION

FORS2 Subsystems

FORS2 physically consists of four parts: a top section (which includes the two internal calibration units), a collimator section, a filter/camera section and the external calibration unit.

- Top section. It contains the focal plane equipment. The motorized functions are an entrance shutter, a MOS (Multi-Object Spectroscopy) unit with 19 movable slits each composed by two blades that can be moved individually and simultaneously, a longslit mask unit with 9 slits and a mask exchange unit for MOS spectroscopy (MXU) consisting of a storage magazine holding up to 10 masks.
- Internal calibration units. They are equipped with the following arc lamps: Ne (2x), Ar (2x), He (1x) and HgCd (1x).

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA030

AN UPGRADE OF THE HARPS-N SPECTROGRAPH AUTOGUIDER AT TNG

R. Cirami †, I. Coretti, P. Di Marcantonio, INAF - OATs, Trieste, Italy
F. Alesina, N. Buchschacher, F. Pepe, OGE, Versoix, Switzerland

Abstract

HARPS-N is a high-precision radial-velocity spectrograph installed on the INAF – TNG in the island of La Palma, Canary Islands. The HARPS-N project is a collaboration among several institutes lead by the Astronomical Observatory of the University of Geneva. The HARPS-N control software is composed by the Sequencer, which coordinates the scientific observations and by a series of modules implemented in LabVIEW for the control of the instrument front end, calibration unit and autoguider. The autoguider is the subsystem in charge of maintaining the target centered on the spectrograph fiber. It acquires target images at high frequency with a technical CCD and with the help of dedicated algorithms keeps the target centered on the fiber through a piezo tip-tilt stage. Exploiting the expertise acquired with the autoguiding system of the ESPRESSO spectrograph installed at the ESO Very Large Telescope (VLT), a collaboration has been setup between the HARPS-N Consortium and the INAF - Astronomical Observatory of Trieste (INAF – OATs) for the design and implementation of a new autoguider for HARPS-N. This paper describes the design, implementation and installation phases of the new autoguider system.

INTRODUCTION

HARPS-N (High Accuracy Radial velocity Planet Searcher - North) is an echelle spectrograph installed on the INAF – Telescopio Nazionale Galileo (INAF - TNG) in the island of La Palma, Canary Islands. It covers the wavelength range between 383 to 693 nm, with a spectral resolution R=115000. The instrument allows the measurement of radial velocities with the highest accuracy currently available in the north hemisphere and is designed to avoid spectral drift due to temperature and air pressure variations thanks to a very accurate control of pressure and temperature. HARPS-N is fiber-fed by the Nasmyth B Focus of the 3.6 INAF - TNG telescope through a Front End Unit (FEU). The two HARPS fibres (which inject the object and sky or a calibration lamp, respectively) have an aperture on the sky of 1"; this produces a resolving power of 115,000 in the spectrograph. Both fibres are equipped with an image scrambler to provide a uniform spectrograph pupil illumination, independent of pointing decentering.

The main scientific rationale of HARPS-N is the characterization and discovery of terrestrial planets by combining transits and Doppler measurements.

The HARPS-NpProject is a collaboration between the Astronomical Observatory of the Geneva University (lead), the Center for Astrophysics (CfA) in Cambridge, the Universities of St. Andrews and Edinburgh, the Queens University of Belfast, and the INAF – TNG Observatory [1].

HARPS-N Autoguider

HARPS-N current auto-guiding system (AG) has been developed in 2012 using the LabVIEW programming language. This choice has been driven by the requirements of controlling several devices in parallel:

- technical CCD (camera)
- piezo controller for the tip-tilt mirror
- 3-axis motion controller for the neutral density filters

The role of the autoguider is to correctly center the star on the fiber during the acquisition phase and to keep the star on the fiber, during the observation phase, by continuously moving the piezo tip-tilt mirror using small computed corrections in addition to the telescope tracking system. To do this, the software reads frames from the guiding camera at the highest possible frequency, computes the barycenter of the star and sends the corresponding corrections to the piezo tip-tilt mirror. The system also provides an integrated image algorithm for the calculation of the fiber hole center, improving the quality of the guiding system along the entire scientific exposure.

After few years of operations, the maintenance of the LabVIEW-based autoguider software proved to be difficult, due to strong dependencies both inside the LabVIEW VIs (Virtual Instruments) and between the AG and the Local Control Unit - LCU (Software which control the calibration unit of HARPS-N). The system startup was also very complicated due to several initialization phases, which slowed down the system considerably. For these reasons it was decided to re-design the entire software from scratch, trying to simplify it based on the experience gathered in several years of operations. Exploiting the expertise of the INAF – OATs, responsible of the design and the implementation of the autoguider system of the ESPRESSO spectrograph installed at the ESO VLT on Cerro Paranal in Chile [2], a collaboration has been setup between the HARPS-N Consortium and INAF – OATs for the design and implementation of a new autoguider for HARPS-N.

HARPS-N AUTOGUIDER DESIGN AND IMPLEMENTATION

The main goal of the HARPS-N autoguider is to maintain the target centered on the fiber hole during the whole scientific exposure. It has also the responsibility to provide the acquisition image to the Sequencer (the software in charge of coordinating the scientific exposure) during the acquisition phase to properly center the object on the fiber hole and to calculate the corrections for offloading the telescope in case of telescope drift/bad tracking.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA031

SOFTWARE AND HARDWARE DESIGN FOR CONTROLS INFRASTRUCTURE AT SIRIUS LIGHT SOURCE

J. G. R. S. Franco†, P. H. Nallin, E. P. Coelho, C. F. Carneiro, R. C. Ito, R. W. Polli, V. S. Pereira, A. R. D. Rodrigues
Brazilian Synchrotron Light Laboratory, Campinas, Brazil

Abstract

Sirius is a 3-GeV synchrotron light source under construction in Brazil. Assembly of its accelerators began on March 2018, when the first parts of the linear accelerator were taken out of their boxes and installed. The booster synchrotron installation has already been completed and its subsystems are currently under commissioning, while assembly of storage ring components takes place in parallel. The Controls System of Sirius accelerators, based on EPICS, plays an important role in the machine commissioning and installations and improvements have been continuously achieved. This work describes all the IT infrastructure underlying the controls system, hardware developments, software architecture and support applications. Future plans are also presented.

INTRODUCTION

Sirius accelerator, the new 4th generation Brazilian synchrotron light source, has been under construction since 2014, along with the full operation of the current facility, UVX. Engineering assemblies and installation, which include controls subsystems, have started in early 2018.

Sirius Controls System conceptual design aimed to be scalable, distributed and easy to maintain [1]. Based on these principles and on open-source solutions, hardware, infrastructure and software concepts have been designed, implemented and installed on Sirius site. Thus, the integration of a large variety of equipment has been achieved.

SYSTEM ARCHITECTURE

Network Topology

Based on EPICS framework, controls systems topology is composed of two Supermicro servers (with dual Intel Xeon E5-2695 processors, 8x 64GB DDR4, 16x 8TB HDD for data storage and 2x 480GB SSD for operational system) and two core switches with 48 SFP+ ports each (Figure 1). Lower level switches, having 4 10GBase-SR ports, 24 1000Base-T ports and PoE+ driver capability, are connected to the main ones in a star topology. Both switch models have redundant power supply, fans and management modules. A ring interconnection will be available in the future in order to have network redundancy.

Figure 1: Core switch Aruba 5412R.

Lower level switches are placed in 24 different areas, such as Sectors Instrumentation Areas, RF Room, Transport Line Area, LINAC Area and Power Supplies Room, and port numbers are extended with 10/100/1000Base-T equipment if needed.

HARDWARE SOLUTIONS

Controls system distributed nodes are based on the open-hardware single board computer (SBC) Beaglebone Black [2], running an embedded Debian distribution operational system.

The inexpensive embedded system has been successfully in operation in LNLS current facility (UVX) since 2016 [3], as a replacement for outdated SBCs and also as a test bench for Sirius nodes.

Main hardware projects aim the extended use of Beaglebones.

SERIALxxCON

Designed to be the main Beaglebone Black baseboard, with over two hundred units distributed on controls cabinets (Figure 2), they are used to interface with several equipment through either RS-485 or RS-232 serial communication. Standard FTDI module is available and also a high-performance serial interface, developed with Beaglebone's embedded Real-Time Processors (PRUs) [4], reaching pre-defined baudrates up to 15 Mbps. This is a requirement for power supplies communication, once the amount of data to be transferred between systems is elevated.

The board also have inputs for both optical and electrical timing system for synchronized operations using high-performance interface, which can be set up in three different main modes:

- Single sequence curve execution, sending setpoint commands;
- Continuous curve execution;
- Single broadcast command.

† guilherme.franco@lnls.br

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA032

BIG DATA ARCHITECTURES FOR LOGGING AND MONITORING LARGE SCALE TELESCOPE ARRAYS*

A. Costa†[1], E. Sciacca [1], G. Tosti[2], J. Schwarz[3],
P. Bruno [1], A. Calanducci [1], A. Grillo [1], F. Vitello [1], V. Conforti[4], F. Gianotti[4],
U. Becciani[1], S. Riggi[1]
[1]INAF, Osservatorio Astrofisico di Catania, Catania, Italy
[2]Università di Perugia, Dipartimento di Fisica e Geologia, Perugia, Italy
[3]INAF, Osservatorio Astronomico di Brera, Brera, Italy
[4]INAF, Osservatorio di Astrofisica e Scienza dello Spazio di Bologna, Bologna, Italy

Abstract

Large volumes of technical and logging data result from the operation of large scale astrophysical infrastructures. In the last few years several "Big Data" technologies have been developed to deal with a huge amount of data, e.g. in the Internet of Things (IoT) framework.

We are comparing different stacks of Big Data/IoT architectures including high performance distributed messaging systems, time series databases, streaming systems, interactive data visualization. The main aim is to classify these technologies based on a set of use cases typically related to the data produced in the astronomical environment, with the objective to have a system that can be updated, maintained and customized with a minimal programming effort.

We present the preliminary results obtained, using different Big Data stack solution to manage some use cases related to quasi real-time collection, processing and storage of the technical data, logging and technical alert produced by the array of nine ASTRI telescopes that are under development by INAF as a pathfinder array for the Cherenkov astronomy in the TeV energy range.

INTRODUCTION

Internet of Things (IoT) is an emerging technology that is becoming an increasing topic of interest among technology giants and business communities. IoT components are interconnected devices over the network, which are embedded with sensors, software and smart apps to collect and exchange data with each other or with cloud/data centres. The data generated by IoT devices is large in volume and random in nature and needs to be analyzed using Big Data analytics engine (see e.g. [1]) in order to extract the critical information or to understand behavioural patterns.

ASTRI [2] (*Astrofisica con Specchi a Tecnologia Replicante Italiana*) started in 2010 to support the development of technologies within the Cherenkov Telescope Array[3]. The first result of the project was the construction of a prototype telescope now installed at the astronomical INAF site at the Etna volcano in Sicily. The next phase of the project, currently underway, is the construction of a series of nine units of ASTRI telescopes (named ASTRI Mini-Array) [4].

This paper presents the logging and monitoring software architecture that is under development for the ASTRI mini-array telescopes that takes advantage of this new technological evolution to be prepared for the challenges related to the operation of the telescopes including the reliability, availability and maintainability of all its sub-systems and auxiliary devices.

ASTRI TELESCOPES

The logging and monitoring system takes into consideration all the telescopes and their subsystems [5]. Each telescope includes the system to control the telescope motion (such as the Mount Control System–MCS for the motion of the mechanical structure and the Active Mirror Control–AMC for controlling the primary and secondary mirrors) and the camera activities. In addition, in the mini-array site, we foresee some auxiliary systems to detect the environmental condition (such as the Weather Station–WS) and to assess the quality of the observations (such as the All Sky Camera–ASC). Figure 1 shows the telescopes sub-systems and auxiliary systems handled by the logging and monitoring software.

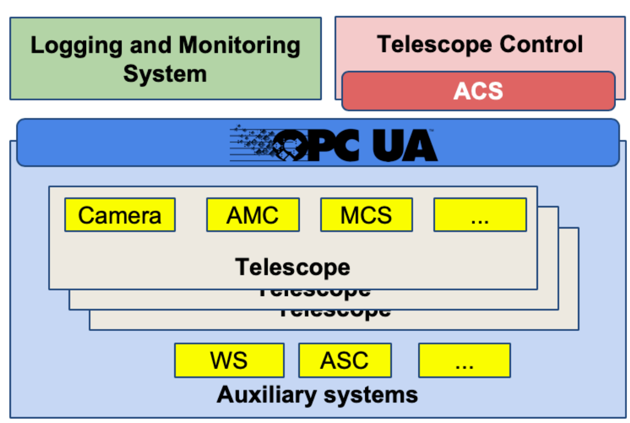


Figure 1: Telescopes sub-systems and auxiliary systems handled by the logging and monitoring software.

Starting from the experiences with the ASTRI prototype, we estimate a technical/operational data load of about 14

---

* This work was partially supported by the ASTRI "Flagship Project" financed by the Italian Ministry of Education, University, and Research (MIUR) and led by the Italian National Institute of Astrophysics (INAF).
† alessandro.costa@inaf.it

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA033

TIMING, SYNCHRONIZATION AND SOFTWARE-GENERATED BEAM CONTROL AT FRIB*

E. Daykin†, M. Konrad, Facility for Rare Isotope Beams, East Lansing, Michigan

Abstract

The Facility for Rare Isotope Beams, once completed, will require hundreds of devices throughout the machine to operate using synchronized timestamps and triggering events. These include, but are not limited to fault timestamps, time-dependent diagnostic measurements and complex beam pulse patterns. To achieve this design goal, we utilize a timing network using off-the-shelf hardware from Micro Research Finland. A GPS time base is also utilized to provide client timestamping synchronization via NTP/PTP. We describe our methods for software-generated event and beam pulse patterns, performance of installed equipment against project requirements, integration with other systems and challenges encountered during development.

TIMING NETWORK TOPOLOGY

The FRIB timing network consists of two main segments: a stable, high-accuracy fiber event and time distribution network, as well as Precision Time Protocol (PTP) [1] and Network Time Protocol (NTP) [2] distribution over the facility LAN. A schematic of this configuration can be seen in Fig. 1. Devices are classified as high-, medium-, or low-accuracy depending on their sensitivity to time drift and jitter. These requirements are detailed below in Table 1:

Table 1: Timing Requirements

Class	Accuracy	Examples
High	$\pm 1\ \mu s$	Beam instrumentation
Medium	± 1 ms	Devices with timestamped interlocks
Low	± 1 s	Most other network devices

Fiber Distribution

Fiber-optic event signals are suitable for devices classified as high-accuracy and/or require event triggers. Events and timestamps are propagated over a three-level, tree-like network of inexpensive Micro Research Finland (MRF) [3] distribution fan-outs. The master fan-out transmits events to nine level 2 nodes, each in turn serving a handful of level 3 nodes within their respective 'region'. Level 3 nodes finally distribute these events to client devices in their immediate operating area.

* Work supported by the U.S. Department of Energy Office of Science under Cooperative Agreement DE-SC0000661
† daykin@frib.msu.edu

Network Time Distribution

Network timing is suitable for devices classified as medium or low-accuracy. For more modern devices in both classes, PTP is preferred due to its superior accuracy, more robust master selection, and more graceful handling of leap seconds.

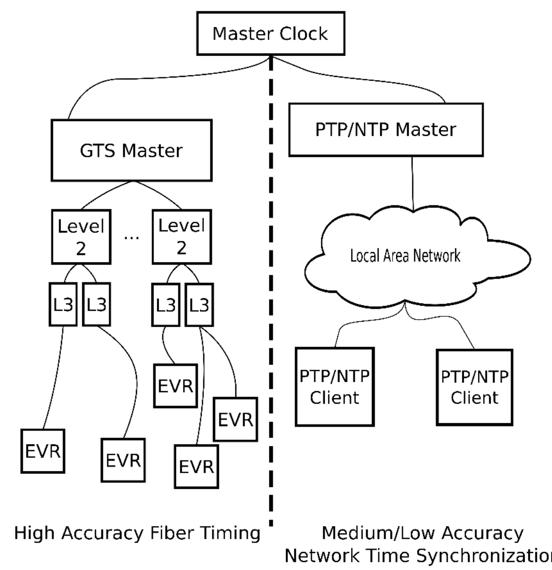


Figure 1: Simplified layout of FRIB's timing network.

MASTER DESIGN

The master time's stability is maintained via widely available timing hardware. This hardware generates precise 1 pulse per second (1 PPS) and 10 MHz signals, which are used for stable event distribution over MRF timing hardware, in addition to providing a phase-locked oscillator (PLO) reference signal for the facilty-wide RF clock. The fine design of this system is detailed in Fig. 2.

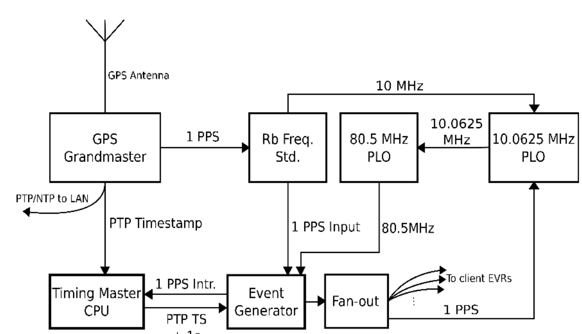


Figure 2: Synchronization and event generation machinery.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA034

SOFTWARE ARCHITECTURE FOR NEXT GENERATION BEAM POSITION MONITORS AT FERMILAB

J. S. Diamond[#], Fermilab, Batavia, USA

Abstract

The Fermilab Accelerator Division / Instrumentation Department develops Beam Position Monitor (BPM) systems in-house to support its sprawling accelerator complex. Two new BPM systems have been deployed over the last two years – one upgrade and one new. These systems are based on a combination of VME and Gigabit Ethernet connected hardware and a common Linux-based embedded software platform with modular components. The architecture of this software platform and the considerations for adapting to future machines or upgrade projects will be described.

INTRODUCTION

The Fermilab Booster is a synchrotron accelerator with a circumference of 474 meters which accepts 400 MeV protons from the Linac, accelerates to 8 GeV in less than 67 milliseconds to be injected into the Recycler. In 2018 the Booster BPM data acquisition system was upgraded to a VME-based system based on in-house developed digitizer and timing modules. At the same time the design of a VME-based BPM system for Fermilab's newest accelerator, the Integrable Optics Test Accelerator (IOTA) was being developed. The decision was made to re-use as much hardware, firmware and software as possible from the Booster BPM upgrade project.

HARDWARE & FIRMWARE

Both the Booster BPM upgrade project and the IOTA BPM project are based on VME 64x crates supplied by Weiner. In the case of the Booster, six crates are utilized to instrument all the BPMs while IOTA uses only one. Each crate contains a Single Board Computer (SBC), a Timing and signal distribution board and multiple digitizer modules. The Booster BPM system uses an Artesyn MVME-8100 SBC with a QorIQ processor and 2GB of system RAM. The IOTA BPM system uses a Concurrent Technologies 405x SBC with an Intel Core Duo processor and 2GB of system RAM.

The Timing module is based on a design originally developed for the Main Injector and Tevatron BPM systems. The timing module decodes the Fermilab site-wide machine clock, TCLK and synchronizes with the machine RF.

Each digitizer module receives ADC clock and trigger signal from the timing module. The raw signals from the BPM plates are passed through an analog transition module and connected to an ADC channel on the digitizer module. The digitizer modules run at 250 MSPS and filter the data through a down converter and into on-board RAM.

In the Booster BPM the analog transition modules are present in the VME crate and use the VME bus for power. The data acquisition software can detect these modules and verify that they are present but otherwise does not interact with them. In the IOTA BPM these modules are external to the VME crate and powered by their own NIM crate and have no interaction with the data acquisition software. Plans for a future BPM system are incorporating a Raspberry Pi-based controller in the NIM crate for controlling attenuation and gain settings on the analog transition modules.

EMBEDDED LINUX STACK

Buildroot is an open source embedded Linux build system that automates the construction of a cross-compile toolchain, Linux kernel and root filesystem. Building a Linux kernel and root filesystem from scratch gives the developer control over the cross-compile toolchain, which support software is present and which kernel options are used. Using Buildroot also allows us to achieve a system footprint of less than 30 MB.

DEVICE DRIVERS

Communication with hardware devices over the VME bus was achieved using the mainline VME driver introduced into the Linux kernel in version 3.10. This driver allows developers to interact with VME attached device from within a Linux kernel module (LKM) much like a PCI or USB device. The mainline VME driver supports both the Universe II and TSI-148 VME bridge chips through a common API.

Each in-house developed hardware device requires an LKM to be developed to facilitate communication between the data acquisition software and the device. In the case of the timing modules this LKM only communicates with one device but the LKM for the digitizer and analog transition modules must support communication with multiple devices. Upon insertion into the kernel the LKM requests access to the VME bus by acquiring a resource from the kernel VME driver. The VME resource is used to scan the VME bus and probe for hardware. Once a hardware device is successfully probed it is registered with the Linux device model as a VME bus device. If the LKM is removed, user space is notified, and the device is removed from the device model as a part of the LKM shutdown procedure.

Access to hardware device registers is provided to user space using the Linux sysfs filesystem. Each register on the hardware is made available as a sysfs attribute that can be read/written through the device's entry in sysfs. This is a useful tool for developers to interact with and

* This work was supported by the DOE contract No. DEAC02-07CH11359 to the Fermi Research Alliance LLC.
# jdiamond@fnal.gov

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-MOPHA039

SLOW CONTROL SYSTEMS AT BM@N AND MPD/NICA DETECTOR EXPERIMENTS

D. Egorov, P. Chumakov, R. Nagdasev, V. Shutov, JINR, Dubna, Russia

Abstract

NICA (Nuclotron-based Ion Collider fAcility) is a new accelerator complex designed at the Joint Institute for Nuclear Research (Dubna, Russia) to study properties of dense baryonic matter. BM@N (Baryonic Matter at Nuclotron) is the first experiment at the complex. It is an experimental setup in the fixed-target hall of the Nuclotron to perform a research program focused on the production of strange matter in heavy-ion collisions. MPD (Multipurpose Detector) is a detector for colliding beam experiments at the complex, and it is being developed to provide: efficient registration of the particles produced by heavy ion collisions; identification of particle type, charge and energy; reconstruction of vertices of primary interactions and the position of secondary particle production. Existing Slow Control Systems for BM@N experiment, assembling, and testing zones of MPD detectors are based on Tango Controls. They provide monitoring and control of diverse hardware for efficient data taking, stable operation of detectors and quality control of assembled modules. Current status and developments as well as future design and plans for MPD Slow Control System will be reported.

INTRODUCTION

The NICA (Nuclotron-based Ion Collider facility) is accelerator facility which is now under construction at Joint Institute for Nuclear Research (JINR, Dubna) [1]. NICA accelerator complex scheme is shown in Fig. 1.

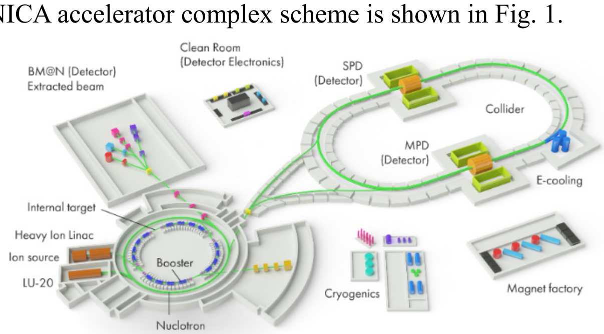


Figure 1: NICA accelerator complex scheme [2].

Two modes of operation are foreseen, collider mode and extracted beams, with two detectors: MPD (MultiPurpose Detector) and BM@N (Baryonic Matter at Nuclotron) [1].

First physical run at BM@N with basic setup was performed in spring 2018, which also included new physics program on SRC studies in collaboration with GSI, MIT, Tel Aviv University etc. BM@N experiment scheme is shown in Fig. 2.

MPD is currently under construction and its first test run is planned on 2021. MPD scheme is shown in Fig. 3.

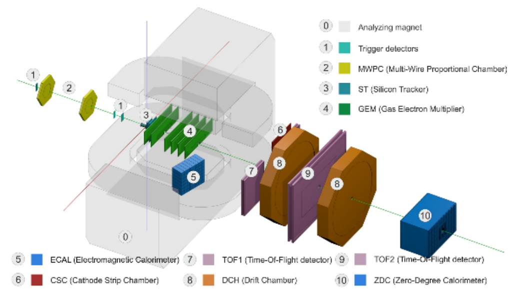


Figure 2: BM@N experiment scheme.

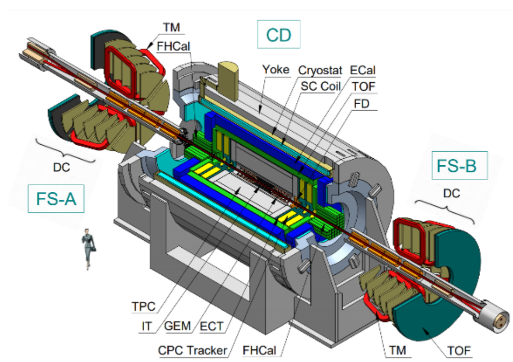


Figure 3: MPD detector scheme [2].

Though these two experiments have different principle of operation, they also have many common parts – same type of some detectors, same hardware and the most important point – same people, who develop and work with it.

So the main tasks of Slow Control Systems, which are described in this paper – monitor the statuses of the diverse hardware, archive the data from the devices in unified format to the database for further physical analysis and provide scalability of the developed system, because the number of modules and channels is different in both facilities.

Tango Controls was chosen as the base for such systems. It is free, open-source and cross-platform framework, which supports multiple programming languages and has tools that simplify the tasks mentioned above [3].

Most of the developments, described in the next section, was tested during multiple technical and physical runs on BM@N, but the same applications will be used on MPD.

CURRENT DEVELOPMENTS

It is important for the shift crew to know the states of detectors' hardware. The shift leader must start data acquisition according to the statuses of devices, proper settings and so on.

Tango device server and the client with graphical layout of detectors was developed for BM@N experiment [4]. The client window is shown in Fig. 4.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA040

BEAM POSITION FEEDBACK SYSTEM SUPPORTED BY KARABO AT EUROPEAN XFEL

V. Bondar*, W. Ehsan, M. Beg, M. Bergemann, C. Carinan, R. Costa, F. Dall'Antonia, C. Danilevski, S.G. Esenov, R. Fabbri, H. Fangohr[1], G. Flucke, D. Fullà Marsà, A. Galler, G. Giovanetti, D. Göries, J. Grünert, S. Hauf, D. Hickin, T. Jarosiewicz, E. Kamil, Y. Kirienko,

A. Klimovskaia, T. Kluyver, D. Mamchyk, T. Michelat, I. Mohacsi, A. Parenti, D. Rueck, H. Santos, R. Schaffer, A. Silenzi, C. Youngman, J. Zhu, P. Zalden, S. Brockhauser[2]

European XFEL GmbH, Holzkoppel 4, 22869, Schenefeld, Germany

[1]also at University of Sauthampton, SO17, 1BJ, Southampton, United Kingdom

[2]also at Biological Research Center of the Hungarian Academy of Sciences, Szeged, Hungary and at University of Szeged, Arpad ter 2, 6701 Szeged, Hungary

Abstract

The *XrayFeed* device of Karabo [1, 2] is designed to provide spatial X-ray beam stability in terms of drift compensation utilizing different diagnostic components at the European XFEL (EuXFEL). Our feedback systems proved to be indispensable in cutting-edge pump-probe experiments at EuXFEL.

The feedback mechanism is based on a closed loop PID control algorithm [3] to steer the beam position measured by the so-called *diagnostic devices* to the desired centred position via defined actuator adjusting the alignment of X-ray optical elements, in our case a flat X-ray mirror system.

Several *diagnostic devices* and actuators can be selected according to the specific experimental area where a beam position feedback is needed. In this contribution, we analyze the improvement of pointing stability of X-rays using different diagnostic devices as an input source for our feedback system. Different types of photon diagnostic devices such as gas-based X-ray monitors [4], quadrant detectors based on avalanche photo diodes [5] and optical cameras imaging the X-ray footprint on scintillator screens have been evaluated in our pointing stability studies.

INTRODUCTION

At the European X-ray Free Electron Laser (EuXFEL) facility there are currently three X-ray optical beam lines which provide soft and hard X-ray photons to six instruments. In order to control both the hardware components of the beamline and the data acquisition from the instruments, EuXFEL has developed in-house a control system, Karabo. Hardware devices and system services are represented in this control system as Karabo software devices, and are distributed among various control hosts, thus making Karabo a distributed control system. The Karabo design is event-driven, offering subscription to remote signals to avoid polling for parameter updates. Devices communicate via a central message broker using language (C++ and Python) agnostic remote procedure calls (RPC) [6, 7]. Here, we focus on the design and usage of one such device, called *XrayFeed*. Its aim is to continuously stabilize the beam position in experiments, removing the need for direct user manipulation to ensure beam stability.

The idea of feedback control is to make a setup that ensures that any deviation of a measured parameter (beam position) from the set point (desired position) will be corrected, thus providing stability (e.g. beam position in a given plane). It is implemented using the PID (Proportional-Integral-Derivative) mechanism.

The *XrayFeed* software device (see Fig. 1) can be used in conditions where external disturbances to the positions of mirrors under control are not predictable and when the PID mechanism satisfies positional accuracy requirements [3]. The proposed device is robust for PID tuning process as well as for PID controller operation. The device allows different actuators and different diagnostic devices to be involved in the feedback control schema and monitoring of the real time behavior of the system under PID control. By so-called diagnostic detector we mean a diagnostic device whose output can be used as a 'measured' signal in feedback control loops. *XrayFeed* has a flexible implementation allowing choice of diagnostic device for feedback control and characterization and optimization of the feedback solution used. In this paper we describe the design of *XrayFeed* software device and present results of its application for precise position feedback control in the flat X-ray mirror system using different actuators and diagnostic devices.

BEAM POSITION FEEDBACK SETUP

Block diagram of *XrayFeed* device implementing PID algorithm is illustrated in Fig. 2. Actual beam position in a plane normal to the beam is measured by a diagnostic device (*Px* signal on diagram) and is inputted into the PID controller whose aim is to minimize beam position displacement error *e(t)*. This error is processed according to the Proportional-Integral-Derivative algorithm using PID gains determined during a tuning process. Resulting PID signal is set in the actuator device as an input voltage *u(t)*.

* valerii.bondar@xfel.eu

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA041

CAUSE-AND-EFFECT MATRIX SPECIFICATIONS FOR SAFETY CRITICAL SYSTEMS AT CERN

B. Fernández *, E. Blanco, M. Charrondiere,
R. Speroni, CERN, Geneva, Switzerland
H. Hamisch, M. Bonet, M. H. de Queiroz,
Universidade Federal de Santa Catarina, Florianópolis, Brazil

Abstract

One of the most critical phases in the development of a Safety Instrumented System (SIS) is the functional specification of the Safety Instrumented Functions (SIFs). This step is carried out by a multidisciplinary team of process, controls and safety experts. This functional specification must be simple, unambiguous and compact to allow capturing the requirements from the risk analysis, and facilitating the design, implementation and verification of the SIFs. The Cause and Effect Matrix (CEM) formalism provides a visual representation of Boolean expressions. This makes it adequate to specify stateless logic, such as the safety interlock logic of a SIS. At CERN, a methodology based on the CEM has been applied to the development of a SIS for a magnet test bench facility. This paper shows the applicability of this methodology in a real magnet test bench and presents its impact in the different phases of the IEC 61511 safety lifecycle.

INTRODUCTION

The European Organization for Nuclear Research (CERN) operates the largest particle physics laboratory in the world. This research laboratory hosts many critical industrial installations that are necessary for the numerous experiments performed here. Some examples are cryogenics plants, cooling and ventilation processes, powering systems, superconducting magnet test benches and many more. A failure in these industrial installations or in their control systems may have catastrophic consequences, such as enormous economic losses, environmental damages or even human causalities. For that purpose, at CERN, many Safety Instrumented Systems (SISs) have been engineered to mitigate the risks of these industrial processes.

Safety Instrumented Systems

A SIS is a prevention mechanism designed to reduce the probability of occurrence of hazardous events. The IEC 61511 standard [1] provides the so-called safety life cycle, which provides guidelines to develop, maintain and manage a SIS for the process industry. Once the unacceptable risks are identified by the risk analysis, the process and safety experts must specify the necessary Safety Instrumented Functions (SIFs) to reduce the probability of occurrence of these risks. A SIF specification must contain at least the following elements:

- The functionality of the SIF: a precise description of the required SIF logic.
- The target Safety Integrity Level (SIL): a quantitative measure of the risk reduction.
- The operation mode required for the SIF: low, high or continuous demand, depending on the nature of the process and risk.

Nowadays, Safety PLCs (Programmable Logic Controllers) are widely used in SISs and the functionality of the SIFs is implemented in the PLC programs.

The specification method to express the functionality of a SIF must be simple, unambiguous and compact to allow capturing the requirements from the risk analysis, and facilitate the design and implementation of the PLC program. There are many specification methods to express unambiguously the functionality of a SIF, for example, a textual boolean expression, a logic diagram or a Cause and Effect Matrix (CEM).

Objectives

This paper presents a real case study at CERN of the usage of a CEM-based specification to express the interlock logic of a magnet test bench installation. The benefits and limitations of this approach in comparison with the previously adopted methods are also summarized.

The paper is structured as follows: first, the paper introduces the basic concepts and adopted CEM semantics for this project. Second, the case study is described, including the process description and an example of the CEM usage. Finally, the analysis and conclusions of this study are presented.

CAUSE AND EFFECT MATRIX

Cause and Effect Matrix is a compact and intuitive graphical representation of boolean expressions. This makes it adequate to represent stateless logic, where a given output depends only on a combination of the current input signals. CEM is generally well accepted to specify interlock logic in the process and manufacturing industries. However there are many variants of CEMs and the companies adopt the semantics that best adapt to their processes and engineering practices. Some PLC providers have included the CEM in their engineering tools. This is the case of Siemens Industrial Automation, which provides the SIMATIC Safety Matrix [2]. This tool allows the use of CEM as a specification mechanism but imposes a specific software architecture,

* borja.fernandez.adiego@cern.ch

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA042

EVALUATING VISTA AND EPICS WITH REGARD TO FUTURE CONTROL SYSTEMS DEVELOPMENT AT ISIS

I. Finch, ISIS, Rutherford Appleton Laboratory, United Kingdom

Abstract

The ISIS accelerators currently use the Vista Controls System software Vsystem for machine control purposes. In this paper we describe our preliminary work in evaluating a possible migration to the EPICS control system, with emphasis on lessons learned in our previous software migration. We outline a Vsystem/EPICS bridge that has been developed to facilitate the trial and any migration following on from that evaluation.

A HISTORY OF CHANGE

The ISIS spallation neutron source saw first beam in 1984 and was formally opened in 1985, with a second target station commissioned in 2008 [1]. Over the course of 35 years the controls system has had to incorporate several generations of computer and embedded systems. The control system has also undergone two migrations of server hardware platforms (GEC → DEC Alpha, DEC Alpha → HP Itanium). The first of these hardware platform migrations was also a migration of operating system (GEC Core OS → OpenVMS) and controls system software (GRACES / BABBAGE → Vsystem) [2].

With the announced discontinuation of the Itanium processor architecture [3], a transition to a commodity x86 platform is in the early stages. This will be combined, again, with a change of operating system as we migrate to Linux. Since the existing Vista Controls System software product Vsystem [4], colloquially called Vista, used to control the accelerators may be run on a mixture of operating systems (OpenVMS, Linux, and Windows) and their compatible hardware platforms, this is anticipated to be complex but soluble.

However, during this computer hardware platform and operating system migration we also plan to conduct an evaluation of the Experimental Physics and Industrial Control System (EPICS) [5]. If this evaluation is successful, we may conduct a second controls system software migration.

VSYSTEM AND EPICS COMPARISON

Vsystem and EPICS both originated in work done during the 1980s to create a control system for the Ground Test Accelerator Controls System (GTACS) at Los Alamos National Laboratory (LANL) [6]. While EPICS eventually became an open-source project developed as a collaboration between multiple accelerator organisations, Vista is a closed-source commercial product with paid support. Both are distributed control systems with common features such as databases, and channels (Vsystem) or process variables (EPICS).

Vsystem is the more centralised of the two with databases existing on central servers, and the expectation that input and output to the hardware will be performed by readers and handlers respectively running on the same servers. In EPICS Input/Output Controllers (IOCs), the equivalent of Vsystem's readers and handlers, are expected to be distributed and localised to the controlled hardware along with the databases.

FRONT END TEST STAND

The EPICS evaluation will be conducted on the ISIS Front End Test Stand (FETS) [7], a hardware development system separate from the main ISIS accelerators. FETS consists of a Penning ion source, a magnetic low energy beam transport (LEBT) to focus the ion beam, a Radio Frequency Quadrupole accelerator (RFQ) to bunch and accelerate the beam, a medium energy beam transport (MEBT) and a chopper line to increase the separation of the bunches ready for injection into a synchrotron.

The existing controls hardware on FETS is a mixture of our in-house developed Controls Standard STE (CSS) and CompactPCI Standard (CPS) systems. The CSS hardware was developed and deployed in the 1990s and is based on the STEbus standard [8]. It consists of an Intel 80188 processor, an Ethernet card, and one or more commercial off-the-shelf or ISIS-designed I/O cards. The CPS system is an evolution of this design based on CompactPCI hardware [9]. It was developed and designed in the 2000s and is our most current system. The CPS hardware is also undergoing a migration to Linux [10].

This CSS and CPS hardware on FETS already has Vsystem controls interfaces implemented. Since these systems were developed in-house, no existing EPICS IOC are available.

PLANNING FOR MIGRATION

Development work for the EPICS deployment on FETS has not begun with the development of IOCs for CSS and CPS hardware. Instead software has been developed which creates an EPICS bridge to the existing Vsystem controls system. This will allow any existing hardware monitored or controlled by Vsystem to be monitored or controlled through EPICS.

This approach was chosen because it allows us to decouple any future migration of the controls system UI from the migration of the controls system hardware interfaces. Each may be done independently. In addition, since Vsystem hardware interfaces exist for all our currently deployed hardware we can apply finite IOC development effort to

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA043

ACCELERATOR CONTROL DATA MINING WITH WEKA*

W. Fu†, K. A. Brown, T. D'Ottavio, P. S. Dyer, S. Nemesure
Brookhaven National Laboratory, Upton, USA

Abstract

Accelerator control systems generates and stores many time-series data related to the performance of an accelerator and its support systems. Many of these time series data have detectable change trends and patterns. Being able to timely detect and recognize these data change trends and patterns, analyse and predict the future data changes can provide intelligent ways to improve the controls system with proactive feedback/forward actions. With the help of advanced data mining and machine learning technology, these types of analyses become easier to conduct. As machine learning technology matures with the inclusion of powerful model algorithms, data processing tools, and visualization libraries in different programming languages (e.g. Python, R, Java, etc), it becomes relatively easy for developers to learn and apply machine learning technology to online accelerator control system data. This paper explores time series data analysis and forecasting in the Relativistic Heavy Ion Collider (RHIC) control systems with the Waikato Environment for Knowledge Analysis (**WEKA**) system and its Java data mining APIs.

INTRODUCTION

The Waikato Environment for Knowledge Analysis (WEKA) [1] system provides more than 80 machine learning algorithms and models in the latest version and third party addon packages. This number continues to increase as more models become available over time. Typically, a data mining process with machine learning involves the following steps as shown in Fig. 1. WEKA performs data mining tasks in a similar fashion.

In the RHIC controls system, time-series data is logged and saved in databases or in files using a self-describing data set (SDDS) format. WEKA supports four types of data that includes numeric, nominal, string, and date-time. WEKA also supports data files in ARFF/CVS formats which is able to embed above named basic data types [2]. Due to the data format and type differences, most RHIC controls data cannot be directly loaded into the WEKA system. Proper data conversion and pre-processing are required.

This data conversion process can be simplified by taking advantage of the Collider-Accelerator Department (C-AD)'s existing Database Management Tool (dmt). This application pre-processes the data from a database into the format needed by WEKA. Once the data is imported into WEKA, all applicable data processing models, training models, visualization tools, and other handy tools available in WEKA can be applied to the data.

For both logged and real-time data, we developed a Java program which implemented the WEKA's Java machine learning APIs for data loading, conversion, pre-processing, and model training, and use the program to load and convert logged/live RHIC controls data. The data is then streamed into the WEKA data mining engine providing for easy access to all of WEKA's data mining models.

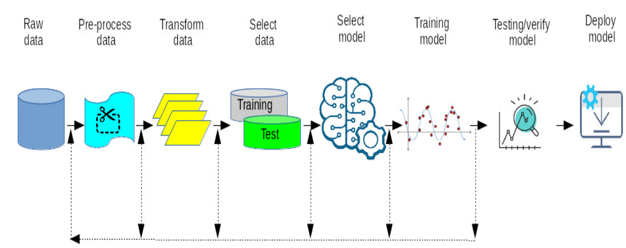


Figure 1: Work flow of data mining with WEKA.

WEKA has a rich set of tools and Java APIs for many kinds of data mining/machine learning tasks. In this paper, we focus on accelerator controls time series data analysis and forecasting.

TIME SERIES DATA FORECASTING

Time series data is a common category of accelerator controls data. A time series data is predictable if:

- The target data series has detectable and recognizable change trends and patterns;
- The target data series has one or more associated data series which have detectable and recognizable change trends and patterns, and these trends and patterns affect the data changes in target data series and make the target data predictable;

Machine learning models can help detect and recognize hidden trends and patterns quickly.

The temporal nature of accelerator time-series data creates the desired need to predict future data trends in order to anticipate and prevent problems and take optimized action ahead of time.

Time series data are typically not suitable for data mining or machine learning techniques which requires each data point to represent an independent observation and independent of data order. Time series analyses [3] use statistical techniques to model a time-dependent series of data. This is usually the better choice for forecasting future data based on historical data. WEKA has a

* Work supported by Brookhaven Science Associates, LLC under Contract No. DE-SC0012704 with the U.S. Department of Energy.
† Email: fu@bnl.gov

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA044

DEVELOPMENT OF ETHERNET BASED REAL-TIME APPLICATIONS IN LINUX USING DPDK

G. Gaio†, G. Scalamera, Elettra-Sincrotrone Trieste S.C.p.A., Trieste, Italy

Abstract

In the last decade Ethernet has become the most popular way to interface hardware devices and instruments to the control system. Lower cost per connection, reuse of existing network infrastructures, very high data rates, good noise rejection over long cables and finally an easier maintainability of the software in the long term are the main reasons of its success. In addition, the need of low latency systems of the High Frequency Trading community has boosted the development of new strategies, such as CPU isolation, to run real-time applications in plain Linux with a determinism of the order of microseconds. DPDK (Data Plane Development Kit), an open source software solution mainly sponsored by Intel, addresses the request of high determinism over Ethernet by bypassing the network stack of Linux and providing a more friendly framework to develop tasks which are even able to saturate a 100 Gbit/s connection. Benchmarks regarding the real-time performance and preliminary results of employing DPDK in the acquisition of beam position monitors for the fast orbit feedback of the Elettra storage ring will be presented.

INTRODUCTION

For today's control systems Ethernet is an attractive option that can compete with and often overtake field-bus technologies. Setting up an Ethernet infrastructure is generally easier and less expensive than other communication networks. In the particle accelerator field, suppliers of devices or systems are replacing the conventional interfaces (direct I/O, serial lines, GPIB, etc.) with Ethernet links, especially in high performing instrumentation.

In time-sensitive applications such as feedback systems and, more recently, performance optimization applications using machine learning, it is fundamental to process synchronous data in real-time because their effectiveness scale linearly with the repetition rate. For this reason interfacing Ethernet based devices to the control system in the most efficient way is becoming even more important than in the past.

Real-time Networking

Since 2005 the new front-end computers installed in Elettra and later in FERMI run the GNU/Linux operating system. The kernel versions span from 2.4.25 running on the oldest PowerPC VME boards to the less older 3.14.58 installed on rack-mount servers.

Until the release of RT_PREEMPT on the mainline (2.6.23), the vanilla Linux kernel was unreliable predicting the execution of a task.

Even today, the network stack is unsuitable for developing time sensitive network applications [1]. The POSIX socket operations (system calls), which transfer control from the application layer to the kernel have significant overheads (e.g. context switch and CPU cache pollution). Moreover in the last fifteen years the network performance has grown faster than the one of the CPUs due the stagnation in the single thread performance. Interrupt moderation techniques try to mitigate the CPU load caused by high-end network interface cards (NIC) but at the cost of increasing the latency.

In order to overcome these limitations, for the FERMI and Elettra front-end computers involved in time critical applications we adopted RTAI and more recently Xenomai, which enable systems to perform real-time tasks. Since the majority of these applications exchange data through Ethernet, the drivers of the on-board NICs were modified to execute the interrupt handler in the RTAI/Xenomai domain and run arbitrary code bypassing the Linux network stack [2].

The downsides of this approach are the development time for patching Ethernet device drivers at every kernel upgrade over different architectures and the complexity of debugging real-time applications that usually run in kernel space. For these reason in the last years we have carried out a campaign to evaluate the real-time capabilities over Ethernet of the latest Linux kernel running in multi-socket servers.

LINUX TUNING

Thanks to the evolution of the Linux kernel, nowadays a system can be easily configured to prioritize low latency over throughput [3]. The most common customizations to perform are:

BIOS

- Enable turbo mode to allow the CPU to reach its maximum clock frequency.
- Disable CPU lower state to avoid the CPU turning to deeper sleep states.
- Disable hyper-threading because the logic cores that share resource with other logic cores can introduce latency.
- Disable virtualization and monitor options because they introduce latency in memory access.

Linux

- Remove a given CPU core from the general kernel Symmetric multiprocessor system (SMP) balancing and scheduler algorithm (*isolcpu*), pin the critical task to the reserved CPU core.

† giulio.gaio@elettra.eu

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA045

A NEW SIMULATION STRUCTURE TO IMPROVE SOFTWARE DEPENDABILITY IN COLLIDER-ACCELERATOR CONTROL SYSTEMS*

Y. Gao[†], T. Robertazzi, Stony Brook University, Stony Brook, USA
K. A. Brown, J. Morris, R. H. Olsen, Brookhaven National Laboratory, Upton, USA

Abstract

The Collider-Accelerator Department (C-AD) at Brookhaven National Laboratory (BNL) operates a world-class particle accelerator – the Relativistic Heavy Ion Collider (RHIC). To ensure its safe and proper operations, C-AD develops its own control systems. It is a large distributed complex system consisting of approximately 1.5 million control points [1]. The system has two physical layers: the front end level and the console level. In work [2], a new simulation structure is proposed which aims to improve the console level codes reliability. The structure enables developers to conveniently do customized testing on ADO[1] codes, specifically ADOs using the General Purpose Interface Bus (GPIB) interface [3]. In this work, a new simulation framework is proposed. It extends the simulation structure in [2] by accommodating new types of ADOs that use the Ethernet connections. Together, they form a more comprehensive simulation environment which enhances the overall controls software dependability.

INTRODUCTION

The Relativistic Heavy Ion Collider (RHIC) at Brookhaven National Laboratory (BNL) is a world-class particle accelerator, which helps scientists to study what the university may have looked like in the first few moments after its creation. RHIC contains two 3.8 kilometers counter-rotating super-conducting rings to carry particle beams which can be collided in 6 crossing regions to provide possible interactions for experimenters to study.

The RHIC is operated by elaborate control systems at the Collider-Accelerator Department (C-AD) of BNL. Instances of the C-AD control systems are also applied in the Linear Accelerator (LINAC), Electron Beam Ion Source (EBIS), Tandem Van de Graff pre-accelerators, the Booster accelerator, and the Alternating Gradient Synchrotron (AGS). C-AD control systems provide operational interfaces to the accelerator complex. Its architecture is hierarchical and consists of two physical layers with network connections: the Front End Computers (FECs) level and the Console Level Computers (CLCs) level, as shown in Fig. 1. The front end level contains more than 500 FECs, each of which running on the VxWorks™ real-time operating system. Every FEC consists of a VME chassis with a single-board computer[2], network connection, and I/O modules [4]. The console level

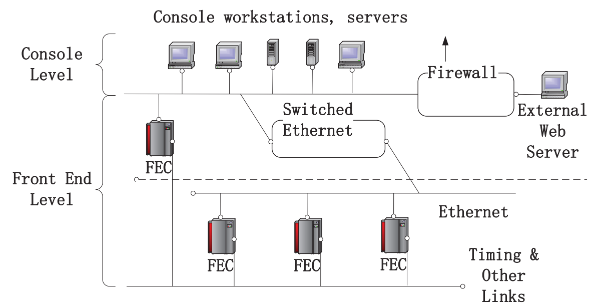


Figure 1: RHIC system hardware architecture.

is the upper layer of the control system hierarchy, which consists of operator consoles, physicist workstations and server processors that provide shared files, database and general computing resources.

There are several fundamental system components in the C-AD control systems.

Accelerator Device Object (ADO) It abstracts features from the underlying devices into a collection of control points (also known as parameters), and provide those parameters to the users of the control systems. ADO designers determine the number and names of parameters based on the needs of the system. ADO parameters can be viewed or edited by the Parameter Editing Tool (PET). ADOs provide the *set()* and *get()* methods as the controls interface to the accelerator devices. The accelerator complex is controlled by users or applications which set() and get() parameter values in instances of the ADO classes. A special preprocessor is used to help to convert ADO ".rad" files[3] into C++ files [5]. It takes care of the necessary details, and allows the ADO designers to focus on the more important parts: the ADO functionalities.

Controls Name Server (CNS) It is a centralized repository where unique name/value pairs can be efficiently managed and queried. Given an object's instance name[4], the CNS will provide enough information so that the associated data can be accessed. The CNS is session oriented which means several copies of it can be run at the same time as long as each of them has its own host. This feature allows developer to have a "private" CNS, which makes it possible to signal a process to look for an ADO instance in a different place from where it normally resides. The proposed

* Work supported by Brookhaven Science Associates, LLC under Contract No. DE-SC0012704 with the U.S. Department of Energy.
† ygao@bnl.gov
[1] ADO stands for Accelerator Device Object, see details below.
[2] They can have different processor architectures, e. g. POWER3E, MV2100, MV3100, XILINX, etc.
[3] It stands for RHIC ADO Definition file.
[4] That object can be an ADO parameter, a Complex Logical Device (CLD), a manager's parameter name, or an alias (a name used by developers which is more human-readable), etc.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA046

A NEW SIMULATION TIMING SYSTEM FOR SOFTWARE TESTING IN COLLIDER-ACCELERATOR CONTROL SYSTEMS*

Y. Gao†, T. Robertazzi, Stony Brook University, Stony Brook, USA
K. A. Brown, M. Harvey, J. Morris, R. H. Olsen, Brookhaven National Laboratory, Upton, USA

Abstract

Accelerators need to be operated in a timely way to successfully accelerate the beam from its creation at its source, to the experiments at its destination. Thus, synchronization among accelerator devices is important. The Collider-Accelerator Department (C-AD) of Brookhaven National Laboratory (BNL) develops their own control systems for their accelerator complex[1]. The synchronization in the C-AD control systems is accomplished by a distribution of timing signals, which are sent out along so-called[2] *time lines* [1] in the form of digital codes. Accelerator devices in the complex which require their times synchronized to the acceleration cycles are connected to the time lines. Those devices are also equipped with time line decoders [2], which allow them to extract timing signals appropriately from the time lines. In this work, a new simulation timing system is introduced, which can generate user-specific timing events for software testing in the C-AD control systems.

INTRODUCTION

Accelerator systems must be synchronized for the proper operations of equipment over a wide area. In order for the beam to have the desired properties (momentum, size, intensity, etc.), devices must act in concert, and evolve together in a particular way [2]. Hence the synchronization among accelerators, and devices within each accelerator is crucial.

In the C-AD control systems, the synchronization is accomplished by a distribution of timing signals around the accelerator facility [2]. Devices which need timing signals synchronized to the acceleration cycles are connected to the time lines. Timing signals are sent out along a time line in the form of digital codes, and those codes (representing the timing signals) can be extracted by devices (equipped with time line decoders) in the accelerator complex from the time line as signals to perform certain operations.

To better understand it, consider the following example. The Booster main magnet power supply is programmed to start to follow a reference function when it receives a time line trigger from the time line. A time line trigger is also typically called a time line event, which corresponds to a specific hexadecimal number and is given a 3-letter acronym. In this case, the hexadecimal number is $000A$, and its acronym is $BT0$, which stands for Booster-Time-zero. This time line event triggers the start of the Booster main magnet cycle, hence this name. Moreover, $BT0$ is distributed to all devices on the time line, therefore any other device who is interested in this event will also be programmed to respond to it.

In the C-AD accelerator complex, there is an elaborate timing system to accomplish the synchronization, which provides a highly reliable, serial timing link to all equipment locations. Events and clocks derived from this link are used to initiate hardware operations including changes in settings, state changes, and data acquisition. Particularly, the synchronization is collectively conducted by three timing systems [3], the event link system, the beam-sync event system, and the Real Time Data Link (RTDL). The event link system provides a reliable serial timing link to equipment locations throughout the RHIC complex. It is a crucial component in the C-AD control systems.

This work mainly focuses on the event link system. Specifically, a new simulation timing system is proposed, which can generate user-defined timing events at specific times on specific event links. Developers can use this simulation system to interact with timing-sensitive applications for testing purposes, hence improving the reliability of the controls timing system.

A more detailed motivation is presented.

Motivation

The occurrences of timing events on the C-AD event links affect the running of controls software in many ways. Some particular timing events trigger the executions of some software methods directly. Other events (such as PPM[3] user codes) establish a context that affects the way software operate.

In the front end level, Front End Computers (FECs) detect events by VME boards with direct connections to an event link. In the console level, ADO managers[4] and other console level processes receive notifications over the network from FECs. FECs use "relMon" ADOs to deliver notifications of events as they happen on the event link. For each event link, a special "evMon" ADO delivers regular reports that summarize all the events that have occurred during a Supercycle (a 4 seconds time period on the RHIC event link, see details in the next section).

In order to test software thoroughly, the software should be run in a variety of timing conditions. For example, we sometimes need to arrange a time to test a piece of software when multiple PPM users are active at the same time in the

* Work supported by Brookhaven Science Associates, LLC under Contract No. DE-SC0012704 with the U.S. Department of Energy.
† ygao@bnl.gov
1 The complex includes the Linear Accelerator (Linac), the Electron Beam Ion Source (EBIS), the Tandem Van de Graff pre-accelerators, the Booster accelerator, the Alternating Gradient Synchrotron (AGS), and the Relativistic Heavy Ion Collider (RHIC).
2 These time lines are like timing buses inside a computer.
3 Pulse to Pulse Modulation, which will be introduced later.
4 Console level servers which hold ADOs. ADOs stand for Accelerator Device Objects, see next section for details.

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-MOPHA047

CERN SECONDARY BEAM LINES SOFTWARE MIGRATION PROJECT

A. Gerbershagen†, B. Rae††, D. Banerjee, J. Bernhard, M. Brugger, N. Charitonidis, G.L. D'Alessandro[1], M.W.U. van Dijk, L. Gatignon, E. Montbarbon, I. Peres[2], M. Rosenthal

European Organization for Nuclear Research (CERN), Geneva, Switzerland
[1]also at Royal Holloway, University of London, Egham, UK
[2]also at Israel Institute for Technology (Technion), Haifa, Israel

Abstract

The Experimental Areas group of the CERN Engineering department operates a number of beam lines for fixed target experiments, irradiation facilities and test beams. The software currently used for the layout of the beam lines (BEATCH), beam optics (TRANSPORT), particle tracking (TURTLE) and muon halo calculation (HALO) has been developed in Fortran in the 1970's and 1980's and requires renovation in order to ensure long-term continuity. The on-going Software Migration Project transfers the beam line description to a set of newer commonly used software tools, such as MADX, FLUKA, G4Beamline, BDSIM and others. This contribution summarizes the goals and the scope of the project. It discusses the implementation of the beam lines in the new codes, their integration within the CERN layout database and the interfaces to the software codes used by other CERN groups. This includes the CERN secondary beam line control system CESAR, which is used for the control of the beam via setting of the magnets, collimators, filters etcetera, as well as readout of the beam instrumentation. The proposed interface is designed to allow a comparison between the measured beam parameters and the ones calculated with beam optics software.

INTRODUCTION

Beam Lines Managed by the Group

The Experimental Areas group of the Engineering department is responsible for the management of the fixed target experimental areas and test beams at CERN. This includes the so-called secondary beam lines, their associated facilities, beam line elements and infrastructure. The group is also ensuring the support in regards to the operation, design and physics studies of secondary beams and the technical and engineering support.

Figure 1 illustrates the CERN accelerator complex layout. The areas and beam lines managed by the EA group include three categories:

1. The **fixed target experiments**, including COMPASS, NA61, NA62, NA63, NA64, CLOUD, UA9 and several others. These are often designed to perform precision studies in the fields of the Standard Model (e.g. quantum chromodynamics) and Beyond Standard Model physics. They require stable beam conditions for prolonged periods of time.
2. The **irradiation facilities** such as HiRadMat, CHARM, IRRAD and GIF++, which are used for measurements of irradiation hardness of different types of materials, electronics or detectors.
3. The **test beams,** used for prototype tests and calibration of detectors, e.g. for LHC, linear colliders, space & balloon experiments. These are also utilized for outreach purposes (e.g. within the Beamlines for Schools programme). Users of the test beams usually require a large spectrum of beam conditions within a few days.

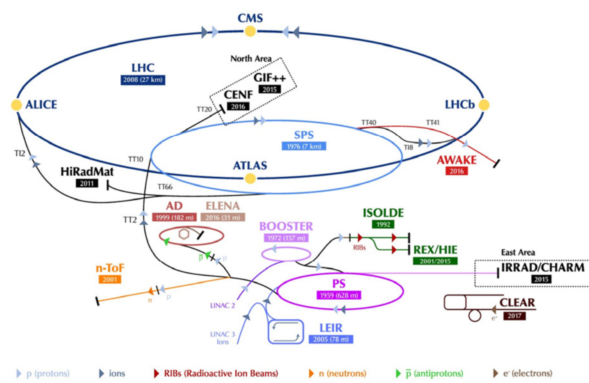


Figure 1: CERN accelerator complex scheme.

Requirements on the Software for the Beam Simulation and Beam Line Control

The group is responsible for a wide spectrum of beam lines mentioned above, which sets broad requirements on the software which is used to describe the beam lines, simulate beams, control beam line elements and read out beam instrumentation data. The ideal set of software has to be able to provide its users with the computation of the particle production at the targets and be able to simulate the beam optics properties (such as momentum selection, transverse beam parameters, beam intensity, particle type etc.) along the beam line. In addition it must provide input (if necessary, via a suitable interface) for other groups at CERN dealing with e.g. radiation protection, civil engineering, ventilation, survey and metrology and material survival studies. Finally, it has to be able to control the beam line elements with the aim that the envisaged beam properties can be achieved, measured and verified.

---
† a.ge@cern.ch
†† bastien.rae@cern.ch

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA048

THE IRRAD DATA MANAGER (IDM)*

B. Gkotse†,1,2, P. Jouvelot[1], G. Pezzullo[2], F. Ravotti[2]
[1] MINES ParisTech, PSL University, Paris, France
[2] Experimental Physics Department, CERN, Geneva, Switzerland

Abstract

The Proton Irradiation Facility (IRRAD) is a reference facility at CERN for characterizing detectors and other accelerator components against radiation. To ensure reliable facility operations and smooth experimental data handling, a new IRRAD Data Manager (IDM) web application has been developed and first used during the last facility run before the CERN Long Shutdown 2. Following best practices in User Experience design, IDM provides a user-friendly interface that allows both users to handle their samples' data and the facility operators to manage and coordinate the experiments more efficiently. Based on the latest web technologies such as Django, JQuery and Semantic UI, IDM is characterized by its minimalistic design and functional robustness. In this paper, we present the key features of IDM, our design choices and its overall software architecture. Moreover, we discuss scalability and portability opportunities for IDM in order to cope with the requirements of other irradiation facilities.

INTRODUCTION

The Proton Irradiation Facility (IRRAD), located in the East Area of the Proton Synchrotron (PS) accelerator complex at CERN, is an infrastructure dedicated to the qualification of materials, electronic systems and detector components for High-Energy Physics (HEP) experiments. A beam of protons with a momentum of 24 GeV/c accelerated by the PS is delivered to IRRAD, in pulses, and used for the irradiation experiments. IRRAD started its operations in 1999 [1]; it went through a major upgrade in 2014 to deal with the increasing demand for irradiation experiments by the HEP community, linked to the development of the High-Luminosity LHC [2]. Every year in IRRAD, hundreds of electronic and detector components, called "samples", are tested and qualified. Dosimetry is an important part of the operation; thus, during the irradiation experiments, aluminum dosimeters are placed together with the samples for defining the actual accumulated fluence [3] through gamma spectrometry. As a result, the registration, planning and follow-up of these experiments require the management of a considerable amount of data.

In the early years of IRRAD, a software application called Sample Manager was developed and used on a local computer for the registration of the samples to be irradiated. For the upgraded IRRAD facility, this system was outdated, running only on local computers with specific drivers, and could not be upgraded. Thus, these facts called for the development of a new web-based system with specific software requirements. These requirements were that the new system should be in line with the new IRRAD facility software needs, integrate into the modern software infrastructure of CERN, and provide better User Experience (UX). Since this application would be online, security was also a crucial issue to be considered. Moreover, after the irradiation experiments, the samples are radioactive and therefore this system had to be also compliant with the CERN traceability procedures and able to send/receive data from the official system used by CERN for the traceability of potential radioactive equipment (TREC) [4]. Last but not least, the scalability and portability of the system were also important aspects taken into consideration for future developments.

In this paper, we describe the overall development of IDM, the new IRRAD Data Manager web application [5], and its key functionalities. More specifically, in the second section, the design life cycle of IDM is described. In the third section, the software choices, the architecture and the deployment methods are presented. In the fourth section, we focus on the functionalities of IDM. Finally, in the fifth section, we conclude by providing some statistics regarding IDM operations during the IRRAD run of 2018, and presenting, as future work, our ideas on extending and adapting the IDM functionalities to other irradiation facilities.

DESIGN LIFE CYCLE

As stated in the previous section, User Experience was an important aspect for the new web application IDM. In particular, IDM should provide an intuitive interface targeting different user groups, such as physicists, engineers and technicians. Therefore, IDM design was based on UX universal principles in order to be intuitive and user-friendly [6]. In particular, we applied a User-Centered Design (UCD) approach, which allows for a better understanding and identification of the users' requirements and goals [7].

Research and Exploration

The first phase of a UCD focuses on the users' needs and tasks. For this reason, we used the CERN Irradiation Facilities online database developed in previous work in order to find information and details about existing irradiation facilities at CERN and worldwide [8]. Then, we interviewed coordinators, operators and users of IRRAD and of some of the irradiation facilities found. This allowed for a better understanding of the current state of the art regarding the used software tools. We thus gained a better insight of the users'

* This work has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement no. 654168 (AIDA-2020).
† Blerina.Gkotse@cern.ch

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA049

TEST-BENCH DESIGN FOR NEW BEAM INSTRUMENTATION ELECTRONICS AT CERN

M. Gonzalez-Berges, J. O. Robinson, M. Saccani, V. Schramm, M. A. Stachon
CERN, Geneva, Switzerland

Abstract

The Beam Instrumentation group has designed a new general-purpose VME acquisition board that will serve as the basis for the design of new instruments and will be used in the renovation of existing systems in the future. Around 1200 boards have been produced. They underwent validation, environmental stress screening and run-in tests to ensure their performance and long term reliability. This allowed to identify potential issues at an early stage and mitigate them, minimizing future interventions and downtime. A dedicated test-bench was designed to drive the tests and continuously monitor each board functionality. One board has more than 45 functions including memories, high speed serial links and a variety of diagnostics. The test-bench was fully integrated with the CERN Asset Management System to allow lifecycle management from the initial production phase. The data captured during these tests was stored and analyzed regularly to find sources of failures. This was the first time that such a complete test-bench was used. This paper presents all the details of the test-bench design and implementation.

INTRODUCTION

The CERN Beam Instrumentation (BI) group is in charge of designing, implementing and maintaining the instruments for the CERN accelerator complex. There are different types of instruments, including Beam Position Monitors (BPMs), Beam Loss Monitors (BLMs), Wire Scanners, etc. Currently there are several thousand instruments deployed in the different accelerators, some of which have been installed for several decades, with many using dedicated electronics. During the construction of the LHC the majority of instrumentation systems [1] were designed to use a common VME acquisition platform, the Digital Acquisition Board (DAB)-64x, which speeded-up developments, provided a standardised software interface and greatly improved maintainability. The largest instrumentation systems using this board are the BLM [2] and BPM [3] systems for the Large Hadron Collider (LHC), with over 4000 and 2500 devices acquired respectively.

Looking towards the future, the group intends to standardize the new electronics systems with the development of another general purpose VME board, the VFC-HD (VME FMC Carrier High Density) [4] Over 1200 units have been produced so far. This board is already being used in the development of new systems, such as the new BPM electronics for the CERN Super Proton Synchrotron, the renovation of the BLM systems in the LHC and its injectors, and is foreseen to be the default acquisition platform for several other new or upgraded instruments.

Since the VFC-HD will be used in many critical systems, the quality and reliability of the production are extremely important. This paper describes the design and implementation of a test-bench that was used to validate the VFC-HD production in terms of both functionality and reliability.

THE VME HIGH DENSITY FMC CARRIER BOARD (VFC-HD)

The VFC-HD (Figure 1) is a generic acquisition board for processing data either from an on-board FMC mezzanine, or via dedicated SFP fibre-optical inputs. Once processed in the on-board FPGA, this data can be stored in an on-board memory or read-out directly via Ethernet or through the VME bus to a host CPU. The high density FMC connector allows instrument specific or standard industrial modules to be hosted, providing great flexibility. The card also has on-board temperature and voltage sensors for self-diagnostics and provides inputs for timing and debugging links such as White Rabbit [5] or Beam Synchronous Time (BST) [6].

The VFC-HD board will be used for the main upgrades of many beam instrumentation systems at CERN in the coming years, offering increased performance, maintainability and reliability. Around 1200 boards have been produced and tested with the following main characteristics:

- Intel ArriaV GX Field Programmable Gate Array (FPGA)
- High Pin Count FPGA Mezzanine Card (FMC-HPC) slot, fully ANSI/VITA compliant:
  - 3 Low/High User Banks A and B (LA, HA & HB)
  - 10 gigabit lanes connected to FPGA transceivers
  - Programmable power supply voltage (Vadj).
- 6x Small Form-factor Pluggable Plus (SFP+) transceiver slots
- 4x GP (General purpose) SFP+ with up to 6.5Gbps
- 1x GP SFP+ with optional clock for White Rabbit Ethernet
- 1x GPSFP+ with optional clock data recovery (CDR) for Beam Synchronous Time (BST) reception
- 2x 4GB DDR3 memories
- Flexible clocking resources: adjustable and programmable Voltage Controlled Oscillators (VCO) and Phase Locked Loop (PLL).
- 30 single ended connections to VME64x P0 to support clock & trigger distribution in (custom) BI LHC VME crates
- 40 single ended (or 20 LVDS) connections to VME P2 available for rear transition modules
- 4 LEMO connector for general purpose input/output on the front panel

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA050

TOWARDS IMPROVED ACCESSIBILITY OF THE TANGO CONTROLS*

P. P. Goryl, M. Liszcz, S2Innovation, Kraków, Poland
A. Götz, R. Bourtembourg, ESRF, Grenoble, France
V. Hardion, MAX IV Laboratory, Lund, Sweden

Abstract

Tango Controls is successfully applied at more than 40 scientific institutions and industrial projects. These institutions do not only use the software but also actively participates to its development. The Tango Community raised several projects and activities to support collaboration as well as to make Tango Controls being easier to start with. Some of the projects are led by S2Innovation. These projects are: gathering and unifying of Tango Controls documentation, providing a device classes catalogue and preparation of a so-called TangoBox virtual machine. Status of the projects will be presented as well as their impact on the Tango Controls collaboration.

DOCUMENTATION

Initial State

Until recently, the Tango documentation consisted mainly of The Tango Book (a big pdf file provided with the source distribution) and many other documents in different formats (pdf, word, html, etc.). The book contained precious information but it was maintained by a single person and it was difficult to get external contributions. In 2015, it was decided to merge all the available documentation in a single location and to make it easy to contribute to. In order to encourage contributions the idea was to write documentation like code [1]. It was then decided to use Sphinx [2] and to convert and publish all the tango-controls related documentation on *readthedocs* [3].

Tools

The existing documentation was converted to *reStructuredText* [4] and imported into a GitHub repository. Keeping the documentation in a git repository allows the use of standard git flow procedure. The contributions are accepted as pull requests which are then reviewed before merging into a published branch. Thanks to Travis [5], the pull requests are validated prior to the merging. The converted documentation is next published with the use of the Sphinx toolset on *readthedocs*. The *readthedocs* service allows for providing multiple versions of the documentation, making them accessible also for the previous releases of Tango Controls.

New Documentation Structure

The book and various other documentation, including tools' manuals, HOW-TOs and tutorials were grouped into sections suitable for users with different roles and different levels of experience. Such a structure allows the reader to find the right section depending on whether she/he is an end-user, a developer or a system administrator. The intended audience is stored as a meta-data within each document.

Community Involvement

The original documentation of the Tango Controls was written by the community members. However, before it was made available in a public repository, only its authors were able to contribute to specific documents. Now, any community member can propose improvements and contribute to the new Tango Controls Documentation.

TANGO CONTROLS DEMONSTRATION VIRTUAL MACHINE

Goal

There are a lot of tools and libraries in the Tango Controls ecosystem. Although most of the tools including Tango Controls kernel have straightforward installation, newcomers may find it difficult to choose the configuration suitable for their needs. The availability of multitude of tools and applications for the Tango Controls makes it overwhelming for new users who would not like to invest much time into investigating technical details. The so-called TangoBox, a Tango Controls demonstration virtual machine image, provides a way to use the ecosystem without the need to spend time on investigation, selection and configuration of most of Tango Controls tools.

The additional goal is to provide a working configuration example. User may look how certain applications are deployed and configured. There are also shell scripts which serves as installation process documentation and can be referenced for custom deployment.

Contents

The TangoBox comes with pre-installed Tango core libraries and tools, including various desktop and web applications. These are available as launchers, directly from its desktop.

JLinac provides a GUI application backed by simulation device servers. It is a demo based on a real application from the ESRF control room. Jive and Astor let the user learn how Tango Controls administration works. Pogo, JDraw, TaurusDesigner allows starting developing for Tango Controls. There are examples of generic GUI applications and widgets provided in Java, Python and C++ (ATK, Taurus, Cumbia). TangoBox contains various device servers, including a Modbus device server. It is also possible to interact with the Tango Controls via web applications, like Waltz, JupyTango and Egiga.

* Work supported by Tango Controls Collaboration

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA051

TOWARDS SPECIFICATION OF TANGO V10*

P. P. Goryl, M. Liszcz, S2Innovation, Kraków, Poland
L. Pivetta, Elettra-Sincrotrone Trieste S.C.p.A., Basovizza, Italy
A. Götz, ESRF, Grenoble, France
V. Hardion, MAX IV Laboratory, Lund, Sweden

Abstract

More than 40 laboratories use Tango Controls as a framework for their control systems. During its 18 years of existence, Tango Controls has evolved and matured. The latest 9.3.3 release is regarded as the most stable and feature-rich version of the framework. However, it makes use of what is today considered as outdated CORBA technology which impacts all the stack, from the low-level transport protocol up to the client API and tools. The Tango Community decided to move forward and is preparing for so-called Tango Controls v10. Tango v10 is meant to be more a new implementation of the framework than a release of new features. The new implementation shall make the code easier to maintain and extendable as well as remove legacy technologies. At the same time, it shall keep the Tango Controls objective philosophy and allows the new implementation to coexist with the old one at the same laboratory. The first step in the process is to provide a formal specification of current concepts and protocol. This specification will be the base for the development and verification of new source code. Formal specification of Tango Controls and its purpose will be presented along with tools and methodologies used.

CONTEXT

After the first release in 2001, the Tango [1] control system framework has been continuously evolving and improving, triggered by the request of new features and the need for better performance.

Each new release has been developed guaranteeing full backward-compatibility. Currently, Tango 9 uses CORBA synchronous and asynchronous communication and ZeroMQ protocols for publish-subscribe data transport. Both CORBA and ZeroMQ are well documented protocols, with clean open-source implementation libraries that provide complete APIs. Thus, the need for a complete product specification in a formal language was not mandatory. The existing documentation, in the form of the Tango Controls manual and the API documentation, and the close cooperation of core developers has been sufficient to keep knowledge and compatibility between versions.

However, aging of certain technologies and libraries used by the framework together with the turnover in the developers team, led to a non optimal understanding of some Tango kernel implementation concepts within the growing community of the Tango Controls collaboration.

* Work supported by Tango Controls Collaboration

TANGO RFC

The Tango Request For Comment (RFC) is the name of the project which aims to define the most important aspects of Tango without being tied to any implementation. This is an attempt to separate what Tango adds compared to CORBA and ZeroMQ.

The idea of specifying Tango came after the Tango Kernel meeting held in 2019 at Solaris, Kraków, Poland (Fig. 1). Tango version 10 is a recurrent discussion in the community. Although everyone agrees to remove the obsolete technologies, the analysis of the code showed that Tango is tightly linked to CORBA, making it very hard to reuse the existing C++ implementation called "libtango9". A complete re-implementation is always a risk especially when the current stable version is heavily used and depended on by many sites.

Figure 1: The Tango Kernel group meeting in Kraków, Poland.

Two attempts were made to advance on v10. The first way was to propose another architecture with a different level of abstraction in a form of a plugin. The migration plan would consist of implementing a CORBA plugin first, to check the compatibility with the former "libtango9".

The second attempt was a prototype where CORBA was replaced with gRPC [2] done by the MAX IV Laboratory. It demonstrates the feasibility by making a PyTango* client communicating with a PyTango server. The prototype was breaking not only the backward compatibility with Tango v9 in terms of wire protocol but also changed some CORBA-specific behaviours.

The conclusion was that Tango needs a new implementation in any case which then raises the question "What makes Tango so Tango?". The answer is the Tango RFC project. The Tango RFC project got inspired by the ZeroMQ RFC [3] after investigating [4] how the other open source projects solve this problem.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA052

EVOLUTION BASED ON MICROTCA AND MRF TIMING SYSTEM

F. Gougnaud†, P. Bargueden, J.F. Denis, G. Ferrand, A. Gaget, F. Gohier, P. Guiho, T. Joannem, A. Lotodé, Y. Lussignol, Y. Mariette, V. Nadot, N. Solenne, Q. Bertrand
IRFU, CEA, Gif'sur-Yvette, France
I. Hoffman Moran, E. Reinfeld, I. Shmuely, SoreqNRC, Yavne, Israel

Abstract

For many years our Institute CEA Irfu has had a sound experience in VME and EPICS. For the accelerator projects Spiral2 at Ganil in Normandy (France) and IFMIF/LIPAc at JAEA/Rokkasho (Japan) the EPICS control systems were based on VME. For 5 years our Institute has been involved in several in-kind collabo-ration contracts with ESS. For the first contracts (ESS test stands, Source and LEBT controls) ESS recommended we use VME based solutions on IOxOS boards. Our close collaboration with ESS, their support and the requirements for new projects have led us to develop a standardized hardware and software platform called Irfu EPICS Environment based on microTCA.4 and MRF timing system. This paper describes the advantages of the combination of these recent technologies and the local control system architectures in progress for the SARAF project.

INTRODUCTION

The CEA Irfu Institute started to use VME in 1983 to upgrade the command control of the Linear Accelerator of Saclay, called ALS, located at Orme des Merisiers very close to Saclay. The associated VME real time system was Versados then VxWorks from 1989. VxWorks was chosen for the accelerator prototype MACSE based on superconductive cavities. In 1993 we came into the EPICS community with the platform VME/VxWorks that we used for different projects and collaborations. For instance, we used this EPICS solution for the collaboration with the Tesla Test Facility at DESY (Germany), for Spiral2 [1] at Caen in Normandy (France) and for the IFMIF LIPAc project at Rokkasho (Japan).

The migration to MTCA.4 was decided in summer 2018. This paper presents the context and the design for this MTCA.4 platform and the application in the SARAF project introduced in [2, 3, 4].

CONTEXT

We started a collaboration with ESS (European Spallation Source) in 2014 essentially based on VME/Linux, IOxOS boards and Siemens 1500 PLC solutions. We were in charge of the control system of the ESS source and LEBT at Catania (Sicily, Italy) and several RF test stands for ESS [5]. The IOxOS Company had already started its migration to MTCA.4 and their acquisition boards presented a definite advantage.

The FPGA Mezzanine Card (FMC) acquisition boards permitted us to upgrade to MTCA.4 by saving the FMC and the cabling.

We noticed that our usual VME companies neglected the VME support. We also felt that IOxOS was becoming more involved in its new orientations, MTCA.4, than in VME. This situation was new.

ESS decided to migrate to MTCA.4 very early on [6]. We could observe their progress on MTCA.4 with the IOxOS boards and the advantages of using both MTCA.4 and the MRF timing system. ESS ICS encouraged us to do such a migration. Our partner SNRC accepted CEA's recommendation to migrate to MTCA.4 for the SARAF control system. Therefore, all together we decided this migration in summer 2018. The CEA team updated and standardized the IRFU EPICS Environment [7] with MTCA.4 solutions based on IOxOS, MRF boards and ESS ICS EPICS drivers to start. See Figure 1. Now, we are beginning to use our own IOxOS EPICS drivers for more flexibility.

MTCA.4 PLATFORM

Generalities

MTCA is an electronic framework for analogue and digital signal processing. MTCA.4 was released as an official standard by the PCI Industrial Manufacturers Group (PICMG) in 2011 and is strongly supported by the xTCA physics groups and electronics manufacturers.

Advantages regarding VME are a larger bandwidth, additional timing and trigger signals in the backplane, introduction of Rear Transition Modules (RTM) allowing to connect cables from the rear and to swap IO and processing boards without the need to remove the rear cables. The RTM can provide low and high speed analogue signals, digital signals, clock signals and management signals.

Crates

We plan to standardize 2 MTCA.4 crate types with several boards. Our first choice is the crate NATIVE-R2, a very compact one.

MCH and CPU boards

The NAT-MCH-PHYS80 offers an 80-port PCIe Gen3 switch and can be combined with the Rear Transition Module with quad-core Intel® Xeon® E3 CPU NAT-MCH-RTM-COMex-E3. Currently, the new PCIe hub module turns the NAT-MCH-PHYS80 into a powerful single-slot solution for management and switching that is available for MTCA.4.

† francoise.gougnaud@cea.fr

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA053

STATUS OF CONTROL AND SYNCHRONIZATION SYSTEMS DEVELOPMENT AT INSTITUTE OF ELECTRONIC SYSTEMS*

M. Grzegrzółka†, A. Abramowicz, A. Ciszewska, K. Czuba, B. Gąsowski, P. Jatczak, M. Kalisiak, T. Leśniak, M. Lipiński, T. Owczarek, R. Papis, I. Rutkowski, M. Sawicka, K. Sąpór, D. Sikora, M. Urbański, Ł. Zembala, M. Żukociński, ISE, Warsaw University of Technology, Poland

Abstract

Institute of Electronic Systems (ISE) at Warsaw University of Technology designs, builds and installs control and synchronization systems for several accelerator facilities. In recent years ISE together with the Deutsches Elektronen-Synchrotron (DESY) team created the RF synchronization system for the European XFEL in Hamburg. ISE is a key partner in several other projects for DESY flagship facilities. The group participated in development of the MTCA.4 standard and designed a family of components for the MTCA.4-based LLRF control system. Currently ISE contributes to the development of the Master Oscillators for XFEL and FLASH, and phase reference distribution system for SINBAD. Since 2016 ISE is an in-kind partner for the European Spallation Source (ESS), working on the phase reference line for the ESS linac, components for 704.42 MHz LLRF control system, including a MTCA.4-based LO signal generation module and the Cavity Simulator. In 2019 ISE became one of the co-founders of the Polish Free-Electron Laser (PolFel) located in the National Centre for Nuclear Research in Świerk. The overview of the recent projects for large physics experiments ongoing at ISE is presented.

INTRODUCTION

Institute of Electronic Systems (ISE) is part of Warsaw University of Technology - one of the leading technical universities in Poland. ISE develops state of the art electronic systems. Among others, ISE collaborates with Deutsches Electronen Synchrotron (DESY) and European Spallation Source (ESS), working on control and synchronization systems for linear accelerators. This contribution presents the overview and status of the most significant on-going projects under development at ISE.

XFEL MASTER OSCILLATOR

In collaboration with DESY, ISE was developing Master Oscillator for European Xray Free Electron Laser (E-XFEL) [1]. ISE was responsible for the component selection, prototyping, design of system modules, installation, and commissioning of the frequency synthesis system. The system currently provides ultra-low-noise 1.3 GHz reference signal at the E-XFEL facility, offering following parameters: < 16 fs rms jitter (10 Hz to 1 MHz bandwidth), < 10^{-12} long-term frequency stability, and +41dBm of output power.

Currently, ISE works on a novel real-time redundancy subsystem [2] (Fig. 1) whose aim is to further improve overall reliability of the Master Oscillator. This solution will provide continuous reference signal even in case of almost any potential failures in the system. It is accomplished through continuous monitoring of signal's parameters, low-latency detection and switching, and energy storage in a high-Q filter.

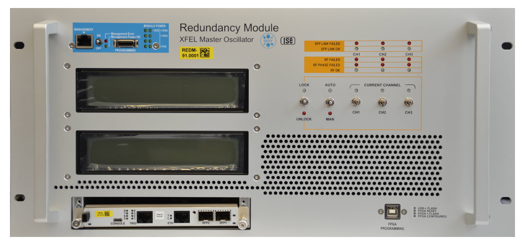


Figure 1: Photo of the XFEL Master Oscillator Redundancy Controller.

UPGRADE OF MASTER OSCILLATOR FOR FLASH

ISE is also involved in development of a new Master Oscillator system for FLASH facility. Contribution of ISE is similar to E-XFEL's case and include preparation of system concept, design and assembly of system modules, as well as participation in installation and commissioning activities.

Main motivation for the efforts is to refresh the dated system and improve the performance. Architecture of the new system will follow the previously developed E-XFEL's Master Oscillator and surpass its performance (expected rms jitter below 10 fs rms). Current activities are concentrated on working out the detailed concept and selection of the most critical components.

SINBAD SYNCHRONIZATION SYSTEM

SINBAD (Short Innovative Bunches and Accelerators at DESY) is an accelerator research facility at DESY. It will be a host for many experiments, related to ultra-short electron bunches and high gradient acceleration techniques.

To ensure proper and stable work of ARES - Accelerator Research Experiment at SINBAD - a high frequency phase reference distribution system had to be designed, manufactured, installed, and tested [3]. This includes temporary distribution system that will be later upgraded to its final form, utilizing active interferometric phase stabilization [4, 5], optionally supported by thermal stabilization.

* Work supported by Polish Ministry of Science and Higher Education, decision number DIR/WK/2016/06 and DIR/WK/2016/03.
† M.Grzegrzolka@elka.pw.edu.pl

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA058

Lua–LANGUAGE–BASED DATA ACQUISITION PROCESSING EPICS SUBSCRIPTION FILTERS*

J. O. Hill, Los Alamos National Laboratory, Los Alamos, USA

Abstract

A previous paper described an upgrade to EPICS enabling client side tools at LANSCE to receive subscription updates filtered selectively to match a logical configuration of LANSCE beam gates, as specified dynamically by control room application programs. This update paper will examine evolving enhancements enabling Lua–language based data acquisition processing subscription update filters, specified by snippets of Lua-language source-code embedded within the EPICS channel-name's postfix. We will discuss the generalized utility of this approach across a wide range of data acquisition applications, projects, and platforms; the performance and robustness of our production implementation; and our operational experience with the software at LANSCE.

LANSCE

The Los Alamos Neutron Science Centre (LANSCE) was originally designed to be a versatile machine for medium-energy (800 MeV) nuclear physics experiments. It had three injectors and could simultaneously accelerate positive hydrogen ions (H+), negative hydrogen ions (H-) and polarized negative hydrogen ions (P-). These three beams could all have different intensities, duty factors, and even different energies – depending on experimental needs. Today LANSCE can simultaneously generate four H- beam types and two H+ beam types. It services several experimental facilities including a proton storage ring, a low-intensity neutron research facility, proton radiography, ultra-cold neutron source, isotope production, and a proposed materials test-station.

Developed during the infancy of computer control systems, the architecture of the original LANSCE control system (LCS) had elements of data-acquisition along with elements of traditional computer control system architectures. One of the more interesting and useful features of the legacy LCS system was its ability to do "Timed" and "Flavoured" reads. A "Timed Read" sampled typically relative to the leading or trailing edge of a beam gate. A "Flavoured Read" refers to the ability to schedule the read for a particular machine cycle containing a desired configuration of beam-gates. A “Flavour” is configured by specifying for each of 13 timing system beam gates whether it must be present, must be absent, or is not relevant. Therefore, there can be up to 3^{13} possible flavour combinations. In practice, only a few (meaningful) combinations of the six beam destination beam-gates along with a handful of diagnostic-trigger-gates are used, but more esoteric flavours for diagnostic and experimental purposes are considered to be essential.

At LANSCE the 120 slot super-cycle of regularly scheduled beam gates repeats at a 1 Hz rate, but there is also a cycle-stealing scheduling anomaly allowing multiple incompatible beam species to be scheduled in the same cycle, and if more than one of them is currently enabled *only* the gates for the highest-priority species will be emitted during that cycle. This allows, for example, a beam species associated with one-shot enabled proton-radiography, to modally assume a cycle assigned also to the beam species used for high repetition rate production-oriented neutron experiments.

A pivotal requirement, imposing unique constraints on implementation of the lowest levels of the real-time embedded system-software, is that configuration of "Timed" and "Flavoured" data acquisition cycles must be dynamically selected by application programs at the situational compulsion of LANSCE operations and tuning staff.

EPICS CONTROL SYSTEM

An EPICS Input Output Controller (IOC) is configured with Database Records implementing function blocks for various purposes including logical IO, numerical calculation, and ordered sequencing. The EPICS Channel Access (CA) internet communication subsystem is based on a publish-and-subscribe communication model where clients subscribe for updates, servers publish updates to subscribed clients, and records post state change events to servers. A channel is a virtual communication link between a client side application program and a process variable (PV) exported by a service. EPICS clients issue asynchronous read, write, and subscribe requests to the process variable in the service. Clients are notified when the network connectivity of a channel changes.

An essential tenet of the original EPICS design was that regular periodic processing of EPICS Records isn't disturbed by influences outside of an IOC thereby guaranteeing time-periodic algorithms such as PID loops, and time-deterministic response by EPICS Records to state changes detected within sensors, are properly maintained. The load induced by Record Processing is constant and predictable. In contrast, externally induced load from network clients is variable. Therefore record processing executes at higher priorities, and CA network services execute at relatively lower priorities. Multiple event-queues, containing subscription updates, communicate in between record-processing, and the server's per-client dedicated threads.

LUA – A BRIEF INTRODUCTION

“Lua”, a language designed specifically to be embeddable within other software, was created in 1993 by members of the Computer Graphics Technology Group (Tecgraf) at the Pontifical Catholic University of Rio de Janeiro, in Bra-

* Work supported by US Department of Energy under contract DE-AC52-06NA25396.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA059

ULTRA-HIGH PRECISION TIMING SYSTEM FOR THE CEA-LASER MEGAJOULE

S. Hocquet†, N. Bazoge, P. Hours, D. Monnier-Bourdin, Greenfield Technology, Bègles, France
T. Falgon, T. Somerlinck, CEA CESTA, Le Barp, France

Abstract

High power laser such as the Laser MegaJoule (LMJ) or National Ignition Facility (NIF) requires different types of trigger precision to synchronize all the laser beams, plasma diagnostics and generate fiducials. Greenfield Technology, which designs and produces picosecond delay generators and timing systems for about 20 years, has been hired by CEA to develop new products to meet the LMJ requirements. More than 2000 triggers are about to be set to control and synchronize all of the 176 laser beams on the target with a precision better than 40ps RMS. Among these triggers, Greenfield Technology's GFT1012 is a 2 or 4-channels delay generator challenging ultra-high performances: an ultra-low jitter between 2 slaves of 4ps RMS and a peak-to-peak thermal drift over 1 week lower than 6 ps due to a thermal control of the most sensitive parts (thermal drift is below 1ps/°C) and specific developments for clock management and restitution. Ongoing investigation should bring the jitter close to 2 ps RMS between 2 slaves.

INTRODUCTION

The Laser MegaJoule (or LMJ), is a high-power laser facility based in France dedicated to study high energy density physics and more particularly inertial confinement fusion [1]. To control and set up all of the 176 laser beams and laser and plasma diagnostics, the LMJ timing system must meet harsh requirements [2]. Recent evolution of the trigger requirements has split the timing system specification into 2 performance categories: standard and high precision timing (SPT/HPT). Requirements are sum up on Table 1 below.

Table 1: LMJ Timing System Requirements

	Range	Jitter (RMS)	Thermal drift (peak-to-peak, over 1 week)	Qty.
SPT Triggers	±1s	><100ps	<2ns	~2000
HPT Triggers	± 50µs	<5ps	<10ps	~500

Greenfield Technology (GFTy) is a company specialized in timing systems for 20 years and has been mandated to give a global solution. To achieve this task with cost control, GFTy has designed a specific timing system for the LMJ from the command-control software to the optical splitters in addition of the master-slave architecture (see Figure 1 for details).

† s.hocquet@greenfieldtechnology.com

ULTRA-HIGH PRECISION SLAVE GFT1012

To achieve ultra-high precision triggers, GFTy has designed the GFT1012 which is a 2 or 4-channels slave. GFT1012 conception has been focused on two major axes:

- Minimize noise to minimize temporal jitter
- Thermal control to maximize stability over time

Noise Control

To be compatible with the rest of the timing system architecture, GFT1012 is linked to the optical network made of a master clock and two stages of optical splitters. GFT1012 is able to decode the same optical signal made of a 1B/2B message at 155,52MHz. However, to minimize instability due to the optical reception sensibility, GFT1012 incoming optical power has been maximized by choosing 1-to-4 optical splitter as second stage of the optical network (instead of 1-to-16 optical splitter for GFT1018, standard slaves also used on LMJ).

Clock management is essential in timing system, that is why a clock and data recovery (CDR) function is implemented after optical reception. This function can be achieved by a PLL (phase-locked loop) such as ADN2817 and ADN2814 from Analog Devices but these components are designed for higher frequency systems (up to 2.7Gb/s for ADN2817 and 675 Mb/s for ADN2814) and are no more dedicated for 155,52MHz timing system frequency. Table 2 shows In/Out clock jitter for different frequencies: at 155 MHz, the jitter is way above 5ps RMS.

Table 2: In/out RMS Jitter for ADN2814 and ADN2817

Frequency	ADN2817	ADN2814
155 MHz	15 ps	8.2 ps
311 MHz	11.5 ps	3.2 ps
622 MHz	5.7 ps	2.7 ps
1.2GHz	3.3 ps	-

Furthermore, GFTy tests on ADN2814 shows that this PLL was not as stable as expected over time and 8ps variation in few minutes have been measured (see Figure 2).

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA062

THE PERSONNEL SAFETY SYSTEM OF ELI-ALPS*

F. Horvath†, L.J. Fulop, Z. Héjja, Sz. Horváth, G. Kávai, T. Kecskés, I. Kiss, V. Kurusa, K. Untener
ELI-ALPS, Szeged, Hungary

Abstract

ELI-ALPS will be the first large-scale attosecond facility accessible to the international scientific community and its user groups. The facility-wide Personnel Safety System (PSS) has been successfully developed and commissioned for the majority of the laboratories. The system has three major goals.

First, it provides safe and automatic sensing and interlocking engineering measures as well as monitoring and controlling interfaces for all laboratories in Building A: emergency stop buttons, interlock and enabling signals, door and roller blind sensors, and entrance control.

Second, it integrates and monitors the research technology equipment delivered by external parties as black-box systems (all laser systems, and some others). Third, it includes the PSS subsystems of research technology equipment developed on site by in-house and external experts (some of the secondary sources).

The gradual development of the system is based on the relevant standards and best practices of functional safety as well as on an iterative and systematic lifecycle incorporating several internal and external reviews. The system is implemented with an easily maintainable network of safety PLCs.

INTRODUCTION

ELI's long-term objective is to become the world's leading user facility utilizing the power of state of the art lasers for the advancement of science and applications in many areas of societal relevance [1]. The main objective of the ELI Attosecond Light Pulse Source (ELI-ALPS) pillar is the establishment of a unique attosecond facility that provides ultrashort light pulses with high repetition rates.

The Personnel Safety System (abbreviated as PSS) of ELI-ALPS is responsible to prevent or reduce any laser related harm or risk to the persons working in the facility. This system is one of the several protection layers aiming to prevent persons from getting life-threatening exposure, injury or damage potentially caused by laser radiation.

The PSS provides safe and automatic sensing and interlocking engineering measures as well as monitoring and controlling interfaces for all non-ionizing Laser- and Low-shielded Target Areas. The system consists of highly reliable sensors and actuators (collectively called field devices), and logic solvers which maintains the data link between the field devices.

* The ELI-ALPS project (GINOP-2.3.6-15-2015-00001) is supported by the European Union and co-financed by the European Regional Development Fund
† ferenc.horvath@eli-alps.hu

The PSS is complemented by other engineering controls: entrance control, mobile laser shielding walls, curtains, organizational and administrative controls (e.g. safety signs, safety trainings) and personal protective equipment (e.g. laser safety goggles).

The PSS communicates and integrates several other safety systems delivered by external parties: the PSS subsystem of the laser source systems, the fire safety system, the Safety Access Control System (Saf-ACS) as well as the Radiation Monitoring System in the future. For the block diagram of the PSS, see Fig. 1.

The system has been developed by following the relevant international standards and recommendations with particular reference to the IEC61511, as well as considering the experiences of similar projects [2-5].

SAFETY REQUIREMENTS

The completely independent Environmental, Health and Safety Division systematically surveyed and still continuously monitors the hazards and risks of the facility with a special focus on research technology related ones. The engineers and scientists are heavily and regularly involved in this work in order to identify and mitigate as risks as possible.

These hazard and risk assessments target all kinds of hazards from fire, electrical to laser radiation and so on. The results of the assessments are the allocation of safety functions to different kind of protection layers, with a special focus on laser safety and safety instrumented systems: requirements against external suppliers (e.g. companies delivering laser systems) as well as requirements against the facility-wide PSS. An important measure related to the PSS systems is the regular training (including exams) of all personnel working in the lab, especially the area managers responsible for the proper and safe operation of the respective laboratories.

RESEARCH TECHNOLOGY EQUIPMENT

The general architecture of ELI-ALPS' research technology includes three major parts. *Laser Source (LaSo)* systems provide the necessary laser beam to the experiments. They arrived as black box systems satisfying the prescribed PSS requirements (internal states, interface). *Beam transport systems* transfer the laser beam to the Secondary Sources. Developed in-house as a white box system. *Secondary Source (SeSo) Systems* are the equipment in which the attosecond pulses are generated. Developed in a collaboration of domain-experts as grey box systems from PSS point of view.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA063

TOWARDS A COMMON RELIABILITY & AVAILABILITY INFORMATION SYSTEM FOR PARTICLE ACCELERATOR FACILITIES

K. Höppner*, Th. Haberer, K. Pasic, A. Peters
HIT, Heidelberg Ion Beam Therapy Center,
University Hospital Heidelberg, Germany
H. Humer, Austrian Institute of Technology, Vienna, Austria
J. Gutleber, A. Niemi, CERN, Geneva, Switzerland

Abstract

Failure event and maintenance record based data collection systems have a long tradition in industry. Today, the particle accelerator community does not possess a common platform that permits storing and sharing reliability and availability information in an efficient way. In large accelerator facilities used for fundamental physics research, each machine is unique, the scientific culture, work organisation, and management structures are often incompatible with a streamlined industrial approach. Other accelerator facilities enter the area of industrial process improvement, like medical accelerators due to legal requirements and constraints. The Heidelberg Ion Beam Therapy Center is building up a system for reliability and availability analysis, exploring the technical and organisational requirements for a community-wide information system on accelerator system and component reliability and availability. This initiative is part of the EU H2020 project ARIES, started in May 2017. We will present the technical scope of the system that is supposed to access and obtain specific reliability statistical information in ways not compromising the information suppliers and system producers.

THE HIT MEDICAL ACCELERATOR

The heavy ion accelerator at HIT is used for raster-scanning radiation of cancer patients (cf. [1, 2] for an overview) with different types of ions from three sources [3] in several treatment rooms, two with horizontal fixed beam exit (operational since 2009) and the heavy ion gantry with rotatable beam exit (operational since 2012 [4]), and a beam exit for experiments (see Fig. 1). Each combination of source and destination may be used for medical treatment, represented within the Accelerator Control System (ACS) by the so-called *virtual accelerator* number. A radiation plan consists of a series of beam pulses chosen from a catalogue of 255 different energy values (88–430 MeV/u for carbon, 48–220 MeV/u for protons), 6 focus sizes, 15 intensity values (2×10^6–5×10^8 particles per second for carbon, 8×10^7–2×10^{10} pps for protons), and 36 exit angles in case of the gantry. These tuples of beam settings are named the MEFI combinations. Both the virtual accelerator as the MEFI combination may be changed from beam pulse to beam pulse (multi-plexed operation).

* klaus.hoeppner@med.uni-heidelberg.de

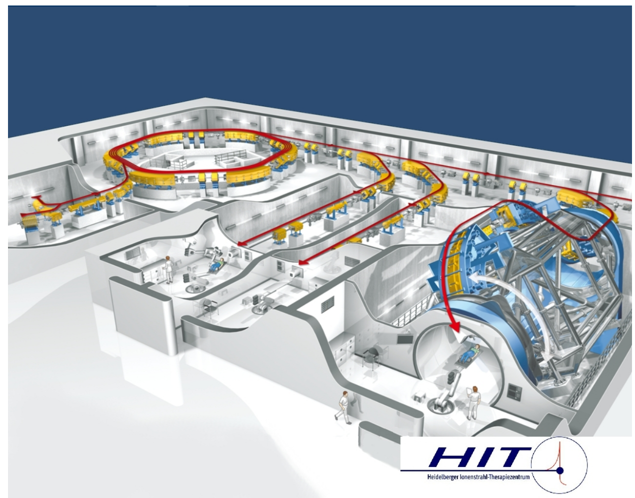


Figure 1: HIT accelerator facility with ion sources, linear accelerator, synchroton, two horizontal beam exits and gantry for medical treatment. The experimental area is not shown.

THE ARIES PROJECT

The ARIES (Accelerator Research and Innovation for European Science and Society) project [5], running since May 2017 until April 2021 and co-funded by the European Commission under its Horizon 2020 programme, brings together a consortium of 42 beneficiaries from 18 countries: accelerator laboratories, technology institutes, universities and industrial partners to jointly address common challenges for the benefit of a number of projects and infrastructures in high-energy physics, as well as in photon and neutron science. By promoting complementary expertise, cross-disciplinary cooperation and a wider sharing of knowledge and technologies throughout academia and with industry, ARIES will significantly enhance the science and technology base for European accelerators.

The main goals of ARIES are linked to developing and demonstrating novel concepts and further improving existing accelerator technologies, providing European researchers and industry with access to top-class accelerator research and test infrastructures, enlarging and further integrating the accelerator community in Europe, and developing a joint strategy towards sustainable accelerator science and technology.

ARIES comprises a strong industrial participation with 8 industrial partners, including three small and medium en-

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA064

AN OFF-MOMENTUM BEAM LOSS FEEDBACK CONTROLLER AND GRAPHICAL USER INTERFACE FOR THE LHC

B. Salvachua*, D. Alves, G. Azzopardi, S. Jackson,
D. Mirarchi, M. Pojer, CERN, Geneva, Switzerland
G. Valentino, University of Malta, Msida, Malta

Abstract

During LHC operation, a campaign to validate the configuration of the LHC collimation system is conducted every few months. This is performed by means of loss maps, where specific beam losses are voluntarily generated with the resulting loss patterns compared to expectations. The LHC collimators have to protect the machine from both betatron and off-momentum losses. In order to validate the off-momentum protection, beam losses are generated by shifting the RF frequency using a low intensity beam. This is a delicate process that, in the past, often led to the beam being dumped due to excessive losses. To avoid this, a feedback system based on the 100 Hz data stream from the LHC Beam Loss system has been implemented. When given a target RF frequency, the feedback system approaches this frequency in steps while monitoring the losses until the selected loss pattern conditions are reached, so avoiding the excessive losses that lead to a beam dump. This paper will describe the LHC off-momentum beam loss feedback system and the results achieved.

INTRODUCTION

In the Large Hadron Collider two counter-rotating beams collide in 4 experiments at a centre-of-mass energy of up to 14 TeV [1, 2]. In order to keep the beams on their desired trajectory, super-conducting dipoles with a nominal field of 8.33 T are used. A collimation system protects the machine against beam losses. The loss of a very small fraction of the circulating beam is a concern because this may induce a magnet quench, where a superconducting magnet enters the normal conducting state, initiating a beam dump or, in extreme cases even damaging machine components.

More than 100 collimators are located in the LHC ring in order to concentrate beam losses in these dedicated areas. The main betatron cleaning occurs in LHC Insertion Region 7 (IR7) and off-momentum particles are cleaned in LHC Insertion Region 3 (IR3). A collimation hierarchy is defined with robust primary collimators with smaller gaps catching primary beam losses, secondary collimators more retracted than the primaries intercepting secondary showers, and absorbers to intercept and absorb the remaining showers. At both sides of each colliding interaction point, tertiary collimators provide local protection to the strong focusing triplet magnets while additional absorbers are used to intercept the physics collision debris.

* belen.salvachua@cern.ch

The hierarchy of the collimation system has to be validated and verified at least every three months during machine operation in order to ensure that any possible drift does not affect the beam cleaning efficiency. The validation is done by means of beam loss maps. Betatron cleaning is validated by transversely exciting a low intensity beam with white noise in a controlled way, generating slow beam losses that are used to verify the collimation hierarchy in IR7 and the residual leakage into the cold magnets. Off-momentum cleaning is verified by shifting the Radio-Frequency (RF) by a fixed amount resulting in an orbit shift for both beams. This shift is more pronounced were the machine dispersion is highest, leading to higher beam losses in the off-momentum collimation system in IR3.

The validation of the LHC configuration through loss maps is performed at every stage in the LHC cycle: at injection energy with injection protection collimators inserted and retracted, at the end of the energy ramp, at the end of the beam squeeze, when the beams are colliding, as well as for special experimental configurations. Off-momentum validation [3] of a single configuration requires two dedicated fills for each configuration (to probe particles with both higher and lower than nominal momentum) where the beams are completely lost on the IR3 collimators. A minimum of 5 to 8 configurations need to be validated in this way. For the betatron cleaning a procedure was developed that allows loss maps to be performed without dumping the beam. However this was not initially the case for off-momentum loss maps, whose time significantly impacted the machine availability for physics. In order to perform the off-momentum loss maps in a controlled way without losing the beam a feedback based on the signal from beam loss monitors located downstream from each collimator was developed, together with a dedicated graphical user interface.

BEAM LOSS MONITORING DATA

The LHC is equipped with more than 3 600 ionisation chambers [4] distributed along the ring and covering the regions were high losses are expected. Downstream from each collimator there is at least one such ionisation chamber.

The beam loss monitors detect secondary shower particles the number of which, for a given energy, are proportional to the number of initiating protons. The signal is read-out in 12 different running sums (RS01 - RS12) ranging from 40 µs to 83.9 s calculated every 40 µs. A Java concentrator publishes the BLM data every 1 Hz for each of the running sums. A special fast data stream of RS06 (integration time of 10.06 ms) is published via FESA (Front-End Software

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA066

ELECTRONICS FOR LCLS-II BEAM CONTAINMENT SYSTEM SHUT-OFF*

R. Kadyrov†, E. Chin, C. Clarke, E. Rodriguez, F. Tao, D. Brown, M. Petree
SLAC National Accelerator Laboratory, Menlo Park, California, USA

Abstract

LCLS-II is a new xFEL facility under construction at SLAC National Accelerator Laboratory. Its super-conducting electron linac is able to produce up to 1.2 MW of beam power. The existing normal conducting LCLS linac can operate concurrently in shared accelerator housing. A Beam Containment System (BCS) is employed to limit the beam power and prevent excessive radiation in case of electron beam loss or FEL breach. Fast and slow shut-off paths are designed for devices with different response requirements. The system is required to shut-off the beam within 200 µs for some of the fast sensors. The fast path is based on custom electronic designs, and the slow path leverages industrial safety-rated PLC hardware. The system spans 4 km of LCLS-II and combines inputs from about 150 sensors of different complexity. The architecture is based on multiple levels starting with summing sensor inputs locally and to converting them into permits for the shut-off devices. Each level is implemented redundantly. Automated and manual tests at all levels are implemented in the system. System architecture, electronics design and cable plant challenges are presented below.

INTRODUCTION

There have been multiple changes to the BCS architecture since it was presented to the community last time [1]. The hardwired shut-off for devices with fast response requirements has grown into the "fast shutoff path" subsystem comprised of custom electronic units with copper and fiber interconnections. The role of the PLC has been expanded to supervising, testing the fast path electronics and shut-off devices, and centralized bypass handling. Not all initiatives for the PLC implementation presented previously in [1] have been implemented, and the final design closely follows de-facto standards of safety systems implementation at SLAC with redundant implementation and diversity in programming.

SYSTEM ARCHITECTURE

The BCS sensors are divided into two categories by the shut-off time required. The first category includes sensors with response requirements of 500 ms or more that are connected to the "slow path" which is based on distributed Siemens failsafe PLC I/O and safety PLC CPUs. Magnet Current Monitors (MCMs), vacuum breach interlock system, cooling water differential pressure switches, bremsstrahlung radiation monitors and position indicators from beam stoppers provide inputs to I/O in the field, PLC CPU processes readings from the field I/O, evaluates interlock conditions, and issues slow path permits for the Shut-off chassis (SOC).

All sensors with shut-off time requirements below 500 ms are connected to the "fast path". The following sensors use the fast shut-off path: Average Current Monitor (ACM), Beam Loss Monitors (BLMs), and the Photon Absorber Burn Through Monitor (BTM). Faults from sensor electronics are summed in digital summary chassis (DSC) and passed to the SOC for permit generation. The subsystem with the fastest response required is the BLM. If beam is not mitigated in 200 µs after the beam loss, thermal damage to personnel protection equipment can occur at beam power above 250 kW.

Both paths are shown in Figure 1. Permits from the shut-off chassis are passed to the interface chassis that translates the signal levels and logic to be compatible with the shut-off devices.

A typical BCS installation combines sensors within reach of PLC I/O and DSC which corresponds to three to five 100-m sectors of the accelerator structure. Installation boundaries are defined based on signal density in each area and optimal cable lengths. Installation locations are selected such that the permitting signals are passed in the upstream direction towards primary shut-off devices to reduce the response time.

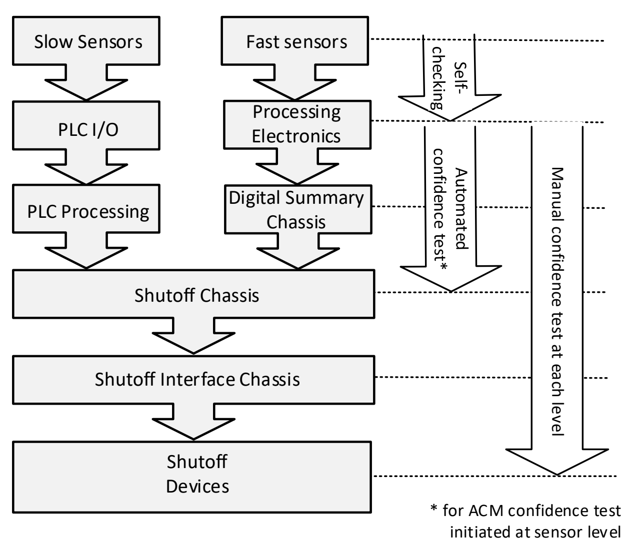


Figure 1: System fast and slow shut-off paths. Fast path diagnostics coverage.

* SLAC is supported by the U.S. Department of Energy under contract DE-AC02-76SF00515
† rkadyrov@slac.stanford.edu

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA067

NEW INJECTION INFORMATION ARCHIVER FOR SuperKEKB

H. Kaji*, High Energy Accelerator Research Organization, Ibaraki, Japan

Abstract

We developed the injection information archiver which records a variety of injection-related data for the SuperKEKB collider. The recorded data are utilized to understand the beam quality and the beam background at the interaction point. Even though the first version of the injection information archiver is working well, it cannot be operated with the maximum injection rate of 50 Hz. Therefore, we develop a new system that collects the injection data via the dedicated optical network. It is realized by expanding the Bucket Selection system. We installed one node which collects data from the network and records them into the archiver. The system worked without any problem and collected injection-related data in the 2019 spring run.

INTRODUCTION

The efficient and stable beam injection is important for the SuperKEKB collider [1]. It operates high current beams, like 2.6 A (positron ring) and 3.6 A (electron ring), to archive the large luminosity.

The detailed understandings of the injection operation and injected beam-pulse are important to discuss the beam quality, collision quality, and the beam background at the interaction point. For this purpose, the injection information archiver system is developed in 2018 [2]. It is successfully operated and the interesting results are evaluated from the recorded data.

However, this system has the issue that data acquisition can not be implemented in the injection rate >25 Hz. And such a high rate injection will be carried out in the future large current operation at SuperKEKB.

We developed the new injection information archiver system which can record the pulse-by-pulse data in 50 Hz. This system utilized the distributed shared memory network to collect the necessary injection related information.

INJECTION INFORMATION

The injection information archiver collects a variety of data. For example, the operation parameters of injector linac (LINAC) are collected and recorded. Besides, the following data are collected with the software process developed for this purpose.

Injected Current

The injected current, ΔI, is defined as follows:

$$\Delta I = I^{\mathrm{aft}}_{\mathrm{bunch}} - I^{\mathrm{bef}}_{\mathrm{bunch}}, \tag{1}$$

where $I^{\mathrm{bef}}_{\mathrm{bunch}}$ and $I^{\mathrm{aft}}_{\mathrm{bunch}}$ are the bunch current of RF-bucket which LINAC injects beam-pulse (Injection-bucket). It is recorded twice, before injection and after injection. The data of the appropriate channel of the bunch current monitor (BCM) is collected and recorded.

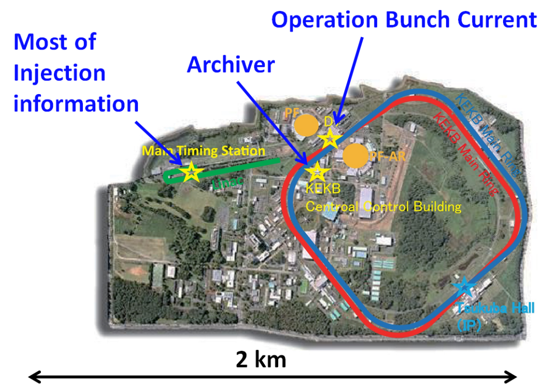


Figure 1: Accelerator layout at KEK and the location of injection-related data.

Operation Current Loss

There is the current loss of the storage bunches caused by the injection operation. The well-known loss is the injection kicker effect. The storage bunches around the Injection-bucket is kicked to approach the injection beam-pulse. This process makes storage bunches oscillating. Therefore, they emitted the synchrotron radiation and their bunch current is decreased.

We can determine and record this effect. The current loss is defined with Eq. (1). However, we change the measured bunch. In the SuperKEKB accelerator case, ±500 RF-buckets around Injection-bucket is affected by the injection kicker. The beam loss can be measured by applying the offset within this RF-bucket region.

Location of Information

The injection-information archiver collects a variety of data from the large beamlines. Figure 1 shows the location of the above information. They are spread on the SuperKEKB accelerator area.

The master IOC of Event Timing System has most of the injection-information since it manages the injection. There are the BCMs at the D7 hall. The master IOC of Bucket Selection located at the Central Control Building (CCB). It decides the next Injection-bucket.

DISTRIBUTED SHARED MEMORY NETWORK

We utilize the distributed shared memory to collect the injection information since their source is broadly spread. In this section, we introduce its module, network configuration, and the interruption functions.

* hiroshi.kaji@kek.jp

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA068

IMPROVING RELIABILITY OF THE FAST EXTRACTION KICKER TIMING CONTROL AT THE AGS*

P. Kankiya†, J. Jamilkowski, Brookhaven National Laborarory, Upton, USA

Abstract

The fast extraction kicker system at AGS to RHIC transport line uses Stanford Research DG535 delay generators to time, synchronize, and trigger charging power supplies and high-level thyratron trigger pulse generators. This timing system has been upgraded to use an SRS DG645 instrument due to reliability issues with the aforementioned model and slow response time of GPIB buses. The new model provides the relative timing of the separate kicker modules of the assembly from a synchronized external trigger with the RF system. Specifications of the timing scheme, an algorithm to load settings synchronized with RHIC real-time events, and performance analysis of the software will be presented in the paper.

INTRODUCTION

The design principle of a fast extraction system is to give the circulating beam a deflection at the kicker position. The RF synchronized discharge trigger is distributed to the alternating gradient synchrotron (AGS) extraction fast kickers and the RHIC injection fast kicker system via fiber links [1].

Two DG535 pulse delay generators are used in each kicker system. The particles with initial conditions at the beginning of G10 are traced through the lattice and receive an appropriate kick at the kicker and an additional kick at the ejector. Extraction bump are produced using two supplies to create bumps centered on H10 septum and G10 kicker. Over time the control of kicker timing using the DG535 trigger generation system has been fallible.

Table 1: DG645 Specifications

Feature	Range
Channels	4 independent pulses controlled in position and width.
Range	0-2000s
Resolution	5 ps
Jitter	Ext Trigger to any output < 100 us

Reasons for Upgrade of Delay Generator

Multiple hardware upsets of DG535 during the run and need for AC resets resulted in a significant amount of down time of AGS operations. This was reported by the pulsed power group as a concern and an upgrade to modern pulse generator was proposed.

Upgrade Strategy

The SRS DG645 has been chosen to serve as a pulse generator. The new software developed for this device runs on Linux as opposed to a Front-End Computer in the case of DG535. Other than the chief benefit of improving reliability , there is also additional benefit of reduced cost of removal of network device used for communicating using a GPIB bus. As DG645 device is capable of communicating using the ethernet. The timing specifications [2] of this model provide a significant improvement in jitter and resolution as described in Table 1.

A new software Accelerator Device Object (ADO) [3] was created to control the DG645 physical device. The new ADO receives event activity from another ADO from a set called aelMon (AGS event link monitor), which is a reflected value of the real-time event generation system. These events are used to load new delay settings to the respective channels as configured for kicker timing. The AGS operation is PPM (Pulse to Pulse Modulation) based, that is, operation of that facility for multiple recipes, with characteristics of beam intensity, species, main magnet cycle, and certain other machine parameters which can vary on successive pulses. The ADO instance is also PPM based which means that there are different delay settings stored for a different profile of AGS operation [4]. These settings are automatically loaded at the change of AGS user occurs, as the software keeps track of the active users.

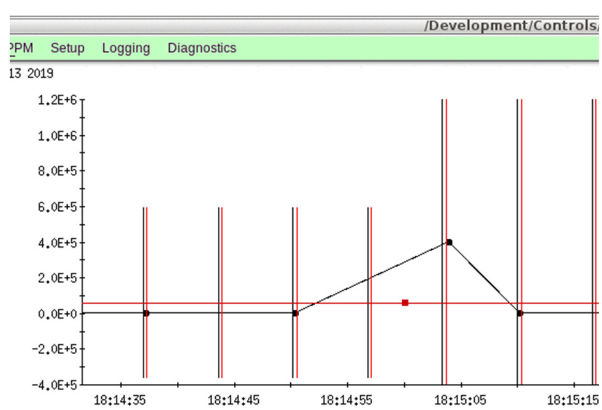


Figure 1: Execution time of new delay settings across seven AGS cycle.

* Work supported by Brookhaven Science Associates, LLC under Contract No. DE-SC0012704 with the U.S. Department of Energy.
† pkankiya@bnl.gov

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA069

AUTOMATION OF THE UNDULATOR MIDDLE PLANE ALIGNMENT RELATIVE TO THE ELECTRON BEAM POSITION USING THE K-MONOCHROMATOR

S. Karabekyan†, S. Abeghyan, W. Freund, European XFEL GmbH, Hamburg, Germany
L. Froehlich, DESY, Hamburg, Germany

Abstract

The correct K value of an undulator is an important parameter to achieve lasing conditions at free electron lasers. The accuracy of the installation of the undulator in the tunnel is limited by the accuracy of the instruments used in surveying. Moreover, the position of the electron beam also varies depending on its alignment. Another source of misalignment is ground movement and the resulting change in the position of the tunnel. All this can lead to misalignment of the electron beam position relative to the center of the undulator gap up to several hundred microns. That, in turn, will lead to a deviation of the ΔK/K parameter several times higher than the tolerance requirement. An automated method of aligning the middle plane of the undulator, using a K-monochromator, was developed and used at European XFEL. Details of the method are described in this article. The results of the K value measurements are discussed.

INTRODUCTION

The European X-ray Free Electron Laser (EuXFEL) produces spatially coherent photon pulses in the energy range from 0.26 to 29.2 keV at electron beam energies of 10.5 GeV, 14 GeV, or 17.5 GeV [1].

It has two hard X-ray Self-Amplified Spontaneous Emission (SASE) undulator systems and one soft X-ray SASE undulator system for producing high brightness laser radiation.

The hard X-ray undulators SASE1 and SASE2 consist of 35 undulator segments, each 5m long with a 40 mm magnet period, separated by 1.1 m long intersections for e-beam steering, focussing, and phase adaptation. In case of the soft X-ray undulator (SASE3) the setup is similar but has 21 segments and a longer magnet period of 68 mm [2].

In order to get a correct K value of an undulator, the middle plane of the undulator gap must coincide with the trajectory of the electron beam. Therefore, it is important to have the ability to adjust the middle plane of the undulator gap relative to the electron beam.

REQUIREMENTS FOR UNDULATOR SYSTEMS

One prerequisite for achieving lasing is the tuning of the K-value of all undulator segments to a very high precision. The undulator segments are characterized by the K-parameter: $K = \frac{eB\lambda_u}{2\pi m_e c}$, where B is the effective magnetic field and λ_u is the undulator period.

From FEL tolerance calculations, it can be shown that the relative error in the produced wavelength must be smaller than the Pierce parameter ρ. ρ is a fundamental scaling parameter and gives a measure of the exponential gain and saturated efficiency of a high-gain FEL, with typical values in the X-ray regime of $10^{-4} \leq \rho \leq 10^{-3}$ [3, 4]. The error in K for the given error in λ at 1 Å (~12 keV photon energy) is approximately $3\cdot10^{-4}$ and determines the required K-measurement accuracy.

This magnetic tuning and calibration of the undulators was done in the lab before the transport to the final position in the tunnels. Mounted in the tunnels there is no possibility to measure the K-values precisely enough by means of magnetic measurements. Photon-based commissioning [5] of the European XFEL undulators requires a precise adjustment of the K-parameters of all undulator segments and phasing between these segments. The undulator commissioning spectrometer, also known as K-Monochromator or K-Mono, which is in essence a hard x-ray monochromator based on a Si(111) channel-cut crystal, makes it possible to measure the spontaneous radiation of the undulator segments for K-tuning and other diagnostic measurements (e.g. pointing, phase matching) [3, 6, 7].

One important goal is the characterization of the undulator segments with a well-defined electron orbit after electron-beam based alignment and once lasing was established. In this case a database of energy spectrum measurements for each undulator segment will make it possible to identify changes due to radiation damage or changes in the mechanical alignment

K VALUE MEASUREMENT METHOD

There are several methods for K-tuning, phase matching, and trajectory alignment for which the K-mono can be used [3, 6, 7]. Here we have used the K-mono with SR-imager, a highly sensitive imager with a big field of view (27 mm x 15 mm), for observing the spatial profile of the spontaneous radiation of single undulator segments [5]. During our tests this appeared to be the fastest method.

When tuning the K-mono to photon energies slightly below the resonant energy of the undulator segments, donut-like rings appear, as the divergence angle of the spontaneous radiation depends on the photon energy and the off-axis spectrum gets 'red-shifted'. For even harmonics the on-axis flux is zero, but flux can be observed at an angle of only a few μrad away from the axis.

† suren.karabekyan@xfel.eu.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA071

INTERGRATED MULTI-PURPOSE TOOL FOR DATA PROCESSING AND ANALYSIS VIA EPICS PV ACCESS*

J.H. Kim†, H.S. Kim, Y. M. Kim, H.-J. Kwon, Y.G. Song
Korea Multi-purpose Accelerator Complex, Korea Atomic Energy Research Institute, Korea

Abstract

At the KOMAC, we have been operating a proton linac, consists of an ion source, low energy beam transport, a radio frequency quadrupole and eleven drift tube linacs for 100 MeV. The beam that users require is transported to the five target rooms using linac control system based on EPICS [1] framework. In order to offering stable beam condition, it is important to figure out characteristic of a 100 MeV proton linac. Then the beam diagnosis systems such as beam current monitoring system, beam phase monitoring system and beam position monitoring system are installed on linac. All the data from diagnosis systems are monitored using control system studio for user interface and are archived through archiver appliance [2]. Operators analyze data after experiment for linac characteristic or some events are happened. So data scanning and processing tools are required to manage and analysis the linac more efficiently. In this paper, we describe implementation for the integrated data processing and analysis tools based on data access.

INTRODUCTION

We have been implemented KOMAC control system based on Experimental Physics and Industrial Control System (EPICS) framework. Control System Studio (CSS) and archive appliance for data archiving that can communicate with EPICS using Channel Access protocol has adopted for managing data. Figure 1 shows the block diagram of KOMAC control system.

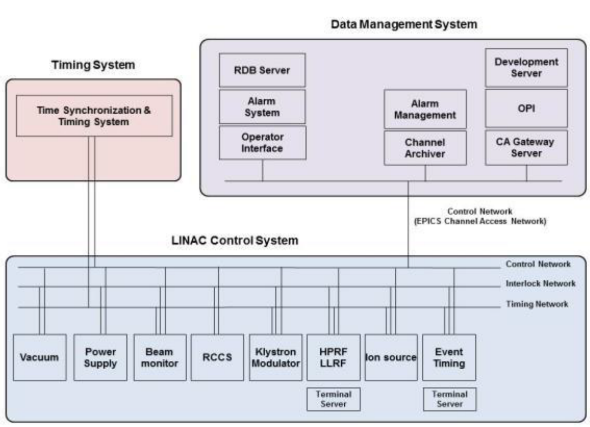


Figure 1: The block diagram of KOMAC control system.

The control system is divided into three types: LINAC control system for monitoring linac status; Timing system to synchronize all the devices; Data Management system for archiving and processing the acquired data. KOMAC has been managing five target rooms for users. To provide proper proton beams it is important to figure out linac characteristics. Therefor various beam diagnostic systems such as Beam Current Monitoring system, Beam Phase Monitoring System, Beam Position Monitoring system and wire scanners, are installed in beamlines and target rooms. And Data processing tools are required to analyse acquired data from beam diagnostic system. So High level applications have been developed to analyse the.

LEBT SCAN TOOL

Low Energy Beam Transport (LEBT) system consists of two solenoids magnets and two steering magnets for high current proton beam and minimization of beam losses, as LEBT system match the ion beam from ion source into the Radio Frequency Quadrupole (RFQ) as power supplies for LEBT system are controlled. To analyse and characterize LEBT system, magnet current need to be changed in various range. It takes a lot of time and effort to do it manually. Therefor we have been implemented LEBT SCAN tool based on CSS and Python [3] script, which can access EPICS Input Output Controller (IOC) using channel access libraries. Figure 2 shows LEBT scan tool.

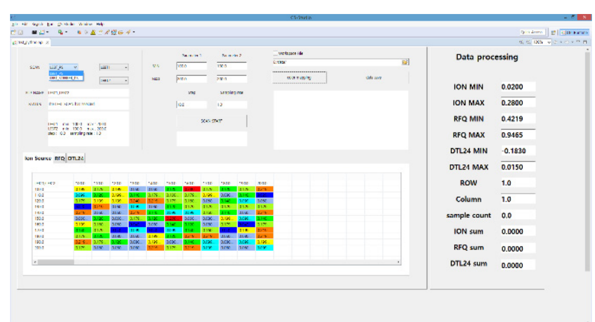


Figure 2: The LEBT scan tool.

The LEBT scan starts when the operators enter a proper value and click the start button. The acquired data that are ion source current, RFQ current, DTL24 current data from Beam current monitoring system, are shown progress table and all the raw data are archive in text format in the designated directory. When scanning is over, the result of LEBT scan is shown in table using color map.

PHASE SCAN TOOL

To increase the beam power, it is important to set proper set points of RF amplitude and phase. The RF phase is measured by 10 stripline-type BPMs. The beam energy is measured using the data from two BPMs. The Phase scan

* Work supported by KOMAC operation fund of KAERI by MSIT
† ygsong@kaeri.re.kr

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA072

AUTOMATION IN NSRC SOLARIS WITH PYTHON AND TANGO CONTROLS

W.T. Kitka*, M.K. Fałowski, A.M. Marendziak, N. Olszowska, M. Zając
SOLARIS National Synchrotron Radiation Centre, Krakow

Abstract

NSRC SOLARIS is a 1.5 GeV third generation light source constructed at Jagiellonian University in Krakow, Poland. The machine was commissioned in April 2016 and operates in decay mode. Two beamlines PEEM/XAS and UARPES were commissioned in 2018 and they have opened for conducting research in fall 2018. Two more beamlines (PHELIX and XMCD) are installed now and will be commissioned soon. Due to small size of the team and many concurrent tasks, automation is very important. Automating many tasks in a quick and effective way is possible thanks to the control system based on TANGO Controls and Python programming language. With facadevice library the necessary values can be easily calculated in real-time. Beam position correction with PID controller at PEEM/XAS and UARPES beamlines, alarm handling in SOLARIS Heating Unit Controller and real-time calculation of various vacuum parameters are shown as examples.

INTRODUCTION

Control system at SOLARIS is based on TANGO Controls. Thanks to tools like PyTango [1] and Taurus [2] and their intuitive APIs creating new controllers and GUIs is fast and easy. In case of data processing, facadedevice library has vital importance. Devices created with it subscribe to events from other, low-level devices and deliver processed signals [3]. With addition of enormous collection of available Python packages, programmers can automate a lot of processes at synchrotron.

PI CORRECTION

SOLARIS operates in decay mode with two injections per day: first one at 7:00 AM and second one at 4:00 PM. Therefore, the position of the beam changes during the shift so beamline operators have to correct it. As this correction requires frequent but very small changes, TANGO device with PI controller was created. To correct the beam position, differences in currents from slits should be minimized. The minimized current difference is described by the formula $\frac{current1-current2}{current1+current2}$. The executive element is the mirror. To ensure that the mirror is not moved too far during the process, the correction is turned off when sum of the moves reaches set threshold. After tests, sample time 10 s and proper values of K_i and K_p were established. With the use of configured PI controller TANGO device, the current difference is kept in range from -0.002 to 0.002 on UARPES beamline. The whole process can be monitored from Taurus-based GUI which is shown at on the Fig. 1.

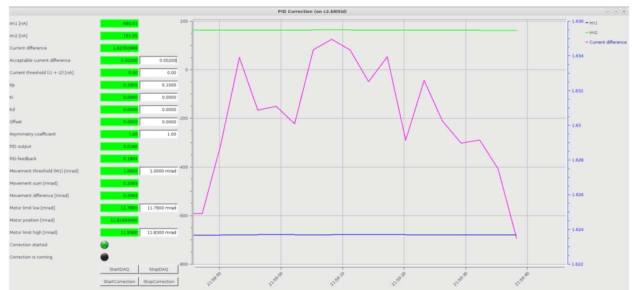


Figure 1: PID correction application.

Beam position correction process' course has been shown on the Fig. 2, Fig. 3 and Fig. 4.

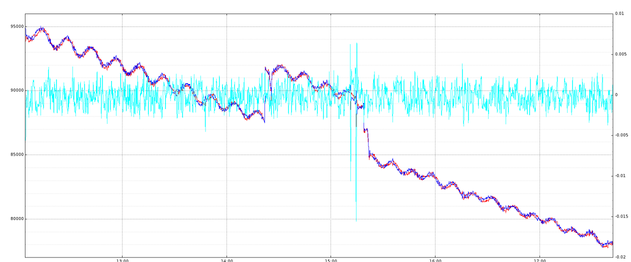


Figure 2: Currents (red and dark blue) and their difference (light blue).

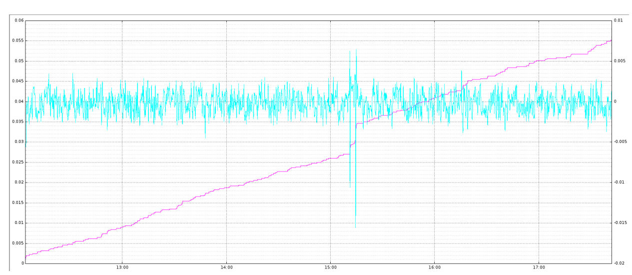


Figure 3: Sumaric movement of mirror's motor (pink) and currents difference (light blue).

HYSTERESIS MODE IN SOLARIS' HUC

The Heating Unit Controller (HUC) is used to bake-out the vacuum elements [4]. In basic mode if average pressure from input analog channels is higher than defined maximum pressure value the heating process is paused. If this alarm is present longer than configured Waiting Time then the heating process is turned off. This works well for leaks,

* wojciech.kitka@uj.edu.pl

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA073

RECENT UPDATES OF THE RIKEN RI BEAM FACTORY CONTROL SYSTEM

M. Komiyama, A. Uchiyama, M. Fujimaki, N. Fukunishi
RIKEN Nishina Center, Wako, Saitama, Japan

Abstract

We report on two of the latest updates for the RIKEN Radioactive Isotope Beam Factory (RIBF) control system. First, the successor of the existing beam interlock system (BIS) has been recently developed. The new interlock system is based on a programmable logic controller (PLC) and uses a Linux-based PLC-CPU. This allows the Experimental Physics and Industrial Control System (EPICS) programs to be executed in addition to a sequence CPU. By using two kinds of CPUs properly in accordance with the speed required for each signal handled in the system, we have succeeded in reducing the response time to less than one third of the existing BIS using a prototype.

Second, a trial was performed to extend coverage of the alarm system. We have applied the Best Ever Alarm System Toolkit (BEAST) in addition to the Alarm Handler over several years (mainly for vacuum components). We have attempted to include the magnet power supplies but found difficulties in treating older power supplies that have large fluctuations of read-out values for their excitation currents. Our trials to overcome this problem are presented in this paper.

INTRODUCTION

The RIKEN Radioactive Isotope Beam Factory (RIBF) is a cyclotron-based accelerator facility aimed at the development of nuclear physics, material science, and life science. RIBF consists of two heavy-ion linear accelerator injectors, five heavy-ion cyclotrons including the world's first superconducting ring cyclotron (SRC). Cascades of the cyclotrons can provide the world's most intense RI beams over the entire atomic mass range by using fragmentation or fission of high-energy heavy-ion beams [1]. For example, a 345-MeV/nucleon ^{238}U beam of 72 pnA and a 345-MeV/nucleon ^{124}Xe beam of 173 pnA have already been obtained.

The components of the RIBF accelerator complex (such as the magnet power supplies, beam diagnostic devices, and vacuum systems) are controlled by the Experimental Physics and Industrial Control System (EPICS) [2] with a few exceptions such as the control system dedicated to RIBF's radio frequency system [3]. However, all the essential operation datasets of EPICS and other control systems are integrated into the EPICS-based control system. In addition, two types of interlock systems that are independent of the accelerator control systems are also operated in the RIBF facility: a radiation safety interlock system for human protection [4] and abeam interlock system (BIS) for hardware protection from recent high-power heavy-ion beams [5].

* misaki@riken.jp

UPGRADE OF BIS

The hardware configuration and the process flow in the existing BIS are shown in Fig. 1. The BIS was developed based on the Melsec-Q series programmable logic controllers (PLCs). It was designed to stop beams within 10 ms after receiving an alarm signal from the accelerator and beam line components. Upon receiving an alarm signal, the BIS outputs a signal to one of the beam choppers that immediately deflects the beam just below the ion sources. It also inserts one of the beam stoppers (Faraday cup) installed upstream of the problem component. After inserting the relevant beam stopper, the beam chopper can be switched off, and beam delivery can resume up to the inserted beam stopper. This feature is particularly useful during beam tuning because beam tuning is conducted in a step-by-step manner from an injector to a final-stage accelerator. The inserted beam stopper can then be extracted from the beam line after the problem is fixed.

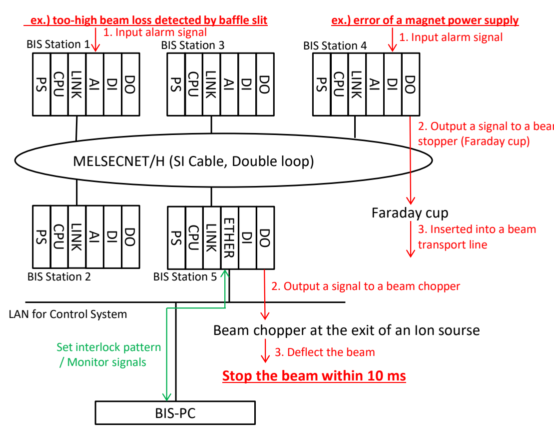


Figure 1: Example of the hardware configuration and process flow in BIS. The green line signifies communication via Ethernet.

The BIS began its operation in 2006. The recent response time is 15 – 20 ms (greater than its design value) because too much information is shared among each station through optical links in the BIS and increasing input signals. The beam power has already exceeded 10 kW, and beam operation at the level of several tens of kilowatts is expected in the near future [6]. To operate higher-power beams more safely, a response speed of 10 ms or less is required for the BIS. In addition, a greater number of components than those included in the present BIS have to be carefully monitored. This is because subtler

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA075

EPICS SUPPORT MODULE FOR EFFICIENT UDP COMMUNICATION WITH FPGAS*

M. Konrad†, E. Bernal, M. Davis
Facility for Rare Isotope Beams, East Lansing, USA

Abstract

The driver linac of the Facility for Rare Isotope Beams (FRIB) contains 332 cavities which are controlled by individual FPGA-based low-level RF controllers. Due to limited hardware resources the EPICS IOCs cannot be embedded in the low-level RF controllers but are running on virtual machines communicating with the devices over Ethernet. An EPICS support module communicating with the devices over UDP has been developed based on the Asyn library. It supports efficient read and write access for both scalar and array data as well as commands for triggering actions on the device. Device-related parameters like register addresses and data types are configurable in the EPICS record database making the support module independent of hardware and application. This also allows engineers to keep up with evolving firmware without recompiling the support library. The implementation of the support module leverages modern C++ features and relies on timers for periodic communication, timeouts, and detection of communication problems. This allows the communication code to be tested separately from the timers to keep the run time of the unit tests short.

INTRODUCTION

FRIB [1] is a project under cooperative agreement between the US Department of Energy and Michigan State University (MSU). It is under construction on the campus of MSU and will be a new national user facility for nuclear physics. Its driver accelerator is designed to accelerate all stable ions to energies >200 MeV/u with beam power on the target up to 400 kW [2]. Commissioning of the second linac segment is currently underway and the accelerator is planned to support routine user operations in 2022 [3].

The FRIB linac requires about 350 low-level RF controllers [4] to actively stabilize the RF field in the accelerating cavities as well as roughly 55 machine protection nodes [5] preventing damage to the machine by turning off the beam within 35 μs in case of a fault. The EPICS support module described by this publication acts as a driver enabling the EPICS Input/Output Controller (IOC) to communicate with these devices.

HARDWARE

Both the LLRF controllers as well as the MPS hardware have been developed in-house and use a low-cost pizza-box design based on a Spartan 6 FPGA. A MicroBlaze soft-core processor [6] implemented in the FPGA allows these devices to be controlled remotely. Since this processor does not provide sufficient resources for running an embedded EPICS IOC, the IOC needs to run on a remote machine instead. A simple C program running directly on the soft-core processor initializes the hardware and handles communication over Ethernet. Due to the lack of an IP stack, the devices' capabilities for handling network communication are limited to UDP (TCP is not supported).

UDP PROTOCOL

The UDP communication protocol specifies that an IOC initiates communication with a device by sending a request packet. The device generally responds with one or more UDP packets. The following commands are supported:

- Read registers
- Write registers
- Read waveform
- Read a block from persistent memory
- Erase a block from persistent memory
- Write a block to persistent memory
- Request write access

The commands for reading and writing registers can transfer multiple consecutive registers at the same time to increase efficiency. Three memory regions are defined for reading and writing registers. They correspond to read-only memory (e. g. for reading out sensor data), read/write memory (e. g. reading/writing set-points) and to "write-once" memory, respectively. The latter address range is used to implement commands which trigger some action on the device (like "start ramp").

The read waveform command transfers an array from the device to the IOC. These arrays can be very large and need to be broken into many UDP packets.

The commands for accessing the persistent memory are acting on a block of memory. They allow flash memories or EEPROMs to be read/modified remotely. The block size usually depends on the capabilities of the memory chip used. The driver automatically selects the most efficient block size if the device supports a range of block sizes.

DESIGN CONSIDERATIONS

The Asyn support module [7] is leveraged to implement the functionality for asynchronously reading/writing registers and for providing the device-support layer. However, some aspects like transfer of large waveforms or firmware in parallel to other communication are not supported by Asyn and thus are implemented by other means in the driver.

* Work supported by the U.S. Department of Energy Office of Science under Cooperative Agreement DE-SC0000661
† konrad@frib.msu.edu

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA076

TIMING SYSTEM UPGRADE FOR MEDICAL LINEAR ACCELERATOR PROJECT AT SLRI

R. Rujanakraikarn†, P. Koonpong, S. Tesprasitte
Synchrotron Light Research Institute, Nakhon Ratchasima, Thailand

Abstract

A prototype of 6 MeV medical linear accelerator has been under development at Synchrotron Light Research Institute (SLRI). Several subsystems of the machine have been carefully designed and tested to prepare for x-ray generation. To maintain proper operation of the machine, pulse signals are generated to synchronize various subsystems. The timing system, based on the previous version designed on Xilinx Spartan-3 FPGA, is upgraded with better timing resolution, easier configuration with more timing channels, and future expansion of the system. A new LabVIEW GUI is also designed with more details on timing parameters for easy customization. The result of this new design is satisfactorily achieved with the resolution of 10 nanoseconds per time step and up to 15 synchronized timing channels implemented on two FPGA modules.

INTRODUCTION

Synchrotron Light Research Institute (SLRI) has been developing a prototype of the 6 MeV medical linear accelerator for cancer treatment. This project has been proposed to help increase the availability of low-cost radiotherapy machines in Thailand. The already-designed linear accelerating structure of the prototype has operating frequency at 2,998 MHz, a 3.1 MW magnetron driven by a solid-state modulator, and a hot-cathode electron gun. The prototype has a fixed-position drive stand without gantry to provide housing for the modulator cabinet for magnetron, electron gun, and an automatic frequency control (AFC) system in order to provide resonant frequency tuning. All of the subsystems, including a collimator motion control system [1], are connected to the Main Control System in a private network as shown in Fig. 1.

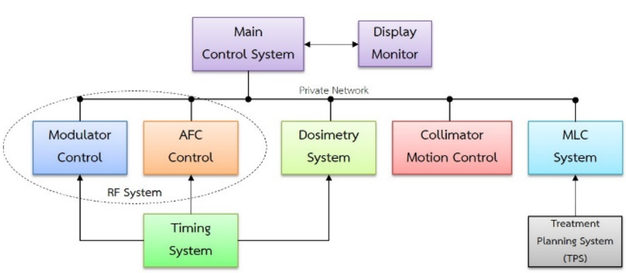


Figure 1: Network diagram of the machine prototype [1].

In order to produce X-rays to obtain the properties of the beam needed for cancer treatment using the prototype, it is necessary to produce the electron beam with the appropriate properties. The process of producing good electron beams depends on the synergy of all the subsystems, from the electron source, RF source, dosimetry system, and error detection and interlocking system. The timing system that provides synchronization of all of the machine subsystems is indispensable, with appropriate accuracy of the timing signals connecting the subsystems. All timing signals required for individual subsystems have their specific time domain requirements. If there are no such signals, it will be very difficult to set other machine parameters in order to produce the required electron beam. This has a direct impact on the quality of cancer treatment as well.

In order to generate the required timing signals one needs to be able to adjust the resolution of the pulse width (in microseconds) and delay time. Signal period can be adjusted to determine the frequency of the main clock. Analog output voltage level can also be determined as needed with enough number of channels as required by the subsystems. Configuration of all timing signals is also important. All of these requirements can be achieved by an easy-to-use GUI that is appropriately designed to communicate with users. Existing timing system [2] has been used for testing the operation of the medical linear accelerator prototype for some time with satisfactory result.

Recently, a new timing system has been designed to achieve better performance and to serve more requirements of the machine tests. In this paper all major upgrades of the timing system for this project are described. System design, both hardware and software, is explained in the next section. Installation and result are presented in the final section.

SYSTEM DESIGN

This section describes the main upgrades and implementation, both hardware and software, of the new timing system developed in this project. New timing signal characteristics and time-domain requirement settings of the signals are also explained in details.

Hardware

The main hardware platform for the new timing system is the Xilinx's Spartan 3 FPGA development board [3, 4], which is based on the same platform chosen for the existing system. The system main clock of the FPGA main board runs at 50 MHz providing the resolution of 20 nanoseconds for time domain characteristics of the timing signals. This main system clock is multiplied by 2 in this new system so that the resolution is doubled to the order of 10 nanoseconds in order to set up the timing signals whose pulse width and delay time are of better resolution. The time-domain characteristics of individual timing signals can be set independently at the outputs of each timing module.

One FPGA module has a maximum capacity of 8 timing outputs. The parameter adjustment on the FPGA depends on serial communication via RS-232 with the single board

† roengrut@slri.or.th

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA078

RENOVATION OF THE SPS PERSONNEL PROTECTION SYSTEM: A CONFIGURABLE APPROACH

T. Ladzinski, B. Fernandez Adiego, F. Havart, CERN, Geneva, Switzerland

Abstract

The renovation of the SPS Personnel Protection System (PPS) comprises the installation of industrial access control solutions and the implementation of a new safety instrumented system tailored to the particular needs of the accelerator. The SPS has been a working horse of the CERN accelerator complex for several decades and its configuration has changed through the many years of operation. The classic solutions for safety systems design, used in the LHC and PS machines, have not been judged adequate for this accelerator undergoing perpetual changes, composed of many sites forming several safety chains. In order to avoid expensive software modifications, each time the accelerator configuration evolves, a configurable safety software design was proposed. This paper presents the hardware architecture of the PLC-based SPS PPS and the configurable software architecture proposed. It further reports on the testing and formal verification activities performed to validate the safety software and discusses the pros and cons of the configurable approach.

INTRODUCTION

The Super Proton Synchrotron (SPS), CERN's second largest accelerator, was put in service in 1976. Its access control and safety interlock system, also known as the Personnel Protection System was upgraded in the beginning of the nineties. However, after nearly three decades of use, it has reached its end of life, with further upgrades no longer possible due to lack of spare-parts and more severe safety constraints.

During the ongoing Long Shutdown (2019-2020) of the CERN accelerator complex the SPS PPS is being fully replaced. This major project will complete the renovation process of the Personnel Protection Systems, providing the same level of safety across the CERN accelerator complex. In the design of the system, feedback from the implementation and operation of personnel protection systems of the Large Hadron Collider (LHC) [1] and Proton Synchrotron (PS) [2] has been taken into account. At the same time, as a new industrial partner was chosen for the implementation, certain parts of the design have been completely reviewed and new solutions proposed. In particular, it was considered that the very rigid implementation of the LHC PPS resulted in high total cost of ownership of the system. The accelerator evolves with time as new access zones are added or existing ones modified. Each zone is a development apart, albeit sharing the same library of components, and implementation of a new site or simply an addition or removal of one door has proven very expensive in the past, especially in terms of testing and validation. In order to improve the situation for the SPS PPS, it was proposed to design a configurable system, expanding the concepts already introduced in the past for the protection of personnel in the secondary beam line experimental areas at CERN [3].

SPS INTERLOCKED ZONES

The SPS accelerator is a circular machine housed in a 7 km long tunnel 60 m under the ground. In addition to the main ring tunnel, the SPS complex includes the transfer tunnels linking the ring with the PS and the LHC machines as well as several experimental areas. The SPS is currently used to deliver beams to the LHC and to three experimental regions: the North Area, as well as the HiRadMat and the AWAKE [4] facilities. In the past the SPS was used as a collider with clockwise and counter clockwise beams circulating in opposing directions.

Access from the outside to the SPS zones is strictly controlled and is only possible using dedicated access points. Once the current project is completed, they will be composed of a Personnel Access Device (PAD) [5] and a Material Access Device (MAD). Sixteen access points are present in the SPS, each leading to an access zone. Access control is managed at access zone level, with authorisations delivered to adequately trained personnel performing approved works in a given access zone.

Several access zones which are always interlocked with the same elements inhibiting operation with beam form a safety chain. In every safety chain there are at least three important safety elements capable of stopping the beam. In general two are of magnetic type and one is a beam line obstructer. Figure 1 represents graphically the relationship between the safety chains (also referred to as "interlocked zones") of the SPS complex.

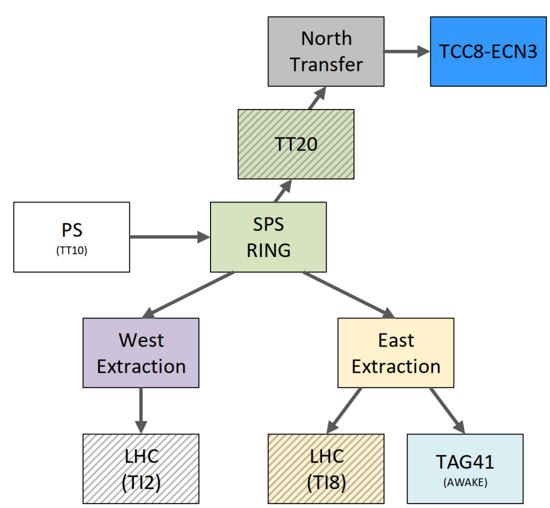


Figure 1: SPS Safety Chains with arrows representing proton beam direction.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA080

AUTOMATIC RECONFIGURATION OF CERN 18 kV ELECTRICAL DISTRIBUTION – THE AUTO TRANSFER CONTROL SYSTEM

J.C. Letra Simoes, S. Infante, F. Marin, CERN, Geneva, Switzerland

Abstract

Availability is key to electrical power distribution at CERN. The CERN electrical network has been consolidated over the last 15 years in order to cope with the evolving needs of the laboratory and now comprises a 210 MW supply from the French grid at 400 kV, a partial back up from the Swiss grid at 130 kV and 16 diesel generators. The Auto Transfer Control System has a critical role in minimizing the duration of power cuts on this complex electrical network, thus significantly reducing the impact of downtime on CERN accelerator operation. In the event of a major power loss, the control system analyses the global status of the network and decides how to reconfigure the network from alternative sources, following predefined constraints and priorities. The Auto Transfer Control System is based on redundant Programmable Logic Controllers (PLC) with multiple remote I/O stations linked via an EtherNet/IP ring (over optical fiber) across three major substations at CERN. This paper describes the system requirements, constraints and the applicable technologies, which will be used to deliver an operational system by mid-2020.

MOTIVATION

Three of the main 18 kV substations, ME9, SEM12 and BE9, located in different CERN geographical sites, are involved in the Auto Transfer control system. These are three key substations supplying the SPS (Super Proton Synchrotron), the North Area experimental areas, the Prevessin site (stable network), the LHC (Large Hadron Collider) 18 kV loop, the Meyrin site, the ATLAS Experiment, the LHC1.8 site (general services and machine networks) and the safety networks. The three substations are, under nominal condition, supplied via the French grid at 400 kV but they can also be supplied by the Swiss grid at 130 kV in case of problems on the French grid.

The ME9 substation is obsolete, more than 40 years old. It is currently undergoing a major renovation, in line with the electrical network consolidation plan and in conjunction with CERN accelerators shutdown phase. The renewal of ME9 implied a complete redesign of the substation and therefore of the 18 kV network, bringing improvements in terms of availability and flexibility of the network.

Additionally, with the existing Auto Transfer being obsolete, difficult to maintain and evolve, it was the right time for re-engineering and replacing the existing, 20 years old Auto Transfer control system.

The system was only partially mastered in terms of hardware and software, as its development and implementation had been outsourced to an external company. This limitation often created difficulties in case of major power outages that involved an auto transfer requiring post-mortem analysis. For this reason, it was decided to go for an internal implementation of the new system.

The project specification was developed in direct contact with the operation team, who has the expertise about the system expected behavior in every possible configuration scenario. The specification covers all the operation needs and allows future evolutions of the network with minor modification at the software level.

SPECIFICATION

The Auto Transfer control system main function consists in automatically reconfigure and resupply the three 18 kV substations in case of a power outage due to failures on the French grid (400 kV), the Swiss grid (130 kV) or internal failures related to transformers, bus bars or inter-substation liaison cables. See Fig. 1 for details. However, resource limitations and system operational constraints shall be taken into account. Examples are the power availability of each source, limited by its upstream transformer, and the maximum power accepted by the inter-substation liaison cables. Theoretically, up to six sources can be used to supply the seven bus bars composing the three substations; none of them can be left in parallel with another, particularly between the French and the Swiss grid. The Auto Transfer control system shall be able to reconfigure the network by opening/closing breakers with a minimum of maneuvers, maximizing the number of supplied bus bars by their nearest source, according to a predefined configuration set by the user (operation team).

THE SOLVER APPROACH

In order to cover the specified requirements, several solutions were envisaged. Former system used a *domino* approach, which consisted in supplying each bus bar as a separate entity of the network: a *domino*. Using this approach, each bus bar had one main (nominal) source and one or several backup sources. The control system managed each *domino* so that it could be powered by one of the breakers directly attached to it, according to predefined priorities. This approach created dependency between *dominos*, leading to unnecessary maneuvers and a need of synchronization between substations. Furthermore, it was tailored to the existing network configuration, limiting its evolution.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA085

CERN CONTROLS OPEN SOURCE MONITORING SYSTEM

F. Locci*, F. Ehm, L. Gallerani, J. Lauener, J. Palluel, R. Voirin
CERN, Geneva, Switzerland

Abstract

The CERN accelerator controls infrastructure spans several thousands of computers and devices used for Accelerator control and data acquisition. In 2009, a fully in-house, CERN-specific solution was developed (DIAMON) to monitor and diagnose the complete controls infrastructure. The adoption of the solution by a large community of users, followed by its rapid expansion, led to a final product that became increasingly difficult to operate and maintain. This was predominantly due to the multiplicity and redundancy of services, centralized management of data acquisition and visualization software, its complex configuration and its intrinsic scalability limits. At the end of 2017, a completely new monitoring system for the beam controls infrastructure was launched. The new "COSMOS" system was developed with two main objectives in mind: firstly, detecting instabilities and preventing breakdowns of the control system infrastructure. Secondly, providing users with a more coherent and efficient solution for development of their specific data monitoring agents and related dashboards. This paper describes the overall architecture of COSMOS, focusing on the conceptual and technological choices for the system.

INTRODUCTION

The CERN Accelerator Control System [1] relies on many components and a substantial infrastructure, which must be available 24 hours a day, 7 days a week. This hardware and software infrastructure needs to be monitored in order to anticipate or detect failures and fix them as quickly as possible. The Controls Open-Source Monitoring System (COSMOS) project was launched in 2017 to renovate the existing in-house solution [2] [3], which was suffering from its hyper-centralized model, the multiplicity of the solution, service overlap and scalability issues.

THE CONTEXT

In monitoring, the term 'host' refers to a device with an IP address (responsive to ping) while 'service' refers to any application, resource or hardware component (network device, module, sensor, etc.) providing a particular function on the host.

The accelerator control system has just under 7000 hosts (Fig. 1), mainly Linux CentOS CERN 7 computers (the use of Windows is declining in the domain of accelerator controls) and specific Ethernet devices (BMCs[1], PLCs[2], etc.). The number of Linux computers is constantly increasing, by 5 to 8% per year, while disk space has increased by a factor of 500 in a decade.

* frank.locci@cern.ch

[1] Baseboard Management Controller

[2] Programmable Logic Controller

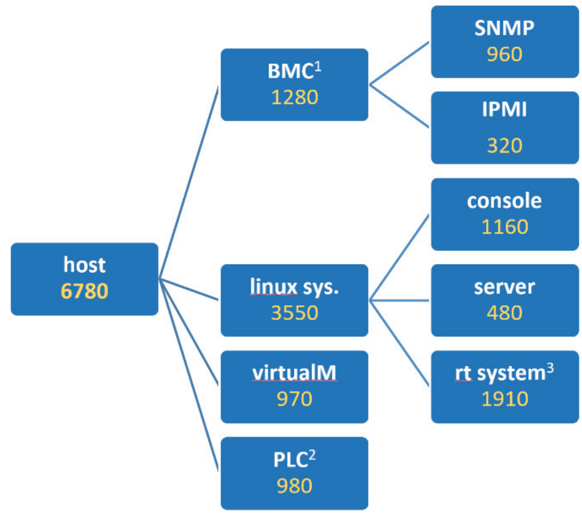


Figure 1: Main types of control system hosts.

OBJECTIVES AND SCOPE

Reviewing the existing system and evaluating major products in the monitoring field (collectd, Icinga2, Zabbix, Prometheus) helped us to define the main objectives of the COSMOS project and laid the foundations for the future solution.

Preliminary Study Recommendation

Recommendations emerging from the preliminary study were the following:

- Align the new monitoring system with CERN IT services (e.g. the central "DB on Demand" service) and industry standards in order to allow us to focus on our core business.
- Use de-facto standard technologies and open-source software as far as possible.
- Propose a new paradigm where specific aspects of the monitoring are delegated to experts who become responsible for collecting their metrics, define alerts and setup their own dashboards.

Scope of the COSMOS Monitoring System

When designing a monitoring system, it is important to consider the origin and the nature of data we want to monitor. We can distinguish at least two types of information intended for users with different objectives:

- Functional monitoring to detect infrastructure related failures, to alert the system administration team or equipment experts and to assist in taking technical decisions.
- Business monitoring focused on operational data and providing support for controlling the accelerator.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA086

THE DESIGN OF EXPERIMENTAL PERFORMANCE ANALYSIS AND VISUALIZATION SYSTEM

J. Luo†, L. Li, Z. Ni, X. Zhou, Institute of Computer Application,
China Academy of Engineering Physics, Mianyang City, China
Y. Gao, Stony Brook University, New York, USA

Abstract

The analysis of experimental performance is an essential task to any experiment. With the increasing demand on experimental data mining and utilization, methods of experimental data analysis abound, including visualization, multi-dimensional performance evaluation, experimental process modeling, performance prediction, to name but a few. Oriented to the high power laser experimental facility, we design and develop an experimental performance analysis and visualization system, consisting of data source configuration component, algorithm management component, and data visualization component. It provides us feasibilities such as experimental data extraction and transformation, algorithm flexible configuration and validation, and multi-viewing presentation of experimental performance. It will bring great convenience and improvement for the analysis and verification of experimental performance.

INTRODUCTON

Large-scale physical experiments are conducted intending to capture experimental data and make further scientific analysis, which plays a significant role in multiple discipline researches, like Hefei Light Source (HLS), European X-ray Free-Electron Laser (XFEL), Large Hadron Collider (LHC), International Thermonuclear Experimental Reactor (ITER), National Ignition Facility(NIF), to name but a few [1-6].

Taking the high power laser facility as an example, a shot experiment often produces hundreds of gigabytes data, ranging from energy scalars to focal spot images. After a shot is conducted, all diagnostic subsystems start to acquire these experimental data and further archive them into the database. Subsequently, scientists utilize the visualization subsystem to observe the ordinary report of the experiment, and make statistical analysis of the experimental data captured offline, to evaluate the performance of this experiment and the status of the facility. Furthermore, fresh experimental data contributes to modify the performance model of each laser beam.

With the development of the equipment, the demand of experimental data analysis increases rapidly, resulting in a variety of data analysis and visualization programs. These programs are conventionally customized for different application scenarios, by realizing different analysis algorithms and different graphical forms. On one hand, they are well-adapted to specific scientific scenarios; on the other hand, this program development pattern leads to a great consumption of manpower involved in program development, a messy data manage framework, a high redundancy of experimental data storage, and also a narrow application range of each program.

To address these issues mentioned above, we designed a unified data management scheme; on this basis, we further realized an experimental performance analysis and visualization system. This system is under commissioning at the high power laser facility.

The next section provides an overview of current data management architecture. Problems of experimental data application are discussed in section III. Section IV illustrates the design of experimental performance analysis and visualization system. Finally, we conclude this paper in section V.

OVERVIEW OF CURRENT DATA MANAGEMENT ARCHITECTURE

The experimental equipment produces a large variety of data. These data are multi-folds, mainly including process data that indicates the real-time circumstances of the facility entities, progress data which supports the automatic scheduling and management of experimental tasks, physical phenomenon data that describes the initial input and final output of an experiment. This paper focuses on the process of the physical phenomenon data, it is also called the experimental data.

As mentioned in the former section, after an experiment is conducted, all diagnostic subsystems start to measure these experimental data and further archive them into the database. Subsequently, these experimental data are mainly utilized for online visualization, offline statistical analysis, and decision supportive computations like the modification of the performance model (see Fig. 1).

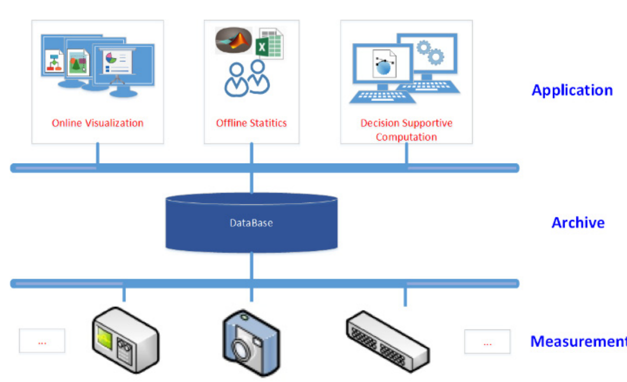


Figure 1: Flow of experimental data.

† email address: luoj1987@caep.cn

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA088

CONSOLIDATION OF RE-TRIGGERING SYSTEM OF LHC BEAM DUMPING SYSTEM AT CERN

N. Magnin, W. Bartmann, C. Bracco, E. Carlier, G. Gräwer, T.D. Mottram, E. Renner,
J.P. Rodziewicz, V. Senaj, C. Wiesner, CERN, Geneva, Switzerland

Abstract

The Trigger Synchronisation and Distribution System (TSDS) is a core part of the Large Hadron Collider (LHC) Beam Dumping System (LBDS). It comprises redundant Re-Trigger Lines (RTLs) that allow fast re-triggering of all high-voltage pulsed generators in case one of them self-triggers, resulting in a so-called asynchronous dump. For reliability reasons, the TSDS relies on many RTL redundant trigger sources that do not participate directly in the execution of a normal dump. After every dump, signals propagating on the RTLs are analyzed by Post Operation Check (POC) systems, to validate the correct performance and synchronisation of all redundant triggers. The LBDS operated reliably since the start-up of LHC in 2008, but during its Run 2, new failure modes were identified that could induce damage for the beam dump block and the dump protection elements. In order to correct these failure modes, an upgrade of the TSDS is realized. This paper reviews the experience gained with the LBDS during LHC operation and describes the new architecture of the TSDS being implemented. Measurements and simulations of signals propagating on the RTL are presented, and the analysis performed by the POC systems are explained.

INTRODUCTION

The LHC Beam Dumping System

The LBDS is a critical system, ensuring safe extraction of the beam from LHC. The beam is sent to the extraction channel using 15 extraction kicker magnets (MKD) and 15 extraction septa (MSD). It is then diluted by 4 horizontal (MKBH) and 6 vertical dilution magnets (MKBV) on the beam dump absorber (TDE). To allow for the rising edge of MKD magnetic field, a particle free Beam Abort Gap (BAG) of 3 μs is maintained in LHC [1].

The High Voltage Pulsed Generators (HVPG) power the MKDs and MKBs. At the reception of a trigger, four Power Trigger Modules (PTM) will start the conduction of two switches composed of Fast High Current Thyristors (FHCT) that discharge capacitors into the magnet.

The Beam Energy Tracking System (BETS) is a surveillance system within the LBDS, continuously checking that the voltages inside each HVPG correspond to the beam energy, to guarantee a correct extraction angle at any time. In case one HVPG exceeds tolerances, the BETS requests a beam dump [1].

The Beam Interlock System (BIS) is connected to thousands of devices around LHC, and in case a problem is reported by one of them, the BIS requests a beam dump [2].

Trigger Synchronisation and Distribution System

The TSDS is built around the two redundant Trigger Synchronisation Units (TSU), responsible for the detection of a dump request (DR) coming from various sources and the dispatch of the Synchronous Beam Dump Trigger (SBDT) to the HVPGs, synchronously with the passage of BAG in MKDs [3].

The TSDS also comprises a Re-Triggering System (RTS), allowing the asynchronous fast re-trigger of all HVPG in case of one MKD HVPG self-triggers.

Each MKD HVPG is equipped with a Re-Trigger Box (RTB), that couples internal pickup signals to the RTL to generate a pulse when the HVPG is triggered, and also to capture the pulses on the RTL, and send them to their PTMs. Each MKB HVPG is equipped with only a Re-Trigger Receiver (RTR), that will capture the pulses on the RTL, and send them to their PTMs.

These RTBs are interconnected by the RTL cable, composed of 4 conductors that are used as two pairs, with a common shielding. One pair is used for the RTL pulses, the other for a Continuity Monitoring (CM), where a constant current is used to check the RTL continuity, see Fig. 1.

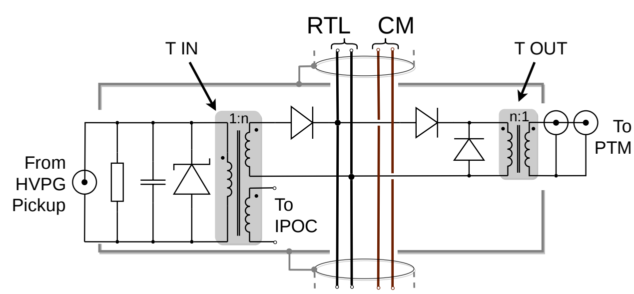


Figure 1: Simplified schema of RTB.

The TSDS was already upgraded before LHC Run 2, to add a redundant direct connection from BIS to the RTLs as a second layer of protection in case of failure of the TSU cards [4, 5].

A simplified view of the current architecture of TSDS is shown in Fig. 2. We can see from left to right the 15 MKD (A-O) HVPG (blue) connected to the RTL through their RTB, and the 10 MKBs (MKBHA-D to MKBVA-F) HVPG connected to the RTL through their RTR. About 300 m separate the MKDs from the MKBs in LHC tunnel.

At reception of a DR, the TSU cards send an Asynchronous Beam Dump Trigger (ABDT) to the RTLs through

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA090

DESIGN OF VESSEL AND BEAMLINE VACUUM AND GAS CONTROL SYSTEM FOR PROTON RADIOGRAPHY*

P. S. Marroquin†, J. D. Bernardin, J. G. Gioia, D. A. Hathcoat, A. Llobet, H. J. Sandin, W. W. Winton, Los Alamos National Laboratory, Los Alamos, USA

Abstract

A new capability for conducting explosively-driven dynamic physics experiments at the Proton Radiographic (pRad) facility at Los Alamos National Laboratory (LANL) is in development. The pRad facility, an experimental area of the Los Alamos Neutron Science Center (LANSCE), performs multi frame proton radiography of materials subjected to an explosive process. Under design is a new beamline with confinement and containment vessels and required supporting systems and components. Five distinct vacuum sections have been identified, each equipped with complete vacuum pumping assemblies. Inert gas systems are included for backfill and pressurization and supporting piping integrates the subsystems for gas distribution and venting. This paper will discuss the design of the independent vacuum control subsystems, the integrated vacuum and gas control system and full incorporation into the Experimental Physics and Industrial Control System (EPICS) based LANSCE Control Systems and Networks.

OVERVIEW

Proton Radiography at LANSCE

The proton radiography capability at the LANSCE Area C pRad facility has been advancing materials science for over 20 years [1, 2]. High energy protons, provided by the LANSCE proton accelerator, are used to produce multi frame radiographs of materials during dynamic experiments. The new beamline will extend pRad capabilities in terms of material size and types, explosive dynamics and imaging. The design incorporates a nested vessel network to contain and observe the dynamic experiment, a beamline to transport and focus the proton beam, a vacuum and gas handling system to provide the necessary environment, and a control system to operate all of it.

New pRad Beam Line Vacuum System

Vacuum performance and isolation requirements resulted in five major isolatable sections of beamline and vessels [3]. To support independent operation of the five sections, and to comply with project assembly/disassembly requirements, five identical vacuum pump carts will be used, each with separate but coordinated control equipment. Various vacuum pumps, isolation valves, vacuum gauges, check valves and other devices are included in the system design.

* Supported by the US Department of Energy, Los Alamos National Laboratory. Managed by Triad National Security, LLC for the DOE National Security Administration (Contract 89233218CNA000001).
† pilar@lanl.gov

CONTROL SYSTEM DESIGN

Mechanical Design

The beamline components include the upstream beam pipe, the containment outer vessel, the confinement inner vessel, and the downstream beam pipe. An assortment of valves allow for the sections to be isolated and to permit independent vacuum pumping, gas handling, and operation. Not all valves, piping and assorted other vacuum components are depicted on Fig. 1 which illustrates the beamline and vessels and associated vacuum carts.

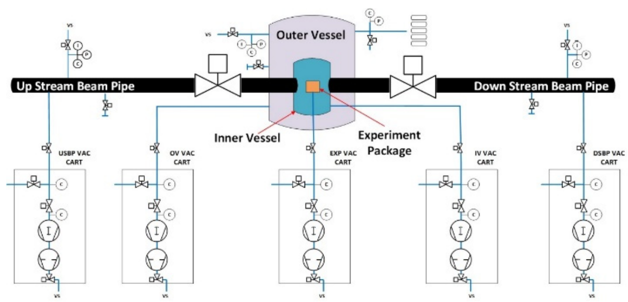


Figure 1: Beamline vacuum components.

The number of vacuum isolation valves totals to 45 of which 12 are gate valves and the remainder are right angle or globe valves. The two largest valves are the beam pipe isolation valves on either side of the outer vessel. For measuring vacuum and/or pressure there are 18 Convectron vacuum gauges, 8 ion gauges and 12 pressure gauges. The target vacuum base pressure is 50mTorr and the design pressure for post-experiment is 97,807 Pag (14.0 psig). Each vacuum cart contains one roots blower and one rotary vane pump. All of these components are read and/or remotely controlled solely by the control system, to work within requirements that during alignment operations with proton beams the beamline area is evacuated of personnel.

Instrumentation and Controls

Valves included in this design vary in sizes and types but all are air actuated and commanded by 24 volt binary signals. Limit switches are used for open/closed indications. Software will monitor travel times and consistency and validity of open/close signals and issue faults when required. All valves fail closed on loss of signal or air.

Rotary vane pumps back roots blowers which in combination, are required to rough out the system to 50 mTorr in 30 minutes.

For vacuum measurements, a number of Convectron and ion gauges are located throughout the system. The ion gauges are used specifically during high vacuum leak checking prior to main operation. Also included are a few

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA091

ESS MEBT CONTROL SYSTEM INTEGRATION

I. Mazkiaran, I. Bustinduy, C. de la Cruz[1], G. Harper, A. Rodríguez Páramo
Consorcio ESS-Bilbao, Zamudio, Spain
J. P. S. Martins, European Spallation Source ERIC, Lund, Sweden
[1] also at Centro Nacional del Hidrógeno CNH2, Puertollano, Spain

Abstract

The high power linac of European Spallation Source, ESS, accelerates 62.5 mA of protons up to 2 GeV in a sequence of normal conducting and superconducting accelerating structures. The normal conducting part is being commissioned in Lund, Sweden. The Medium Energy Beam Transport (MEBT) line has been designed tested and mounted at ESS Bilbao premises to guarantee tight requirements are met. Installed in Lund during summer 2019, and its commissioning is foreseen to the second quarter of 2020.

The main purpose of this 3.62 MeV MEBT is to match the RFQ output beam characteristics to the DTL input requirements both transversally using quadrupoles, and longitudinally RF buncher cavities. Additionally, the beam is also cleaned by efficient use of halo scrapers and pulse shape by means of a fast chopper. Besides, beam characterization (beam current, pulse shape, size, emittance) is performed using a comprehensive set of diagnostics.

Therefore, firstly, control integration of magnets and steerers power supplies, for quadrupoles, as well as synchronism, triggering, linked to high voltage pulsers within the chopper control, is part of the commitment for the present work. Secondly, the control developments of beam instruments such as Faraday Cup and Emittance Meter Unit will be described. All the integrations are based on ESS EPICS Environment (E3) [1].

INTRODUCTION

The ESS MEBT is the section of the accelerator where proton beam is of 3.6 MeV and current of 62.5 mA with pulsed shape of 2.8 ms length at 14 Hz.

At these energy levels, specific interceptive beam instrumentation can be still inserted into the beam trajectory, whereas at higher energy stages, beam radiation and high temperatures does not allow their use. Eleven (11) quadrupoles are set for matching the proton beam transversally, 11 steerers will redirect the trajectory of the beam, 3 RF buncher cavities are set for conditioning the beam longitudinally, a stripline fast chopper shapes the beam pulses, and a Beam Dump.

As depicted in Fig. 1, the beam instruments inserted tightly in the 3.81 m length MEBT. The layout includes 3 Wire Scanners (WS), 8 Beam Position Monitors (BPM), which are housed inside 8 of the quadrupoles with very strict tolerances. In addition, one Fast Chopper, 3 x 2 Scraper blades (SC), 1 Faraday Cup (FC), 1 Emittance Meter Unit (EMU) provided by one pair of horizontal and vertical slits, and one pair of horizontal and vertical grids, 2 Beam Current Monitors (BCM), Fast Current Beam Monitor (FCBM), Non-Invasive Profile Monitor (NPM) and Bunch Shape Monitor (BSM) are part of the Beam Diagnostics set for MEBT.

In terms of control, ESS-Bilbao is responsible for the FC, EMU, motion of the SC, Fast Chopper, Quadrupoles, Steerers and Buncher Cavities. Therefore, Fig. 1 is representing only interfaces that affect to these elements.

ESS-Bilbao is also in charge of the control management of cooling and local protection system interlocks as a part of the control of those elements.

HW and SW architectures are thoroughly studied to provide a consistent standardized system for the control development.

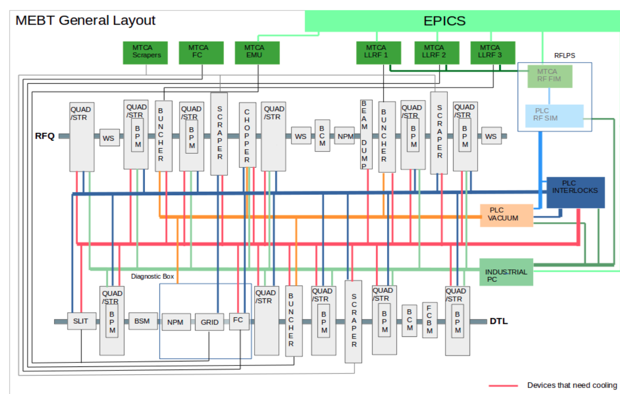


Figure 1: Control layout for ESS MEBT.

CONTROL ARCHITECTURE

MicroTCA has been the technology adopted for high speed processing and communication interfaces. MicroTCA crate provides a fundamental module Micro Carrier Hub (MCH) which adds management and communication functionality to the system such as Intelligent Platform Management Interface (IPMI), clock distribution and generation, switching functionality, increasing architecture robustness with the possibility of MCH redundancy.

The fact of using MicroTCA.4 standards, provides the architecture versatility of Advance Mezzanine Card (AMC) and Rear Transition Module (RTM) boards along with precise time characteristics.

An initial stage, while waiting for the maturity of some MicroTCA HW boards, have been developed using VME technology. It has been properly studied as a previous step to MicroTCA migration.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA092

PROTOTYPING THE RESOURCE MANAGER AND CENTRAL CONTROL SYSTEM FOR THE CHERENKOV TELESCOPE ARRAY

D. Melkumyan*, I. Sadeh, T. Schmidt, P. Wegner, DESY, Zeuthen, Germany
S. Sah, M. Sekoranja, Cosylab d.d., Ljubljana, Slovenia
I. Oya, M. Füßling, CTAO gGmbH, Heidelberg, Germany
DI. Schwanke, Humboldt-Universitaet zu Berlin, Berlin, Germany
DII. Schwarz, INAF – Osservatorio Astronomico di Brera, Milan, Italy
for the CTA Observatory†

Abstract

The Cherenkov Telescope Array (CTA) will be the next generation ground-based observatory for gamma-ray astronomy at very-high energies. CTA will consist of two large arrays with 118 Cherenkov telescopes in total, deployed in the Paranal (Chile) and Roque de Los Muchachos (Canary Islands, Spain) Observatories. The Array Control and Data Acquisition (ACADA) system provides the means to execute observations and to handle the acquisition of scientific data in CTA. The Resource Manager & Central Control (RM&CC) sub-system is a core element of the ACADA system. It implements the execution of observation requests received from the scheduler sub-system and provides infrastructure services concerning the administration of various resources to all ACADA sub-systems. The RM&CC is also responsible of the dynamic allocation and management of concurrent operations of up to nine telescope sub-arrays, which are logical groupings of individual CTA telescopes performing coordinated scientific operations. This contribution presents a summary of the main RM&CC design features, and of the future plans for prototyping.

INTRODUCTION

The Cherenkov Telescope Array (CTA) is the planned next generation observatory for ground-based very-high-energy gamma-ray astronomy [1]. In order to provide full sky coverage, CTA will consist of two large arrays with 118 Cherenkov telescopes in total, to be deployed in the Southern (Paranal, Chile) and Northern (La Palma, Spain) Hemispheres. The southern array will be sensitive to entire energy range of CTA, covering gamma-ray energies from ~20 GeV to above 300 TeV. The northern array will focus on low- and mid-energy ranges from ~20 GeV to ~20 TeV. In addition to a large number of scientific instruments, CTA will comprise three types of telescopes covering different energy ranges: Small Sized Telescopes (~4 m diameter) to cover the highest energy gamma-rays, Medium Sized Telescopes (~12 m) to cover the core energy range of CTA and Large Sized Telescopes (~23 m) sensitive to low-energy gamma-rays (SSTs, MSTs and LSTs, respectively).

The Array Control and Data Acquisition (ACADA) system provides the means to execute observations and to handle the acquisition of scientific data in CTA [2, 3].

In this proceeding, we focus on the design and implementation of one of the top-level sub-systems of ACADA, the Resource Manager and Central Control (RM&CC) system. The purpose of RM&CC is to execute observations requests received from the Short-term Scheduler sub-system and to provide to all ACADA sub-systems infrastructure services concerning the administration of various resources.

ARRAY CONTROL AND DATA ACQUISITION SYSTEM

ACADA will provide the functionality required to monitor and control all telescopes and auxiliary instruments in CTA; to perform observations and calibration procedure; to handle, filter and store data from all the telescope and auxiliary instruments; and to produce status and quality reports.

The ACADA architecture was designed using the Software Platform Embedded System (SPES) methodology [4]. A combination of OMG systems Modeling Language (SysML) [5] and Unified Modeling Language (UML) [6] was employed to model the ACADA system architecture [7] (system requirements, system behavior and subsystems). The ACADA system is composed of several closely interrelated sub-systems (*Short-term Scheduler, Transient Handler, Resource Manager and Central Control, Array Data Handler, Human Machine Interface, Science Alert Generation Pipeline, Array Alarm, Configuration, Reporting, Monitoring and Logging*). These subsystems are split up further into individual components. Each sub-system and each component serves a well-defined easily comprehensible purpose. This architectural approach allows us to implement a flexible use-case driven software development approach thanks to the traceability from use cases to the logical software elements.

ACADA will be implemented as a distributed software system using the ALMA Common Software (ACS), which is a set of application frameworks built on top of CORBA middleware [8]. ACS is based on a container-component model and supports the programming languages C++, Java and Python. Each CTA site will contain one instance of the ACADA system.

* david.melkumyan@desy.de
† https://cta-observatory.org

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA095

STATUS OF OpenXAL AT ESS

N. Milas*, J. F. Esteban Müller, E. Laface, Y. Levinsen
European Spallation Source ERIC, Lund, Sweden

Abstract

The Open XAL accelerator physics software platform is being developed through international collaboration among several facilities since 2010. The goal of the collaboration is to establish Open XAL as a multi-purpose software platform supporting a broad range of tool and application development in accelerator physics and high-level control. This paper discusses progress in beam dynamics simulation and updated application framework along with new generic accelerator physics applications for the ESS branch of the collaboration. We present the current status of the project, a roadmap for continued development and an overview of the future developments needed for ESS future commissioning work.

INTRODUCTION

Open XAL [1, 2] is a generic open source software platform for accelerator physics. Open XAL is written in the Java programming language but is accessible via any JVM based scripting language. It is used world-wide by different accelerator labs including: China Spallation Neutron Source (CSNS) in Dongguan, China; European Spallation Source ERIC (ESS) in Lund, Sweden; Spiral2 program at the Grand Accélérateur National d'Ions Lourds (GANIL) in Caen, France; and Spallation Neutron Source (SNS) in Oak Ridge, TN, USA.

The decision to use Open XAL at ESS was taken in 2015 [3] taking into account the similarities between SNS and ESS and the possibility of re-using part of the work already developed. Between 2015 and 2019 a fair amount of changes to the core part of the code were done in order to fit Open XAL to site specific needs at ESS. The main efforts were directed towards improving the site specific online model (JELS) and in the development of applications required for the first stages of the accelerator commissioning. Further changes were done in the applications framework, that was shifted from Swing to JavaFX, on the data saving format with the addition of HDF5 as the new standard among many others.

In this work a summary of the specific development of Open XAL functionalities will be described as well as the first set of application used for the Ion Source and LEBT commissioning between 2018 and 2019. An overview of the future developments will also be outlined.

* natalia.milas@esss.se

CORE AND EXTENSIONS

Development and Deployment

Open XAL has been upgraded to Java 12 with minor modifications and it will be available at the Control Room for the next commissioning phase starting on the second quarter of 2020. At ESS we use a Gitlab server to have the sources locally stored, and builds are now done using Gitlab-CI. We split the deployment process for Open XAL core and applications, so that each application can follow its own deployment schedule. During the commissioning of the LEBT and the Ion Source in 2019 we noticed that the deployment process of Open XAL as a single unit made the process of debugging and releasing new features for the Applications cumbersome. We now have several Git repositories, one for the core and services, one for the lattice, and for the applications.

Model and Machine Description

The machine description is now imported from the official lattice repository, which uses the TraceWin [4] format, and the SMF xml files are generated accordingly. The definition of apertures has been extended and now each element can have an array describing its aperture, and an integer to set the aperture shape (e.g. rectangular, elliptical). Markers can incorporate an aperture bucket to describe the aperture of the drift spaces. The AcceleratorSeq class includes methods to get the aperture for all its elements.

We also simplified some of the existing elements. Dipole bending and steering magnets had different classes for vertical and horizontal orientations, and they have been merged. The field map classes have been completely refactored to allow quick development of new field map classes. This new field map model has a 4^{th} Runge-Kutta integrator for the equations of motion of a particle in electromagnetic fields and allows the superposition of field maps.

Open XAL had a simple model for misalignments implemented for some elements. This has been refactored to take into account all elements, except for dipole magnets, and is valid only for small angles, when the approximation $\tan\theta \approx \theta$ holds. We use the standard yaw-pitch-roll convention (Tait–Bryan angles). For the misalignment, sequence type elements like the RF create a global misalignment and single elements inside the sequence can have their own parameters, the final quantities used are the sum of all misalignments for each single element.

Originally, Open XAL was designed for bunched beams only. We included a new algorithm in the EnvelopeTrackerBase class to simulate space-charge effects for continuous beam [5], which is needed to model the ESS LEBT. This algorithm can be activated by enabling a flag either while

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA096

ESS DRIFT TUBE LINAC CONTROL SYSTEM ARCHITECTURE AND CONCEPT OF OPERATIONS

M. Montis, L. Antoniazzi, A. Baldo, M. Giacchini, INFN-LNL, Legnaro, Italy
T. Fay, ESS, Lund, Sweden

Abstract

The Drift Tube Linac (DTL) of the European Spallation Source (ESS) [1] is designed to operate at 352.2 MHz with a duty cycle of 4% (3 ms pulse length,14 Hz repetition frequency) and will accelerate a proton beam of 62.5 mA pulse peak current from 3.62 to 90 MeV. According to the project standards, the entire control system is based on the EPICS framework [2]. This paper presents the control system architecture designed for the DTL apparatus by INFN-LNL [3], emphasizing in particular the technological solutions adopted and the high-level control orchestration, used to standardize the software under logic design, implementation and maintenance points of view.

INTRODUCTION

The entire ESS linear accelerator and, as consequence, the DTL require dedicated equipment and strategies for the control (Figure 1). DTL system is interfaced with other different apparatus composing the normal conducting linac and transversal systems and services, such as RF system, Machine Protection System (MPS) and Personal Protection System (PPS). Because of the complexity of the project and the number of persons involved at different layers, the DTL Control System (CS) design and implementation must follow precise strategies and solutions, with the aim of optimizing costs and time during the stages of the installation campaign in Sweden.

DTL CS is in charge to provide the software and hardware layer required to operate the apparatus. Not all the functional sub-systems composing the DTL are in charge of the DTL CS Group, so the control system architecture exposed here can cover only a part of them.

All the external sub-systems (e.g. RF system) must exchange data and information with DTL CS. Due to that, a big effort is required at level of coordination among the different parts (ESS and stakeholders).

CONTROL SYSTEM ARCHITECTURE

The architecture realizes the canonical 3-layer structure (Figure 2) where:

- At the lower level there are all the DTL functional subsystems. Under CS aspects, the lower layer defining the field where the I/O signals come from.
- The middle layer defines the set of controllers used to perform the logic and the automation required by the application (Hardware and Software). In this layer all the control units (EPICS Input/Output Controllers - IOCs) run both the low-level interface applications and the high-level state machines (Control System Core).
- At the highest level, all the services provided by ESS-ERIC to performs the normal tasks to operate the Linac. The principal services and tools are:
  - <u>HMI tool</u>, which defines the set of control panels used to control the DTL apparatus. For this purpose, a six-monitors screen is used and EPICS Control System Studio (CSS) application is devoted to implement the control panels.
  - <u>Archiver service</u> provides to the DTL CS the set of tools to archive and retrieve EPICS process variables (EPICS PVs) according to the strategies provided by INFN-LNL.
  - <u>Alarm service</u> provides the interface to the operator to supervise DTL status, indicating cases of warnings (not interlocks).

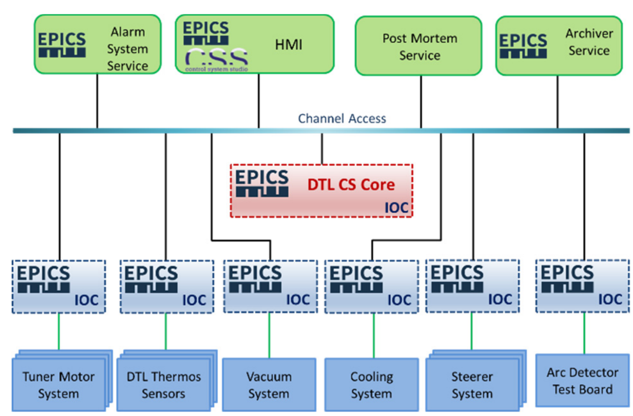


Figure 2: DTL Control System Architecture.

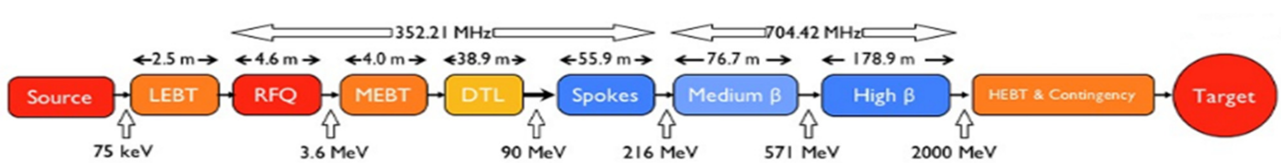


Figure 1: Schematic of the ESS Linac.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA097

EPICS BASED CONTROL SYSTEM FOR SPES TAPE STATION FOR BEAM CHARACTERIZATION: MOTION SYSTEM AND CONTROLS

M. Montis[1], M. Giacchini[1], T. Marchi[1], INFN-LNL, Legnaro, Italy
B. Genolini[2], L. Vatrinet[2], D. Verney[2], IPN, Orsay, France
J. Abraham[3], iThemba LABS, Somerset West, South Africa

Abstract

The SPES [1] [2] Tape Station (STS) for Radioactive Ion Beams (RIBs) characterization is under construction at LNL. This tool will be used to measure the actual composition of the radioactive ion beams extracted from the SPES-β ion source and to optimize the source's parameters. STS will provide beam diagnostic information by determining the beam composition and intensity. At the same time, it will be able to measure the target release curves needed for the source's characterization and development. The core part of the system, the related motor and controls are being designed and constructed in synergy with IPN Orsay (France), iThemba Laboratories (South Africa) and the Gamma collaboration (INFN-CSN3). In particular, the mechanical part is based on the existing BEDO [3] tape system operated in ALTO while the control system for motion is an EPICS [4] base application under implementation by iThemba and INFN, result of an upgrade operation required to substitute obsoleted hardware and update logic and algorithm.

INTRODUCTION

In order to characterize the radioactive ion beam provided by the SPES ion source, the STS uses particular techniques mainly based on γ-ray spectroscopy of the β-decaying radioactive nuclei.

The ions are implanted on an aluminated mylar moving tape and either measured in-situ or transported to a shielded decay location. The detection of single γ-ray spectra or β-γ coincidence spectra provides then the required information. The further movement of the tape allows removing any residual long-lived activity before a new measurement cycle is started (Figure 1).

The radioactive ions of interest are produced, extracted, ionized and pre-selected at the SPES target location. Two tape stations are foreseen in the present layout. A first Tape Station (STS1) is placed just out of the production and preselection bunkers and serves as the first feedback for the source operation. A second Tape Station (STS2) is installed after the High-Resolution Mass Spectrometer (HRMS) to operate as feedback for the HRMS itself. For non-reaccelerated beams (with and without the use of the HRMS), STS1 is used for beam composition determination before delivery to the experiments.

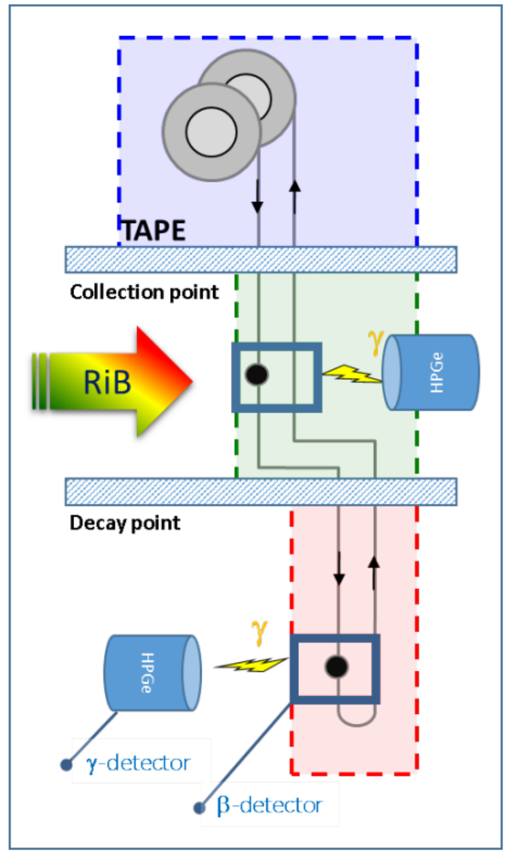


Figure 1: Schematic view of the Tape Station.

THE STS CONTROL SYSTEM

In Legnaro Laboratories a tape station was used to perform similar diagnostic analysis for dedicated experiments, but the status of mechanical, electronics and controls let us decide to invest in a new version: obsolete devices such as motors and electronics components not more available on the market is a crucial point for guarantee the maintenance of the system. At the same time, the old LabVIEW based control software must be updated in order to eventually manage new sensors and actuators.

With the focus to provide a solution fully integrated in the main SPES control system, a new EPICS control system has been chosen.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA098

A NEW COMMUNICATION INTERFACE FOR THE EUROPEAN SOUTHERN OBSERVATORY (ESO)'s VERY LARGE TELESCOPE TECHNICAL DETECTOR CONTROL SYSTEM USING ARAVIS, AN OPEN-SOURCE LIBRARY FOR GenICam CAMERAS

K. F. Mulholland*, Observatory Sciences Ltd., Kirkcaldy, U.K.
J. Knudstrup, F. Pellegrin, European Southern Observatory, Garching, Germany

Abstract

The European Southern Observatory's [1] Very Large Telescope (VLT) [2] Core Control System (CCS) [3] provides support for high-performance industrial cameras with its Technical Detector Control System (TDCS). Until now, TDCS has used a communication interface based on an API from Allied Vision Technologies (AVT) [4], which only supports cameras made by AVT. As part of the VLT 2019 release, a new communication interface has been developed for TDCS using Aravis [5], the open-source library for GenICam [6] cameras.

Aravis has been independently developed to provide support for cameras from any vendor, although this is not guaranteed. It reads the GenICam interface of a GigE Vision camera to enable control. It also has capabilities for USB3Vision cameras.

With this new communication interface, support for other manufacturers is now possible. It has been tested with cameras from AVT and Basler [7], and further tests using a CameraLink camera with a GigE Vision adapter are planned. This paper will discuss the capabilities of Aravis, considerations in the design of the communication interface, and lessons learnt from the implementation.

INTRODUCTION

The VLT [2] is one of the most advanced facilities for ground-based astronomy in the world. It is composed of four 8.1-m Unit Telescopes (UTs) and four 1.8-m Auxilliary Telescopes (ATs). The VLT CCS [3] is a software framework providing general operational tools and facilities for communication, logging, an online database, and an alarm system. It also includes a Real-Time Display (RTD) system, interfaces for instrument control, and toolboxes to build applications, amongst other functionality.

Although the VLT first started operations in the late 1990's, it is still constantly evolving to incorporate new techniques and technologies. For example, operating high-performance industrial cameras that transmit high-speed video over a Gigabit Ethernet (GigE) network, with the GigE Vision standard, has been achieved with the Technical Detector Control System (TDCS).

Cameras that implement the GigE Vision standard use the Generic Interface for Cameras (GenICam) [6]. This aims to provide a generic interface for cameras, irrespective of the connection they use. For example, as well as GigE Vision, GenICam is also used by USB3 Vision and Camera Link cameras. The GenICam standard mandates that the camera parameters are stored in a self-describing XML file, which maps the features to the camera's registers. An application API can then easily read and set parameters in the XML file, and GenICam is able to retrieve or set the matching values from the camera's registers.

Up until now, TDCS was only able to communicate with cameras from Allied Vision Technologies (AVT) [4]. It has used the PvAPI SDK, which is no longer supported. While changing the communication interface to be based on the new Vimba SDK, also made by AVT, was considered, an alternative, Aravis, was chosen because Aravis is:

- open source
- actively maintained
- vendor independent

TDCS

TDCS is designed to be compliant with the overall VLT CCS. It allows the controlling and configuration of GigE Vision cameras, streaming the camera data to a real-time display, the application of customised data processing recipes, and storage of data in FITS files. It implements a state machine which is also generally compatible with the VLT's CCS architecture.

TDCS is implemented in C++ and is multithreaded. Several threads are used in order to maximise performance. There is one permanent thread, which is used to handle commands. Other threads perform specific tasks such as acquiring data from the camera, or process the data. The overall architecture of TDCS is shown in Fig. 1.

TDCS currently uses a communication interface based on AVT's PvAPI SDK to communicate with GigE Vision cameras. Messages are sent to TDCS via the VLT CCS messaging system. These messages might be to change the state of TDCS, to start or stop acquisition, or to change parameters, etc. TDCS parses commands to determine what to forward to the camera, and "translates" the commands into the camera's API calls. Similarly, it receives frames from the camera and is able to *publish* them to e.g. the Real Time Display (RTD) module of VLT CCS, convert and store them into a FITS file format, and/or apply customisable *recipes* to perform user-defined data processing.

TDCS is currently used by ESPRESSO [8], and will be used by four other new instruments currently under construc-

* kfm@observatorysciences.co.uk

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA099

XCHEM LABORATORY PUCK SCANNER – ALGORITHM AND RESULT VISUALISATION

U. M. Neuman*, J. D. O'Hea, Diamond Light Source Ltd, Didcot, UK
K.Ward, I. Rey, Tessella, Abingdon, UK

Abstract

Macromolecular Crystallography (MX) facilities are known for using many samples and require software tools which can scan, store and help to track samples' Data Matrix codes and to maintain the correct sample processing order. An open source Data Matrix code scanning program, Puck Scanner, developed at Diamond Light Source (DLS) is introduced, its scanning algorithm explained and the continuous visualisation of results presented. Scanned codes are stored together with date, time, and the number of valid codes within a puck. This information is crucial for researchers as it allows them to match the sample with X-ray scanning results. The software is used in Diamond's XChem laboratory on a day to day basis and has started to be adopted by other facilities.

INTRODUCTION

An MX beamline experiment collects the diffraction patterns produced by an X-ray beam sent through a crystal sample [1]. Crystal samples are prepared in a laboratory, placed in a sample holder and moved to the beam. Both in-house and external laboratories may be used to prepare the samples. The scope of this paper is the DLS [2] internal sample preparation laboratory XChem. The XChem laboratory sample preparation process is described in [3]. The process is assisted by various software tools: Shifter [4], PanDDA [5], XChemExplorer [6] and the discussed Puck Scanner.

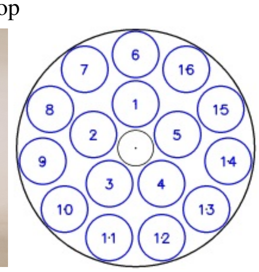
Puck Scanner is used to scan the Data Matrix [7] codes on top of each of the sample pins and on the side of the 'puck' in which the pins are stored. The decoded Data Matrix codes are stored on a disk by Puck Scanner together with the date, time and the number of successfully scanned codes before the sample puck is sent to the beamline. Scanning pucks before sending them allows researchers to track the pucks. Not all codes need to be successfully scanned for reserchers to identify a puck.

COMPONENTS

The XChem laboratory uses standard sample holders - unipucks (universal pucks) (Fig. 1) - produced by companies like Crystal Positioning Systems [8] and MiTeGen [9] with an additional Data Matrix on side. This side Data Matrix code is a crucial addition. Before this was introduced researchers experienced lots of problems related to mismatching the pins and pucks. There are 16 slots in each puck arranged in two rings. Each slot has a number assigned

(a) Unipuck top

(b) Unipuck side



(c) Puck template

Figure 1: Unipuck top 1a and side1b, and puck template 1c.

to it. A characteristic cut in the edge of the puck - the notch - is visible in the right bottom corner of Fig. 1a.

Standard SPINE [9, 10] sample pins with Data Matrix codes printed on top are used by the XChem laboratory. Crystals are held on the pins using small loops attached to the tips. Fig. 2 shows a pin from two different perspectives, the side and the top with the Data Matrix codes visible.

A laboratory stand designed and 3D printed at DLS supports the puck scanning procedure (Fig. 3). It has two camera slots - one for a camera pointing at the side code and one for a camera pointing at the top of the puck. The notch ensures that the puck is positioned correctly and the side code is visible to the camera. Without the notch there would be no guarantee this side code is seen by the side camera. The cameras are connected to the laboratory PC using USB 2.0. Basic microscope pencil shaped cameras by Supereyes are used by XChem [11].

* urszula.neuman@diamond.ac.uk

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA100

quasar: THE FULL-STACK SOLUTION FOR CREATION OF OPC-UA MIDDLEWARE

P. P. Nikiel*, P. Moschovakos, S. Schlenker, CERN, Geneva, Switzerland

Abstract

quasar (Quick OPC-UA Server Generation Framework) started as OPC-UA server generation framework. The project evolved into a software ecosystem providing OPC-UA support for distributed control systems. OPC-UA servers can be modeled and generated and profit from tooling to aid development, deployment and maintenance. OPC-UA client libraries can be generated and published to users. Client-server chaining is supported. *quasar* was used to build OPC-UA servers for different computing platforms including server machines, credit-card computers as well as system-on-chip solutions. *quasar* generated servers can be integrated as slave modules into other software projects written in higher-level programming languages (such as Python) to provide OPC-UA information exchange. *quasar* supports quick and efficient integration of OPC-UA servers into a control system based on the WinCC OA SCADA platform.

The ecosystem is adapted to different OPC-UA stack implementations and thus can be used as fully free and open-source solution as well as with and for commercial applications.

The contribution will present an overview and the evolution of the ecosystem along with example applications from the ATLAS Detector Control System (DCS) and beyond.

PREVIOUS WORK

The OPC-UA Standard

The OPC-UA standard [1] is very attractive for information exchange between nodes of a distributed control system. The prime advantages are: information modeling, firewall friendliness, portability to different computing platforms, usage of open standards, security and scalability. The standard defines how information is to be exchanged but it leaves the corresponding aspects of software engineering undefined. Therefore software engineers are left without tools to make OPC-UA compliant software. In addition it bears the risk of multiple incompatible software architectures and implementations which are difficult to maintain in a big distributed system which is expected to be used for a decade or longer.

Quasar

quasar was born as an OPC-UA server generation framework [2, 3] to standardize the process of OPC-UA software creation. It was used to create, develop and maintain multiple OPC-UA server projects, primarily for the controls of the ATLAS [4] and other LHC experiments at CERN. Open-sourcing in 2015 [5] permitted its application to numerous projects beyond CERN, e.g. to create OPC-UA servers for commercial power supplies [6].

quasar profits from the Model-Driven Architecture. The models used in *quasar* are called *quasar designs*. *quasar designs* are on a conceptually higher level than OPC-UA information models. A number of dependent artifacts is generated from *quasar designs*: source code, model visualizations, documentation, SCADA integration, among other.

MOTIVATION FOR EVOLUTION

Due to the positive experience with a growing number of OPC-UA applications, the new standard gained wide interest. Consequently, new applications (and thus requirements) followed. Among the requirements the following were of the highest importance: liberation from restrictive software licensing, embedding of OPC-UA software components on embedded targets, distributing processing logic to a chain of serially or parallely connected OPC-UA components, quick integration into SCADA systems and integration of OPC-UA software components into other programming languages.

EVOLUTION OVERVIEW

Liberation from Restrictive Licensing

Initially, the Unified Automation's C++ OPC-UA SDK [7] (further referred to as *UA-SDK*) was supported as the only OPC-UA implementation. However, the *UA-SDK* requires a paid license to develop with, which is considered to be relatively costly, especially for multiple developers. This was considered a significant barrier to wider adoption of OPC-UA. In addition, the source code is closed-source, which poses a significant limitation for build platforms requiring access to the source code, like *Yocto* (detailed in the next subsection).

Thus an attempt was made to find a substitute for the *UA-SDK*. Among free and open-source OPC-UA protocol stacks, *open62541* [8] was considered the most promising choice. A compatibility library called *open62541-compat* [9] was made to adapt the API of *open62541* to the one offered by *UA-SDK*. As the result, while building OPC-UA software components made with quasar, it is possible to select the protocol stack: either *UA-SDK* (paid) or *open62541* (free and open-source).

The *open62541-compat* library covers most of the features achievable with the *UA-SDK* . However, at the time of writing, certain features were not yet available. For example, the internal architecture of the *open62541* leaves much less freedom on how to process incoming OPC-UA requests than the *UA-SDK*. This limits the distribution of requests processing into multiple threads and prevents job queuing. Nevertheless only few applications known to the authors

* Piotr.Nikiel@cern.ch

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA103

THE PLC CONTROL SYSTEM FOR THE RF UPGRADE OF THE SUPER PROTON SYNCHROTRON

J. C. Oliveira†, L. Arnaudon and A. Diaz Fontalva
CERN, Geneva, Switzerland

Abstract

During the CERN Long Shutdown 2 (LS2), the 200 MHz main acceleration system of the Super Proton Synchrotron (SPS) is being upgraded. Two cavities will be added to reach a total of six. Each new cavity will be powered by Solid State Power Amplifiers (SSPA) grouped into 16 "towers" of 80 modules each, in total 2560 modules. This paper describes the newly developed control system which uses a master PLC for control and interlock of each cavity and the slave PLC controllers for each of the solid state amplifier towers. The system topology and design choices are discussed. Control and interlocking of all subsystems necessary for the operation of an RF cavity are detailed, and the interaction between the master and slave PLC controllers is outlined. We discuss some preliminary results and performance of the test installation.

INTRODUCTION

During the long shutdown 2 (LS2), the LHC injection chain is being updated to increase the beam intensity supplied to the LHC in order to increase the collision luminosity [1].

In the SPS, the 200 MHz main acceleration system is being upgraded for this purpose.

It will go from 4 TWC (traveling wave cavities) with 2x4 sections (Fig. 1) + 2x5 sections to 6 TWC of 2x4 sections + 4x3 sections (Fig. 2)

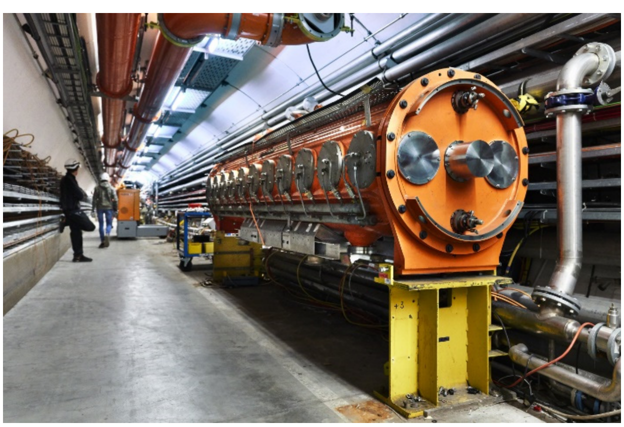


Figure 1: One section of 200MHz TWC.

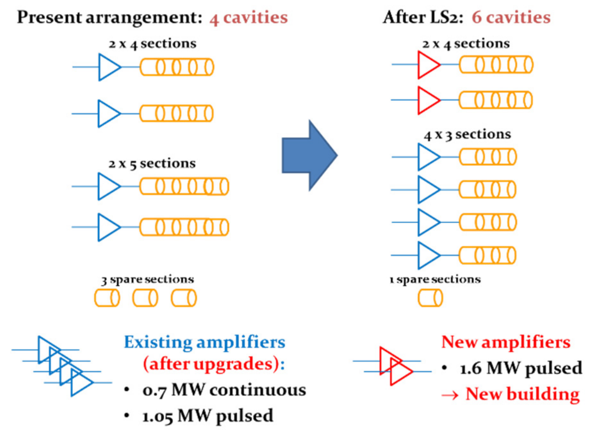


Figure 2: Renovation of 200MHz TWC system.

The two new cavities, of four sections each, will be powered by a new solid-state amplifier chain that will provide 1.6 MW to each cavity.

The other 4 cavities, of 3 sections each, will continue to use the old tube amplifiers with 1.05 MW each.

The whole system (Cavity, amplifiers, etc.) is controlled by PLCs (programmable logic controllers).

In this article, we will focus on the control system of the solid-state amplifiers.

THE NEW SOLID STATE AMPLIFIER

The new amplifier chain is composed as follows (Fig.3):

- The RF signal is generated by the beam control (Low Level RF system)
- Two fast RF switches, one on the Low Level side (LL RF Switch), one on the high level side (HL RF Switch)
- 16 pre-drivers (TTis) of type TTi Norte SSPA (Solid State Power Amplifier) delivering 1250 Wp each (Fig. 4). The RF drive signal is split in 16 to feed the 16 pre-drivers.
- 16 Thales towers, with 80 SSPAs each, which deliver 115 kWp per tower. The output signals of the 80 SSPAs are combined to power the cavity.
- The RF signal to the cavity will be 1.6MWp

THE NEW CONTROL SYSTEM

The new control system is composed as follows (Fig. 5):

- A master PLC of the brand Beckhoff
- Sixteen tower controllers of the brand Beckhoff
- An in-house designed fast-interlock system coupled with an analog acquisition system to monitor RF signals
- A FESA server running on FEC

† joao.carlos.oliveira@cern.ch

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA105

ADAPTATION OF CERN POWER CONVERTER CONTROLS FOR INTEGRATION INTO OTHER LABORATORIES USING EPICS AND TANGO

S. Page, J. Afonso, C. Ghabrous Larrea, J. Herttuainen, Q. King, B. Todd
CERN, Geneva, Switzerland

Abstract

Modern power converters (power supplies) at CERN use proprietary controls hardware, which is integrated into the wider control system by software device servers developed specifically for the CERN environment, built using CERN libraries and communication protocols. There is a growing need to allow other laboratories to make use of power converters that were originally developed for CERN and, consequently, a desire to allow for their efficient integration into control systems used at those laboratories, which are generally based upon either of the EPICS and TANGO frameworks.

This paper gives an overview of power converter equipment and software currently being provided to other laboratories through CERN's Knowledge and Technology Transfer programme and describes differences identified between CERN's control system model and that of EPICS, which needed to be accounted for. A reference EPICS implementation provided by CERN to other laboratories to facilitate integration of the CERN power converter controls is detailed and the prospects for the development of a TANGO equivalent in the future are also covered.

THE CERN KNOWLEDGE TRANSFER PROGRAMME

The Knowledge Transfer (KT) group at CERN aims to engage with experts in science, technology and industry in order to create opportunities for the transfer of CERN's technology and know-how. The ultimate goal is to accelerate innovation and maximise the global positive impact of CERN on society. This is done by promoting and transferring the technological and human capital developed at CERN.

CERN POWER CONVERTERS

Over the years, CERN has designed various families of power converters. The recent designs all work with the third-generation CERN-designed Function Generator/ Controller (FGC3) and five switched-mode 4-quadrant converter families and five fast-pulsed converter designs are now available under license [1]. The switched-mode converters range from the CUTE (±12.5 A ±15 V) to the SIRIUS (±450 A ±450 V peak, ±200 A ±80 V RMS), which can be combined in series or parallel combinations. The fast-pulsed converters range from 50 A to 3000 A with pulse durations of at least 5 ms and typical repetition rates of 1-2 Hz. The 320 A MaxiDisCap design can operate faster with support for pulse rates up to 10 Hz.

FGC3.1 AND FGC3.2

The current third-generation Function Generator/ Controller is FGC3.1 [2]. This small control computer was de-signed between 2007 and 2011 and went into operation in 2012. More than 2700 have been produced. It runs at 10 kHz and supports a current or field regulation period of 100 µs or multiples of 100 µs. This limits the regulator bandwidth for the rejection of perturbations to around 1 kHz.

Component obsolescence makes further production of the FGC3.1 difficult and some applications need more bandwidth than can be provided with 10 kHz regulation, so the FGC3.2 development was started in 2018 for operation from 2022 [3]. The FGC3.2 will be plug-compatible with the FGC3.1, except for increased power consumption on the +5 V. This may require an upgrade of the PSU, depending on how many other cards are sharing the supply.

If a lab or company wishes to license a CERN power converter design or use the FGC3 controls with an existing power converter, then FGC3.1s are available for small quantities that do not need more than 10 kHz bandwidth. For large quantities or high-performance applications, it will be necessary to wait for the FGC3.2 design to be ready.

Firmware and FPGA Programming

In both cases, CERN will provide the firmware, FPGA programming and support during integration and commissioning as well as long-term updates as part of the license agreement. All the firmware and FPGA programming source code will be available for review, but CERN will not give permission to compile them locally nor to create derivative versions or distribute this code. CERN invites bug reports, bug fixes and feature requests from licensees.

Despite the hardware reaching the end of its production life, the FGC3.1 programming will continue to be supported for at least 20 years. The FGC3.2 should be producible until at least 2027 and it will also be supported at CERN for at least 20 years. We anticipate future refreshes of the design after 2027 to replace obsolete components.

The step from FGC3.1 to FGC3.2 includes a significant change in architecture. The older design is based on the combination of a Microcontroller Unit (MCU) + Digital Signal Processor (DSP) + Field Programmable Gate Array (FPGA), while the newer design will use a multicore ARM-based System-on-Chip (SoC) with an FPGA. The FGC3.1 was a "bare-metal" design with a tiny real-time micro-kernel on the MCU and no operating system on the DSP. With FGC3.2, the System-on-Chip (SoC) will run Linux with the PREEMPT_RT patch.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA106

FGC3.2: A NEW GENERATION OF EMBEDDED CONTROLS COMPUTER FOR POWER CONVERTERS AT CERN

S. Page, C. Ghabrous Larrea, Q. King, B. Todd, S. Uznanski, D. Zielinski,
CERN, Geneva, Switzerland

Abstract

Modern power converters (power supplies) at CERN are controlled by devices known as Function Generator/Controllers (FGCs), which are embedded computer systems providing function generation, current and field regulation, and state control. FGCs were originally conceived for the Large Hadron Collider (LHC) in the early 2000s, though later generations are now increasingly being deployed in the LHC Injector Chain (Linac4, Booster, Proton Synchrotron and Super Proton Synchrotron).

A new generation of FGC known as the FGC3.2 is currently in development, which is intended to provide for the evolving needs of the CERN accelerator complex, and other High Energy Physics (HEP) laboratories via CERN's Knowledge and Technology Transfer programmes. This paper describes the evolution of FGCs, summarises tests performed to evaluate candidate components for the FGC3.2 and details the final hardware and software architectures chosen. The FGC3.2 will make use of a multi-core ARM-based System-on-Chip (SoC) running an embedded Linux operating system in contrast to earlier generations which combined a microcontroller and Digital Signal Processor (DSP) with software running on “bare metal”.

EVOLUTION OF FGC POWER CONVERTER CONTROLLERS

The first Function Generator/Controller, FGC1, was an evolution of the controls developed in the 1980s for the Large Electron Positron (LEP) collider power converters, with one small controller embedded in each converter. In 1997, the MCU was updated and a Texas Instruments C32 DSP was added to support digital regulation of the current. This evolved into a second version, FGC2, used in the LHC with error corrected memory to improve radiation tolerance. In 2007, development started on a third generation, FGC3, with the same MCU + DSP architecture but with newer and more powerful components. The resulting FGC3.1 was put into operation in 2012 [1]. Some 2700 FGC3.1s have been produced to date, for use in the CERN accelerator complex.

NEW REQUIREMENTS

The FGC3.1 embeds a Renesas RX610 32-bit MCU running at 100MHz for global interfaces, a Texas Instruments C6727 DSP running at 300MHz for function generation and regulation and a Xilinx Spartan 3AN FPGA for glue logic. This hardware platform is capable of running at 10 kHz and supports a current or field regulation period of 100 µs or multiples of 100 µs. This limits the regulator bandwidth for the rejection of perturbations to around 1 kHz.

In 2018, the FGC3.2 development project was launched for two main reasons:

1. The FGC3.1 uses components which are now end-of-life, making procurement progressively more difficult and expensive.
2. Certain circuits at CERN need more than 1 kHz current regulator bandwidth.

The goal for the FGC3.2 is to be a plug-compatible replacement for the FGC3.1, ready for operation from 2022, that can be manufactured until at least 2027 and which can provide a significantly higher regulator bandwidth.

There is not a specific bandwidth requirement, rather the objective is to create the fastest possible controller for a similar price to the FGC3.1. The new design will also increase the size of the memory, upgrade the networking speed and will address other issues, such as the complexity of writing software for two different processors.

HARDWARE CHOICES

It was evident that a multi-core System-on-Chip (SoC) would provide the processing resources in the FGC3.2. Extensive market research was carried out, followed by feasibility studies on a few selected SoCs. The following processor families were considered: ARM, Intel Atom, AMD Embedded (G) and High Performance (R), Power Architecture, Intel Quark, AVR32, AVR and PIC microcontroller. Some of these are better suited to small, low-performance embedded systems and only ARM and Intel SoCs were studied further. In total, twenty-one were compared using twenty-seven criteria. Only ARM-based SoCs were retained due to their widespread use in embedded systems, their extensive community support and their superior Thermal Design Power (TDP) to performance ratios. The ARM family consists of three main series: M, R and A. M and R series parts were predicted to be too slow for FGC3.2’s requirements, thus leaving series A (Cortex) as the preferred choice. After comparing the ARM Cortex-based SoCs, the four listed in Table 1 were selected for further testing.

Table 1: SoCs and Evaluation Kits

SoC	Evaluation Kit	Primary Cores
Xilinx Zinq XCZU9EG	EK-U1-ZCU102-G	4 x ARM A53 @ 1.50 GHz
TI AM5728	Beaglebone Black	2 x ARM A15 @ 1.50 GHz
NXP LS1046A	LS1046ARDB-PB	4 x ARM A72 @ 1.80 GHZ
HiSilicon Kirin 960	Lemaker HiKey 960	4 x ARM A73 @ 2.36 GHz

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA109

PYTHON BASED APPLICATION FOR BEAM CURRENT TRANSFORMER SIGNAL ANALYSIS*

M.C. Paniccia†, D.M. Gassner, A. Marusic, A. Sukhanov,
Brookhaven National Laboratory, Upton, USA

Abstract

There are a variety of beam current transformers that are used at all accelerator facilities for current and bunch charge measurements. Transformer signals are traditionally measured using integrator electronics followed by a digitizer. However, integrator circuits have a limited bandwidth and are susceptible to noise. By directly digitizing the output of the transformer, the signal bandwidth is limited only by the transformer characteristics and the digitizing platform. Digital integration and filtering can then easily be applied to reduce noise resulting in an overall improvement of the beam parameter measurements. This paper describes a python-based application that performs the filtering and integration of a current transformer pulse that has been directly digitized by an oscilloscope.

INTRODUCTION

Current transformers have been the standard instrument for current and bunch charge measurements across all accelerator facilities. At the BNL Low Energy RHIC electron Cooling (LEReC) facility [1], a Bergoz Integrating Current Transformer (ICT) is utilized for calibrated bunch charge measurements. The LEReC beam contains a com-plex bunch structure defined by the 704 MHz fiber laser producing individual electron bunches of about 40 ps full length which are grouped together into a single macro-bunch. The macrobunches contain thirty individual bunches and macrobunch trains are separated at ~9 MHz frequency, as shown in Fig. 1.

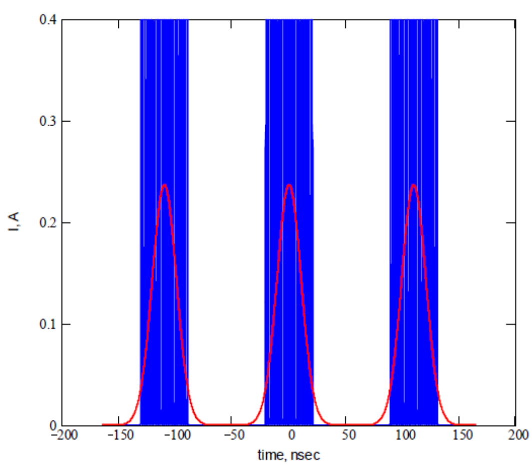


Figure 1: The LEReC beam structure. Thirty electron bunches (blue) spaced by 1.4ns placed on a single ion bunch (red), with ion bunch repetition frequency of 9 MHz.

* Work supported by the US Department of Energy under contract No. DE-AC02-98CH10886.
† mpaniccia@bnl.gov

The ICT provides precise macrobunch measurements down to the pC level and provides a pulse proportional to the bunch charge with a fixed width of approximately 70 ns [2]. Initial configuration of the ICT measurement system used a Bergoz Beam Charge Monitor Integrate/Hold/Reset (BCM-IHR) integrator module. When the LEReC program transitioned to long macrobunch trains and continuous wave (CW) beams, the integrator could no longer be used to calculate beam charge. Therefore, to utilize the ICT during these operational modes, the output of the ICT is directly digitized with an oscilloscope. A custom Python application integrates each pulse to provide bunch-by-bunch charge calculation and estimated beam current.

HARDWARE SETUP

The ICT is a specially designed current transformer that integrates the charge of the passing beam pulse and then outputs the results. The immediate integration allows for the measurement of beam pulses with picosecond pulse widths. In addition, it reduces the effects of pulse-to-pulse crosstalk and eddy-current loss [3]. At LEReC, the ICT is an in-flange module with 5:1 turn ratio and is located in the Injection section of the LEReC accelerator beam line [4]. The output is carried over LDF1-50a, ¼” Heliax cable to the monitoring electronics 100 meters away. A SPDT relay is used to connect the ICT to either the BCM-IHR electronics or the Keysight DSO-X-3024A oscilloscope and can be controlled remotely. Controls for the BCM-IHR gain settings and calibration settings are remotely available using a VME-based digital output module and the output is digitized on a custom Zynq-based digitizer chassis. Both the oscilloscope and the BCM-IHR electronics run off an external 1 Hz trigger.

BCM-IHR vs. Oscilloscope

Macrobunch charge is determined by integrating the output pulse from the ICT. Integration in the BCM-IHR electronics module is performed in two stages: first, a positive integration window, followed by a negative integration window. The two windows are summed together and provide a DC level proportional to the charge captured. The two-window configuration performs a baseline subtraction that reduces the effects of baseline noise and baseline droop. Integration window length can be adjusted using a potentiometer and is limited to 0.1 - 7µs; both windows have the same duration. The module can operate at a maximum repetition rate of 10 KHz [5]. At LEReC, the integration windows are set to 7 µs in order to capture up to 60 macrobunches. However, the BCM-IHR integrator is limited to 40 nC before saturating, limiting the number of macrobunches to 10 at nominal individual electron bunch charges of 130 pC.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA111

EASING THE CONTROL SYSTEM APPLICATION DEVELOPMENT FOR CMS DETECTOR CONTROL SYSTEM WITH AUTOMATIC PRODUCTION ENVIRONMENT REPRODUCTION

I. Papakrivopoulos[1,2], G. Bakas[1], U. Behrens[3], J. Branson[4], P. Brummer[2,5], S. Cittolin[4], D. Da Silva Gomes[2,6], G. L. Darlea[7], C. Deldicque[2], M. Dobson[2], N. Doualot[2,6], J.R. Fulcher[2], D. Gigi[2], M.S. Gladki[2], F. Glege[2], G. Gomez-Ceballos[7], J. Hegeman[2], W. Li[3], A. Mecionis[6,8], F. Meijers[2], E. Meschi[2], R.K. Mommsen[6], K. Mor[2], S. Morovic[4], V. O'Dell[6], L. Orsini[2], C. Paus[7], A. Petrucci[4], M. Pieri[4], D. Rabady[2], K. Raychino[2], A. Racz[2], A. Rodriguez Garcia[2], H. Sakulin[2], C. Schwick[2], D. Simelevicius[2,8], P. Soursos[2], A. Stahl[3], M. Stankevicius[6,8], U. Suthakar[2], G. Tsipolitis[1], C. Vazquez Velez[2], A. Zahid[2], P. Zejdl[2,6]
[1]National Technical University of Athens, Athens, Greece
[2]CERN, Geneva, Switzerland
[3]Rice University, Houston, USA
[4]University of California San Diego, San Diego, USA
[5]Karlsruhe Institute of Technology, Karlsruhe, Germany
[6]Fermi National Accelerator Laboratory, Batavia, USA
[7]Massachusetts Institute of Technology, Cambridge, USA
[8]Vilnius University, Vilnius, Lithuania

Abstract

The Detector Control System (DCS) [1], [2] is one of the main pieces involved in the operation of the Compact Muon Solenoid (CMS) experiment at the LHC. The system is built using WinCC Open Architecture (WinCC OA) and the Joint Controls Project (JCOP) framework [3] which was developed on top of WinCC at CERN. Following the JCOP paradigm, CMS has developed its own framework which is structured as a collection of more than 200 individual installable components each providing a different feature. Every one of the systems that CMS DCS consists of, is created by targeting and installing a different set of these components. By automating this process, we are able to quickly and efficiently recreate systems both in production, but also, to create development environments identical to the production ones. This latter one results in smoother development and integration processes, as the new/reworked components are developed and tested in production-like environments. Moreover, it allows the central DCS support team to easily reproduce systems that the users/developers report as being problematic, reducing the response time for bug fixing and improving the support quality.

INTRODUCTION

Control and real-time monitoring are essential for the successful operation, wellbeing assurance and efficient data taking of the CMS detector. This is where the DCS comes in play. Similar systems are used in all the LHC experiments but the design, structure and implementation differ between them. In CMS the DCS is implemented as a big distributed system [4] where each node, which is called a system or a project, plays a distinct role either providing general infrastructure or having a dedicated role interfacing and interacting with a single subsystem of the experiment.

The DCS community in CMS consists of a central team that is responsible for the control system that handles all the common and generic infrastructure of the experiment and a set of sub-detector teams, each of them being responsible for one or more systems that handle the specific needs of every individual subsystem of the experiment. Apart from being in charge of the general infrastructure, the central team provides support to the rest of the experiment in control system related topics as well as tools that can be reused by the other teams or implement common experiment-wide functionality. Finally it provides administration and operation of all the control system related server infrastructure of the experiment in terms of OS and generic software like WinCC, OPC servers etc.

The experiment's control system is designed in a way of small reusable software entities that are called components. As mentioned above, each system and as an extension its role, is defined by the set of components that are installed into it. All systems start from a minimal initial point differing only by a little and diverge from one another with the installation of the components taking their final state and altogether forming the control system of the experiment. Out of the 200 individual components around half are provided by the central team while the rest are system specific and are developed by the sub-detector teams. Most of the central components, offer generic functionality that can be used and even extended while others implement a specific feature, like the communication with a certain type of device. This modular architecture allows for better maintenance of the system, but at the same time makes it highly dependent on the ability to constantly be able to install components. Furthermore, it renders CMS independent of the specific

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA112

IMPROVING PERFORMANCE OF THE MTCA SYSTEM BY USE OF PCI EXPRESS NON-TRANSPARENT BRIDGING AND POINT-TO-POINT PCI EXPRESS TRANSACTIONS

L.Petrosyan*, DESY, Hamburg, Germany

Abstract

Increasing Performance of the MTCA System by use of PCI Express Non-Transparent Bridging and Point-To-Point PCI Express Transactions.

The PCI Express provides one of the highest data transfer rates today. However, with increase in number of modules in a MTCA crate and client programs, and also with complication of modules and increase in number of module registers, as well with increase in amount of data requested by the users the performance of the whole system plays an important role.

Distributed systems are gaining popularity as they fill the need of next generation systems.

Multiprocessor systems provide not only the ability to increase processing bandwidth, but also allow greater system reliability through host failover.

The use of non-transparent bridges in PCI systems to support intelligent adapters in enterprise systems and multiple processors in embedded systems is well established. In these systems, the non-transparent bridge functions as gateway between the local subsystem and the backplane. Such applications can be ported to PCI Express by the use of non-transparent bridges, with the non-transparent bridge integrated into a PCI Express switch in place of one of the transparent bridges.

PCIE NON-TRANSPARENT BRIDGE

MTCA systems use the PCIe as a central bus. Adding a second CPU to the existing MTCA system, using Non-Transparent Bridging, will allow to increase the performance of the system.

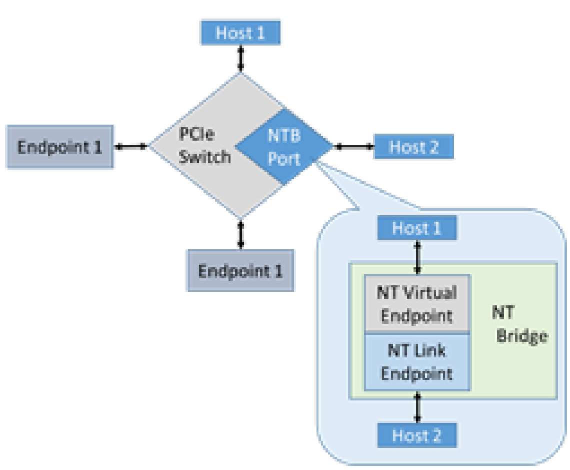


Figure 1: Non-Transparent Bridge.

* ludwig.petrosyan@desy.de

PCIe has to have only one Root Complex, PCIe Bus configuration and memory mappings has to be done by the one Host to avoid the Bus numbering and memory mixing.

Non-Transparent Bridging used to connect two independent address/Host domains; it allows to connect second Host to the existing PCIe bus. A Non-Transparent Bridge consist of two back-to-back PCIe endpoints, a Virtual and Link side endpoints (Fig. 1). A Non-Transparent Bridge isolates the address spaces of different Hosts by appearing as an endpoint to each side.

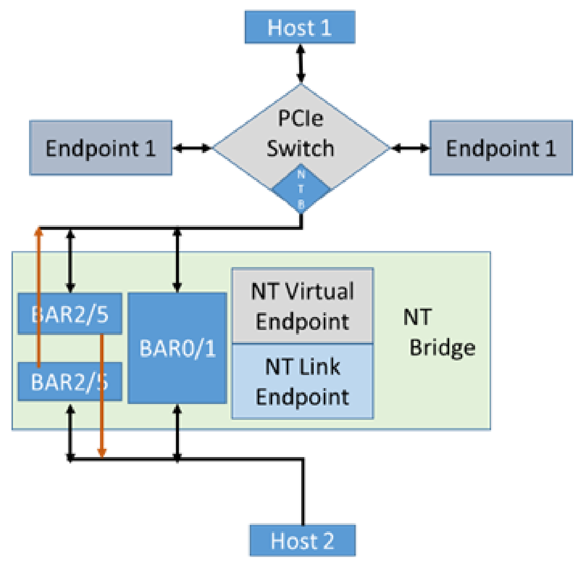


Figure 2: Key elements of the Non-Transparent Bridge.

The Key elements of the NTB (Fig. 2):

- BAR0/1 used for NTB configuration, visible from both sides of the NTB
- Up to 4 BARs, individually enabled
  - BAR 2/5 are aperture into the address space on the far side of the other endpoint, provides ad-dress transaction from one side to other
- 8 Scratchpad Registers (BAR0)
  - Provide a means of communication between two Hosts over a non-transparent bridge. They are readable and writeable from both sides of the NTB
- 16 Doorbell Registers (BAR0)
  - The doorbell registers are used to send inter-rupts from one side of the NTB to other.

The PCIe packets are routed by address and Bus number, so to send packets from one side to other the addresses and bus number have to be translated.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA113

LINUX-BASED PXIe SYSTEM FOR THE REAL-TIME CONTROL OF NEW PAINTING BUMPER AT CERN

M. P. Pimentel*, C. Chanavat, E. Carlier, G. Gräwer, N. Magnin, N. Voumard, T. Gharsa
CERN, Geneva, Switzerland

Abstract

In the framework of the LHC Injectors Upgrade Project, the new connection from Linac4, injecting a 160 MeV H- beam into the Proton Synchrotron Booster (PSB) requires a set of four slow kicker magnets (KSW) per PSB ring to move the beam on a stripping foil, remove electrons and perform phase space painting. A new multiple-linear waveform generator based on a Marx topology powers each KSW, allowing adjustment of the current discharge shape with high flexibility for the different beam users.

To control these complex power generators, National Instruments (NI) PXIe crates fitted with a set of modules (A/D, D/A, FPGA, PROFINET) are used. Initially, control soft-ware developed with LabVIEW has validated the test bench hardware. A full software re-engineering, accessing the hard-ware using Linux drivers, C APIs and the C++ framework FESA3 under Linux CentOS7 was achieved for operational deployment.

This paper describes the hardware used, and the integration of NI PXIe systems into CERN controls environment, as well as the software architecture to access the hardware and provide PSB operators and kicker experts with the required control and supervision.

BACKGROUND

The PSB comprises four superposed rings, each equipped with four KSW magnets located at the injection area. A dedicated multi-waveform pulsed current generator individually powers each magnet [1, 2].

A single of those generators is composed of a master controller module, referred as KSW interlock card (KIC), and five capacitor charger/discharger (CCD) modules, 1x120V and 4x1200V. All of the mentioned modules are equipped with Anybus®-IC interfaces providing PROFINET communication interface [3]. To receive fast control commands, the generator power amplifier receives instructions by means of a hardwired connection with the FPGA controller board on the PXI crate. Finally, a feedback controller board receives an analog reference signal of the required waveform to be played in order to compare it with the discharge feedback and compensate any deviations perfecting the discharge shape. In addition to that, a pickup signal is available to read the final generators discharge.

A single Front-End Computer (FEC) controls and monitors all the mentioned peripherals on four generators corresponding to a ring. Fig. 1 is a simplified illustration of the described system.

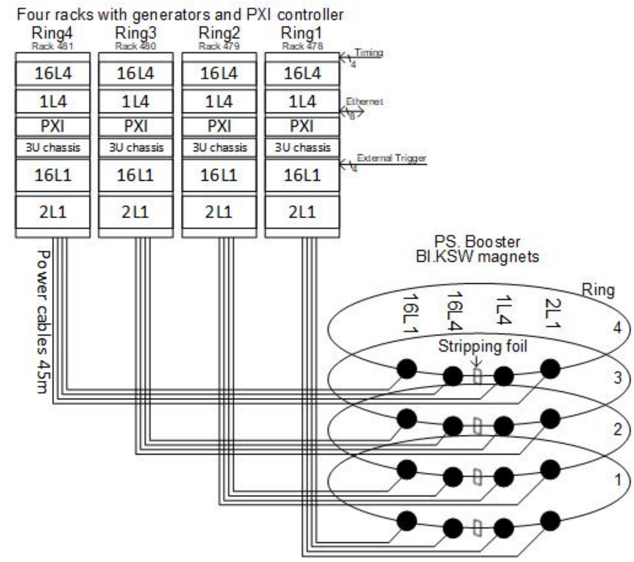


Figure 1: BI.KSW system.

CONTROL HARDWARE

The final FEC setup selected, to control four of those generators, consists in a NI PXIe-1082 chassis equipped with a PXIe-8821 controller. To interface with the PROFINET modules, the control crate is equipped with a Kunbus® DF-Profinet-IO operating as bus controller. The control crate hosts as well a field-programmable gate array (FPGA), NI PXIe-7841R, controlling four power amplifiers acting di-rectly in synchronisation with timing triggers. To ensure in-ternal processes synchronisation with the central timing, and generate part of the necessary timing synchronous triggers, a CERN custom hardware module, central timing receiver (CTR), incorporates the control crate. To generate the analog reference waveform, the crate also features an NI PXIe-6358 DAC module. In order to read the magnet discharge shape, a NI PXI-6132 digitizer is also incorporated.

OBJECTIVES

For early testing and validation of the prototype pulsed current generator, a test bench has been set up with all the components constituting a single ring. During this phase of prototyping and development, a first expert control system has been developed using LabVIEW™ as development environment.

In an effort to optimise the integration within the CERN accelerator control system and profit of all the existing generic tools, a Linux based approach has been retained for the final operational control system design. The software

* CERN TE-ABT-EC, marco.pimentel@cern.ch

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA114

ACHIEVING OPTIMAL CONTROL OF LLRF CONTROL SYSTEM WITH ARTIFICIAL INTELLIGENCE *

R. Pirayesh
Department of Mechanical Engineering, University of New Mexico, Albuquerque, NM, USA
J. A. Diaz Cruz, S. G. Biedron[1], M. Martinez-Ramon, S. I. Sosa
Department of Electrical and Computing Engineering,
University of New Mexico, Albuquerque, NM, USA
[1] also at Department of Mechanical Engineering,
University of New Mexico, Albuquerque, NM, USA

Abstract

Microphonics and Lorentz Force Detuning are common sources of detuning in Superconducting Radio-Frequency (SRF) cavities and continuous wave (CW) systems. Requirements as tight as 10 Hz, are common in such systems and are equivalent to change in cavity length of only a few nanometers. Traditional approaches to mitigate detuning in SRF cavities consist of mechanical modifications of the cavity/cryomodule environment and advance control techniques such as active compensation. In this research, we explore Artificial Intelligence (AI) techniques that can improve existing control systems to ensure better performance and lower detuning. Machine learning (ML), as part of AI, can learn the complexity and non-linear behaviour of the system. Deep Learning (DL), as one of the greatest algorithms in ML, can scale and distribute well on the cores of high-performance computers (HPC). This enables the controller to learn the huge amount of data coming from diagnostic instrumentation. Furthermore, Gaussian Process (GP) can be used in parallel with DL to increase the performance of ML. We describe such AI implementation on a computer model of the RF control of an SRF cavity for LCLS-II. This model, called Cryomodule-on-Chip (*CMOC*), was developed at LBNL. It is our main goal to implement such AI-supported control for the compensation of microphonics in LCLS-II SRF cavities.

INTRODUCTION

The quality of the electron beam affects the quality of the X-rays produced in Free Electron Lasers. The amplitude and phase of the electromagnetic fields in accelerating cavities are controlled by the Low Level RF (LLRF) control system. In particular, for the Linac Coherent Light Source upgrade (LCLS-II), the LLRF must provide high stability of the phase and amplitude to produce narrow-band hard X-rays [1].

In this contribution, we investigate the use of AI as a tool to enhance the stability of the LLRF system. We utilize *CMOC*, a software developed at LBNL, to model the different noises present in the system; later on *CMOC* is also used to model the LLRF. AI and ML have been previously used in conjunction with control systems for numerous applications: We utilized formation flying control systems to keep two satellites in formation [2–4]. We also used different AI frameworks to enhance the performance of the control [5–7], similar approaches can be implemented for the control of particle accelerators and their sub-components, e.g. the LLRF. We have previously explained the challenges of applying these AI techniques to LLRF [8].

* The study at the University of New Mexico was supported by DOE Contract DE-SC0019468

LCLS-II AND LLRF MODEL

LCLS-II is composed of 35 cryomodules, each with 8 SRF cavities used for staged acceleration of an electron beam. The RF power driving each of these 280 accelerating cavities is provided by Solid State Amplifiers (SSA) [9] and controlled by the LLRF. The current LLRF framework is a proportional and integral (PI) controller that is implemented in an FPGA [10] and is sketched in Fig. 1.

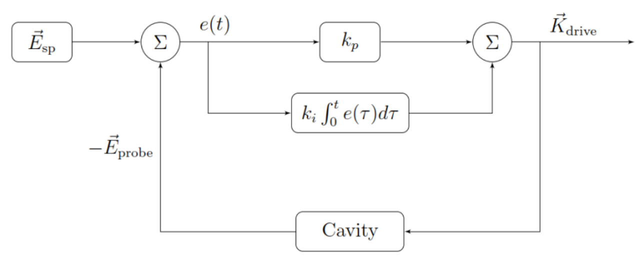


Figure 1: Diagram of a PI Controller.

The chosen gains for this PI controller are $k_p = 1200$ and $k_i = 3.8{\times}10^7$ [11]. These gains can further be tuned dynamically with ML algorithms, according to the cavity probe signal, forward power, reverse power, and other system parameters, e.g. minimizing reflected power and losses. As a result, the gain coefficients will take values from a set of optimal parameters that minimize errors depending un the particular state of the system.

Cavity Model

A model of the system encompassing the SRF cavity, the LLRF, and the cryomodule was developed by the LLRF team at LBNL and has been used to study the electrodynamic performance of the system. For a cavity with several electromagnetic modes, each mode can be represented by a resonant circuit model, see Fig. 2.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA115

CODE GENERATION TOOLS AND EDITOR FOR MEMORY MAPS

P. Plutecki*, B. Bielawski, A. Butterworth
CERN, Geneva, Switzerland

Abstract

Cheburashka, a toolset created in the Radio Frequency Group at CERN, has become an essential part of our hardware and software developments. Due to changing requirements, this toolset has been recently rewritten in C++ and Python. A hardware developer, using the graphical editor, defines a memory map, which is subsequently used to ensure consistency between software and hardware. The memory map file is an input for a variety of tools used by the hardware engineers, such as VHDL code generators. In addition to aiding the firmware development, our tools generate C++ wrapper libraries. The wrapper provides a simple interface on top of a Linux device driver to read and write registers by exposing memory map nodes in a hierarchical way, performing all low-level bit manipulations and checks internally. To interact with the hardware, a software that runs on a front-end computer is needed. Cheburashka allows us to generate FESA (Front-End Software Architecture) classes with parts of the operational interface already present. This paper describes the evolution of the graphical editor and the Python tools used for C++ code generation, along with a description of their main features.

INTRODUCTION

Cheburashka [1], developed in Java, has been serving both as an editor of XML memory maps and a code generation tool for FESA [2] (Front-End Software Architecture) and the driver wrapper. For the VHDL code generation, Gena, a tool written in Python, has been used. At some point it has been decided to rewrite the code generators in Python, with extensive use of templates. The file format of the memory maps has been changed to YAML [3] (YAML Ain't Markup Language), also to work with a new tool called Cheby [4], a VHDL generator developed by the Hardware and Timing section, Controls group at CERN, the successor of Gena. Despite YAML being more human-readable than the original format (XML), considering the complexity of memory maps for devices which have been developed at the Radio Frequency group and users' habits, a new memory map editor was needed.

MEMORY MAP EDITOR

The new GUI (Graphical User Interface) has been developed to work with new file format (YAML) for memory maps. Since many code generation tools written in Python were already production ready, instead of having a monolithic program for editing memory maps and generating code, it has been decided to have the editor split into two parts:

- a generic C++ core, that, based on a schema file, can work with a single document YAML file,
- a set of Python scripts that will provide features specific for the memory map editor and possibility of running external Python tools, such as code generators.

To achieve this, a Python 3.6 interpreter has been embedded in the editor, using pybind11 [5] library.

As it is not a trivial task to design an intuitive interface, the GUI is composed of dockable widgets (see Fig. 1), which can be freely rearranged by the user. These widgets can be hidden, detached, tabbed and resized. All changes done to the layout are saved in the user's configuration file, so they are persistent.

Core

The core of the editor has been written in modern C++, using Qt5 [6] libraries. For parsing YAML files, yaml-cpp [7] has been selected, as it is a C++11 library that supports the latest YAML standard (1.2).

The editor needs a schema file, which defines the structure of a YAML file that is being edited. The schema, also a YAML file, specifies allowed mappings and sequences. Moreover it defines basic validation rules for scalars:

- if it is required,
- its type:
  - boolean,
  - integer,
  - hexadecimal integer,
  - floating-point,
  - string,
  - enumeration,
  - file.
- regular expression pattern matching,
- its default value,
- in case of numerical types, a range (minimum and/or maximum value).

In most cases, these validation rules are sufficient, although sometimes it is necessary to define more sophisticated checks. For this, a validation function written in Python can be bound to a scalar or a mapping.

In addition to the validation, auxiliary attributes can be defined in the schema, such as a tooltip text, if a scalar can be added, removed or edited by the user. The latter are

* p.plutecki@cern.ch

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA117

BIG DATA ARCHIVING FROM ORACLE TO HADOOP

I. Prieto Barreiro, M. W. Sobieszek. CERN, Geneva, Switzerland.

Abstract

The CERN Accelerator Logging Service (CALS) is used to persist data of around 2 million predefined signals coming from heterogeneous sources such as the electricity infrastructure, industrial controls like cryogenics and vacuum, or beam related data. This old Oracle based logging system will be phased out at the end of the LHC's Long Shut-down 2 (LS2) and will be replaced by the Next CERN Accelerator Logging Service (NXCALS) which is based on Hadoop. As a consequence, the different data sources must be adapted to persist the data in the new logging system. This paper describes the solution implemented to archive into NXCALS the data produced by QPS (Quench Protection System) and SCADAR (Supervisory Control And Data Acquisition Relational database) systems, which generate a total of around 175,000 values per second. To cope with such a volume of data the new service has to be extremely robust, scalable and fail-safe with guaranteed data delivery and no data loss. The paper also explains how to recover from different failure scenarios like e.g. network disruption and how to manage and monitor this highly distributed service.

INTRODUCTION

CALS is used to persist and retrieve billions of data acquisitions per day and is considered a mission critical service. The data is coming from heterogeneous sources such as the CERN accelerator complex, related subsystems and experiments [1]. The logging system was initially designed for a throughput of 1 TB/year and it is currently operating far over its design limits, storing over 1.2 TB/day. The underlying storage of CALS is based on Oracle Database and uses an old monolithic architecture which is difficult to scale up [2]. CALS will be phased out at the end of the LHC's Long Shut-down 2 (LS2) in 2020 and will be replaced by the Next CERN Accelerator Logging Service [3] (NXCALS) which is based on Hadoop.

The replacement of the logging system will have an impact on many client applications which use a database to database transmission mechanism to archive the data in the logging system. This is the case for QPS (Quench Protection System) and SCADAR (Supervisory Control And Data Acquisition Relational database) [4], which are first storing their data in a temporary database and then transferring a sub-set of the data to CALS for permanent storage using PL/SQL. As NXCALS is based on Hadoop, this approach of data transmission is technically not possible and therefore it was necessary to implement a new mechanism to feed the new logging system with the data.

This paper describes the solution implemented to archive in NXCALS the data produced by QPS and SCADAR. The client systems of NXCALS must register the signals before archiving their data values, and these two systems have registered over 1.6 million signals in the logging system which generate a total of around 175,000 values per second. Therefore, the implemented solution must cope with big data volumes and it was designed to be extremely robust, scalable and fail-safe with guaranteed data delivery and no data loss. The paper also explains how to recover from different failure scenarios like e.g. network disruption and how to manage and monitor this highly distributed service.

SYSTEM OVERVIEW

The tasks performed by the new data source for NXCALS, named WinCC OA Data Source (WCCDS), are conceptually simple:

1. Get the list of signals registered in the logging system.
2. For each registered signal, get the list of values not yet transmitted to the logging system and send them to NXCALS.
3. Wait for the reception acknowledge of the transmitted values and mark them as logged.
4. Allow data re-transmissions on user demand.
5. Manage the registration, removal and modification of the signals in the logging system.

The implementation of the WCCDS service was split into two main processes: Datasource process for the first four aforementioned tasks and Metadata process for the last task. Figure 1 shows the system overview; the different QPS and SCADAR applications store their data and metadata in two different database schemas. The data schema contains the values and timestamps produced by all the signals of the system. The metadata schema keeps track of the signals registered in the logging system and contains additional information, like the name of the table where the signal values are stored or the timestamp of the last values archived in NXCALS for each signal. The WCCDS service queries both schemas and transmits the data and metadata to NXCALS. When the service receives the reception acknowledge for the transmitted data, it marks the data as transferred to NXCALS.

Data Partitioning and Scaling

The volume of data to be transmitted to the logging system is too big to be handled by a single JVM (Java Virtual Machine) process and the WCCDS service profits from the data partitioning defined by CALS, where a signal belongs to a data category and a data category belongs to a data transfer group. For example, QPS has over 135,000 signals registered in the logging system which are partitioned into 53 data categories. The data categories are spread over 20 data transfer groups.

This data partitioning allows a simple distribution of the data transmission among different JVM processes and machines depending on the capacity of the available resources.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA118

IMPROVING ALARM HANDLING FOR THE TI OPERATORS BY INTEGRATING DIFFERENT SOURCES IN ONE ALARM MANAGEMENT AND INFORMATION SYSTEM

M. Bräger*, M. Bouzas Reguera, U. Epting, E. Mandilara, E. Matli, I. Prieto Barreiro, M. P. Rafalski, CERN, Geneva, Switzerland

Abstract

CERN uses a central alarm system to monitor its complex technical infrastructure. The Technical Infrastructure (TI) operators must handle a large number of alarms coming from several thousand equipments spread around CERN. In order to focus on the most important events and improve the time required to solve the problem, it is necessary to provide extensive helpful information such as alarm states of linked systems, a geographical overview on a detailed map and clear instructions to the operators. In addition, it is useful to temporarily inhibit alarms coming from equipment during planned maintenance or interventions. The tool presents all necessary information in one place and adds simple and intuitive functionality to ease the operation with an enhanced interface.

INTRODUCTION

Alarm systems are an essential component in all environments where technical equipment is supervised by control systems. At CERN, different alarm system solutions have been designed over time and were mostly dedicated to cover precise expert systems. With the grouping of several control rooms in the CERN Control Center (CCC) and the increasing infrastructure for LHC operation, more equipment needed to be supervised by the Technical Infrastructure (TI) operators. The existing tools and their technology were developed in the last millennium and it was difficult to maintain these with young developers and modern tools.

Today's alarm system integrates state of the art technologies and provides the operators with all functionality for efficient alarm handling. Each alarm has information about precise location, fault cause and consequences and the actions to be done. The information is concentrated in a tool that easily allows to identify alarm avalanches and provides detailed information for single alarms. The official CERN map is used to give a geographical overview with animated live alarm information. Extensive filtering possibilities have been introduced, which makes the tool usable not only for the TI operators, but also for the equipment specialists and other services like the CERN fire brigade.

The paper provides an overview of the chosen architecture and design decisions for ALIS to deliver an improved alarm handling experience to the end-users.

* Matthias.Braeger@cern.ch

MOTIVATION AND FUNCTIONALITY

Today's core requirements for a sophisticated alarming tool have not changed much since the early 90's [1]. Only data variety and velocity have tremendously increased. In first place stakeholders want a service that is agnostic to all kind of errors and helps to quickly get an overview in critical situations. Due to the regular feedback of the TI operators CERN's in-house alarm systems became over the years more and more sophisticated and today we are technically able to provide a state-of-the-art ALarm Information System (ALIS) that integrates many long-requested features and consolidates the current diversity of tools. To fulfill modern user needs ALIS must be accessible from all kind of devices including mobile devices. This naturally lead to the choice of creating a web application instead of a traditional industrial SCADA desktop program.

The application comes with an intuitive search interface to browse through more than 170'000 TI alarms. The biggest alarm sources are electrical devices, security related equipments such as access doors or smoke- and fire detectors, and pre-calculated alarms from other SCADA devices. As speed was one of the main requirements to the new system, search results are non-blocking and provided in between 300ms (search for all active alarms) to maximum few seconds. The filtered alarms can then be displayed either as a list or on an interactive map including live updates. Furthermore, the user can consult all alarm details from the so-called Single Alarm Page that is opened when selecting a particular alarm.

To prevent unauthorised access to the sensible content the tool makes use of the CERN OAuth2 Single-Sign-On service.

ARCHITECTURAL OVERVIEW

The ALIS system is based on an existing open-source infrastructure called CERN Control and Monitoring Platform (C2MON) [2] [3] [4] (see also Fig. 1). C2MON is used at CERN as back-end for the Technical Infrastructure Monitoring (TIM) [5] service which is acquiring and storing sensor data for a multitude of SCADA applications and user groups. TIM also provides live updates for the previously mentioned 170'000 TI alarms, which are evaluated out of 230'000 input tags coming from various hardware and software sources. The service has a separate web-based and domain specific workflow-driven data entry system called MoDESTI [6]. It allows to enter and review all relevant data for a proper alarm treatment. MoDESTI takes care of validating the entered information before storing into the TIM

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA121

GENERIC DATA ACQUISITION INTERFACES AND PROCESSES IN SARDANA

Z. Reszela†, J. Andreu, G. Cuni, T. M. Coutinho, C. Falcon-Torres, D. Fernandez-Carreiras, R. Homs-Puron, C. Pascual-Izarra, D. Roldan, M. Rosanes-Siscart, ALBA-CELLS, Barcelona, Spain
M. T. Nunez Pardo de Vera, DESY, Hamburg, Germany
A. Milan-Otero, MAXIV Laboratory, Lund, Sweden
G. W. Kowalski, SOLARIS, Krakow, Poland

Abstract

Users visiting scientific installations aim to collect the best quality data frequently under time pressure. They look for complementary techniques at different sites and when they arrive to one they have limited time to understand the data acquisition architecture. In these conditions, the availability of generic and common interfaces to the experimental channels and measurements improve the user experience regarding the programming and configuration of the experiment. Here we present solutions to the data acquisition challenges provided by the Sardana scientific SCADA suite. In one experimental session the same detector may be employed in different modes *e.g.*, getting the data stream when aligning the sample or the stage, getting a single time/monitor controlled exposure and finally running the measurement process like a step or continuous scan. The complexity of the acquisition setup increases with the number of detectors being simultaneously used and even more depending on the applied synchronization. In this work we present recently enriched Sardana interfaces and optimized processes and conclude with the roadmap of further enhancements.

SARDANA INCREMENTAL AND ITERATIVE DEVELOPMENT

ALBA Synchrotron [1] was designed, constructed and commissioned in 2003 - 2012. In the selection process of its control system, commercial SCADAs competed against open source projects: EPICS [2] and Tango [3]. At that time these last two had been already successfully applied in numerous scientific installations. Finally the Tango Control System was selected [4] however its direct application to the experiment control was lacking many specific features like, for example, generic interfaces of the laboratory equipment, flexible sequencer or turn-key and customizable scans. Therefore, the Sardana project [5, 6], highly inspired on SPEC [7], was started in 2007. Its initial scope was mainly focused on the motion control, data acquisition and storage for the needs of step scans or similar measurement procedures. Works on the Taurus project [8, 9] dedicated to graphical user interfaces (GUI) started at a similar time. Both projects together compose the Sardana SCADA Suite which consists of Python libraries extendable with plugins and exported to the Tango control system.

In 2013, DESY, MAXIV and SOLARIS together with ALBA established a collaboration on Sardana and Taurus. At the same time Taurus started gaining popularity within the Tango collaboration and became the *de facto* standard for building client applications in Python. From then on, many enhancements were proposed and implemented by the community of developers *e.g.*, native integration of diffractometers, hardware and software synchronized continuous scans or improved integration of one and two dimensional fast detectors (1D and 2D). In the next chapters we will concentrate on the data acquisition possibilities offered by Sardana.

STRIVE FOR OPTIMAL USER EXPERIENCE

When synchrotron users arrives to the laboratory they have limited time to understand the beamline components and the data acquisition architecture. At the same time they would like to tune the experiment parameters to achieve the best quality of data. Similarly, beamline staff newcomers would like to understand the data acquisition setup as quickly as possible to be able to exploit it to the maximum and prepare it in the best way for the upcoming users. For both use cases, a standard experiment control software is highly commendable.

The user interacts with the beamline control system viagraphical or command line interfaces (CLI). Each beamline at ALBA has, among others, one Taurus GUI application with a *synoptic view* of the line - a high level navigation tool enabling single-click access to the involved instruments. When interacting with the synoptic the *instrument panels* show all the relevant control and acquisition points represented by the *element widgets*. Thanks to the generic interfaces of the Sardana elements, any type of *moveable* (anything that can be controlled by commanding a set point *e.g.*, a motor, a temperature controller or a power supply) is represented by a *motor widget*. The same applies to the experimental channels: the *channel widget* provides a basic acquisition control interface to a counter, 2D detector, etc., hiding the complexity of the specific configuration behind an optional form with expert parameters. The live-view of the acquisition points *e.g.*, either motor's position or channel's current value, is as easy as drag-and-dropping from the element widget to plots or trends. The flexibility, immediacy and ease-of-use of this set of tools is specially welcome in highly dynamic contexts such as manual

† zreszela@cells.es

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA123

VACUUM CONTROLS CONFIGURATOR: A WEB BASED CONFIGURATION TOOL FOR LARGE SCALE VACUUM CONTROL SYSTEMS

A. Rocha†, I. Amador, S. Blanchard, J. Fraga, P. Gomes, G. Pigny, P. Poulopoulou, C. Lima
CERN, Geneva, Switzerland

Abstract

The Vacuum Controls Configurator (vacCC) is an application developed at CERN for the management of large-scale vacuum control systems. The application was developed to facilitate the management of the configuration of the vacuum control system at CERN, the largest vacuum system in operation in the world, with over 15,000 vacuum devices spread over 128 km of vacuum chambers. It allows non-experts in software to easily integrate or modify vacuum devices within the control system via a web browser. It automatically generates configuration data that enables the communication between vacuum devices and the supervision system, the generation of SCADA synoptics, long and short term archiving, and the publishing of vacuum data to external systems. VacCC is a web application built for the cloud, dockerized, and based on a microservice architecture. In this paper, we unveil the application's main aspects concerning its architecture, data flow, data validation, and generation of configuration for SCADA/PLC.

INTRODUCTION

In the early 2000's, during the construction of the LHC and anticipating a considerable increase in the number of vacuum devices to be controlled, a software application (vacDB-Editor) and a set of databases (vacDB) were developed to homogenize and automate the configuration of the SCADA and PLCs for CERN's vacuum systems. Figure 1 shows a simplified overview of this application with its main building blocks.

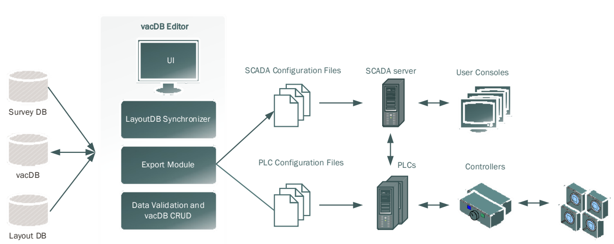


Figure 1: vacDB-Editor overview.

Users interact with the vacDB-Editor user interface, where they can modify the configuration of the control system (adding/removing equipment, modifying equipment attributes, configuring vacuum sectors, alarms). After validating user input, data is persisted in vacDB, and from it, an export functionality generates the configuration files that are used by both the SCADA (WinCC-OA) and PLCs (Siemens S7/TIA). These files allow PLCs to communicate with device controllers, enable the communication between the SCADA and PLCs, and configure all SCADA functionalities such as automatically-generated graphical interfaces, archiving, alarms, and sharing of data with external systems.

In addition, the vacDB-Editor provides a functionality to import equipment and sectorization data from CERN's Layout database. All of the described functionalities (user interface, configuration exporter, Layout DB synchronizer, and data validation & persistence) are packaged into a single monolithic application, which runs on each user's desktop computer.

The Need for Upgrading the vacDB-Editor

Over the past years it has become increasingly difficult to maintain and upgrade the existing vacDB-Editor application. This is due to its obsolete technology stack, written in Java 6, using an old version of Oracle's ADF framework, whose development is dependent on a no longer supported IDE, JDeveloper 10g, released in 2007 [1]. Because of mandatory upgrades of the Java runtime environment at CERN, the vacDB-Editor, using older versions of Java, became more and more unstable, with frequent bugs and crashes reported by the users.

In late 2018 it was announced at CERN that a mandatory update had to be performed to upgrade Oracle databases [2], from version 11g to 18c, on all of CERN's production instances, affecting vacDB. Since Oracle 18c requires a more recent JDBC driver, not available in JDeveloper 10g, an unsupported migration of the vacDB-Editor was performed to support the new driver. While this migration was extremely difficult to perform, it appears to have been successful. It is however impossible to be sure that it will continue working as new Java runtime environments get deployed at CERN. Combining the technical reasons above with the scarce user base of ADF and JDeveloper, it was decided to rewrite the vacDB-Editor application using modern technologies, on a micro-service architecture, allowing us to be more resilient to technological advancements in the future. The new version of the vacDB-Editor is called vacCC, short for *Vacuum Controls Configurator.*

HIGH LEVEL ARCHITECTURE

vacCC is based on a microservice architecture, where each functionality is handled by an independently deployable application [3]. Although more complex to implement due to the increasing number of software parts, interactions, and underlying infrastructure, the usage of microservice architectures has been shown to bring important advantages over monolithic applications. We

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA124

LOCAL OSCILLATOR REAR TRANSITION MODULE FOR 704.42 MHz LLRF CONTROL SYSTEM AT ESS*

I. Rutkowski, K. Czuba, M. Grzegrzółka, Warsaw University of Technology, Poland

Abstract

This paper describes the specifications, architecture, and measurements' results of the MTCA-complaint Local Oscillator (LO) Rear Transition Module (RTM) board providing low phase noise clock and heterodyne signals for the 704.42 MHz Low Level Radio Frequency (LLRF) control system at the European Spallation Source (ESS). The clock generation and LO synthesis circuits are based on the module presented at ICALEPCS 2017. The conditioning circuits for the input and output signals must simultaneously achieve the desired impedance matching, spectral purity, output power as well as the phase noise requirements. The reference conditioning circuit presents an additional challenge due to input power range being significantly wider than the output range. The circuits monitoring the power levels of critical signals and voltages of supply rails for remote diagnostics as well as the programmable logic devices used to set the operating parameters via Zone3 connector are described.

INTRODUCTION

Modern linear particle accelerators subject charged subatomic particles (or ions) to electormagnetic field in order to increase their energy. A Low Level Radio Frequency (LLRF) control system stabilizes the electromagnetic field inside accelerating modules using the amplitudes and phases of the cavity RF signals detected by down-converting them to an intermediate frequency. The signals are then sampled and digitized by high-speed precise analog-to-digital converters. The architecture necessitates synthesis of clock and LO signals. Quality of those signals influences the performance of the field detection.

The European Spallation Source (ESS) LLRF control system is based on the Micro-Telecommunications Computing Architecture (MTCA) [1]. The two aforementioned signals are generated by the LO board having a rear transition module (RTM) form factor and supplying 4 neighboring LLRF systems [2]. Important parts of the design are based on a module designed for the Cavity Simulator project [3] known as CS-LOG and described in [4].

The next section lists requirements of the LO-RTM project.

REQUIREMENTS

The input reference frequency is 704.42 MHz and the nominal input power is +3 dBm ± 2 dB. The clock to reference frequency ratio is 1/6 and the IF to reference frequency ratio is 1/28 or 1/22 (corresponing to 117.40 MHz and 25.16 or 32.02 MHz, respectively). LO and clock signals' phase noise requirements are presented in Table 1 and they are dependent on the sufficient quality of the reference signal. Other requirements:

- compliant with the MTCA.4 RTM specification,
- form factor: RTM (mid-size or full-size),
- combined power good signal for all power supply voltages and information about board temperature shall be provided to the MTCA.4 management,
- 1 reference output (power level: +13 dBm ± 1 dB),
- 4 main LO outputs (power level: +15 dBm ± 1 dB),
- 4 main clock outputs (power level: +15 dBm ± 1 dB),
- the main signals connector type: SMA,
- 1 monitoring LO output and 1 monitoring clock output (power level: > -20 dBm),
- the monitoring output signals connectors type: MMCX,
- RF signals connectors' location: front panel,
- clock and LO signals spectrum: sine waves with maximum harmonic spurious level of -60 dBc,
- the non-harmonic spurious not greater than -60 dBc for clock and not greater than -50 dB for LO,
- the VSWR on each port (Reference Input, LO Outputs, CLK Outputs) not greater than 1.5 (corresponding to return loss of -14.0 dB),
- the maximum absolute error of the power detection: 1 dB,
- resolution of power detection: 0.05 dB or better,
- the module shall be controlled by the AMC module through Zone3 connector,
- the pinout of the Zone3 connector: compliant with the DESY digital class D.1.0 [5],
- 1 clock signal shall be provided at the Zone3 connector,
- Zone3 signaling standard: LVDS,

DESIGN

This section describes circuits introduced in LO-RTM to fulfill the new requirements.

LO Signal Conditioning and Distribution

The selection of LO frequency synthesis scheme as well as selection of the IF divider and amplifier were presented in [4]. Direct Analog synthesis scheme was selected for the design because it introduces no systemic frequency error and reduces the far-from-carrier uncorrelated phase noise.

The circuit verified on CS-LOG was modified to achieve the desired output power level at 4 outputs. A 1W HBT linear amplifier is used as the final driving stage in addition to the low-noise preamplifier (see Fig. 1). An attenuator sets the second stage's point of operation. A coupler and resistive power divider provide signal to the power detector and the monitoring output.

* Work supported by Polish Ministry of Science and Higher Education, decision number DIR/WK/2016/03

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA131

WASTE HEAT RECOVERY FOR THE LHC COOOLING TOWERS: CONTROL SYSTEM VALIDATION USING DIGITAL TWINS

B. Schofield*, M. O. Peljo, W. Booth, E. Blanco, CERN, Geneva, Switzerland

Abstract

In order to improve its energy utilization, CERN will deploy a Waste Heat Recovery (WHR) system at one of the surface sites of the Large Hadron Collider (LHC) which will provide heating power to a local municipality. To study the effects that the heat recovery plant will have on the cooling system, a 'digital twin' of the cooling plant was created in the simulation tool EcosimPro. The primary question of interest was whether the existing control system of the cooling plant would be capable of handling transients arising from a sudden shutdown of the heat recovery plant.

The simulation was connected via the communication protocol OPC Unified Architecture (OPC-UA) to a Programmable Logic Controller (PLC) implementing the cooling plant control system. This 'virtual commissioning' setup was used to study a number of scenarios representing different cooling loads, ambient temperature conditions, and heat recovery plant operating points. Upon completion of the investigation it was found that the current cooling plant control system will be sufficient to deal with the transients arising from a sudden stop of heat recovery plant operation. In addition, it was shown that an improvement in the controls could also enhance the energy savings of the cooling towers.

INTRODUCTION

To minimize the environmental impact of CERN's activities, an environmental commitment has been agreed on [1]. WHR has been identified as a key measure to increase the energy efficiency [2]. Preparations have started for installing WHR on the cooling sites, and the project is starting with a pilot at LHC point 8, providing heat from the primary cooling water for use by the nearby municipality of Ferney-Voltaire [2]. During the early design phase of the WHR plant, the question was raised as to whether the existing cooling plant and its associated control system would be capable of rejecting disturbances in the cooling water temperature caused by a failure of the WHR plant. The cooling towers supply primary cooling water to cryogenic refrigeration plants, which are critical for operation of the LHC. Transient events in the cooling plant could cause the secondary cooling circuits for the cryogenics to shut down, which would then halt the operation of the LHC. In order to investigate the effects that the WHR plant might have on the primary cooling water system, it was decided to use a virtual commissioning approach to verify the performance of the existing control system under a set of temperature transients caused by a sudden loss of the WHR.

* brad.schofield@cern.ch

VIRTUAL COMMISSIONING USING DIGITAL TWINS

The principle of virtual commissioning is to connect a production-ready control system implementation with a simulation model (a 'digital twin') of the process to be controlled. This practise allows engineers to detect errors earlier in the development process, and facilitates the final commissioning phase. Lee and Park [3] provide an overview of virtual commissioning as used in manufacturing processes. They identified that the main obstacle for wider application of virtual commissioning is the model building, which requires in-depth expertise, both in modeling and control engineering. Model validation is then a critical step in order to give credibility to the results. However, once a model has been developed, its usefulness does not end at the commissioning phase, as it may be used continuously for operator training, and evaluation of updates to control strategies. At CERN, a process simulation of cryogenic refrigeration plants of the LHC has been developed by Bradu, Gayet, and Niculescu [4]. The simulation model connects to the existing control and supervision systems, and is used extensively for operator training. Virtual commissioning has also been employed by Booth, Blanco Viñuela, Bradu, and Sourisseau [5] for the development of the heating, ventilation and air conditioning (HVAC) system of the Compact Muon Solenoid (CMS) experimental cavern.

In this paper, the development of a model of the primary cooling water plant at LHC Point 8 is presented. The model was implemented in EcosimPro, a multidomain modeling and simulation tool. The EcosimPro model was then connected to a PLC running a copy of the currently operational version of the cooling plant control system using OPC-UA, and a number of simulated scenarios identified by the process experts were evaluated.

MODELLING AND IDENTIFICATION OF EVAPORATIVE COOLING TOWERS

The focus of this work was the development of a mathematical model of the main element of the primary water cooling plant, namely the evaporative cooling towers. The primary water cooling plant at LHC point 8, known as Surface Fluid 8 (SF8), has five main cooling towers, as well as two backup towers. A schematic of the cooling plant is shown in Fig. 1. The five main towers receive the combined return water from three primary cooling circuits, and collect the cooled water in a common basin from where it is supplied to these three circuits. One of the circuits (corresponding to the cryogenics equipment) can be rerouted to the backup towers; however modelling of these towers was

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA132

CONTROL SYSTEM INTEGRATION OF MAX IV INSERTION DEVICES

J. Lidón-Simón, P. Sjöblom, A. Dupré, G. Todorescu, H. Al-Sallami, N. Al-Habib, A. Thiel, M. Lindberg, V. Hardion, MAX IV Laboratory, Lund, Sweden

Abstract

During the last 3 years, MAX IV has installed and commissioned in total 15 insertion devices out of which 6 are new in vacuum undulators, 1 in vacuum wiggler, and 7 in-house developed and manufactured Apple II elliptically polarized undulators. From the old lab, MAXLAB, 1 PU is also reused. Looking forward, 3 additional insertion devices will be installed shortly. As MAX IV only has one Control and IT group, the same concept of machine and beamline installation have been applied also to the insertion devices, i.e. Sardana, Tango, PLC, and IcePAP integration. This has made a seamless integration possible to the rest of the facility in terms of user interfaces, alarm handling, archiving of status, and also future maintenance support.

INTRODUCTION

The purpose of a synchrotron facility is to create a flux of photons that can interact with atom electron shells of a variety of samples that range from proteins crystals to semiconductors. The photon flux is created by forcing a highly relativistic beam of electrons to deflect back and forth over its ideal golden orbit by making them travel along a row of permanent dipole magnets with alternating polarity.

The devices hosting the magnet arrays and the mechanics to move them around the beam are called insertion devices (IDs). If the magnet arrays are arranged to give a broad photon energy spectrum, the system is called wiggler. A different arrangement named undulator gives a more narrow photon beam spectrum, and within that group a subset called elliptically polarized undulators allow to adjust the phase characteristics of the output beam. The energy of the photon spectrum is controlled by changing the distance between the permanent dipole magnets and the electron beam while changing the relative position of the magnet arrays modifies the polarity of the photon beam.

All the insertion devices share the same mechanical principle of two girders, an upper one and a lower one, Fig. 1. The distance between these girders defines the gap mentioned above. Besides that EPUs girders are divided in two subgirders that can be shifted longitudinally with respect to each other, Fig. 2. The mechanically simplest insertion devices are the IVUs at Cosaxs, Danmax and Femtomax beamlines which have a single motor that acts on the distance between girders; more complex setups like the Wiggler in Balder beamline have two motors that act on the gap at both girder ends, allowing to increase the gap at both ends independently, giving the option to tilt the girders; one last type of IVUs, present in Biomax and Nanomax beamlines, have four independent motors at each girder end, adding the posibility of introducing a different offset between each girder and the photon beam plane. Ultimately, the EPUs are

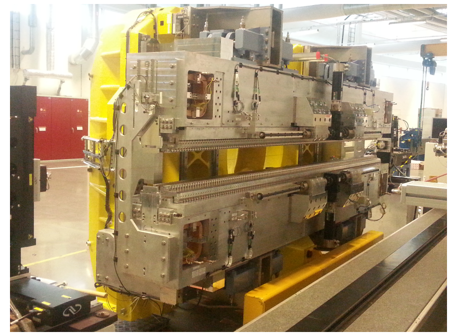


Figure 1: Softimax EPU under test and assembly in MAX IV magnet lab.

Figure 2: Each EPU girder has two subgirders that host the magnet arrays and can move relative to each other and to the subgirders in the other girder. Together they determine the polarization of the light.

equipped with 4 extra motors (8 in total) that allow to move independently their 4 subgirders in the horizontal plane. The girders can be up to 4 m long and weight up to 13.5 tons. The massive structures in Fig. 1 are needed as the maximum attractive force between the top and bottom girder is 46 kN while the maximum repellent force is 30 kN dependent on subgirder phases.

The maximum longitudinal force is 35 kN and the maximum transversal force is 7 kN. As the force between permanent magnets is not linear with distance, closing the gap will cause an ever increasing rate of attractive force as the gap is closed. This causes the entire casted iron structure to bend dependent on gap and phase. From a mechanical point of view, the magnets on the girders can be moved to 0.5 mm far from the beam vacuum chamber, but as this would create too high heat load is not allowed in operation.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA133

STABLE OPERATION OF THE MAX IV LABORATORY SYNCHROTRON FACILITY

P. Sjöblom*, S. Padmanabhan, M. Klingberg, V. Hardion, V. Martos, A. Freitas, A. Milan-Otero, P. Bell, D. Erb, A. Amjad, A. Nardella, H. Petri, J. Rosenqvist, D. Spruce
MAX IV Laboratory, Lund University, Lund, Sweden

Abstract

MAX IV Laboratory, inaugurated in June 2016, has for the last 8 months accepted synchrotron users on three beamlines — NanoMAX, BioMAX and Hippie — while simultaneously pushing towards bringing more beamlines into the commissioning and user phases. As evidence of this, the last call issued addressed 10 beamlines. As of summer 2019, MAX IV has reached a point where 11 beamlines simultaneously have shutters open and are thus receiving light under stable operation. With 16 beamlines funded, the number of beamlines will grow over the coming years. The Controls and IT group has performed numerous beamline system installations such as a sample changer at BioMAX, Dectris detector at Nanomax, and End Station at Hippie. It has additionally developed processes, such as automated IT infrastructure with a view to accepting users. We foresee a focus on end stations and detectors, as well as data storage, data handling and scientific software. As an example, a project entitled "DataStaMP" has been recently funded aiming to increase the data and metadata storage and management system in order to accommodate the ever increasing demand for storage and access.

INTRODUCTION

MAX IV is a fourth generation synchrotron facility with one linear accelerator and two storage rings at 1.5 GeV and 3.0 GeV respectively. The ring lattices are designed as multi bend acromats in order to reach emittances in the ultra-low few hundred pm rad range and hence achieve ultra-high brightness and transverse coherence. Inaugurated in June 2016, the facility has accepted users for three beamlines — Nanomax, BioMAX and Hippie — while design and commissioning for 13 beamlines is ongoing. A milestone was reached in June 2019, when 11 beamlines had their beam shutters open and receiving light simulatneously, Fig. 1.

The facility performance is dependent on the availability and reliability of its accelerators. Without a stable electron beam, the possibility to build state of the art beamlines is very limited. MAX IV is therefore constantly monitoring the availability and reliability of its accelerators, and presents the result to users through a web interface, Fig. 2. For the year to date (mid September, 2019) the up-time and meantime between failures has been 98%, 61 h for the 1.5 GeV ring, 98%, 43 h for 3 GeV ring, and 98%, 42 h for the short pulse facility (located at the end of the linac).

* peter.sjoblom@maxiv.lu.se

At MAX IV, the Controls and IT group has the responsibility to find solutions and coordinate the work involving control systems for the machine and beamlines. The group is divided into 5 collaborating subgroups, whose latest activities are summarised in the subsequent sections. As a group, we value taking part in collaborations with other facilities and using open and shared technology and knowledge.

Figure 1: Screenshot of the MAX IV machine status web interface showing 11 open shutters during top-up mode operations of the 1.5 GeV and 3 GeV rings at 250 mA.

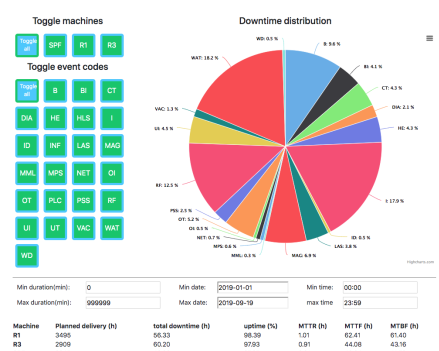


Figure 2: Statistics on the availability and reliability of the accelerators at MAX IV.

INFORMATION MANAGING SYSTEM

The Information Management System (IMS) team has a broad foot-print across MAX IV and strives to find efficient ways to support all the software and tools that falls into the information management and web category. In 2018, a lot of effort was put into making the development and delivery processes more efficient and less error prone with fewer manual intervention. Dockerizing [1] applications and building an automated continuous delivery process were central to this goal. Dockerization in particular has proved

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA134

PyDM - STATUS UPDATE

H. H. Slepicka*, M. Gibbs, SLAC National Accelerator Laboratory, Menlo Park, USA

Abstract

PyDM (**Py**thon **D**isplay **M**anager) is a Python and Qt-based framework for building user interfaces for control systems providing a no-code, drag-and-drop system to make simple screens, as well as a straightforward Python framework to build complex applications. Here we report the state of PyDM as well as the new functionality that has been added in the last year of development, including full support for EPICS PVAccess and other structured data sources, and also the features targeted for release in 2020.

PROJECT STATE

Over the past year PyDM had a couple of feature and bug fix releases which included infrastructure work along with new widgets and enhancements for the user experience while creating screens on the Qt Designer.

INFRASTRUCTURE

PyDM supports Linux, macOS and Windows and Python 2.7, 3.6 and 3.7. In order to ensure that the codebase is healthy and new code added does not break the existing features and platform compatibility, project relies on continuous integration (CI). PyDM initially relied on the cloud-based tools TravisCI (for builds and tests on Linux and macOS), and AppVeyor (for Windows builds and tests). An effort was made during the past year to migrate away from two different systems to the new Azure Pipelines provided by Microsoft.

Azure Pipelines

Azure Pipelines is free for open-source projects and offers 10 parallel builds with support for the three platforms in which PyDM works. This allowed us to reduce the build and test time and also to enhance the operations performed when new code is added to the master branch and new tags are released.

When new code is added to the master branch a new version of the package is uploaded to the "pydm-dev" channel at Anaconda and is immediately available for users for early tests and development. If a new release tag is created, the pipeline creates a new version of the package and uploads it to the "pydm-tag" channel at Anaconda, deploys the documentation to the GitHub Pages website (https://slaclab.github.io/pydm) and also uploads a source package to the Python Package Index (PyPI).

OBTAINING THE PACKAGE

PyDM is available via PyPI and Anaconda channels, as well as source via GitHub.

PyPI

Since PyDM is now available at PyPI, it can be easily added as a dependency to other python packages as well as installed at computers via "pip install pydm".

Anaconda

The PyDM package is available to Anaconda users at the following channels:

– pydm-dev: latest development build;

– pydm-tag: official release versions;

– conda-forge: official release versions;

NEW WIDGET - TEMPLATE REPEATER

The *PyDMTemplateRepeater* widget was added in version 1.7 of PyDM.

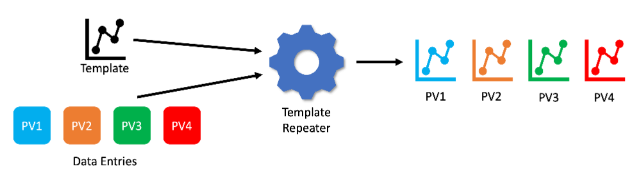


Figure 1: PyDM Template Repeater Widget design concept

The *PyDMTemplateRepeater* lets users specify a display file as a "template", and then specify a JSON file as a "data source" that will fill in multiple instances of that template with values. The widget can lay out the instances vertically, horizontally, or in a 'flow' layout, which is similar to a grid layout that wraps to new rows as the row fills. If used in a python-based display, users can directly supply a list of dictionaries to act as a data source - expand the widget capabilities and allowing it to be hooked up to databases, web APIs, etc. This widget was built to create huge displays with lists of controls for dozens of devices, in a dynamic way, without having to write any code.

LIVE PROTOTYPING WITH QT DESIGNER

In order to provide higher fidelity for users developing displays with the Qt Designer, PyDM now offers an option to connect widgets to live data and render embedded displays while inside Qt Designer. Previously, widgets would only connect to the data plugins when running the screen.

The *PYDM_DESIGNER_ONLINE* environment variable controls this feature and when this variable is defined the live prototyping will be enabled.

It proved to be useful and reduce the amount of work that users have due to resizing and rearrangement of widgets at the screen due to the data being displayed when connected to the data plugin.

* slepicka@slac.stanford.edu

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA135

PyDM - EXTENSION POINTS

H. H. Slepicka*, M. Gibbs, SLAC National Accelerator Laboratory, Menlo Park, USA

Abstract

PyDM (**Py**thon **D**isplay **M**anager) is a Python and Qt-based framework for building user interfaces for control systems providing a no-code, drag-and-drop system to make simple screens, as well as a straightforward Python framework to build complex applications. PyDM developers and users can easily create complex applications using existing Python packages such as NumPy, SciPy, Scikit-learn and others. With high level interfaces for data plugins and external tools, PyDM can be extended with new widgets, integration with facility-specific tools (electronic log books, data logger viewers, et cetera) as well as new data sources (EPICS, Tango, ModBus, Web Services, etc) without the need to recompile or change the PyDM internal source.

PyDM

In 2015, studies were performed to evaluate software to be used as the next-generation display manager[1] at SLAC.

Based on the output of the study and evaluation of potential candidates, it was concluded that a new framework was required to fulfil the demands from scientists, operators and engineers for user interfaces and application development.

PyDM is an open-source Python-based framework for control system graphical user interfaces (GUIs) intended to span the range from simple displays without any dynamic behavior, to complex high level applications, with the same set of widgets.

It provides a system for the drag-and-drop creation of user interfaces using Qt Designer[2] and it also allows for the creation of displays driven by Python code.

Since PyDM is based on Python, it can be leverage the scientific Python ecosystem (see Fig. 1).

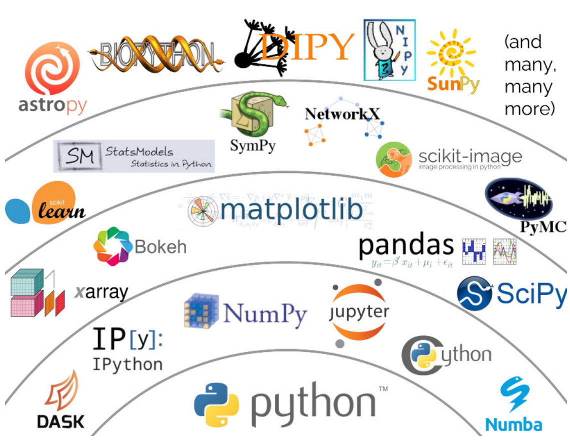


Figure 1: Scientific Python Ecosystem. (Credit: Jake VanderPlas, "The Unexpected Effectiveness of Python in Science", PyCon 2017)

* slepicka@slac.stanford.edu

BEYOND DRAG AND DROP

As many other display managers, PyDM allow users to create synoptic displays via drag-and-drop of widgets at a form in a WYSIWYG (What You See Is What You Get)[3] fashion.

While this is a great solution for synoptic displays in which process variables are presented in a very well defined and static way, it imposes limitations on what can be done when business logic and client-side data processing are desired for more intelligent displays.

To overcome this limitation, PyDM takes advantage of the Qt Framework and also from the Python language to allow users to develop the view using the Qt Designer (Fig.-2) and after that, users can create a Python class (see Listing-1) that allows them to add code to the displays and interface with the widgets created using the Qt Designer through code.

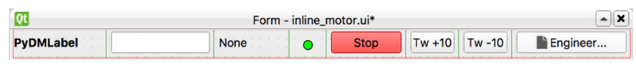


Figure 2: UI developed via drag-and-drop using Qt Designer

```python
from pydm import Display

class MyDisplay(Display):
  def __init__(self, parent=None, args=None,
    macros=None):
    super().__init__(...)
    # Interact here with your ui..
    self.ui.form.setTitle('Hello ICALEPCS')

  def ui_filename(self):
    return 'inline_motor.ui'
```

Listing 1: Example of a Display class

Another possibility is to develop the whole display or application using just Python code, which is again possible due to fact that PyDM is based on Python and Qt. This approach imposes a higher learning curve since users must be familiar with the composition of layouts, the instantiating of widgets as well as the configuration of their properties via Python code.

The PyDM Tutorial[4] covers all three ways of making displays and walks the user through each step with details.

NEW WIDGETS MADE EASY

PyDM widgets are data source agnostic, this means that they have no knowledge of the origin of the data coming to them.

The widgets provided with PyDM rely on seven key pieces of information from the data-sources: connection sta-

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA136

INTEGRATION OF OPTICAL BEAM LOSS MONITOR FOR CLARA

W. Smith†, A. D. Brynes[1], F. Jackson[1], STFC, Sci-Tech Daresbury, Warrington, UK
J. Wolfenden[1], University of Liverpool, Liverpool, UK
[1]also at Cockroft Institute, Warrington, UK

Abstract

The detection of beam loss events in accelerators is an important task for machine and personal protection, and for optimization of beam trajectory. An optical beam loss monitor (oBLM) being developed by the Cockcroft Institute at Daresbury Laboratory required integration with the rest of the controls and timing system of the site's electron accelerator, CLARA (Compact Linear Accelerator for Research and Applications) [1]. This paper presents the design and implementation of an inexpensive solution using a Domino Ring Sampling device from PSI. Signals from the oBLM are acquired and can be processed to resolve beam loss events to a resolution of 0.2m.

INTRODUCTION

Beam losses refer to fractions of the beam which deviate from the nominal beam trajectory and impinge on accelerator components. Beam Loss Monitors (BLMs) are radiation detectors located along an accelerator line in order to observe particle showers. Unlike conventional beam loss monitors, which detect beam loss at discrete locations, oBLMs can detect along the whole length of the machine. This provides much better localisation of beam loss events [2].

An oBLM system consists of a fibre placed along a beamline with photodetectors and readout hardware at the end of the fibre. The operating principle of an oBLM is based on Cherenkov radiation generated as a result of electromagnetic radiation crossing the fibre. This occurs when the particle beam hits any obstacle. The flash of Cherenkov radiation travels down a quartz fibre and is then detected by a silicon photomultiplier. By comparing the arrival time of the response from either end of the fibre it's possible to localise the position of the beam loss relative to the middle of the fibre.

Since the group velocity of the pulse of Cherenkov light in the quartz fibre is approximately 2/3rd the speed of light, the distance of the beam loss event from the middle of the fibre is given by the equation:

$$D = \frac{1}{3}\Delta T c \tag{1}$$

Where ΔT is the difference in the arrival time of the pulse at either end of the fibre (see Fig. 1) [3-4]. The magnitude of the response is proportional to the amount of charge lost.

oBLM Signal

The impulse response from each detector of the oBLM is a damped 20MHz oscillation (see Fig. 2). Multiple beam losses result in a signal that is the sum of impulse responses with different delays and magnitudes. These delays and magnitudes must be determined and processed to gain information about the locations and sizes of beam losses.

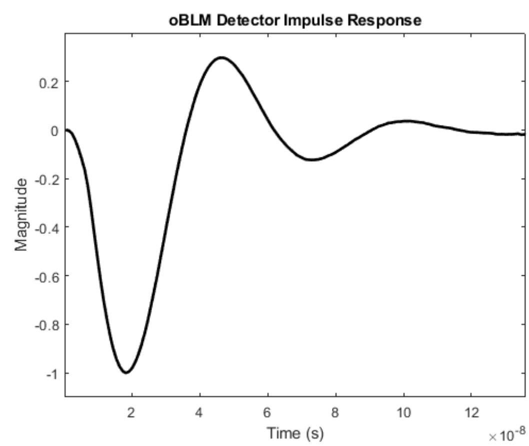


Figure 2: oBLM detector impulse response.

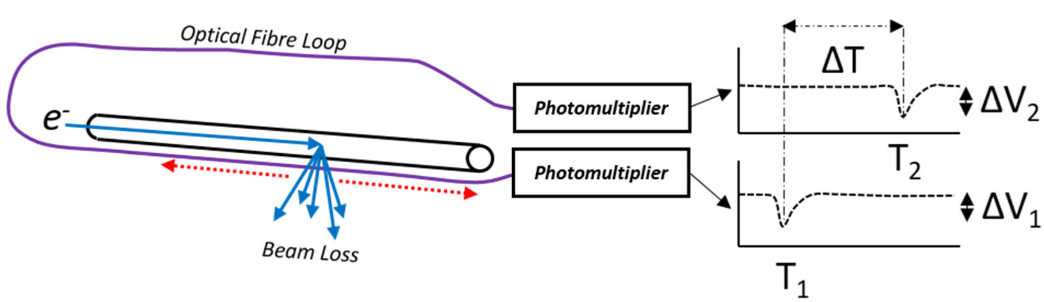


Figure 1: Principle of operation. Cherenkov radiation causes peaks in the voltage from a photo multiplier, the amplitude and delay of which is proportional is to the magnitude and position of the loss respectively.

† william.smith@stfc.ac.uk

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA137

TIMING SYNCHRONIZATION AND CONTROLS INTEGRATION FOR ESS DETECTOR READOUT

W. Smith†, STFC, Sci-Tech Daresbury, Warrington, UK
J. Nilsson, S. Alcock, ESS, Lund, Sweden

Abstract

The European Spallation Source (ESS) is a new facility being built in Lund, Sweden, which when finished will be the world's most powerful neutron source. STFC has an in-kind project with the Detector group at ESS to provide timing and control systems integration for the detector data readout system. This paper describes how time is synchronised and distributed to the readout system from the ESS timing system, and how EPICS is used to implement a controls interface exposing the functionality of detector front ends.

INTRODUCTION

ESS will be the leading facility in Europe in neutron science [1]. Neutrons are produced at the facility through the process of spallation when protons collide with a tungsten target. These neutrons are guided along beamlines to scientific instruments. The detectors that form part of these instruments generate neutron events which are then timestamped.

Data produced by detectors is acquired by detector readout front end nodes. These are connected in a ring or star topology to a detector readout master. The readout master runs firmware on an FPGA that controls the front ends, ingesting the data they produce and sending it to the DMSC (Data Management and Software Centre. The readout master is monitored and controlled by an EPICS IOC (Input Output Controller) [2]. (See Fig. 1)

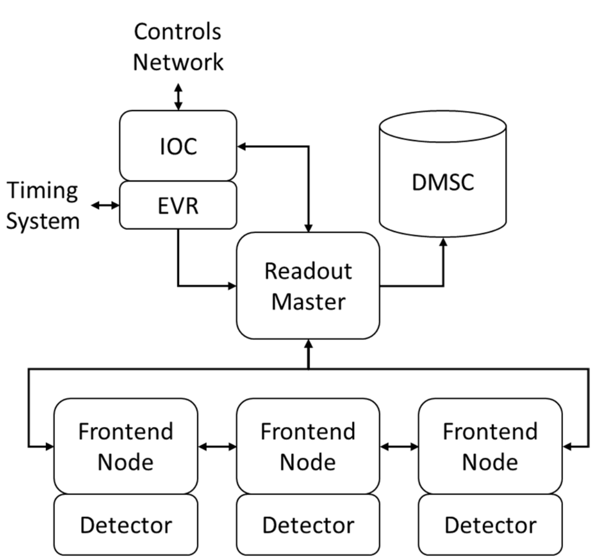


Figure 1: Detector readout system ring topology.

† william.smith@stfc.ac.uk

The readout system does not specify anything about the front ends. Instead the readout master exposes a register space to the control system and forms packets of data to be sent to the DMSC. The EPICS support module defines the communications between the control system and the detector. This definition is used to along with a register map for a specific detector implementation to create an EPICS IOC. Integration with the control system allows parameters on the detectors and the readout master to be monitored and controlled, alarms generated, logged and responded to, and data archived in the EPICS Archiver Appliance [3][4].

Timing System

ESS uses an event timing system provided by Micro Research Finland (MRF) to distribute an 88.0525 MHz clock, triggers, timestamps, and other data about the accelerator [5]. This clock and time must be transferred to the detector readout master and forwarded to the front end. They are then used to timestamp responses from detectors for time of flight measurements and more generally correlating data across many different systems. The readout system aims to provide timestamps with a precision of at least 100ns.

TIMING SYNCHRONISATION

The timing system briefly comprises of event generators (EVM) and receivers (EVR) connected by a fibre network. A GPS receiver provides a 1Hz signal to an EVM which transmits this along with an 88.0525MHz event clock. EVMs transmit event codes and EVRs recover the clock and then decode and respond to event messages. Timestamps are transmitted by sending the current NTP time (Network Timing Protocol) [6] from the EVM to each EVR and then sending the 1Hz signal to validate the time.

Time and a synchronous clock could be provided to the detector readout system by embedding an event receiver in the detector readout master. This would require either using transceivers on the readout master FPGA [7]which would reduce the bandwidth of the readout system, or developing an FMC (FPGA Mezzanine Card) based EVR. It was decided early on in the project not to do this because it required too much effort to develop. Instead the time and clock on a separate PCIe EVR would be synchronised with the time and clock on the readout master.

Due to limitations of the PCIe EVR, the event clock can only be supplied through a prescaler. As such the highest clock that can be transferred is 44.02625MHz. This is sent from one of the outputs of the EVR to the detector readout master. Having provided this clock the synchronisation process involves sending a time in the future and then sending a pulse from another EVR output to validate that time. When the detector readout master receives the time valid

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA138

BEAM GATE CONTROL SYSTEM FOR THE PROTON INJECTOR AND BEAMLINES ON KOMAC*

Y.G. Song†, H.S. Kim, J.H. Kim, H.-J. Kwon
Korea Atomic Energy Research Institute, Gyeongbuk, Korea

Abstract

The Korea Multi-purpose Accelerator Complex (KOMAC) 100 MeV proton linear accelerator operates with the programmable timing system to change real-time timing parameters for low and high-flux proton beam utilization. The main requirements are to synchronize the operation of the facility including accelerator, target, and instruments, to provide the base frequency of up to 60 Hz on a variable beam repetition rate, and to support post-mortem analysis when a beam trip occurs. The MRF event timing system, which consists of one event generator and eleven event receivers, is configured to control the beam gate and beam distribution to transmit the proton beam to the target beam line. The timing events are distributed from the event generator to local event receivers. Local subsystem specific timing can be adjusted locally from beam monitoring and RF modulators. Corresponding to user's demands, beam gate should be automatically controlled, and the beam distribution must be precisely synchronized with the main reference signal for beam gate and accelerator operation. The timing system is configured with sequence logic for beam gate control, and the timing events can trigger the software to perform actions including beam on or off, post-mortem data acquisition, and beam distribution on the beam lines. The results of the timing control system for the beam gate and beam distribution are presented.

INTRODUCTION

The KOMAC linac and multi-beam lines were designed to provide users with a proton beam under various beam operation conditions. Representative specifications of the linac are maximum beam energy of 100 MeV, and peak beam current of 20 mA, and the adjustable repetition rate is up to 120 Hz. The specifications of the KOMAC 100 MeV linac are summarized in Table 1. The KOMAC has four beam extraction points at 20 and 10 MeV for proton beam utilization [1, 2].

The proton accelerators are capable of accelerating the beam under stable operating conditions of various components. The accelerator operating conditions must be kept constant. This can be achieved by changing the beam pulse width and repetition rate for beam service conditions while the accelerator conditioning is stable. To change the beam repetition rate, it is limited to the high frequency repetition rate fixed in the accelerator operating condition. As the number of beam user increases, various beam conditions are required. In order to satisfy this problem, the beam extraction repetition rate can be changed regardless of the high frequency operation repetition rate.

Table 1: The Summarized Specifications of KOMAC 100 MeV Proton Linac

Parameters	20	100
Extraction energy (MeV)	1 ~20	20 ~ 100
Beam current (mA, peak)	20	20
Beam duty (%)	24	8
Beam current (mA, average)	4.8	1.6
Pulse length (μS)	2,000	1,333
Repetition rate (Hz)	120	60
Beam power (kW, average)	96	160

The KOMAC timing system is an MRF event timing system. Event Generator (EVG) is responsible of creating and sending out timing events to an array of Event Receiver (EVR) [3]. Event sequences provide a method of transmitting sequences of events stored in random access memory with defined timing. Using the event sequence, the beam pulse is synchronized with the operation of high frequency and the beam pulse repetition rate can be used without limitation.

KOMAC TIMING SYSTEM

The timing system was fabricated using a versa module eurocard (VME) system composed of an event generator (EVG-230) and event receiver (EVR-230RF) based on EPICS software [4, 5]. The EVG is responsible for generating and sending out event codes over 2-Gbits/s fiber optic links to an array of EVR, which is programmed to decode specific event codes. The EVR generates trigger pulses for the linac components, such as the beam diagnostics, high-power RF system, and high-voltage power supply through the fan-out board. One EVG and one fanout are installed in the control server room, and 12 EVRs are installed in the klystron gallery and beam experiment hall. Figure 1 shows a schematic layout of the timing system with a 300-MHz external reference signal.

For beam extraction, timing was added to the beam injector with a BNC565 delay generator. The delay generator receives a 20 MHz clock and trigger signal from the EVR. The trigger signal determines the repetition rate for the beam service request, and the delay generator operates in burst mode to synchronize the external clock and the trigger and generate the trigger as requested pulses. The beam trigger is limited to a divisor of the rf repetition rate.

* Work supported by KOMAC operation fund of KAERI by MSIT
† ygsong@kaeri.re.kr

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA139

IMPLEMENTATION OF THE PLC BASED MACHINE PROTECTION SYSTEM FOR MAGNETS AT ESS

D. Sánchez-Valdepeñas*, M. Carroll, A. Nordt, M. Zaera-Sanz
European Spallation Source ERIC, Lund, Sweden

Abstract

The European Spallation Source ERIC (ESS) will be the most powerful neutron source in the world, aiming at an average beam power of 5 MW. One of the ESS goals is to deliver neutrons with a 95% overall availability for the 22 research instruments that will be installed at the ESS. To reach this goal, the Protection Systems Group (PSG) develops several interlock systems. One of these systems is the Machine Protection System (MPS) for Magnets (MPSMag), which collects signals from the Local Protection Systems (LPS) of each of the 150 quadrupole magnets distributed along the 600 meters long Linear Accelerator (LINAC). When MPSMag detects a failure of any quadrupole magnet systems, it triggers a beam stop.

The MPSMag design implements requirements from the stakeholders of the LPS for quadrupoles (LPSMag), operations and accelerator teams.

The concept of operation, software design and hardware architecture will be presented in this paper.

INTRODUCTION

Machine Protection supports ESS in reaching its reliability, availability and maintainability goals, in the following way [1, 2]: protect the facility by not allowing proton beam production if a condition could lead to beam induced damage; protect the beam by generating beam interruptions only if strictly necessary; support identifying the initial failure.

Protection functions that span across several systems, are called global protection functions. Protection functions that are part of only one system, are called local protection functions. The time response requirements for executing a protection function, can be very different, ranging from a few microseconds (fast response time) to several hundreds of milliseconds (slow response time). The Beam Interlock System (BIS) implements the logic for global protection functions and consists of several PLC based Machine Protection Systems (MPSs) to implement slow global protection functions and the Fast Beam Interlock System (FBIS) to execute fast global protection functions. The FBIS is the link between the MPSs and the actuator systems and only the FBIS triggers a beam stop. The FBIS has a time response of microseconds. The relation between Local Protection Systems, the Beam Interlock System and the Actuator systems is shown in Figure 1.

The first MPSMag prototype has been implemented using Siemens industrial Programmable Logic Controllers (PLCs), the PROFINET real-time fieldbus communications protocol and Siemens Totally Integrated Automation (TIA) Portal software.

* david.sanchezvaldepenas@esss.se

CONCEPT OF OPERATION

The concept of operation for the MPS [3] is inspired by the ISO 15289 standard. MPSMag has interfaces with different systems to fulfil the protection functions and a description of the signals exchange and interfaces is given in Figure 2.

MPSMag exchanges software signals such as Proton Beam Mode, Proton Beam Destination, configuration and all the Process Values (PV) related to machine protection with Experimental Physics and Industrial Control System (EPICS) through an Input Output Controller (IOC).

The Timing system has a Network Time Protocol (NTP) for MPSMag PLC events time stamping.

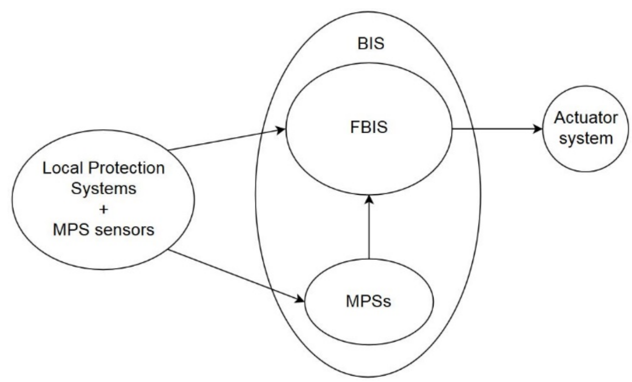


Figure 1: Relation between local Protection Systems, the Beam Interlock System and the actuator systems. The Beam Interlock System consists of PLC based Machine Protection Systems (MPSs) and the Fast Beam Interlock System (FBIS).

The Local Protection Systems for the quadrupole magnets (LPSMag's) protect the magnets from overheating or faults in the powering system. The LPSMag's send Local Beam Permit signals, OK or NOT OK (NOK) to MPSMag. The LPSMag in the Normal Conducting Linac (NC-Linac) is a Programmable Logic Controller (PLC) which sends one Local Beam Permit signal for each quadrupole in the Medium Energy Beam Transfer line (MEBT) to MPSMag. In the Super Conducting Linac (SC-Linac), the LPSMag is implemented on the level of the power converters. Each power converter sends a Local Beam Permit signal for each quadrupole to MPSMag.

In case a Local Beam Permit signal of a quadrupole upstream of the Proton Beam Destination is NOK, the MPSMag initiates a beam stop or prevents beam operation sending a Magnet Beam Permit (MBP) signal NOK to the FBIS. In addition, MPSMag sends the configured Proton Beam Mode and Proton Beam Destination, Keep alive signal, and

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA141

DYNAMIC SYSTEM RELIABILITY MODELLING OF SLAC's RADIATION SAFETY SYSTEMS

F. Tao†, K.W. Belt, SLAC National Accelerator Laboratory, Menlo Park, USA

Abstract

When the LCLS-II project is completed in 2020, there will be three major Department of Energy (DOE) beam programs occupying the same 2-mile long accelerator tunnel, e.g. LCLS, LCLS-II and FACET-II. In addition to the geographical overlap, the number of beam loss monitors of all types has been also significantly expanded to detect power beam loss from all sources. All these factors contribute to highly complex Radiation Safety Systems (RSS) at SLAC. As RSS are subject to rigorous configuration control, and their outputs are permits enabling beam production and transportation, even small faults can cause a long down time. As all beam programs at SLAC have the 95% beam availability target, the complex RSS's contribution to overall beam availability and maintainability is an important subject worth detailed analysis. In this paper, we apply the reliability engineering techniques to analyze the RSS reliability for all three beam programs. Both qualitative and semi-quantitative approaches are used to identify the most critical common causes, the most vulnerable subsystem as well as areas that require future design improvement for better maintainability.

INTRODUCTION

At SLAC National Accelerator Laboratory, there are multiple beam programs taking up part of the same famous two mile long linear accelerator (linac) constructed over 50 years ago. In 2020, when the second generation free electron x-ray laser powered by superconducting electron beam, e.g. LCLS-II, starts operation, there will be three large scientific research user facilities in SLAC, e.g. LCLS, LCLS-II and FACET-II. All those beam programs have their own dedicated beamline components, and share some infrastructure and supporting systems as well.

Being a user facility implies that the SLAC should deliver the beam to the user for their experimental use at a high availability. The availability target for both LCLS and LCLS-II is 95%. Unlike simpler synchrotron radiation facility such as SSRL in SLAC, LCLS and LCLS-II's x-ray laser are driven by an electron accelerator using normal conducting and superconducting RF technologies respectively. Those system are very complex as many subsystem must function in a coordinated manner for the successful system operation.

Among those systems, Radiation Safety Systems (RSS) including Personnel Protection System (PPS) and Beam Containment System (BCS) are carrying out safety critical functions, and play an important role in the system availability. Those reasons include:

- The outputs of RSS are permit signals, which are vital for the overall system operation. Without those permit signals, the beam cannot be generated, accelerated, and delivered to the end users.
- RSS are safety-critical systems with rigorous configuration control, the failed parts have to be isolated and/or repaired to restore the system to normal state. Bypassing faulty parts is generally not allowed, as this will disable the safety function or increase the potential radiation risks, which should be carefully evaluated.
- Unlike other critical systems such as RF, which has some level of redundancy, radiation safety systems are usually configured as one out of two (1oo2) or one out of one with diagnostics (1oo1D), any failure in the single chain will stop the beam operation, make it unavailable for user experiments.
- When RSS are tripped off for some reason, it usually implies there is something wrong, either lower level of control/protection system fails or there is some procedural violation. It generally requires operators to find out the cause of the trip, rather than simply pressing the Reset button to resume the operation. For this reason, the system restoration time is longer.
- With the rigorous configuration control, any invasive diagnostic/repair work requires "Radiation Safety Work Control Form", and need approval from various stakeholders before the work are permitted. This also contributes to the longer restoration time.

For reasons lists above, it is important to evaluate the RSS reliability as early as in the design stage, and constantly re-assess the situation during the entire lifecycle of the system, including operation, maintenance, and proof testing periods. Although the reliability assessment is generally for random failures, continuous assessment and failure analysis can help to identify the systematic failure causes. As a return, it can enhance the overall system safety, which definitely takes priority over the system availability.

Though both PPS and BCS are both safety-critical RSS, two systems are fundamentally different by following criteria:

- System topology: PPS is mostly a de-centralized system, is built upon each zone or region, higher level system will look at each area for decision making; BCS is a centralized system, individual sensor equally contributes to the system action.
- Technology used: PPS is not required to be fast, so they are typical electrical or programmable electrical systems. Electronics are often contained inside commercial-off-the-shelf (COTS) sensor and the signal

† Email address: fengtao@slac.stanford.edu

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA142

FACET-II RADIATION SAFETY SYSTEMS DEVELOPMENT

F. Tao†, B.M. Bennett, N. Lipkowitz, SLAC National Accelerator Laboratory, Menlo Park, USA

Abstract

Facility for Advanced Accelerator Experimental Tests (FACET)-II is an upgrade of the FACET. It uses the middle third of SLAC's 2-mile long linear accelerator to accelerate the electron beam to 10GeV, with positron beam to be added in the Stage 2 of the project. Once the first stage completes in late 2019, it will be operated as a Department of Energy (DOE) user facility for advanced accelerator science studies. In this paper, we will describe the Radiation Safety Systems (RSS) design and implementation for the FACET-II project. RSS including Personnel Protection System (PPS) and Beam Containment System (BCS). Though both systems are safety critical, different technologies are used to implement safety functions. PPS uses Siemens PLC as the backbone for control but legacy CAMAC for data acquisition; while BCS develops customized electronics for faster response to protect safety devices from radiation induced damage, and depends on VME I/O modules for operational data communication.

INTRODUCTION TO FACET-II PROJECT

FACET used first two third of the SLAC's two mile long linear accelerator to generate high density beam of electrons and positrons. The initial motivation for constructing FACET was plasma wakefield acceleration, but the operation results shows that it achieved far more than that.

In 2016, FACET completed its mission to make way for the LCLS-II project, which takes up the first one third of the accelerator originally taken by FACET. The new FACET-II will be a major upgrade over the previous FACET, taking up the middle one third of the accelerator. The beam energy of the FACET-II is 10GeV with the repetition rate up to 30Hz.

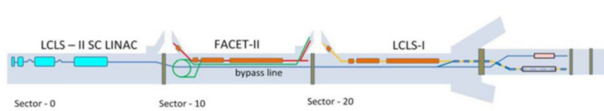


Figure 1: Schematic layout of FACET-II.

This project will be implemented in two stages. In the first stage, a new photoinjector and two bunch compressor in the linac will be restored; and a new positron damping ring with injection and extraction line will be installed in the second stage.

As the radiation hazards assessment completed by radiation protection physicists is based on the stage 1 baseline design, we will limit our discussion in this paper only for this baselined design. Modification of the accelerator and associated experimental area of FACET-II later may require additional modification of the radiation safety systems, and is beyond the scope of this paper.

† Email address: fengtao@slac.stanford.edu

Key parameters of the FACET-II project are listed in Table 1. As this paper is focused on the radiation safety aspects of the project, only those parameters related to radiation safety are listed in the table.

Table 1: Key Parameter of FACET-II [1]

Parameter	Unit	Nominal	Range
Rep Rate	Hz	30	1 - 30
No. of bunches/pulse	-	1	1
Bunch charge	nC	2	0.5 - 5
Final beam energy	GeV	10	4 - 13
Final peak current	kA	72	10 - 130
Final beam power	W	600	2 - 1950

A detailed illustration of the FACET-II beamline layout and the RF accelerating structure is shown in Figure 2:

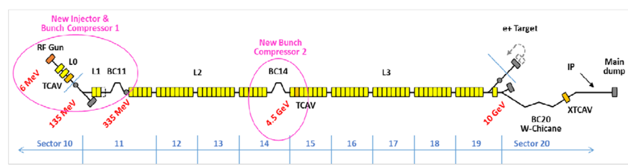


Figure 2: FACET-II accelerator layout.

RADIATION SAFETY SYSTEMS

In SLAC, major radiation hazard is ionization radiation, which comes from electron or positron beam. In addition to the passive shielding, Radiation Safety Systems (RSS) are designed and deployed to further mitigate the risk. RSS include two active control systems built on two distinct principles:

- Personnel Protection System (PPS): keep people away from beam
- Beam Containment System (BCS): keep beam away from people

Although two systems are both safety critical, but they are designed to prevent/mitigate radiation risk for different hazardous scenarios. As a result, the response time required and the technology deployed are significantly different.

The subject of PPS is people, whose movement is slow. Therefore, the typical PPS response time is in terms of seconds. In SLAC's recent engineering practice, a 5 seconds response time has been specified for the worst case scenario, as the real response time of PPS may differ case by case, depending the location of the fault and the signal transmission path to the shutoff devices. Radiation protection physicists are required to use this time constant in their radiation risk assessment. Mechanical engineers are also required to put sufficient material behind the BTM correspondingly in the case of burn-through.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA143

MOTION CONTROL DEVELOPMENT OF THE MATERIAL HANDLING SYSTEM FOR INDUSTRIAL LINAC PROJECT AT SLRI

R. Rujanakraikarn†, P. Koonpong, S. Tesprasitte
Synchrotron Light Research Institute, Nakhon Ratchasima, Thailand

Abstract

The prototype of industrial linac for food irradiation application using x-ray has been under development at Synchrotron Light Research Institute (SLRI). Several subsystems of the machine are carefully designed for proper operation. Material handling system with its motion control and its relationship with a beam scanning system is explained in this paper. Hardware selection and software development together with a networked control system is described. This system is being developed and tested with the object detection system to monitor and control the position and velocity of materials on a conveyor belt.

INTRODUCTION

Synchrotron Light Research Institute (SLRI) has been developing a prototype of linear accelerator for industrial applications. One of the main purposes of this new project is for food irradiation application. Since agricultural products are Thailand's primary economy, this newly proposed project is aimed to increase the availability of the low-cost machines for domestic uses. This accelerator-based system is one of the platforms that can provide good facility for food irradiation. There are three key elements in this accelerator-based system, an accelerator system to deliver the energetic beam, a scanning system to provide uniform beam coverage of the product, and a material handling system that moves the product through the beam in a precisely controlled manner [1].

In the designed prototype there are several main components for each key element. Firstly, the accelerator system consists of an electron linear accelerating structure of the S-band standing wave type, a 3.1 MW magnetron driven by a solid-state modulator, and a hot-cathode electron gun. Secondly, the scanning system comprises a beam scanning magnet and a scanning horn to provide full coverage for irradiation to products. Lastly, the material handling system is composed of conveyor system, motor drive system, and electronic control system. The diagram of this accelerator-based irradiation facility prototype can be shown in Figure 1.

The primary goal of the irradiation facility is to deliver the specified amount of required radiation to the products without unnecessary, wasteful, and excessive dose. Thus, monitoring and control of the process parameters and the information of objects to be scanned are important in order to effectively spread the beam with appropriate field strength. Once the parameters and object information are known, the motion of the conveyor belts in the material handling system must be precisely controlled.

This is to ensure that each product moves through irradiation zone to receive desired dose without slippage or gaps. This can effectively improve the efficiency of the irradiation facility.

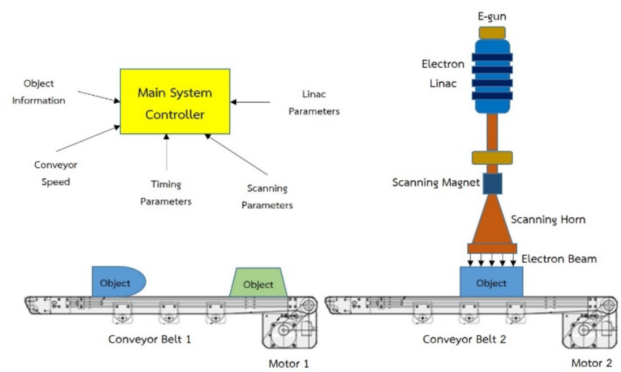


Figure 1: A prototype of accelerator-based irradiation facility.

In this paper, a real-time motion control development of the material handling system together with its relationship to the beam scanning system and object detection system is described. Hardware selection and software development together with a networked control system are explained. Elementary result of the design and tests is briefly explained with some discussion on proper machine operation. Concluding remark is summarized at the end of the paper.

SYSTEM DESIGN

This section explains the brief description of the prototype system design. An overall system diagram showing the beam scanning and the material handling systems is shown in Figure 2. In this diagram, all subsystem controllers are connected in a private network to form a networked control system. These controllers are responsible for motion control of the conveyor belt, camera control for object detection [2], and scanning magnet control for the generation of time-dependent magnetic field deflection of the beam. The main purposes of implementing these subsystem controllers are for process parameter measurement and sensing, and reporting information to and receiving control setpoints from a main system controller which is connected to the same network. The main system controller is designed to implement control algorithms in order to oversee the operation and stability of the whole system. The personal computer in the diagram is connected to the network to receive and display all process parameters and send settings and responses to the main system controller's various requests.

† roengrut@slri.or.th

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA145

EVOLUTION OF THE CERN LINAC 4 INTENSITY INTERLOCK SYSTEM USING A GENERIC, REAL-TIME COMPARATOR IN C++

A. Topaloudis*, J. C. Allica Santamaria, CERN, Geneva, Switzerland

Abstract

During the commissioning phase of Linac 4, three watchdog interlock systems were used to protect the accelerator and its equipment. These systems cut the beam if losses, calculated by combining the intensity measurements at various locations, exceed some predefined thresholds. While the existing systems were designed to be simple and robust to ensure safety, the future connection of the linac to the Proton Synchrotron Booster (PSB) requires new instances of these systems with additional requirements. Such requirements include the remote communication of the watchdogs with the intensity measurement systems to decouple any physical dependency between the two systems, and the arithmetical / logical combination of the measured data based on the watchdog location. As the Controls Interlocks Beam User (CIBU) hardware interface to the Beam Interlock Controller (BIC) is simple, the software part of the system can be redesigned to be generic and application agnostic giving a single decision (beam permit in this case) after performing a configurable set of comparisons. This paper describes the transformation of the existing watchdog interlock system into a generic comparator by its software upgrade, enabling its usage for other applications.

INTRODUCTION

While the accelerators of the proton injector chain at CERN deliver beams to the Large Hadron Collider (LHC) within specification, the requirements for the upgraded High-Luminosity LHC (HL-LHC) exceed their capabilities. The LHC Injectors Upgrade (LIU) project aims to address this by upgrading the proton injectors to deliver the high brightness beams needed by the HL-LHC [1].

In the framework of the LIU project the PSB is undergoing a profound upgrade including the connection to the new Linac 4 injector. Linac 4, a normal conducting H^- ion accelerator, started being constructed in October 2008 and is foreseen to be connected to the PSB in 2020 [2]. The linac is foreseen to produce many different beam types which will be sent to various different locations throughout the subsequent injector complex, such as beams to the LHC, the CERN North and East experimental areas, the CERN antimatter facility and the ISOLDE facility. During its construction period, Linac 4 was commissioned with beam in several stages including a Half-Sector Test (HST) in 2016/17 where half of the PSB injection chicane was temporarily installed and tested [3].

In order to provide safe and successful commissioning and operation, numerous beam diagnostic devices that measure the various parameters of the beam have been developed [4]. Amongst others, several Beam Current Transformers (BCT) have been installed along the linac to provide continuous beam intensity information via a dedicated digital acquisition module - the Transformer Integrator Card (TRIC) [5]. By combining the intensity information in two locations of the accelerator, the beam transmission between those locations can be derived. Such information is essential not only for ensuring the quality of the delivered beam, but also the safety of the accelerator and its equipment as low transmission indicates particle losses.

Linac 4 is protected by a Beam Interlock System (BIS) that comprises a mixture of various hardware, software and external conditions in order to cut the beam and prevent any damage in case of a failure [6]. The BIS is based on the evaluation of several interlock signals from User Systems that are collected via the dedicated user interface to the BIC, the CIBU [7]. To enable the proper operation of the new linac and maximise the beam availability, the user systems should take the future destination of the linac beam pulse into account while providing their interlock signals.

One such user system is the Beam Current Transformer Watchdog that monitors the beam transmission in key locations of the accelerator and provides its interlock signal to the CIBU via a custom VME module. Such a signal is derived from a combination of high loss detection for a particular destination and a bad pulse counter on a beam type basis. To fulfil the requirements and provide the needed flexibility to the operators a hybrid design was chosen by complementing the hardware based watchdog with real-time C++ software responsible for implementing the business logic of the system.

MOTIVATION

During the commissioning phase of the Linac 4, three watchdogs were installed to protect its source, its dump and the temporary HST. Inter-Process Communication (IPC) structures were used to transmit the intensity and time stamp information from the BCT acquisition software to the watchdog making the sharing of the hosting CPU by the two systems a prerequisite [8].

As Linac 4 gets its final form with all the lines being constructed and in view of its future connection to the PSB, new instances of the Watchdog system with additional requirements are needed to protect the full injection chain [9]. As it can be seen in Fig. 1, the temporary watchdog that protected the HST is no longer needed and will be removed, whereas the ones installed at the source and the dump will remain. Two additional watchdogs will be installed to protect the full accelerator line and the beam transverse emittance measurement line (LBE).

* athanasios.topaloudis@cern.ch

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA147

INTEGRATING THE FIRST SKA MPI DISH INTO THE MeerKAT ARRAY

S. N. Twum*, A. F. Joubert, K. Madisa, SARAO†, Cape Town, South Africa

Abstract

The 64-antenna MeerKAT interferometric radio telescope is a precursor to the SKA which will host hundreds of receptor dishes with a collecting area of 1 sq km. During the pre-construction phase of the SKA1-MID, the SKA DSH Consortium plans to build, integrate and qualify an SKA1-MID DSH Qualification Model (SDQM) against MeerKAT. Before the system-level qualification testing can start on the SDQM, the qualified Dish sub-elements have to be integrated into the SDQM and set to work. The SKA-MPI DISH, a prototype SKA dish funded by the Max Planck Institute, will be used for early verification of the hardware and the control system. This prototype dish uses the TANGO framework for monitoring and control while MeerKAT uses the Karoo Array Telescope Control Protocol (KATCP). To aid the integration of the SKA-MPI DSH, the MeerKAT Control and Monitoring (CAM) subsystem has been upgraded by incorporating a translation layer and a specialised SKA antenna proxy that will enable CAM to monitor and command the SKA dish as if it were a MeerKAT antenna.

INTRODUCTION

MeerKAT Overview

The MeerKAT telescope is a 64 dish interferometric[1] array which is located in the northern cape of South Africa. Lessons learned from the KAT-7 telescope, a seven dish interferometric array engineering prototype built prior to MeerKAT construction, were used to build a better and much more sensitive MeerKAT telescope. First light was received from the DEEP2 [1] commissioning field with 4 dishes in May, 2016. DEEP2 was then observed with a progression of 16, 32 and finally 64 dishes. The resolution achieved for these DEEP2 observations were distinctively sharp, exposing finer details for each array progression.

MeerKAT was launched in July 2018 revealing a very detailed image of our galactic centre and has been fully operational doing different science studies with a variety of subarray configurations. The receptors, encompassing the antenna structure with the main reflector, sub-reflector and all receivers, digitisers and other electronics installed, are controlled and monitored using the KATCP - a communications protocol built on the TCP/IP layer (see Fig. 2). The CAM subsystem interfaces with all these subsystems to provide health, state and alarm information, and to execute overall control.

* stwum@ska.ac.za

† South African Radio Astronomy Observatory

[1] Radio interferometry is a technique which simulates telescopes with diameters equal to the longest baseline

SKA Dish Prototypes

MeerKAT is a pathfinder telescope and will become part of SKA Phase 1. Phase 1 marks the commencement of the construction of SKA1-MID Dish array. This array will constitute 133 15-m diameter dishes and 64 13.5-m MeerKAT dishes spread over approximately 150 km in the Karoo region.

The DSH consortium has delivered two prototype dishes for the SKA1-MID array. The first SKA dish prototype (SKA-P) was built and installed in China. It was put under a series of structural tests to identify discrepancies with the designs. The SKA-MPI dish, the second of the two prototype dishes, is funded by the Max-Planck Institute and is currently being assembled on the SKA site in South Africa.

The SKA DSH Consortium will integrate and qualify the SDQM on the Karoo site as part of the System Critical Design Review (CDR), [2]. Preceding this will be the verification of the dish sub elements using the prototype dish. Part of the SKA DSH consortium's effort in qualifying the SQDM is to make optimal use of the existing MeerKAT Receptor Test System (MKAT-RTS) [3] by leveraging the qualified SKA1-MID and MeerKAT hardware:

- SKA1-MID Dish Structure (DS)
- SKA1-MID Local Monitoring and Control (LMC)
- SKA1-MID Single Pixel Feeds (SPF) - Bands 1 and 2, SPF Controller and Services
- SKA1-MID Dish Fiber Network
- RTS Ku-Band Receiver
- MeerKAT L-band and UHF Digitisers
- MeerKAT CAM and Data Switches

In the following sections we explain the activities executed to deliver the MeerKAT CAM DSH proxy software which takes advantage of a translator to connect to the SDQM LMC whilst providing a common interface to the rest of CAM.

INTEGRATING THE DSH LMC INTO THE MEERKAT RECEPTOR TEST SYSTEM

The SARAO CAM team has been tasked, by the SARAO Software Engineering (SE) team, to support the DSH consortium by developing a KATCP/TANGO translator that will be used between the MeerKAT Receptor Test System (MKAT-RTS) and the SDQM LMC in qualifying the prototype dish. This is one of the two translators that was first developed as part of a proof of concept whilst experimenting with the TANGO framework in the context of a real telescope system [4], MeerKAT.

The SKA-1 MID Architecture

SKA is made up of multiple components, named elements. One of those elements, the Telescope Manager (TM), acts

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA149

ACCELERATOR SCHEDULE MANAGEMENT AT CERN

B. Urbaniec, C. Roderick, CERN, Geneva, Switzerland

Abstract

Maximizing the efficiency of operating CERN's accelerator complex requires careful forward planning, and synchronized scheduling of cross-accelerator events. These schedules are of interest to many people helping them to plan and organize their work. Therefore, this data should be easily accessible, both interactively and programmatically. Development of the Accelerator Schedule Management (ASM) system started in 2017 to address such topics and enable definition, management and publication of schedule data in generic way. The ASM system currently includes three core modules to manage: Yearly accelerator schedules for the CERN Injector complex and LHC; Submission and scheduling of Machine Development (MD) requests with supporting statistics; Submission, approval, scheduling and follow-up of control system changes and their impact. This paper describes the ASM Web application (built with Angular, TypeScript and Java) in terms of: Core scheduling functionality; Integration of external data sources; Provision of programmatic access to schedule data via a language agnostic REST API (allowing other systems to leverage schedule data).

INTRODUCTION

In almost any field, effective schedule management is key when it comes to being efficient and maximizing the use of the time available. This also includes the organization of the activities related to the operation of CERN's particle accelerator complex. A lot of effort has always been put into the CERN accelerator schedules, however the schedule data has never been available in a programmatic manner.

Launched in February 2017, the Accelerator Schedule Management project resulted in a new web application that is comprised of three modules to manage:

- The official accelerator schedules.
- Control system changes.
- Detailed planning for Machine Development periods.

Each of the modules is described in more details in the following sections.

SCHEDULE MANAGEMENT

As the name suggests, the Schedule Management (SM) module centralizes the creation, edition and publication of the schedules. In ASM, everything is based on access roles and is fully data driven, therefore, to satisfy the CERN requirement for separate schedules for the LHC and Injector Complex, the person with the role of Schedule Manager has simply configured new schedules accordingly using the forms in the application. Recognizing the utility and ease of use, other types of schedules haven been subsequently added such as for different types of commissioning and facility schedules as shown in Fig. 1.

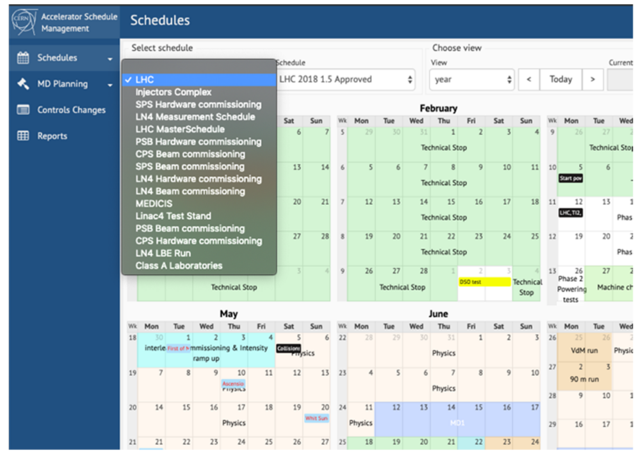


Figure 1: ASM schedules configured for a variety of cases.

The SM module facilities creation and edition of schedules with its point and click user interface allowing to easily add different types of events such as:

- *Background events* – long running periods such as Technical Stops, Physics runs, Machine Development blocks etc.
- *Foreground events* – specific types of events that last for a relatively short time with respect to their corresponding background events e.g. Controls Maintenance Days within an extended Technical Stop.
- *Punctual events* – events that are foreseen to occur at a specific moment in time without a specific ongoing duration e.g. Start of Physics.
- *Floating events* – events for which the exact time and duration are not known in advance, which are therefore scheduled within an approximate time window.

The ASM application validates events as they are scheduled, for example, ensuring they do not overlap. Once a schedule is considered final, the person with the schedule manager role can click to publish the schedule as a new major or minor release. This will automatically update the version number of the schedule and make it available in the overall list of schedules.

Any authenticated user can go to the ASM web application and consult schedules in either yearly (Fig. 2), monthly (Fig. 3) or weekly views (Fig. 4).

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA151

FEASIBILITY OF HARDWARE ACCELERATION IN THE LHC ORBIT FEEDBACK CONTROLLER

L. Grech*, G. Valentino†, University of Malta, Msida, Malta
D. Alves, S. Jackson, J. Wenninger, CERN, Geneva, Switzerland

Abstract

Orbit correction in accelerators typically make use of a linear model of the machine, called the Response Matrix (RM), that relates local beam deflections to position changes. The RM is used to obtain a Pseudo-Inverse (PI), which is used in a feedback configuration, where positional errors from the reference orbit as measured by Beam Position Monitors (BPMs) are used to calculate the required change in the current flowing through the Closed Orbit Dipoles (CODs). The calculation of the PIs from the RMs is a crucial part in the LHC's Orbit Feedback Controller (OFC), however in the present implementation of the OFC this calculation is omitted as it takes too much time to calculate and thus is unsuitable in a real-time system. As a temporary solution the LHC operators pre-calculate the new PIs outside the OFC, and then manually upload them to the OFC in advance. In this paper we aim to find a solution to this computational bottleneck through hardware acceleration in order to act automatically and as quickly as possible to COD and/or BPM failures by re-calculating the PIs within the OFC. These results will eventually be used in the renovation of the OFC for the LHC's Run 3.

INTRODUCTION

Figure 1 illustrates the schematic of the Orbit Feedback Controller (OFC) as it is implemented today in the LHC. The red and blue arrows in Figure 1 show the data paths of the Beam Position Monitors (BPMs) and the Closed Orbit Dipoles' (CODs) signals respectively. The OFC uses BPM measurements throughout the machine in order to automatically adjust the average beam position by performing adequate changes to the COD currents [1].

The OFC was designed in C++ using ROOT libraries prior to the LHC start-up in 2008. ROOT is a scientific software framework originating from CERN with the purpose of being used for the analysis of experimental data related to high energy and nuclear physics [2]. The Service Unit (OFSU) was designed using CERN's Front-End Software Architecture (FESA) and serves as an interface to the OFC from which the operators can change certain parameters as well as control the operation of the OFC itself. The OFSU also serves as a proxy of the incoming measurements collected and calculations done by the OFC [3, 4].

The currents flowing in the CODs are related to the beam position as measured by the BPMs by a Response Matrix (RM), which essentially describes changes in positions as a function of COD deflections. The main principle behind the orbit correction is to (a) measure the beam position using the BPMs, (b) calculate the error with respect to the reference orbit and then (c) calculate the change in current needed in each COD to correct the beam position.

* leander.grech.14@um.edu.mt
† gianluca.valentino@um.edu.mt

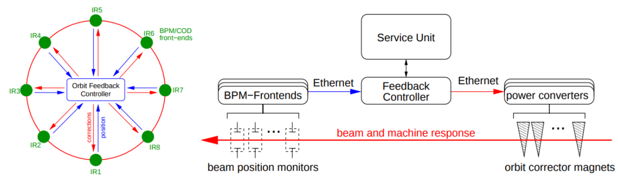


Figure 1: Schematic of orbit feedback controller [1].

To calculate the required change in current in each COD the OFC uses Singular Value Decomposition (SVD) algorithm, where the RM is decomposed into three more manageable matrices, from which the so called Pseudo-Inverse (PI) is found. This PI then directly relates the required COD deflections as a function of the measured orbit error [1].

The following is a basic description of the PI and the SVD algorithm. The PI of a non-square matrix A, is a matrix which when multiplied to A produces an identity matrix. The PI is calculated after computing the SVD of A, which evaluates A as a multiplication of three sub-matrices with special properties, making it trivial to find the PI. Below is the result after SVD:

$$A = U\Sigma V^*$$

where U and V are unitary matrices and Σ is a rectangular diagonal matrix, with non-negative real numbers on its diagonal. The PI, A^{-1}, is calculated as follows:

$$A^{-1} = V\Sigma^+ U^*$$

where Σ^+ is Σ with the diagonals inverted, i.e. $\Sigma^+_{ii} = \frac{1}{\Sigma_{ii}}$

In the OFC, the SVD is used to compute the PIs of the horizontal and vertical RMs, which have around 1150 rows (number of BPMs) and 550 columns (number of CODs in one plane). Apart from this there might be different configurations of the LHC (optics) in which the magnets are used, for which RMs and accompanying PIs have to be computed as well.

PROBLEM DEFINITION

In the current implementation of the OFC, multiple response matrices (e.g. during the RAMP, 13-14 RMs are used) for different machine configurations (optics) are used. It might be the case that there is a change in the LHC operation due to a malfunction in the BPMs or the CODs and

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA152

USE OF MULTI-NETWORK FIELDBUS FOR INTEGRATION OF LOW-LEVEL INTELLIGENT CONTROLLER WITHIN CONTROL ARCHITECTURE OF FAST PULSED SYSTEM AT CERN

N. Voumard*, C. Boucly, M.P. Pimentel, L. Strobino, P. Van Trappen, CERN, Geneva, Switzerland

Abstract

Fieldbuses and Industrial Ethernet networks are extensively used for the control of fast-pulsed magnets at CERN. With the ongoing trend to develop increasingly more complex low-level intelligent controllers near to the actuators and sensors, the flexibility to integrate these within different control architectures grows in importance. In order to reduce development efforts and keep the fieldbus choice open, a multi-network fieldbus technology has been selected for the network-interfacing part of the controllers. Such an approach has been successfully implemented for several projects including the development of high voltage capacitor chargers/dischargers, the surveillance of floating solid-state switches and the monitoring of a power triggering system that, today, are interfaced either to Profibus-DP or Profinet networks. The integration of various fieldbus interfaces within the controller and the required embedded software/gateware to manage the network communication are presented. The gain in flexibility, modularity and openness obtained through this approach is also reviewed.

INTRODUCTION

In the late 1980s, the PLC-based automation systems became increasingly popular in the manufacturing industries aiming to improve their productivity. The need to reduce costs and downtime while ensuring a high level of flexibility has led to the search for innovative decentralized solutions. Some automation manufacturers decided to join their efforts on a fieldbus project to find out a homogeneous common solution. The Profibus standard was born. Today, the PI Organisation (Profibus and Profinet International) counts over 1400 member companies worldwide [1]. Subsequently, a protocol based on Ethernet, Profinet, was designed with a first version available from 2001 onward [2]. Meanwhile, the ongoing trend to develop increasingly more complex low-level intelligent controllers near to the actuators and sensors and the flexibility to integrate these within different control architectures has grown in importance. The integration of fieldbuses into the actuators and sensors themselves proves this, reducing development efforts, as well as cabling costs.

CONTEXT

The CERN Accelerator Beam Transfer group (TE-ABT) is responsible for beam injection, extraction and dump systems for the whole CERN accelerator complex. TE-ABT has based its slow-control architecture on fieldbuses during the last two decades using off-the-shelf PLC components from Siemens. As a choice of fieldbuses, TE-ABT has focused its developments on Profibus-DP and Profinet networks. However, other fieldbuses may be used in the future and a multi-network fieldbus technology is currently envisaged as the best solution. A few commercial companies offers such multi-network device modules to integrate within custom hardware: Hilscher, HMS Industrial Networks, Kunbus are examples. TE-ABT has decided to integrate Anybus products from HMS Industrial Networks in its hardware.

MOTIVATION

As a need to decrease costs in cabling and get rid of PLC-based deported I/O controls which are greedy in terms of space, complex and challenging to modify or add functionalities, TE-ABT has embedded Profibus-DP interface modules in its hardware. One major concern about using commercial off-the-shelve fieldbus modules is their lifetime. As an example, the old Power Trigger Controller card (described below) that has been developed fifteen years ago was integrating an earlier Profibus-DP module from HMS Industrial Networks. This module is now obsolete and it has been necessary to replace it by the CompactCom M40 module in the new version of the card. Thus, the old Profibus-DP modules will be thrown away because no compatible Profinet modules can be ordered in such quantities (only spares in low quantities can still be ordered).

The new modules offer more flexibility than HMS's previous version (no need to redesign the front panel and replacing the module is easier). However, their expected lifetime won't be more than twenty years. The same problem of obsolescence is found with FPGA devices or other sensible components anyway, forcing designers to redevelop a complete PCB.

As Profibus in existing TE-ABT slow control architectures is being replaced by Ethernet-based fieldbuses during the next few years, HMS CompactCom M40 have been selected to move from one to the other without having to redesign and re-validate the hardware. For new projects, only Ethernet-based fieldbuses are being used.

INTEGRATION EXAMPLES

Capacitor Charger/Discharger HV Power Supply

As a first example, a Profinet module has been integrated into in-house development of a capacitor charger/discharger high voltage power supply (CCD HVPS, see Fig. 2) for the renovation of the Proton Synchrotron Booster (PSB) injection transverse painting bumpers (KSW) [3, 4]. To power the four magnets of each of the four rings of the PSB

* nicolas.voumard@cern.ch

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA153

SoC TECHNOLOGY FOR EMBEDDED CONTROL AND INTERLOCKING WITHIN FAST PULSED SYSTEMS AT CERN

P. Van Trappen*, E. Carlier, M. Gauthier, N. Magnin, E. J. Oltedal, J. Schipper
CERN, Geneva, Switzerland

Abstract

The control of pulsed systems at CERN requires often the use of fast digital electronics to perform tight timing control and fast protection of high-voltage (HV) pulsed generators. For the implementation of such functionalities, a field-programmable gate array (FPGA) is the perfect candidate for the digital logic, however with limited integration potential within the control system.

The market push for integrated devices, so called System on a Chip (SoC), i.e. in our case a tightly coupled ARM processing system with specific programmable logic in a single device, has allowed a better integration of the various components required for the control of pulsed systems. This technology is used for the implementation of fast switch interlocking logic, integrated within the CERN control framework by using embedded Linux running a Snap7 server. It is also used for the implementation of a lower-tier communication bridge between a front-end computer (FEC) and a high fan-out multiplexing programmable logic for timing and analogue low-level control.

This paper presents these projects where SoC technology has been used, proposes system deployment & booting solutions and discusses possible further applications within distributed real-time control architecture for distributed pulsed systems.

SYSTEM ON A CHIP TECHNOLOGY

A SoC is an integrated circuit that tightly integrates several components and peripherals of an electronic system, which were once seen as separate chips on a printed circuit board (PCB). The focus of this paper is on SoCs that integrate programmable logic (PL) with one or more computing processors, from now on referenced as the processing system (PS). Most commonly used are ARM® architecture processors which are well known for their use in mobile products because of low cost and efficient power consumption. The PS uses processors that are supported by the free and open-source software (FOSS) Linux mainline kernel which eases the development of software applications.

The programmable logic is composed of the typical FPGA cells and group functionality together in so-called Intellectual Property (IP) cores. These interconnect easily to the PS using various types of the AMBA® ARM® AXI4 protocol buses, enabling e.g. high-performance memory-mapped or streaming data transfer. Custom logic that is implemented in these cells can be clocked at frequencies higher than 100MHz for fast and deterministic behavior. This is a strong advantage but FPGAs often suffer from integration problems, i.e. mainly communication-wise, into existing control systems and middle-ware. Now having a closely coupled PS to configure and treat the PL system data makes that integration almost straight-forward. In addition these SoCs embed various additional peripheral cores (e.g. Ethernet, I2C) with mainlined drivers which significantly eases the PCB design while reducing complexity during gateware and software development.

SWITCH FAST INTERLOCK DETECTION SYSTEM PROJECT

Project Description

Pulsed kicker magnet systems are powered by high-voltage and high-current pulse generators with adjustable pulse length and amplitude. To deliver this power, fast high-voltage switches such as thyratrons or GTO stacks are used to control the fast discharge of pre-stored energy. To protect the machine and the generator itself against internal failures of these switches, a modular digital Fast Interlock Detection System (FIDS) has been developed for the consolidation of existing systems. A Xilinx Zynq®-7000 SoC has been selected for implementation of the required functionalities [1].

HV switch malfunctioning is referred to as *fast interlocks*, which need to be detectable over the system's full dynamic range. In addition to interlocking, they have often a corrective action such as pulsing other magnets in order to fully deflect the beam and to avoid beam losses, thus to protect the machine.

Before the 2020 restart of CERN's accelerator complex, the FIDS will be installed at four places: at the PS Booster distributor, PS injection, SPS dump extraction and LHC injection installations.

Implementation

High bandwidth pick-up signals (>100MHz) are sampled from a HV pulse generator, for which full-scale digitizing would not be cost-effective and also not required for the FIDS functionality. Instead fast discrete comparators are used to produce a 1-bit signal that indicate the presence of a current or voltage. This simplification will not limit the FIDS' performance because the signal amplitude itself is generally of no importance. It is important however that the signal-to-noise ratio (SNR) is high for the full dynamic range of the input signals. This was implemented by having the comparator reference follow the generator's pulse forming network (PFN) charge in order to fix the threshold level to e.g. 50% of its flattop. Certain generators operate with PFN charges between 10kV and 80kV and the pick-up signals

* CERN TE-ABT-EC, pieter.van.trappen@cern.ch

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA154

DATA ACQUISITION SYSTEM DEPLOYMENT USING DOCKER CONTAINERS FOR THE SMuRF PROJECT

J. Vasquez, SLAC National Accelerator Laboratory, Menlo Park, USA

Abstract

The SLAC Microresonator Radio Frequency (SMuRF) system is being developed as a readout system for next generation Cosmic Microwave Background (CMB) cameras. It is based on a FPGA board where the real-time digital processing algorithms are implemented, and high-level applications running in an industrial PC. The software for this project is based on C++ and Python and it is in active development. The software follows the client-server model where the server implements the low-level communication with the FGPA while high-level applications and data processing algorithms run on the client. SMuRF systems are being deployed in several institutions and in order to facilitate the management of the software application releases, dockers containers are being used. Docker images, for both servers and clients, contain all the software packages and configurations needed for their use. The images are tested, tagged, and published in one place. They can then be deployed in all other institutions in minutes with no extra dependencies. This paper describes how the docker images are designed and build, and how continuous integration tools are used in their release cycle for this project.

THE SMURF PROJECT

The next generation of cryogenic CMB (Cosmic Micro-wave Background) cameras [1] require densely instru-mented sensor arrays. These arrays have large number of sensors, in the order of 10,000 to 100,000 per camera. The readout of this large number of sensors is a big challenge that requires substantial improvements in highly-multi-plexed readout techniques.

The SMuRF system is being developed as a readout sys-tem for this next generation CMB cameras. It aims to read 4000 microwave channels between 4 and 8 GHz, in a com-pact form factor. The system reads out changes in flux in resonators by monitoring the change in transmitted ampli-tude and frequency of RF tones produced at each resona-tor's fundamental frequency.

The SMuRF system is unique in its ability to track each tone while minimizing the total RF power required to read out each resonator, thereby significantly reducing the line-arity requirements of the system.

SLAC COMMON PLATFORM

The SMuRF system is based on the SLAC Common Platform Hardware, Firmware, and Software.

SLAC Common Platform Hardware

The SLAC Common Platform hardware is based on the ATCA (Advanced Telecommunication Computing Archi-tecture) standard.

The carrier card contains the FPGA (Xilinx KU15P Ultrascale+) as well as all the digital, management, and power distribution devices. The analog RF devices are located on two double-wide AMC (Advanced Mezzanine Cards) daughter cards. An RTM (Read Transition Module) card contains slow speed analog devices.

The FPGA has 8x 12.5Gbps uplink and downlink JESD204b interfaces to each AMC card, SPI buses to the RTM, as well as a 10Gbps Ethernet link to the ATCA crate's backplane.

All these three boards (carrier, AMCs, and RTM) are installed in one slot of an ATCA crate. The crate provides a dual-star Ethernet backplane, cooling, power distribution, and a management network based on IPMI (Intelligent Platform Management Interface).

Figure 1 shows all the component of the SLAC common platform hardware.

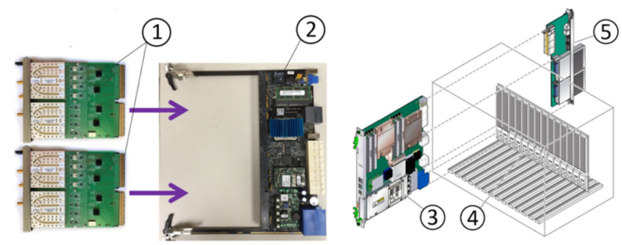


Figure 1: SLAC common platform hardware. 1) AMC daughter cards, 2) carrier card, 3) assembly of a carrier card with AMC daughter cards, 4) ATCA crate, 5) RTM card.

SLAC Common Platform Firmware

The SLAC Common Platform Firmware is a set of VHDL libraries which contain protocols, device access and commonly used modules for all applications that use the SLAC Common Platform hardware.

The SMuRF firmware application uses these set of libraries, as well as an application specific digital signal processing module. The final application digitizes and processes up to 400 channels in a 4 GHz bandwidth.

SLAC Common Platform Software

Rogue [2], a C++ library with Python bindings, is used as a framework to write the low-level software application that communicates directly with the FPGA.

SMURF SYSTEM ARCHITECTURE

A SMuRF system is formed by an ATCA crate and an external industrial PC. The ATCA crate has one or more carrier cards in it. Each carrier card has his own FPGA, which run the SMuRF firmware application, a set of AMC daughter cards and an RTM card. The slot number 1 of the ATCA crate is reserved for an Ethernet switch card.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA156

THE LINUX DEVICE DRIVER FRAMEWORK FOR HIGH-THROUGHPUT LOSSLESS DATA STREAMING APPLICATIONS*

K. Vodopivec†, E. Breeding, J. Sinclair, Oak Ridge National Laboratory, Oak Ridge, USA

Abstract

Many applications in experimental physics facilities require custom hardware solutions to control process parameters or to acquire data at high rates with high integrity. These hardware solutions typically require custom software implementations. The neutron scattering detectors at the Spallation Neutron Source at Oak Ridge National Laboratory transfer custom protocols over optical fiber connected to a PCI Express (PCIe) read-out board. A dedicated kernel-mode device driver interfaces the PCIe read-out board to the software application. The device driver must be able to sustain data bursts from a pulsed source while acquiring data for long periods of time. The same optical channel is also used as bi-directional low-latency communication link to detector electronics for configuration, real time health monitoring and fault detection. This article presents a Linux device driver design, implementation challenges in a low-latency high-throughput setup, kernel driver optimization techniques, real use case benchmarks and discusses the importance of clean application programming interface for seamless integration in control systems. This generic framework has been extended beyond neutron data acquisition, thus, making it suitable for new and diverse applications as well as rapid development of field programmable gate array (FPGA) firmware.

INTRODUCTION

In this article we present a low-level software framework that facilitates rapid PCIe-based FPGA firmware development and fast integration into control systems such as Experimental Physics and Industrial Control System (EPICS). At the Spallation Neutron Source (SNS), custom hardware and electronics are commonly developed to implement functionality that is not commercially available. Throughout the design and implementation, FPGA firmware interfaces and functionality are subject to change, and thus, software is needed to drive test cases and verify functionality. When firmware development is completed, new equipment needs to be integrated into the control system for long term operations. Software must provide a means to control firmware parameters, monitor functionality and read out data at very high sustained data rates. Our software framework, as shown in Figure 1, consists of a Linux kernel device driver for tight integration with PCIe devices, the user space library providing powerful but elegant application programming interfaces and several generic tools for swift prototyping and verification.

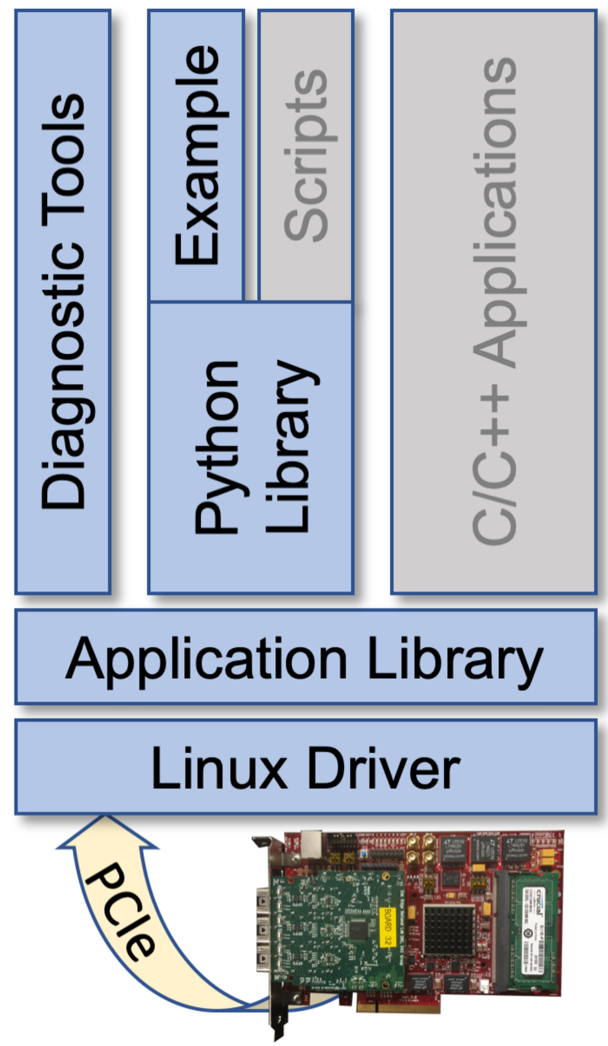


Figure 1: Device driver framework architecture.

LINUX DEVICE DRIVER

Detector data from SNS neutron scattering instruments are sent as packets over optical fiber to multiple PCIe read-out boards hosted in a Linux server. The read-out board uses a Xilinx FPGA to handle both the PCIe interface and the packet transfers with the detector electronics. The theoretical

* This manuscript has been authored by UT-Battelle, LLC under Contract No. DE-AC05-00OR22725 with the U.S. Department of Energy. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes. The Department of Energy will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan(http://energy.gov/downloads/doe-public-access-plan).
† vodopiveck@ornl.gov

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA157

GLOBAL INFORMATION MANAGEMENT SYSTEM FOR HEPS

C.H. Wang , P.Chu, Institute of High Energy Physics, Chinese Academy of Science, Beijing, China
H.H. Lv , Shanghai Institute of Applied Physics, Chinese Academy of Science, Beijing, China

Abstract

HEPS is a big complex science facility which consists of the accelerator, the beam lines and general facilities. The accelerator is made up of many subsystem and a large number of components such as magnets, power supply, high frequency and vacuum equipment, etc. Variety of components and equipment with cables are distributed installation with distance to each other. These components during the stage of the design and construction and commissioning will produce tens of thousands of data. The information collection and storage and management for so much data for a large scientific device is particularly important.

This paper describes the HEPS database design and application from the construction and installation and put into operations generated by the uniqueness of huge amounts of data, in order to fully improve the availability and stability of the accelerator, and experiment stations, and further improve the overall performance.

INTRODUCTION

High Energy Photon Source (HEPS) has constructed at suburban areas of Beijing in the end of June in this year. HEPS is a big complex science facility which consists of the accelerator, the beam lines and general facilities. The accelerator is made up of many subsystem and a large number of components such as magnets, power supply, high frequency and vacuum equipment, etc. Variety of components and equipment with cables are distributed installation with distance to each other. These components during the stage of the design and construction and commissioning will produce tens of thousands of data. The information collection and storage and management for so much data for a large scientific device is particularly important.

Database system will provide global database for lighting project and application services, including accelerator, beam line, HEPS project management, application software and database, thus ensuring HEPS whole big science device from construction, installation and put into operations generated by the uniqueness of huge amounts of data, in order to fully improve the availability and stability of the accelerator, and experiment stations, and further improve the overall performance.

This system designs the related database according to the comprehensive requirements of physics and management, and develops the Application Programming Interface (API) and some database Application software. After passing the test stage and initial operation, the database and application software will be deployed to the computer room of the computing and network communication system, and professional database managers will be responsible for daily operation management and performance optimization.

All database design and application source code will be stored in AcceleratorDatabase software set of GitHub server under modern software management specifications for domestic and international cooperation, and stored in HEPS internal Git server to ensure software security.

DATABASE DESIGN

The database and its related application software architecture also follow the overall architecture of the upper software in the accelerator control. The whole HEPS database covers a large range, and it is difficult to complete the design and construction in one time with limited human resources. Therefore, the database can be functionally broken up and modularized into several smaller databases to be developed separately and done in collaboration with other labs. Each module is connected by simple modification of Primary key and Foreign key, data acquisition API or software service.

The overall architecture of the accelerator and beam line (non-experimental data) database set was based on IRMIS[1] (Integrated Relational Model of Installed System) v3, and was designed based on the collaborative work of the former DISCS[2] international accelerator database. This part of the database will be open source database such as MySQL. Most of the databases related to project management are based on MS SharePoint, so most of them are stored in MS SQL database built in SharePoint in the form of document whole and content decomposition data.

For the mysql-based accelerator and beam line database, MySQL Workbench or other visual schema editing tools as shown in the Fig. 1 are used to design the database schema according to user requirements. After the schema preliminary design is completed, it is deployed to the MySQL server under test.

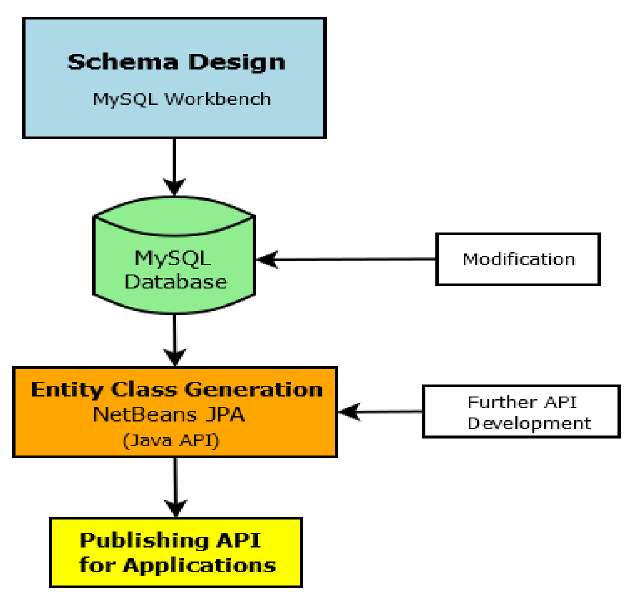


Figure 1: Typical relational database development.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA158

COMPACT ELECTRONIC LOGBOOK SYSTEM

L. Wang†, Y. L. Zhang, P. Chu, X. Wu, F. Q. Guo, Y. C. He, P. Zhu,
M. T. Kang, Z. Zhao, J. Liu, D.P. Jin
Institute of High Energy Physics, Chinese Academy of Sciences, Beijing, China

Abstract

Compact Electronic Logbook System (Clog) is designed to record the events in an organized way during operation and maintenance of an accelerator facility. Clog supports functionalities such as log submission, attachment upload, easy to retrieve logged messages, RESTful API and so on, which aims to be compact enough for anyone to conveniently deploy it and anyone familiar with Java EE (Enterprise Edition) technology can easily customize the functionalities. After the development is completed, Clog can be used in accelerator facilities such as BEPC-II (Beijing Electron/Positron Collider Upgrade) and HEPS (High Energy Photon Source). This paper presents the design, implementation and development status of Clog.

INTRODUCTION

In China, several accelerator facilities are being constructed right now or will be constructed in the near future, so there are strong needs for an electronic logbook system adopting new technologies and architecture to meet the requirements of modern accelerator controls and operations.

Clog is a web-based electronic logbook system, which is developed using Java EE [1] framework, PrimeFaces [2] component library, MySQL relational database and RESTful web service technologies. It forks from the source code of Olog [3] and adopts the display style of Elog [4]. Clog provides log submission, log query, log update and log deletion functionalities via the PrimeFaces UI (User Interface), it also provides RESTful API to communicate with CS-Studio and mobile web UI using HTTP (Hypertext Transfer Protocol) protocol.

SYSTEM OVERVIEW

Clog is developed using Java EE framework and deployed as a web application in GlassFish container. It implements a PrimeFaces UI to be accessed directly by any popular Web browsers and RESTful API to interface with CS-Studio and mobile web UI. The system architecture is shown in Fig. 1.

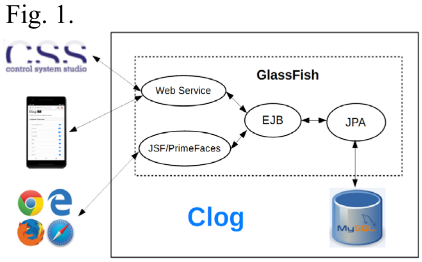


Figure 1: Architecture of Clog.

Clog is developed based on the source code of Olog and has made the following modifications to simplify the implementation:

- Replaced the HTML/Bootstrap/JQuery web UI with JSF/PrimeFaces in order to simplify the UI development, deployment and maintenance.
- Removed BitemporalLog data type, which is very complicated and unnecessary for a logbook system.
- Replaced the Jackrabbit attachment storage mechanism with plain file system, which is straightforward and will facilitate data backup in the future.
- Replaced the JPA (Java Persistence API) Criteria API with JPQL (Java Persistence Query Language) in order to improve the readability of the source code and easier to program database queries.
- Added "subject" field into log entry, which is useful to summarize the log entry.
- Removed "property" table from the database, which is not commonly used.
- Removed Flyway database migration tool, which is not necessary for a logbook system.

BACKEND IMPLEMENTATION

Database Design

Clog has five tables in the database as shown in Fig. 2, the description for the tables are as follows:

- The "entry" table corresponds to log entries.
- The "log" table corresponds to different log versions of a log entry.
- The "logbook" table is shared by both logbook and tag, and they are distinguished by the "is_tag" field.
- The "sysuser" table saves user information.
- The mapping between entry and log are one-to-many, which means one entry includes multiple logs and one log belongs to one entry.
- The mapping between log and logbook is many-to-many, which means one log can belong to multiple logbooks and one logbook can include multiple logs.

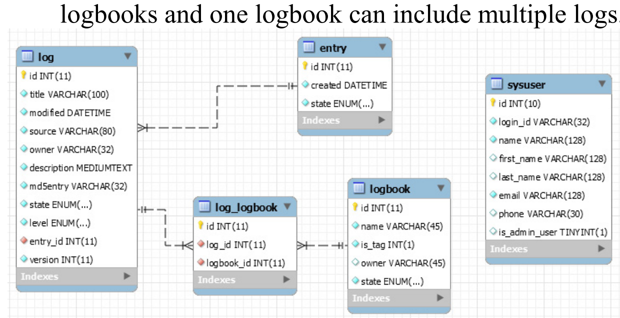


Figure 2: Clog database schema.

† wanglin@ihep.ac.cn

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA160

ENABLING DATA ANALYTICS AS A SERVICE FOR LARGE SCALE FACILITIES

K. Woods, R. Clegg, N. Cook, R. Millward, Tessella Ltd, Abingdon, UK
F. Barnsley, C. Jones, STFC/RAL, Didcot, UK

Abstract

The Ada Lovelace Centre (ALC) at the Science and Technology Facilities Council (STFC) is an integrated, cross-disciplinary data intensive science centre, for better exploitation of research carried out at large scale UK Facilities including the Diamond Light Source, the ISIS Neutron and Muon Facility, the Central Laser Facility and the Culham Centre for Fusion Energy (CCFE). ALC will provide on-demand, data analysis, interpretation and analytics services to worldwide users of these research facilities.

Using open-source components, ALC and Tessella have together created a software infrastructure to support the delivery of that vision. The infrastructure comprises a Virtual Machine Manager (VMM), for managing pools of virtual machines (VM) across distributed compute clusters; components for automated provisioning of data analytics environments across heterogeneous clouds; a Data Movement System (DMS), to efficiently transfer large datasets; a Kubernetes cluster to manage on demand submission of Spark jobs. In this paper, we discuss the challenges of creating an infrastructure to meet the differing analytics needs of multiple facilities and report the architecture and design of the infrastructure that enables Data Analytics as a Service.

INTRODUCTION

Advanced scientific research facilities in the United Kingdom (UK), such as ISIS, Diamond, Central Laser Facility (CLF) generate huge amounts of data. For example, for the past 4 years Diamond has generated approximately 4PB of data per year. This is expected to rise to 15PB per year within the next two years [1]. Scientists need advanced computing infrastructure, products and services to interpret and manage the data they obtain during their research. STFC's Scientific Computing Department (SCD) manages high performance computing facilities, services and infrastructure to support such research.

However, for many scientists, making use of advanced, high performance computing infrastructure can be difficult. For example, users of such facilities may be expected to know how to select the most appropriate compute resource for their work, how to set up and configure virtual machines, or how to move data between archives and local storage on compute clusters. Without appropriate experience and training, such tasks can be daunting.

ALC, which is part of SCD, has a remit to support STFC's science programme by building capacity in advanced software infrastructure for the handling, analysis, visualisation, integration, modelling, and interpretation of experimental data [2]. In pursuit of that goal, ALC and Tessella have developed new software components, Piezo, the VMM and DMS, designed to hide the complexity of using such infrastructure. Piezo, the VMM and DMS are designed to free researchers and scientists from the complex mechanics of managing datasets and environments, enabling them to focus on the analysis and interpretation of their data.

Piezo has been designed to support researchers using bespoke modelling and simulation codes, often developed by the researchers themselves. The VMM and DMS are software infrastructure components designed to work in together in support of researchers using mainstream data analysis tools.

Piezo, the VMM and DMS have all been created using readily available open-source technologies.

PIEZO

Scientists at CCFE have access to a single Spark [3] cluster for running modelling and simulation codes. Unfortunately, a small number of large codes can easily consume all of the available cluster resources, meaning that smaller codes, especially those in development, are effectively shut out. CCFE scientists needed a way to run experimental jobs, easily, with guaranteed availability and rapid turn-around. STFC has compute resources that it can make available to CCFE scientists; the problem faced by ALC and Tessella was how to make those resources available to CCFE scientists, while shielding them from the mechanics of transferring data and job management on remote resources.

Our solution, called Piezo, creates on-demand, single-user Spark clusters. The process of spinning up a Spark job is relatively straightforward; spinning up a Spark cluster on demand to process that job requires a lot of detailed technical knowledge, which many scientists do not possess. To orchestrate the management of these Spark clusters, we chose Kubernetes [4], because it provides facilities which simplify the automated set up of the Spark cluster. Piezo operates on two underlying systems:

1. a high-performance compute cluster.
   - a Kubernetes instance running the cluster to administer Spark jobs.
2. a storage platform with an S3 interface.

Spark jobs are run as Docker [5] containers to ensure consistency and repeatability between different execution platforms. We created a simple web Application Programming Interface (API) to shield scientists from the arcane technical details of creating and managing individual Spark jobs. The web API is responsible for:

- creating Spark jobs.
- managing Spark jobs.
- managing the job results and output files.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA163

THE DETECTOR CONTROL SYSTEM OF THE MUON FORWARD TRACKER FOR THE ALICE EXPERIMENT AT LHC

K. Yamakawa*, Hiroshima University, Higashi-Hiroshima, Japan
for the ALICE collaboration

Abstract

The ALICE experiment is presently finalizing its upgrade program toward the LHC Run 3 starting in 2021. The Muon Forward Tracker (MFT) is one of the new detectors developed in the framework of the ALICE upgrade program. In addition, the experiment is upgrading its whole online and offline (O^2) framework in order to dispose of the data volume from all ALICE detectors. In the context of the O^2, the slow control data are also processed together with the physics data. Therefore, the ALICE central Detector Control System (DCS) must be upgraded to operate within the O^2. The MFT DCS has been developed following the new ALICE central DCS strategy. The Finite State Machine (FSM) for hierarchical control is designed as a part of the MFT DCS. It was confirmed that the FSM has correct behaviour by changes of element states by a simulator. Furthermore, the MFT DCS successfully operates using a simplified but realistic test bench based on the final MFT elements.

INTRODUCTION

ALICE (A Large Ion Collider Experiment) is one of the LHC experiments at CERN. It focuses on the heavy-ion program to understand the Quark-Gluon Plasma (QGP), a state of deconfined quarks and gluons described by Quantum Chromodynamics.

During the first ten years of operation, the ALICE experiment has produced precise and original data in order to characterize the QGP formed in Pb-Pb collisions at LHC. Its upgrade programs toward the forthcoming LHC Run 3 in 2021 are being developed to realize measurements of rare proves allowing further study of the QGP. In the framework of this upgrade program, the Muon Forward Tracker (MFT) [1] is a new detector being installed in the forward region. In addition, the O^2 [2] is a new computing system to handle the data readout and reconstruction from all ALICE detectors together with the detector monitoring and slow control data. Finally, a totally new Detector Control System (DCS) for the MFT is being developed. In this report, we present the latest status of the MFT DCS development.

MUON FORWARD TRACKER

The MFT is a new silicon pixel detector devoted to the muon detection at forward rapidity in the range $-3.6 < \eta < -2.5$. The MFT will be installed between the collision point and the hadron absorber in front of the current muon spectrometer. The MFT improves the capability of muon tracking by track matching with the present muon spectrometer. Thanks to the high accuracy of determination of the displaced muon generation vertices by the MFT, prompt J/ψ and secondary J/ψ from B-meson decays can be separated. Moreover, a better resolution of the di-muon invariant mass spectrum can be achieved especially in the low-mass region.

The MFT consists of 2 half cones made of 5 disks (Fig. 1). Each being composed of 2 detection planes. Each plane is made of hybrid integrated circuits housing Monolithic Active Pixel Sensors (MAPS), which are a new type of sil-icon pixel detectors integrating both sensor and readout electronics in a single detection device based on the CMOS technology [3]. The MAPS allow a low material budget and effective signal collection.

A ladder has typically 2-5 sensor chips assembled on a flexible circuit made in Aluminum. A total of 280 ladders equipped with 936 sensor chips is needed for the full MFT. Each half detection plane of a disk is expediently separated into 4 zones consisting of 3 or 4 ladders. The power generation is handled locally in a Power Supply Unit equipped with DC-DC converters. A Readout Unit (RU) board collects the data from each zone.

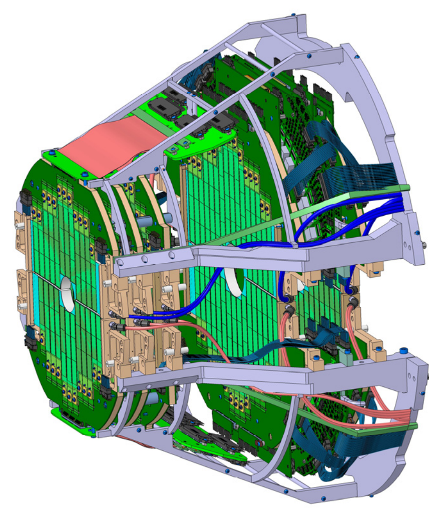


Figure 1: Schematic view of the MFT.

ALICE DCS IN THE CONTEXT OF O^2

The O2 is a new computing system in which online and offline systems are commonly merged. The slow control data share the data stream in the O2 facility with physics data from detectors. Detector conditions for each event are one of the ingredients for the data reduction in online reconstruction. Online data reduction is a very important process because the O2 farms need to handle raw data at a rate

* kosei.yamakawa@cern.ch

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA164

WIRE SCANNER FOR HIGH INTENSITY BEAM PROFILE DIAGNOSTICS*

J. Yan, J. Gubeli, K. Jordan
Thomas Jefferson National Accelerator Facility, Newport News, USA
B. Bailey, University of Tennessee, Knoxville, USA

Abstract

A control and data acquisition system of a high speed wire scanner is developed for high intensity beam profile diagnostics. The control system of the wire scanner includes two IOCs, a Soft IOC and a VME IOC. The Soft IOC connects with an Aerotech Ensemble motor drive through EPCIS motor record and controls the movement of the wire scanner. An Electrical Input card samples the real-time position of the wire through an incremental encoder, and generates a pulse to synchronize a VME ADC data acquisition card, which digitizes and samples the beam-induced signal after pre-amplification. A VME Relay Output card is installed to control the Brake Solenoid and Actuator Solenoid. All the VME I/O cards are installed on one VME crate and controlled by the VME IOC. The system configuration and software of the wire scanner are under development.

INTRODUCTION

The beam profile measurement device, commonly known as a harp, consists of a 45 degree thin wire and two orthogonal thin wires which pass in a controlled manner through the electron beam[1,2,3]. When the wire intercepts the beam, a beam-induced current, as a function of wire position, produces a functional x-y plot. By plotting the number of steps against the charge collected by an analog-to-digital convertor (ADC), the beam's profile is determined when the data is fitted to a Gaussian distribution. Wire scanners are among the most reliable diagnostic tools presently available at electron- and ion-beam accelerator facilities and are crucial to their operation. Wire scanners are interceptive devices and their wires have damage thresholds that can be exceeded with intense beams, causing them to melt or break. RadiaBeam Technologies has designed and manufactured a robust high speed wire scanner for high intensity beams using a new wire material: boron-nitride nanotubes (BNNT)[4]. The RadiaBeam wire scanner (RWS) will be installed in the Low-Energy Recirculator Facility (LERF) at Jefferson Lab to investigate the behavior of wires in a high-energy electron beam. This paper details the mechanical structure of the scanner, the control, and data acquisition system.

THE WIRE SCANNER SYSTEM

The complete RadiaBeam wire scanner system includes linear motion, vacuum chamber, actuation, detection, and associated controls. Figure 1 shows the picture of the RWS system. The Aerotech linear motor is driven by an Aerotech Ensemble HPe 75 high power PWM network digital drive which provides deterministic behavior, auto-identification, and easy software setup. The actuator, attached to the linear motor, holds the thread fork and moves the wires inside the chamber without breaking vacuum. As the conductive wire passes through the beam, it collects a very small charge, this current leaves the vacuum assembly via an SMA connector and travels to a pre-amplifier through a cable, and eventually is measured by an ADC DAQ card. However, BNNT wire is non-conductive and it can't pick up a charge, so a high speed photodiode (PD) or a photomultiplier (PMT) is used to investigate the beam profile as the wire passes through the beam. Two types of encoders, an absolute encoder and an incremental encoder, are installed to measure the position of the wires. The absolute encoder sends the position information to the motor drive while the incremental encoder sends it to the control system. During the beam test, the wires will be optically imaged as they move into the beam and recorded in the process. A CCD video camera and a Si photodiode are installed on a viewport of the vacuum chamber so that light emitted from the wires can be collected for imaging and intensity monitoring simultaneously.

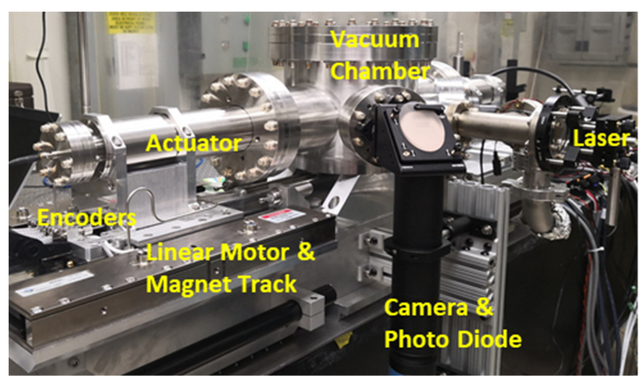


Figure 1: The picture of the RWS system.

SYSTEM CONTROL AND DATA PROCESSING

The control and data processing of the wire scanner system mainly consists of two parts, a software driver for motion control and hardware interface cards for device control and data acquisition. EPICS applications and firmware for the whole system were developed. Figure 2 shows the block diagram of the system.

* Work Authored by Jefferson Science Associates, LLC under U.S. DOE Contract No. DE-AC05-06OR23177.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA165

AN EMBEDDED IOC FOR 100 MeV CYCLOTRON RF CONTROL

Z.G. Yin†, X.L. Fu, X.T. Lu, T.J. Zhang, China Institute of Atomic Energy, Beijing, China
X.E. Mu, North China University of Technology, Beijing, China

Abstract

An ARM9 based embedded controller for 100MeV cyclotron RF control has been successfully developed and tested with EPICS control software. The controller is implemented as a 3U VME long card, located in the first slot of the LLRF control crate, as a supervise module that continuously monitors the status of the RF system through a costume designed backplane and related ADCs located on other boards in the crate. For high components density and signal integrate considerations, the PCB layout adopts a 6 layer design. The Debian GNU/ Linux distribution for the ARM architecture has been selected as an operating system for both robustness and convenience. EPICS device support, as well as Linux driver routings, has been written and tested to interface database records to the onboard 12 multichannel 16bits ADCs and DACs. In the meantime, a chip selecting encoding-decoding strategy has been implemented from both software and hardware aspects to extend the SPI bus of the AT91SAM9g20 processor. The detailed software, as well as hardware designed, will be reported in this paper.

INTRODUCTION

CYCIAE-100 cyclotron is the proton-driven accelerator of Beijing Radioactive Ion Facility. It provides a continuously adjustable high-intensity proton beam with energy up to 100MeV and intensity up to 520uA [1]. The CYCIAE-100 cyclotron radiofrequency system consists of two sets of room temperature resonators, two 100kW power amplifiers and two sets of Low-Level RF systems [2-5]. The LLRF crate is located in the RF power amplifier room, typically with no operators nearby. So, it is required to supervise the RF system parameters such as the dee voltage amplitude, phase, incident power, and reflected power, etc. in the cyclotron control room, during daily operations. In the field of accelerator control, the EPICS software package is widely adopted to build control system and gain remote access over ethernet. Generally speaking, EPICS IOC control software can run on various kinds of hardware and operating systems. For example, it runs on x86 PC with windows and arm SOCs with Linux operating system. In order to reduce the volume and power consumption, the reported IOC was designed based on arm9 SOC chip with embedded Linux OS. Thus, it can be installed in the LLRF chassis as a long 3U VME card, as shown in Figure 1. The hardware, firmware and software design of this IOC module will be reviewed in the following section of this paper.

HARDWARE

ARM9 series processors combine the advantages of high computational power with a small footprint. It also provides advantages such as high reliability and low power consumption. These features make it ideal for embedded controller design. In recent years, IOC based on ARM9 processor has been widely adopted for the control of many large scientific devices around the world. For the hardware designs of the IOC described in this paper, an AT91SAM9g20 processor has been selected as the central processing unit. Besides, the hardware design also includes 64MB SDRAM as memory, selects NAND FLASH and SD card to store OS and data. Other related hardware resources are one USB bus, two serial ports, two SPI interfaces, one Ethernet, 40 GPIO, etc. The block diagram of the hardware design of the embedded IOC is shown in Figure 2.

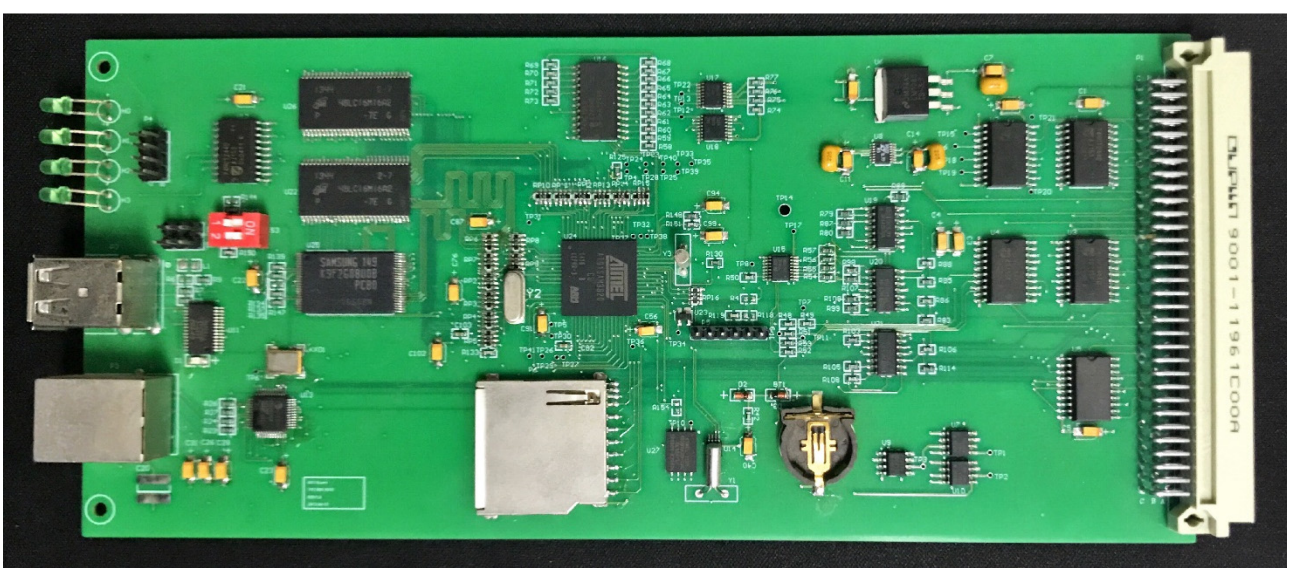


Figure 1: The embedded IOC as VME 3U module.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA167

CLOUD COMPUTING PLATFORM FOR HIGH-LEVEL PHYSICS APPLICATIONS DEVELOPMENT *

T. Zhang† and D. Maxwell
Facility for Rare Isotope Beams, Michigan State University, East Lansing, USA

Abstract

To facilitate software development for the high-level applications on the particle accelerator, we proposed and prototyped a web-based computing platform, the so-called 'phyapps-cloud'. Based on the technology stack composed by Python, JavaScript, Docker, and Web service, such a system could greatly decouple the deployment and development, that is the users only need to focus on the feature development by working on the infrastructure that served by 'phyapps-cloud', while the service provider could focus on the development of the infrastructure. In this contribution, the development details will be addressed, as well as the demonstration of developing Python scripts for physics tuning algorithm on this platform.

INTRODUCTION

With the rapid evolution of information technology, the development of high-level controls software in the accelerator community has been dramatically changing. To apply the most state-of-the-art technology to the accelerator controls application development is like standing on the shoulders of giants, the modern application could always be expected.

High-level physics controls software is to solve the problems regarding how to control the machine with some complex physics tuning algorithms, could depend on complicated physics model, or some generic data crunching routine. The development of the high-level physics applications usually requires a backend service running which can be treated as the data source to the applications, including to accept the input from the applications and to respond to the applications with output, such service typically is a so-called virtual accelerator, which is driven by a specific physics model engine.

The data communication between the application and virtual accelerator should be well abstracted, such that the developed application can work with real accelerator rather than the virtual one. For example, the implemented EPICS-based [1] virtual accelerator is an IOC application, which also exists in the controls network, the high-level application should be able to work with either of them.

The conventional way to develop the high-level physics applications usually requires the developer to install all the required software in a Linux workstation or make use of the well-packed VirtualBox [2] appliance in the VirtualBox application in any host OS. At FRIB, all the requirements have already been packaged into Debian packages, the developer can easily install all of them by 'apt-get' commands on any computer running Debian 8 or 9 OS, and the development work could start. While there is still the maintenance overhead to keep the packages updated and troubleshoot the configuration issues. So, the web-based platform for physics applications named as 'phyapps-cloud' was designed and implemented [3].

Here is the significant advantage of 'phyapps-cloud'. On one hand, the users (the app developers and users) can do the development in the web browser from anywhere, there is no requirement for the development environment. On the other hand, the platform developer can keep 'phyapps-cloud' always updated, with little maintenance effort. By utilizing Docker [4] to containerize the physics applications development environment, the maintenance effort can be further reduced. In this paper, the design, implementation and use case of 'phyapps-cloud' is presented.

SYSTEM ARCHITECTURE

The designed architecture for 'phyapps-cloud' is sketched in Fig. 1. Below lists the main components to compose 'phyapps-cloud':

- `Gateway`: REST web service features the main entrance of 'phyapps-cloud', and the center for the users and services management;
- `Configurable Proxy`: Web service features configurable proxy, which provides a way to update and manage a proxy table by REST API;
- `Service-`*i*: Private service created by the users, currently, two different services are supported, both features with Jupyter-Notebook service and FRIB physics application development environment, but one also comes with the FRIB virtual accelerator service, while the other does not.

The `Configurable Proxy` web service is maintained by the opensource community [5], which is one part of the JupyterHub project [6], while the others are developed at FRIB as a byproduct of 'phantasy-project' [7], they are an alternative method of deployment for physics applications development based on web technology.

`Gateway` REST web app is developed with FLASK [8], which is a lightweight WSGI web application framework for Python [9]. It serves as the main portal where the user connects to 'phyapps-cloud', as Fig. 2 shows. All the new users can sign up for the first time to get onto the platform, then

* Work supported by the U.S. Department of Energy Office of Science under Cooperative Agreement DE-SC0000661, the State of Michigan and Michigan State University.
† zhangt@frib.msu.edu

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA169

DESIGN OF VACUUM CONTROL SYSTEM FOR SUPERCONDUCTING ACCELERATOR

J. M. Zhou, A. L. Li, K. N. Li, C. H. Peng, J. Zheng
China Institute of Atomic Energy, Beijing, China

Abstract

A linear superconducting accelerator is being constructed in our institute. Its vacuum control system should be convenient and reliable. We intend to concentered the control of each vacuum unit into a control box that implement the simple hard interlocking logic and the final action output of the vacuum device and the complete interlocking logic between the vacuum devices is realized in the PLC. Operators can perform local operation through the front panel of the control box or remotely control through the computer by switching the local/remote switch. In addition, the control flow of vacuum extraction and the protection flow when leakage occurs are also given in this paper.

INTRODUCTION

Based on the introduced linear superconducting accelerator, our institute (CIAE) is carrying out technical upgrading to restore its normal working state, and further increasing the beam energy provided by HI-13 tandem accelerator to meet user's requirements.

The design indicators of this sub-item are as follows:

The average energy per nucleus of mass 25-240 heavy-ions is increased by 4 MeV.

Energy resolution $\Delta E/E <= 5 \times 10^{-4}$

Beam intensity>=20 Pn A

Bunching width T≈30 Ps

The main design work for this sub-item includes:

High frequency system

liquid helium cryogenic system

Vacuum system

Beam transport system

Accelerator control system

Beam measurement system

Superconducting acceleration module

Auxiliary system

VACUUM SYSTEM

A total of 31 sets of vacuum units are installed on the linear superconducting accelerator. Each unit consists of a pre-stage dry scroll pump, a solenoid valve, a molecular pump, a vacuum valve and a measuring vacuum gauge. The vacuum gauge provides two measured values, that is, a low vacuum resistance gauge measurement, its range is from 1×10^{5} to 1×10^{-1} Pa [1], and the measurement range of the ionization gauge is from 1×10^{-1} to 1×10^{-7} Pa [1]. The connection is illustrated in Fig. 1.

The vacuum system of linear superconducting accelerator needs to consider the vacuum situation at the constant temperature cabinets and beam pipes, the pre-stage pump adopts 8 L/s turbine dry pump [2], which can pump the vacuum degree of the constant temperature cabinet from atmospheric state to about 1 Pa in one hour, and the main pump adopts 400 L/s compound molecular pump [3], which can pump the vacuum degree from 1Pa to 10^{-4} Pa in three hours.

The beam pipeline is a stainless-steel pipe with a diameter of 100 mm. The vacuum requirement of the pipeline can be satisfied if the interval between the two units is not more than 14.4 meters. At the same time, 12 vacuum valves are installed in different positions of the beam pipeline, which can isolate different parts of the pipeline.

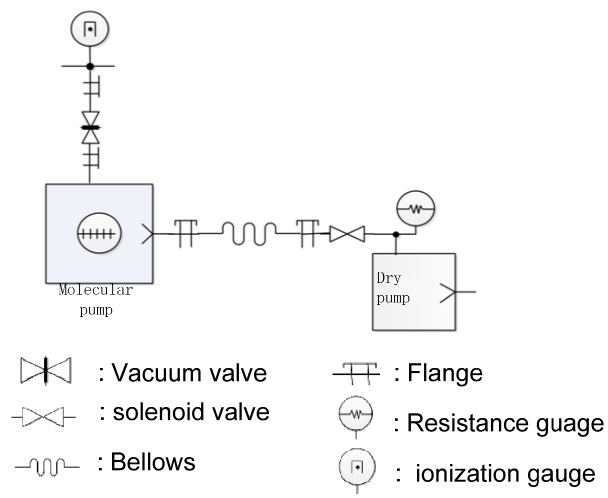


Figure 1: Vacuum unit.

Control Scheme of Vacuum Unit

An on-site control box is designed, which integrates the start and stop functions of each device. The operator can operate the vacuum device through the buttons on the front panel. At the same time, the status signal and control signal of the device and the control signal of the PLC are connected into the control box, in which the basic interlock logic and control outputs of the device are implemented, while the status signals of the device are transferred to the PLC. The system control provides three modes: debugging mode, local mode and remote mode. The debugging mode is to operate the vacuum device in the field to verify the validity of the device without PLC; the local mode is to transmit the operation signal of the site to the PLC. After the interlock logic is executed in the PLC, the action command is sent to the execution relay in the control box to make the device operate, such as starting and stopping; The remote mode means that the operator sends an instruction through the operation interface on the remote computer, performs the same action as the local mode in the PLC, and finally controls the running state of the device. At the same time, after the local mode and the remote mode are

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOPHA173

GRAPHICAL USER INTERFACE PROGRAMMING CHALLENGES MOVING BEYOND JAVA SWING AND JavaFX

S. Bart Pedersen, S. Jackson, CERN, Geneva, Switzerland

Abstract

Since Oracle has decided [1] to stop supporting Java Swing and JavaFX in 2018, replacing Java as our language for the development of graphical user interfaces has posed numerous challenges. Many programmers in the CERN Accelerator Sector will have to adapt and to be re-trained in whatever the next technology will be. Performance of the new GUIs will have to be at least as fast as the existing ones. In addition, programming environment, code versioning, repositories, dependencies and documentation need to be considered as well when choosing the new technology.

First, this paper provides an overview of the research done by the software section of the CERN Beam Instrumentation Group related to comparing GUI languages and explains the reasons for selecting PyQt as a possible future technology. Secondly, the basis for starting a project in PyQt is defined as a guideline for programmers and providing recommendations for the Python [2] environment to be supported.

INTRODUCTION

The software section of the CERN Beam Instrumentation Group (BI) has the mandate to implement real-time servers in C++ that control and monitor instruments developed for beam diagnostics located throughout the CERN accelerator complex. These servers are designed and implemented using an in-house software framework called FESA (Front-End Software Architecture) [3].

The section is also mandated to provide expert graphical user interfaces (GUI) which until now have been developed almost exclusively in Java. These GUIs allow hardware experts easy access to their equipment for parameter setting, signal visualization, error diagnostics, calibration, data post processing and so on. This relies on the underlying, low-level software architecture, middleware (ex: Communication MiddleWare CMW [4]) and the various Java component libraries.

Due to the decision from Oracle to stop supporting Java Swing and JavaFX in 2018, alternative software platforms are currently being tested to replace Java for writing new expert GUI applications. This gives us the opportunity to plan for a good technological solution from a long-term perspective, addressing new needs highlighted by the GUI users, and the limitations identified in the current Java-based solutions.

In defining what a good graphical interface should be, it is important to differentiate between the choice of a programming technology versus the functionalities and performance of the resulting GUI. The user should be presented with features similar to what is available with the current Java implementations, whilst providing improved performance if possible. A comparison of different languages and a list of mandatory graphical functionalities are essential to make a fair analysis and the right final choice.

GUI USAGE AND EXPECTED GRAPHICAL PERFORMANCES

Context and Usage

The main purpose of an expert GUI is to give access to the hardware of any beam instrumentation system through a FESA interface. The GUI is mainly targeted to hardware or software specialists although these GUIs are sometimes also used for machine operation.

Most expert applications are written in Java Swing and more recently JavaFX with a few applications developed with Qt (C++ and PyQt [5]). The applications only run on the non-public technical network (TN) at CERN and, thanks to Java, they execute under both the Windows and Linux operating systems with the only difference being the graphical rendering performance. Some experts have expressed the need to run expert applications on other platforms (mobile telephones, tablets...) and from the CERN general public network (GPN). As it is, most expert GUIs are only connected to FESA devices, but this could be extended to direct access to low-level hardware in the future, which means a direct connection to the hardware driver. A handful of applications are already communicating directly to the hardware through TCP/IP and UDP using in-house protocols, but these applications are rare. Interfaces to retrieve data both from the logging database and from the LHC post-mortem data [6] files are also available. Figure 1 shows all the different communication links between an expert GUI or a Python script and the hardware (HW).

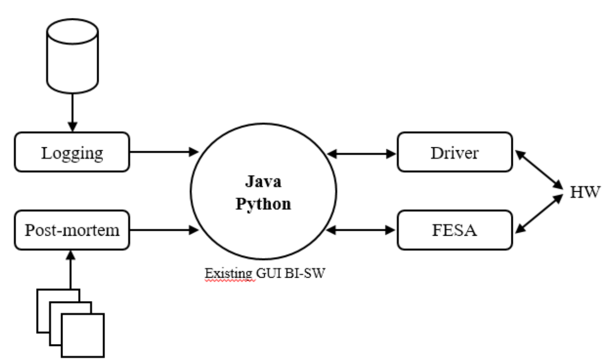


Figure 1: Snapshot of the actual links between the expert programs, the hardware and the databases.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOSH1001

CURRENT STATUS OF KURAMA-II*

M. Tanigaki†
Institute for Integrated Radiation and Nuclear Science, Kyoto University,
Kumatori, Osaka, Japan

Abstract

KURAMA-II, a successor of a carborne gamma-ray survey system named KURAMA (Kyoto University RAdiation MApping system), has become one of the major systems for the activities related to the nuclear accident at TEPCO Fukushima Daiichi Nuclear Power Plant in 2011. The development of KURAMA-II is still on the way to extend its application areas beyond specialists. One of such activities is the development of cloud services for serving an easy management environment for data management and interactions with existing radiation monitoring schemes. Another trial is to port the system to a single-board computer for serving KURAMA-II as a tool for the prompt establishment of radiation monitoring in a nuclear accident. In this paper, the current status of KURAMA-II on its developments and applications is introduced.

INTRODUCTION

The magnitude-9 earthquake in eastern Japan and the following massive tsunami caused a serious nuclear disaster for the Fukushima Daiichi nuclear power plant. Serious contamination by radioactive isotopes was caused in Fukushima and surrounding prefectures, but the existing radiation-monitoring schemes were incompetent for this situation due to damage and chaos caused by the earthquake.

KURAMA [1] was developed to overcome difficulties in radiation surveys and to establish air dose-rate maps during and after the incident. The design of KURAMA was intended to enable a large number of in-vehicle apparatuses to be prepared within a short period of time by using consumer products. The in-vehicle part of KURAMA consists of a conventional radiation survey meter, a laptop PC, a USB-type GPS dongle, and a 3G pocket wifi router. The data-sharing scheme based on a cloud-technology has enabled high flexibility and scalability in the configuration of data-processing hubs or monitoring cars. KURAMA succeeded in the simultaneous radiation monitoring extended over a wide area such as Fukushima prefecture and eastern Japan, in contrast to other conventional carborne survey systems lacking scalability.

As the situation became stabilized, the main interest in measurements moved to the long-term (several tens of years) monitoring of radiation from radioactive materials remaining in the environment. KURAMA-II [2] was developed for such purpose by introducing the concept of continuous monitoring from vehicles moving around residential areas, such as local buses and postal motorcycles. The ruggedness, stability, autonomous operation and compactness were well taken into consideration in its design, and an additional measurement capability of pulse-height information along with location data was also introduced. KURAMA-II has been successfully introduced to the continuous monitoring in residential areas and other monitoring activities [3–6], and the accumulated knowledge is summarized and standardized in the radiation monitoring in Japan [7].

In this paper, the outline and the current status of KURAMA-II along with some results from its applications are introduced.

KURAMA-II

Measurement Unit

KURAMA-II consists of a group of measurement units connected over the network [2]. Each measurement unit is based on CompactRIO by National Instruments to obtain sufficient ruggedness, stability, compactness, and autonomous operation feature. The radiation-detection part of KURAMA-II is the C12137 series by Hamamatsu Photonics [8], a MPPC-based CsI(Tl) detector series characterized by its compactness, high efficiency, direct ADC output and USB bus power operation. The ambient air dose rate, $H^*(10)$, is calculated from the pulse height spectrum obtained for each measurement point by using the G(E) function method [9–11]. Since the energy dependence of detector efficiency is properly compensated by G(E) function, more reliable results for environmental radiations are expected than those from GM counters, which just count the number of incoming γ-rays without identifying the energy of each γ-ray.

Data Management in KURAMA-II

The file transfer protocol used in KURAMA-II has been designed to comply with the standard protocols widely used in today's networks, such as Web Services. In this protocol, two timestamped files, a text file for the air dose rates and a 32-bit binary file for the pulse-height spectra, are separately produced for every three measurement points. Generated files are transferred to a remote server by the POST method. All communications between measurement units and a remote server are based on RESTful API. Unsent files inside a measurement unit are archived as a single zip file for the next available network connection. The server generates the data files of radiation and energy spectrum based on the received data for respective measurement units. A new visu-

* This work is supported by a grant for the Advanced Agriculture and Forestry Robot Research and Development Project from Agriculture, Forestry and Fisheries Research Council, MAFF, Japan, and by a research fund of Radiation Safety Research Promotion Project by Nuclear Regulation Authority, Japan.
† tanigaki@rri.kyoto-u.ac.jp

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOSH1002

adviewer: THE EPICS AREA DETECTOR CONFIGURATOR YOU DIDN'T KNOW YOU NEEDED *

Kenneth Lauer†, SLAC National Accelerator Laboratory, Menlo Park, CA, USA

Abstract

EPICS areaDetector connects area detector cameras to plugin pipelines through the standard flat namespace that EPICS provides. Visualizing and re-configuring this port connectivity in AreaDetector can be confusing and - at times - painful. adviewer provides a Qt-based interactive graph visualization of all cameras and plugins, along with per-plugin configuration capabilities and integration with an image viewer. adviewer is built on Python, ophyd, typhon, qtpynodeeditor, and Qt (via qtpy).

BACKGROUND

areaDetector

Each areaDetector [1] related device class from ophyd [2] redresents a single 'port', i.e., a camera or plugin. All plugins have a source port, indicating the location from which their input data will be retrieved. Cameras do not have a source port, as the information is sourced from a lower level (i.e., the C++ driver support).

To connect one camera to a plugin, one would need to set the plugin's source port to that of the camera. An ordered set of camera and any number of plugins is referred to in this document as a chain. A full configuration of plugins and their corresponding cameras can be represented in several intuitive ways, including as a directed acyclic graph comprised of one or more chains, or as a tree in which the parent of an item indicates its data source where a depth-first traversal reveals the individual chains.

adviewer provides a graphical user interface with both representations of detector configurations, an interactive tree or graph.

areaDetector DEVICES IN ophyd

ophyd makes available many device abstractions from the broader EPICS [3] community - from the motor record, scalers, and so on. There is first-class support for areaDetector detectors and plugins.

Camera and plugin support is summarized in Table 1. Support for unlisted cameras and plugins are welcomed in the form of Pull Requests to the main ophyd repository [4].

Versioning

All ophyd devices can be versioned[1] with user-provided metadata at the device definition time. Versioning is done in a class hierarchy, such that later versions are subclasses of previous versions and are marked as versions of the base.

Table 1: areaDetector Support in ophyd

Plugins	Cameras
AttrPlotPlugin	AdscDetectorCam
AttributePlugin	Andor3DetectorCam
CircularBuffPlugin	AndorDetectorCam
CodecPlugin	BrukerDetectorCam
ColorConvPlugin	DexelaDetectorCam
FFTPlugin	FirewireLinDetectorCam
FilePlugin	FirewireWinDetectorCam
GatherPlugin	GreatEyesDetectorCam
HDF5Plugin	Lambda750kCam
ImagePlugin	LightFieldDetectorCam
JPEGPlugin	Mar345DetectorCam
MagickPlugin	MarCCDDetectorCam
NetCDFPlugin	PSLDetectorCam
NexusPlugin	PcoDetectorCam
Overlay	PerkinElmerDetectorCam
OverlayPlugin	PilatusDetectorCam
PluginBase	PixiradDetectorCam
PosPlugin	PointGreyDetectorCam
ProcessPlugin	ProsilicaDetectorCam
PvaPlugin	PvcamDetectorCam
ROIPlugin	RoperDetectorCam
ROIStatPlugin	SimDetectorCam
ScatterPlugin	URLDetectorCam
StatsPlugin	
TIFFPlugin	
TimeSeriesPlugin	
TransformPlugin	

Users may select individual versions manually or programmatically request the most compatible class with a specific release.

This is especially of use for areaDetector as the PV interface to each of the cameras and plugins has changed significantly over the many releases from R1-9 through R3-7.

Cameras

Cameras in areaDetector generally differ from detector to detector. ophyd provides a base class that all cameras are derived from, along with individual cameras for each detector.

Standard Plugins

The standard plugins provided in areaDetector ADCore are all made available, based on a standard 'PluginBase' class, separated by plugin class and versioned by ADCore. As of the ophyd v1.4.0rc2, all listed plugins are supported from areaDetectorR1-9-1 through R3-4, inclusive.

* Work supported by U.S. D.O.E. Contract DE-AC02-76SF00515.
† klauer@slac.stanford.edu
[1] As of the v1.4 release candidate

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOSH3001

AN EPICS CHANNEL ACCESS IMPLEMENTATION ON SIEMENS PLCs

M. Boros†, R. Fernandes, European Spallation Source, Lund, Sweden
B. Péceli, G. Singler, evopro Innovation Kft, Budapest, Hungary

Abstract

At the European Spallation Source (ESS), a neutron research facility in Sweden, most of the controls are based on PLCs and layered in the following (traditional) way: field equipment ↔ PLC ↔ EPICS IOC ↔ high-level applications. In many situations, the EPICS IOC layer will not implement control logic *per se* and is only used for converting PLC tags into EPICS PVs to enable the usage of high-level applications such as CS-Studio, Archiver Appliance, and Alarm System.

To alleviate this (traditional) way of doing controls, we propose a simpler approach: implementation of the Channel Access (CA) protocol in the PLC layer for the latest family of Siemens PLCs to remove the EPICS IOC layer. We call it S7EPICS. S7EPICS respects version 13 of the CA protocol specification, and supports multiple EPICS-based client connections at the same time – e.g. CS-Studio, Archiver Appliance – without a noticeable service degradation (i.e. delays).

In this paper we introduce this implementation, its architecture and workflow, benchmarking results of tests performed, and future developments that could be pursued such as authentication & authorization mechanisms using, e.g., the Arrowhead Framework.

INTRODUCTION

Integrating a Siemens PLC with EPICS [1] high-level applications requires an EPICS Input/Output Controller, or IOC, properly configured with EPICS modules capable of communicating with the PLC using, e.g., s7plc-comms [2] or OPC UA [3]. Configuring an IOC, even for small projects, demands EPICS programmer skills, time and other valuable resources – e.g. a machine prepared to run the IOC. In some (control) use-cases, it would be simpler and more effective if the PLC could "talk" the Channel Access (CA) protocol [4] itself so that it could be interfaced directly with high-level applications.

In this paper, we describe an open source PLC code called S7EPICS [5], which allows a Siemens S71500 family PLC instance to declare and handle EPICS Process Variables (PVs) that EPICS aware applications may consume directly without an actual IOC running.

S7EPICS is a PLC code implemented as a TIA Portal library. When the library is imported into an existing TIA project and it is called in the main PLC cycle, the PLC becomes a fully-fledged CA server capable of receiving and sending EPICS CA protocol messages through UDP (to discover producers – i.e. servers – of PVs) and TCP packets (to exchange PVs data between producers and consumer – i.e. servers and clients).

† miklos.boros@esss.se

DESCRIPTION

Typically, a common control system scenario encompasses one or more field equipment that are controlled/measured by a PLC. The PLC, eventually equipped with I/O cards, is located in a control cabinet and field equipment are connected to either the PLC CPU or to one of its I/O cards. The business logic is implemented at the level of the PLC, working as a self-contained and independent (hard) control system. In addition, field equipment's signals – interfaced by the PLC – are usually consumed by high-level applications to solve domain specific issues.

In case of the (soft) control system is based on EPICS (and consequently high-level applications too – e.g. CS-Studio for OPI screens designing, Archiver Appliance for signals archiving), the integrator has to create and configure an EPICS IOC that 1) communicates with the PLC CPU or its I/O cards and 2) "converts" the PLC tags into EPICS PVs in a bidirectional communication. The IOC does not implement any control logic though, and (only) has as a main function to map PLC variables into corresponding EPICS PVs. To implement this mapping, the addresses and, sometimes, the offsets of all communication variables have to be specified in the EPICS IOC configuration, which is time consuming and prone to error. Moreover, the IOC has to be maintained and executed on a (physical or virtual) machine, adding an extra layer of complexity to the control system as a whole. Figure 1 illustrates the traditional PLC-EPICS integration involving an IOC layer between the PLC and EPICS high-level applications.

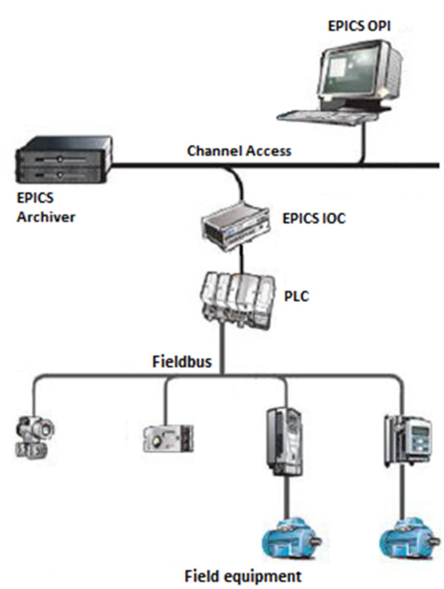


Figure 1: Traditional PLC-EPICS integration.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOSH4001

A LIBRARY OF FUNDAMENTAL BUILDING BLOCKS FOR EXPERIMENTAL CONTROL SOFTWARE

M. Scarcia†, R. Borghes, M. Lonza, M. Manfredda, E. Pedersoli, R. Mincigrucci
Elettra Sincrotrone Trieste S.C.p.A., Trieste, Italy

Abstract

In many experimental facilities there is a rising interest by users and beamline scientists to take part in the experiment control software development process. This necessity arises from the flexibility and adaptability of many beamlines, that can run very different experiments, requiring changes in the software even during experiments.

On the other side, we still need a professional and controlled approach in order to be able to maintain the software efficiently.

Our proposed solution is to exploit the object oriented nature of programming languages to create a library that provides a uniform interface both to the different controlled devices and to experimental procedures. Every component and procedure can be represented as an object, a building block for experiment control scripts.

INTRODUCTION

The scientific data acquisition workflow at the FERMI [1] facility of Elettra Sincrotrone is organized around two software components. The FermiDAQ [2] takes care of the actual data sources acquisition (including corresponding metadata) and of its synchronization with the information gathered from the FEL bunches. The Executer runs different experimental procedures in the form of Python scripts.

Initially some standard experiment templates were defined for each beamline by its staff and were implemented through Python scripts. These scripts were meant not be modifiable by beamline scientists and users, but only managed by the software developers in order to maintain a secure and robust approach to the underlying control and data acquisition systems.

This approach proved in time to be impractical on the FERMI FEL, where experimental setups and procedures were observed to change sometimes even within a single beamtime. Furthermore, even after the commissioning of the machine was officially concluded a lot of internal research has been carried out by the machine physics and beam transport groups, which is logistically usually based on beamline software infrastructure. This type of work often requests a rather different workflow than user oriented research.

The secondary motivation for this work derives from a different problem we encountered in day-to-day operations. The Fermi control infrastructure, which is based on the TANGO framework [3], comprises different software devices that have different origins and usually different development histories. This means in practice that different devices of the same type (e.g. motor controllers) have different command names for the same action (e.g. "Move" instead of "MoveTo"), requiring specific code to be written for each different device even if the actions to be performed are the same. A uniform upper level interface for every type of "abstract" device in the experiment control and data acquisition would therefore be a welcome tool.

OVERVIEW AND IMPLEMENTATION

Two main lines of development are currently being merged in this project. The first regards the creation of elementary "building blocks" with which experimental procedures can be built with greater ease and simplicity. The second concerns producing a uniform interface and the possibility of grouping different devices performing similar tasks. The goal is to have a model that follows experimental flow rather than the control system organization. Figure 1 shows a schematic view of the proposed library and its interaction with existing components.

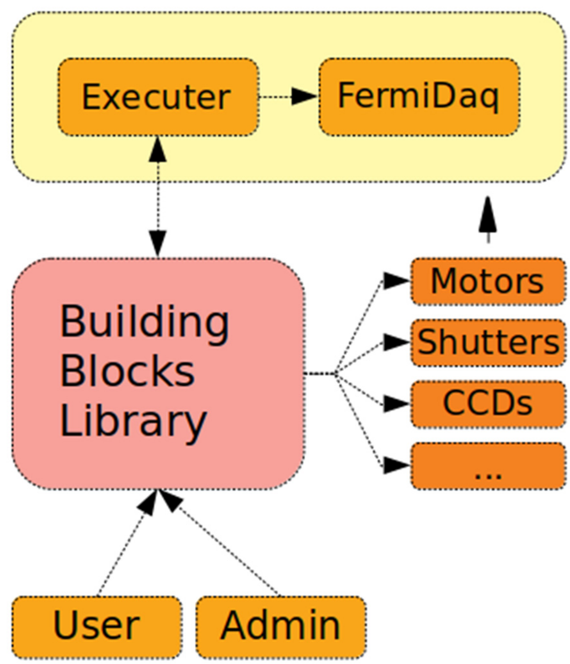


Figure 1: Library interactions.

Reusable Experimental Sequences

After the workflows on FERMI beamlines became more stabilized we realized that the abstraction level where every script is the equivalent of an experiment was too high for many common cases of beamline operation. The natural solution in order to decrease the level of abstraction was to find the common elements in the workflow and codify them as Python objects.

† martin.scarcia@elettra.trieste.it

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-MOSH4002

A CLOUD BASED FRAMEWORK FOR ADVANCED ACCELERATOR CONTROLS*

J. P. Edelen†, M. Keilman, P. Moeller, R. Nagler, RadiaSoft LLC, Boulder CO, USA

Abstract

Modern particle accelerator facilities generate large amounts of data and face increasing demands on their operational performance. As the demand on accelerator operations increases so does the need for automated tuning algorithms and control to maximize uptime with reduced operator intervention. Existing tools are insufficient to meet the broad demands on controls, visualization, and analysis. We are developing a cloud based toolbox featuring a generic virtual accelerator control room for the development of automated tuning algorithms and the analysis of large complex datasets. This framework utilizes tracking codes combined with with algorithms for machine drift, low-level control systems, and other complications to create realistic models of accelerators. These models are directly interfaced with advanced control toolboxes allowing for rapid prototyping of control algorithms. Additionally, our interface provides users with access to a wide range of Python-based data analytics libraries for the study and visualization of machine data. In this paper, we provide an overview of our interface and demonstrate its utility on a toy accelerator running on EPICS.

INTRODUCTION

As the demands on accelerator facilities increase so does the need for automated tuning algorithms and control to maximize uptime with reduced operator intervention. "More uptime means more physics" is a growing proverb for the industry aiming to optimize machine cycles and increase efficiency. One obstacle to these efforts is that existing tools are insufficient to meet the broad demands on controls, data visualization, and data analysis.

To meet these needs, we are building a web-based toolbox that features a generic virtual accelerator control room for the development of automated tuning algorithms and the analysis of large complex datasets. This package can be integrated with any accelerator control system to assist with complex data analysis tasks and development of control algorithms. This includes loading archive data and direct streaming of settings and readings from the control system to the toolbox. Our prototype interface is currently available from any web browser. Machine specific interfaces can be installed behind accelerator firewalls and accessed from anywhere on the controls network. Our toolbox bridges the gap between EPICS-based [1] control systems and in-house control systems by creating a common platform for analysis and simulation. By using a flexible software framework, we will have the ability to include machine learning libraries in the future. At this time, the computational power available to particle accelerator operations is increasing. The customer demands on the particle accelerator industry are expanding. Providing meaningful tools to the operators, scientists, and engineers will help particle accelerator facilities manage efficiency and ensure supply meets demand.

* This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Nuclear Physics under Award Number DE-SC0019682.
† jedelen@radiasot.net

A CLOUD-BASED DATA ANALYSIS TOOLBOX

Improved data analytics capabilities are essential for modern accelerator control rooms. Some facilities now incorporate jupyter notebooks into their daily operations with direct connections to machine parameters and data archiving tools. While this represents a significant step forward in accelerator operations, these modules can at times lack the intuitive feel of point-and-click analysis tools. As part of our framework we are developing a suite of analysis and visualization tools that allow users to upload, manipulate, and analyze data directly in an intuitive browser interface. Here we detail three primary features of our prototype toolbox and describe our plans for future work in this area.

Clustering

Clustering tools have been commonly used for many years, however their presence in accelerator analysis and controls has only recently seen wider adoption. scikit-learn [2] provides a simple gateway to a variety of clustering tools, however, users are required to build post-processing and cluster selection tools on their own. Our interface connects with four popular clustering algorithms and provides users with point-and-click operations on the dataset for cluster selection and further evaluation.

Figure 1 shows a sample dataset in two dimensions with results from running the clustering algorithm k-means. Users can select the clustering algorithms and modify the hyper parameters as needed. Users can click on different clusters and open them in a new plot which allows for further analysis to be performed. For example additional clustering, frequency analysis, or curve fitting.

Frequency Analysis

We have also implemented visualizations for numpy's frequency analysis tools [3]. Here users can easily perform a 1-D frequency analysis on imported datasets and identify the spectrum of a given input signal. Figure 2 shows an example of how the user interacts with the frequency analysis tools and visualizations.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUAPP01

HARDWARE-IN-THE-LOOP TESTING OF ACCELERATOR FIRMWARE*

C. Serrano†, M. Betz, L. Doolittle, S. Paiagua, V. K. Vytla, LBNL, Berkeley, USA

Abstract

Continuous Integration (CI) is widely used in industry, especially in the software world. Here we propose a combination of CI processes to run firmware and software tests both in simulation and on real hardware that can be well adapted to FPGA-based accelerator electronics designs. We have built a test rack with a variety of hardware platforms.

Relying on source code version control tools, when a developer submits a change to the code base, a multi-stage test pipeline is triggered. Unit tests are run automatically, bitstreams are generated for the various supported FPGA platforms and loaded onto the FPGAs in the rack, and tests are run on hardware. Reports are generated upon test completion and notifications are sent to the developers in case of failure.

OVERVIEW

FPGA-based accelerator instrumentation is based on the coherent design of firmware, controls software, hardware and communication links in between. All those layers are usually not static entities but continuously evolve to provide more features or improve performance. In addition to this coherence problem, FPGAs are becoming larger and more complex and, as software, firmware designs follow a fairly complex layered approach in themselves.

In this paper we describe some of the practices we have found useful at Berkeley based on common version control tools and CI setup. Adopting these processes was a rewarding experience and proved easier to deploy than it seemed at first. It has facilitated the task of developing complex, layered designs where these layers are tightly coupled and need to be tested in conjunction [1]. When one of us is working on a particular layer in the design and commits a change in version control, the CI setup automatically runs unit tests, builds FPGA bitstreams and runs fully integrated tests on hardware, verifying that the entire design works as expected as a whole.

Although, we have been using version control software and been writing several tests for our firmware and software designs, testing on the bench, or updating firmware on deployed electronics usually involved verifying many changes all at once. With the CI process in place, not only can we catch failures more often but the process is automated and if something goes wrong after committing a change the developer who made the breaking change is setup to be notified by email.

* This material is based upon work supported by the U.S. Department of Energy, Office of Science, under Contract No. DE-AC02-76SF00515.

† CSerrano@lbl.gov

MOTIVATION

While our group in particular handles specific technical challenges and integration issues [2], the general development processes can be extended to those in other fields of accelerator instrumentation. Those include:

- Growing complexity of multi-layered designs.
- Issues related to interconnection of design layers and overall coherence and integration.
- Collaborative development among organizations where different groups, sometimes in separate institutes, are responsible for layers within a common design.
- Lack of quality assurance and automated testing processes.

Continuous integration is a development practice used to prevent integration problems while modifying a small part of a larger design. Industry practices have in the past forced developers to run unit tests locally before committing changes to shared repositories. Today, this process has been widely adopted and some of this tedious testing and verification steps have been streamlined and automated.

Automated verification and continuous integration practices become particularly important when dealing with complex systems with many integrated parts, especially when multiple groups work together on a design or parts of those designs are shared among different projects.

The issues related to design complexity are not at all unique to accelerator instrumentation but are in fact common in electronics and software industries [3]. However, instrumentation and controls groups in accelerator facilities usually lack solid development process and quality control, let alone automated testing processes.

Deployment is also a very important step in accelerator electronics, usually carried our by software controls groups who have more established release and deployment processes. A usual example of integration difficulties encountered in accelerators is guaranteeing the coherence between FPGA firmware designs and their associated software low-level drivers and high-level applications. If an FPGA designer adds or removes a feature-or even renames a register that is expected by controls software-there needs to be a process for those parts of the design to stay in sync. CI processes help facilitate those steps by testing the entire system as a unit and by providing "artifacts" readily available to all parties. These artifacts can be FPGA bitstreams or software executables which are automatically generated by the CI tools every time a developer submits code changes. These artifacts can be set to be deployed automatically as well.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUAPP02

DEVELOPMENT OF THE MTCA.4 I/O CARDS FOR SPring-8 UPGRADE AND NEW 3 GeV LIGHT SOURCE

T. Fukui, N. Hosoda, RIKEN SPring-8 Center, Hyogo, Japan
M. Ishii, JASRI/SPring-8, Hyogo-ken, Japan
H. Maesaka, T. Ohshima, E. Iwai, RIKEN SPring-8 Center, Sayo-cho, Sayo-gun, Hyogo, Japan

Abstract

We will start a full energy injection from the SACLA to the SPring-8 from next year as a part of the SPring-8 upgrade. For this, we developed several I/O cards with the MTCA.4 form factor. One of the key issues is a timing synchronization between SACLA and SPring-8. We implemented required functions on the FPGA logic of a commercially available I/O card. We develop a module to distribute a trigger and clocks. We also developed cards used for the beam position monitor (BPM) and low-level RF system (LLRF). Those are included two types of cards. One is a 16-bit digitizer used for LLRF for the SPring-8 since 2018 march. We will use the card for the BPM with modified FPGA logic. Second is an implementation of functions with the pulsed RF signals processed on the FPGA logic of a commercially available card. These functions are used for the BPM of the beam transport line from the SACLA to SPring-8. The existing system is used 1 Hz beam repetition but we need more than 10 Hz to achieve an injection time less than 20 minutes to maximize user time. We will report the performance of the MTCA.4 cards, the upgrade plan of the SPring-8, and the construction of the 3 GeV Light Source.

INTRODUCTION

In the last twenty years, SPring-8 has been providing bright X-ray to users as a large-scale third-generation synchrotron radiation facility with the highest electron energy in the world. The SACLA project, which was aiming to provide an X-ray free electron laser to users, started in 2006 with a five-year construction schedule and it has been in operation for user experiments since 2012. We also built a beam transport line between the linear accelerator of SACLA and the storage ring of SPring-8 for an optional operation; a full-energy injection to the SPring-8 storage ring. An ultralow emittance electron beam delivered from SACLA should be compatible with the future upgraded SPring-8 facility. [1]

SACLA delivers pulsed X-ray laser beams whose pulse duration is as short as a few femtoseconds. The peak brilliance of SACLA is extremely high. The complementary use of the upgraded storage-ring light sources and pulsed X-ray laser is essential for opening new frontiers in science and technology. In SPring-8-II the dynamic aperture will be markedly narrower than that in the current SPring-8. We cannot use the existing injector system in SPring-8 without large-scale modification. In addition, because of the long injection interval during top-up operation it is necessary to keep the injector system in a standby condition, which will increase the operation cost. On the other hand, the linac of SACLA is always running for user experiments independently from SPring-8. Therefore, if the injection beam is delivered from SACLA, the operation cost will be minimized. To enable operation by SACLA users experiments and beam injection to SPring-8-II in parallel, it is necessary to control the beam energy and peak current on a pulse by pulse.

Last few years another project was started to build new 3 GeV Light Source at Sendai located in the north-east part of Japan. The beam current and emittance of the storage ring are designed to be 400 mA and 1 nm.rad. We designed equipment of the 3 GeV Light Source with those developed by the SPring-8-II project because the parameter of the linac and the storage ring is very similar to the SACLA and the SPring-8-II. The 3 GeV Light Source will use a top-up operation mode with a low emittance electron beam delivered from a linac. The linac consists of a thermionic electron gun, beam, prebuncher cavity (238 MHz), booster cavity (476 MHz), S-band (2856 MHz) linacs, and main C-band linacs.

Prior to build the 3 GeV Light Source we will build a small linac which will deliver an electron beam to the New SUBARU storage ring located near the transport line from the SACLA to the SPring-8. Figure 1 shows the SACLA and the SPring-8 accelerator complex and New SUBARU. At present the injector linac of the SPring-8 delivers the electron beam to the SPring-8 and the New SUBARU in parallel. We are planning to shutdown the injector linac after confirming the steady operation of the injection from the SACLA because of reduction of operation cost. For this reason, the New SUBARU will need a new injector linac dedicated for the New SUBARU. It is beneficial to build a new linac for the New SUBARU because it becomes a prototype for the injector of the 3 GeV Light Source. The linac consists of an electron gun, beam, prebuncher cavity (238 MHz), booster cavity (476 MHz), S-band (2856 MHz) linacs, and main C-band linacs.

These projects will use MTCA.4 as the standard form factor of an accelerator control system. And it will use for the low-level RF (LLRF) system, timing system, beam position monitors and beam current monitors.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUAPP03

LOW-COST MODULAR PLATFORM FOR CUSTOM ELECTRONICS IN RADIATION-EXPOSED AND RADIATION-FREE AREAS AT CERN

G. Daniluk*, C. Gentsos, E. Gousiou, L. Patnaik, M. Rizzi
CERN, Geneva, Switzerland

Abstract

The CERN control system is comprised of multiple layers of hardware and software that extend from the hardware deployed close to the machine, up to the software running on computers that operators use for control and monitoring. A new centrally supported service is being developed for the layers closest to the accelerator: Distributed I/O and the Fieldbus, targeting both radiation-free and radiation-exposed areas. A key aspect of this project is the selection of industrial standards for the layers, which are currently dominated by custom, in-house designed solutions. Regarding the Distributed I/O layer, this paper describes how we are adapting an industrial crate standard to be suitable as the low-cost modular hardware platform for remote analog and digital I/O applications in radiation-exposed as well as radiation-free areas. We are designing a low cost 3U chassis with a standardized backplane accompanied by a radiation tolerant, switched-mode power supply and an FPGA-based System Board that houses the Fieldbus communication interface.

INTRODUCTION

In the years 2024–25, CERN's Large Hadron Collider (LHC) will undergo a major upgrade [1]. The High Luminosity LHC (HL-LHC) will provide instantaneous luminosities a factor of five larger than the LHC nominal value. To achieve the physics targets several new technologies will be introduced to the accelerator and many systems will be renovated to increase the overall machine availability by 20 %.

The HL-LHC will place challenging demands on data acquisition to/from the accelerator components which need to be controlled and diagnosed, such as the new Nb_3Sn magnets. The need for larger amounts of diagnostics information will result in a requirement for more throughput in the lower layers of the control system and will therefore affect the electronics in this tier and the communication links used to send the information up the controls stack. The current custom electronics–based controls architecture comprises the following three hardware layers (Fig. 1):

- **Front-end Tier:** a powerful computer in various form factors (VME, PICMG 1.3, MTCA.4, *etc.*) that can host a variety of reusable electronic cards to control accelerator components by sending and receiving data and carrying out calculations in real-time.

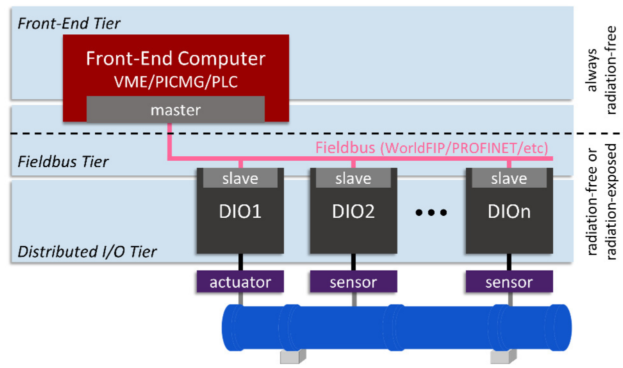


Figure 1: Lowest layers of CERN's accelerator control system.

- **Fieldbus Tier:** a networking solution that ensures communication between the master in the front-end layer and a set of slaves in the distributed I/O tier
- **Distributed I/O Tier:** electronic modules that interface directly with an accelerator equipment in radiation-exposed or radiation-free areas, controlled by the master in the front-end tier over the fieldbus. These are usually FPGA–based boards sampling digital and analog inputs, driving outputs and performing various safety-critical operations.

Historically, in the front-end and fieldbus layers there has been a lot of standardization, sharing and reuse of design effort between equipment groups. However, in the distributed I/O layer, the lack of a generic, modular solution with simple and robust inter-module communication resulted in many custom-made developments in different form factors around CERN. With the Distributed I/O Tier (DI/OT) project we are designing a low-cost, reusable, modular hardware platform that can be easily customised to serve the requirements of various accelerator subsystems in both radiation-exposed and radiation-free areas. Introducing a standardisation in the distributed I/O layer of the control system automatically allows all users to benefit from modules designed by other groups or institutes. The main difference from the front-end tier modular electronics is less complexity, a simpler interface between boards hosted within a single crate and the provision of a radiation-tolerant variant of the hardware kit.

DISTRIBUTED I/O TIER HARDWARE KIT

A fundamental requirement for the DI/OT hardware platform, is basing it on existing industrial standards. The study

* grzegorz.daniluk@cern.ch

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUAPP04

EXTENDING THE LIFE OF THE VME INFRASTRUCTURE AT BNL*

W. Pekrul†, C. Theisen, Brookhaven National Laboratory, Upton, USA

Abstract

A large installation of VME controllers have been used to control and monitor the RHIC Accelerator complex at BNL. As this equipment ages a number of upgrade options are being pursued. This paper describes an FPGA based VME controller board development being undertaken to provide a upgrade path for control applications that reuses existing racks and power supplies and a catalogue of custom application boards. This board is based on a Xilinx Zynq that includes an ARM-9 and a large FPGA fabric. The board includes DRAM, SPI-Flash, ethernet, SD card, USB, SFP, FMC and an Artix FPGA to support the VME bus protocol. The first application of a magnet quench detector will also be described.

INTRODUCTION

The RHIC complex is a long chain of sources and accelerators at BNL. Its control system includes a large installation of hardware built to the VME (Versa Module Eurocard) bus standard. The standard was adopted in the 1980s and has gone through several iterations. As the standard ages and the availability of new boards drops the RHIC complex is still in need of replacements and upgrades and controls for new applications. The ZVC (Zynq VME Controller) was developed to provide an answer to some of these issues. In addition it takes advantage of the FMC (FPGA Mezzanine Card) manufacturer catalogues, provides an architecture that avoids operating over the slow VME backplane and provides the potential for fast deterministic response time.

FEATURES

- Xilinx Zynq 7000 XC7Z045-2FFG900I AP SoC
- XilinxArtixXC7A50T-2FGG484I
- VME32 Bus Interface
- 1-VITA 57.1 FMC HPC connector
- 1-VITA 57.1 FMC LPC connector
- 1 GB DDR3 DRAM memory (four 256 Mb x 8 devices) connected to Zynq PS (processing system)
- 2-128 Mb Quad-SPI (QSPI) flash memory chips connected to Zynq PS (32 MB)
- 128 Mb Quad-SPI (QSPI) flash memory chip connected to Artix (16 MB).
- IIC EEPROMs connected to Zynq and Artix (32 Kb) Stores MAC address, serial numbers, etc
- USB 2.0 ULPI transceiver with USB A Connector
- microSD (Micro Secure Digital) Card Carrier
- JTAG interfaces to Zynq and Artix via 14 pin headers
- Clock sources:
  - Fixed 33.33 MHz LVCMOS Zynq PS oscillator
  - Fixed 200 MHz LVDS oscillator to Zynq and Artix
  - Fixed 100 MHz LVDS oscillator to Zynq and Artix Transceivers
  - I2C 10 to 800 MHz programmable LVDS Zynq PL oscillator
  - External PL Clock on SMC connector
  - External Transceiver Clock on SMC connector
  - I2C 10 to 800 MHz programmable LVDS to SFP
- Ethernet PHY RGMII Interface with RJ-45 Connector on Front Panel
- RS232 Interfaces via USB A Connectors to Zynq and Artix on Front Panel
- Small Form-Factor Plugable Plus (SFP+) Connector on Front Panel
- GTX (Gunning Transceiver) Support:
  - FMC LPC connector (one GTX transceiver)
  - FMC HPC connector (eight GTX transceivers)
  - SFP connector (one GTX transceiver)
- I2C Bus Multiplexed to:
  - 1-to-16 TCA6416APWR port expander
  - M24C08 EEPROM
  - RTC-8564JE real time clock
  - FMC HPC connector
  - FMC LPC connector
  - SFP+ connector
  - Programmable Clocks
- Eight Status LEDs (front panel):
  - Power Good(s)
  - FPGA DONE(s)
  - One Zynq user LED
- Three “Blue Hose” Differential Link Connections
- 16-3.3V GPIOs on unused VME pins.
- Six additional power/ground pairs on spare VME pins to allow maximum utilization of XC7Z045. (45W x 2)
- AP SoC PS Reset Push button on front panel.
- Configuration options:
  - Dual Quad-SPI flash memory
  - 14-pin PL JTAG header
  - Secure Digital (SD) micro card
- On-board temperature, voltage and current monitoring
- Sequenced power up/down.

* Work supported by Brookhaven Science Associates, LLC under contract no. DE-SC0012704 within the U.S. Department of Energy.
† wpekrul@bnl.gov

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUAPP05

PandABlocks - A FLEXIBLE FRAMEWORK FOR Zynq7000-BASED SoC CONFIGURATION

G. B. Christian*, M. Abbott, T. Cobb, C. Colborne, A. M. Cousins, P. Garrick, T. Trafford,
Diamond Light Source Ltd, Oxfordshire, UK
Y. M. Abiven, J. Bisou, F. Langlois, S. Minolli, G. Renaud, G. Thibaux, S. Zhang,
SOLEIL, Gif-sur-Yvette, France
I. S. Uzun, STFC, Oxfordshire, UK

Abstract

The PandABlocks framework comprises the FPGA logic, TCP server, webserver, boot sources and root filesystem, developed for the PandABox platform by Diamond Light Source and Synchrotron Soleil, for advanced beamline scanning applications. The PandABox platform uses a PicoZed System-on-Module, comprising a Zynq-7030 SoC, coupled to a carrier board containing removable position encoder modules, as well as various input and outputs. An FMC connector provides access to ADC/DACs or additional I/O, and gigabit transceivers on the Zynq allow communication with other systems via SFP modules. Specific functions and hardware resources are represented by functional blocks, which are run-time configurable and re-wireable courtesy of multiplexed data and control buses shared between all blocks. Recent changes to the PandABlocks framework are discussed which allow the auto-generation of the FPGA code and tcl automation scripts, using Python and the jinja2 templating engine, for any combination of functional blocks and SFP/FMC modules. The framework can target hardware platforms other than PandABox and could be deployed for other Zynq-based applications requiring on-the-fly reconfigurable logic.

INTRODUCTION

Many x-ray beamlines conduct experiments which involve moving a motor and synchronously acquiring data from a detector. When detector speeds are slow, it is sufficient to step scan the motor, allowing it to move and settle between each detector frame. For many modern beamlines, with detector frame rates in the hundreds or thousands of hertz range, continuous scanning is required, where the motor constantly moves through a trajectory while the detector is acquiring. This technique reduces detector dead-time, but requires precise synchronisation between motion control systems and detectors. To achieve this, the PandABox (Position and Acquisition Box) platform was developed by Diamond Light Source (DLS) and Synchrotron Soleil as a FPGA-based solution for motion encoder and detector trigger processing [1].

The PandA collaboration began in 2015 to develop a common platform to replace their previous generation of in-house systems for beamline synchronisation (Fig. 1): Zebra at DLS; and SPIETBOX at SOLEIL. Both systems are based around a Xilinx FPGA; a Spartan-6 in the case of Zebra, and Spartan-3 in the case SPIETBOX. The goals of PandABox were to develop a flexible system to address the increasingly demanding requirements for beamline scanning, and to overcome concerns with component obsolescence and technical limitations of Zebra and SPIETBOX. A secondary goal was to share resources between DLS and SOLEIL, with SOLEIL taking primary responsibility for the electronic and mechanical design and DLS developing the FPGA firmware, software and user interface. The hardware design for PandABox is freely available on the Open Hardware Repository (OWHR) [2], and is commercially available from Quantum Detectors [3].

Figure 1: The PandABox collaboration between Diamond Light Source and SOLEIL.

THE PandABox HARDWARE

The PandABox hardware is shown in Fig. 2 [4, 5]. The complete assembly is designed to fit within a 19 inch, 1U rack. The system comprises a custom carrier board and encoder daughter cards, and an off-the-shelf PicoZed Z7030 System-on-module (SoM), produced by Avnet [6]. The PicoZed SoM contains a Xilinx Zynq-7030 System-on-Chip (SoC), as well as various on-board peripherals such as DDR3 and QSPI flash memory, ethernet and USB interface chips. The Zynq-7030 comprises Kintex-equivalent FPGA programmable logic (PL) fabric, alongside an embedded dual-core ARM Cortex A9 processor system (PS). Signals from the PS, PL, and peripherals, are brought out on micro-headers on the underside of the SoM PCB, where they connect to the carrier board. A Xilinx Spartan-6 FPGA is also fitted to the carrier to provide addition I/O and monitoring capabilities, due to a lack of sufficient pins on the Zynq SoC [1]. The Spartan-6, referred to as the 'Slow

* glenn.christian@diamond.ac.uk

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUBPL01

AUTOMATIC WEB APPLICATION GENERATION FROM AN IRRADIATION EXPERIMENT DATA MANAGEMENT ONTOLOGY (IEDM)*

B. Gkotse[†,1,2], P. Jouvelot[1], F. Ravotti[2]
[1]CERN, Geneva, Switzerland
[2]MINES ParisTech, PSL University, Paris, France

Abstract

Detectors and electronic components in High-Energy Physics experiments are nowadays often exposed to harsh radiation environments. Thus, to insure reliable operation over time, their radiation tolerance must be assessed beforehand through dedicated testing experiments in irradiation facilities. To prevent data loss and perform accurate experiments, these facilities need to rely upon a proper data management system.

In prior work, we provided a formal description of the key concepts involved in the data management of irradiation experiments using an ontology (IEDM). In this work, we show how this formalisation effort has a practical by-product via the introduction of an ontology-based methodology for the automatic generation of web applications, using IEDM as a use case. Moreover, we also compare this IEDM-generated web application to the IRRAD Data Manager (IDM), the manually developed web application used for the data handling of the CERN Proton Irradiation facility (IRRAD). Our approach should allow irradiation facility teams to gain access to state-of-the-art data management tools without incurring significant software development effort.

INTRODUCTION

Ontologies have been used in Artificial Intelligence (AI) for years for various purposes such as knowledge formalisation, interoperability, complex querying and inference [1]. Nowadays, some companies choose to base their information systems on ontologies [2] and knowledge graphs [3], the descendants of ontologies, in order to allow for better data integration and communication. Ontologies have been shown to be suitable for domain-specific knowledge formalisation, and are broadly used in various domains such as biomedicine [4], bioinformatics [5] and law [6]. In addition to formalisation, ontologies can have many practical applications, which were not explored deeply so far. We focus here on one particular application, namely the management of ontology-related data.

Data management is an important issue in several scientific domains [7], and in the High-Energy Physics (HEP) as well. This is true when running physics experiments at the CERN Large Hadron Collider (LHC) or similar accelerator infrastructures, but also at earlier stages, during their development. Detectors and electronic components used in HEP experiments are often exposed to harsh radiation environments. Thus, to insure reliable operations over time, their radiation hardness must be assessed beforehand through dedicated testing experiments in irradiation facilities. To prevent data loss and perform accurate experiments, these facilities need also to rely upon a proper data management system.

In our previous work [8], we introduced the domain ontology for Irradiation Experiment Data Management (IEDM), which formalises the concepts involved in irradiation testing experiments. In this paper, we introduce both a new methodology for generating automatically web applications from domain ontologies and our ontology-driven, data management web application generator (GenAppi). In this way, we enable non-computer scientists to easily build and deploy such applications. We use IEDM as a representative example to illustrate our approach.

An interesting by-product of our proposal is that GenAppi-generated web applications end up being actually enriched beyond their initial intended goal by the presence of a new underlying ontology, the Ontology-based Web Application Ontology (OWAO), and other web semantic technologies. More specifically, these generated web applications can rely upon well-established open standards, while user data are not only stored in private databases but also as interconnected knowledge graphs. This opens opportunities for both developers and domain experts, for instance in terms of inference or coherence checking.

The structure of the paper is as follows. In the second section, we provide some background information about ontologies, describe relevant user-interface ontologies and discuss related work about ontology-based user interface generation. In the third section, our new methodology is described, focusing on the OWAO ontology designed to enable the description of the concepts and axioms integrated in our web application generator (GenAppi). In the fourth section, we present the IEDM ontology, used as an example ontology to demonstrate the functionalities and User Interface (UI) features of automatically generated web applications. In the fifth section, we compare the IEDM-derived web application to the IRRAD Data Manager (IDM), a custom-made web application currently used in the CERN proton irradiation facility (IRRAD)[1] [9]. Finally, in the sixth section, we conclude our work and present our ideas regarding possible

* This work has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement no. 654168 (AIDA-2020).

† Blerina.Gkotse@cern.ch

1 http://cern.ch/ps-irrad

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUBPL02

ENABLING OPEN SCIENCE FOR PHOTON AND NEUTRON SOURCES*

A. Götz, J. Bodera Sempere, A. Campbell, A. de Maria, M. del Rio,
R. Dimper, J. Kieffer, A.Solé, T. Vincent, ESRF, Grenoble, France
S. Caunt, J. Hall, J. F. Perrin, ILL, Grenoble, France
N. Carboni, A.Hafner, R. Pugliese, CERIC-ERIC, Trieste, Italy
M. Bertelsen, T. H. Rod, T. S. Richter, J. Taylor, ESS, Copenhagen, Denmark
J. C. E, H. Fangohr, C. Fortmann-Grote, T. Kluyver, R. Rosca, EuXFEL, Schenefeld, Germany
F. Gliksohn, L. Schrettner, ELI-DC, Brussels, Belgium

Abstract

Photon and Neutron sources are producing more and more petabytes of scientific data each year. At the same time scientific publishing is evolving to make scientific data part of publications. The Photon and Neutron Open Science Cloud (PaNOSC project is an EU financed project to provide scientific data management for enabling Open Science. Data will be managed according to the FAIR principles. This means data will be curated and made available under an Open Data policy, findable, interoperable and reusable. This paper will describe how the European photon and neutron sources on the ESFRI roadmap envision PaNOSC as part of the European Open Science Cloud. The paper will present the objectives of the project on the issues of data policy, metadata, data curation, long term archiving and data sharing in the context of the latest developments in these areas.

INTRODUCTION

Photon and neutron sources are hitting a data analysis wall with the huge increase in data volumes, new techniques and new user communities. Users are limited by the difficulty in exporting huge data from the source to their home labs and by the lack of easy access to data analysis programs and services. At the same time science is becoming more open by sharing the data, methods and publications - the so-called **Open Science** movement [1]. Making the data used in publications easily available enables others to reproduce the analysis and findings. This has been one of motivations for neutron and photon sources to adopt open data policies. Another motivation has been the need to alleviate the data analysis bottleneck by implementing modern data management so that data analysis services can be built on top of data catalogues. Lack of adequate data management limits the data services which can be be offered to users.

PaNOSC has been financed by the H2020 INFRAEOSC-04 call as part of the EOSC project to bring FAIR data to ESFRI Photon and Neutron sources and to share the outcomes with all national photon and neutron sources. The PaNOSC partners are ESRF project coordinator (Grenoble, France), ILL (Grenoble, France), EuXFEL (Schenefeld, Germany), ESS (Lund, Sweden), CERIC-ERIC (Trieste, Italy), ELI-DC (Bruxelle, Belgium), EGI (Amsterdam, Netherlands). In addition to the internal partners, the following external partners will assist PaNOSC in its missions - GÉANT (Amsterdam, Netherlands), DESY (Hamburg, Germany), CESNET (Prague, Czech Republic), STFC (Hartwell, United Kingdom).

* This project has received funding from the European Union's Horizon 2020 Research and Innovation programme under Grant Agreement No 823852.

DATA POLICIES 2.0

In the past (10 years ago) scientific data was produced without any clear policy on the ownership, life cycle or license of the data. Since 2010 it has become standard practice to define a clear data policy for all scientific data produced in research institutes. This is partly due to the large quantities of data produced but also in order to manage the life cycle of the data better. In order to invest in and manage data for a longer period the ownership and license of the data need to be specified.

All of the PaNOSC partners have adopted or are in the process of adopting (ELI-DC) a data policy. The data policies are all based on the PaNdata-Europe data policy. Adopting a data policy is only the first step. Implementing it is much more work. See [2] for a detailed description of how one of the partners has addressed the challenge of implementing a data policy.

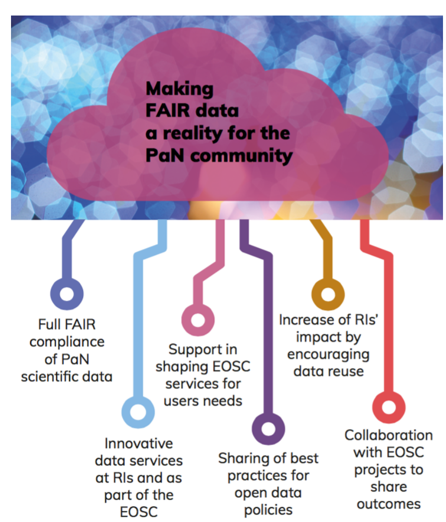


Figure 1: PaNOSC FAIR objectives

One of the main objectives of PaNOSC (see Fig. 1) is to update the PaNdata-Europe data policy framework to include the FAIR (Findable, Accessible, Interoperable, Reusable)

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUBPL03

EXPERIMENTAL DATA TRANSFER SYSTEM BENTEN AT SPring-8

T. Matsumoto†, S. Yokota, T. Matsushita, K. Nakada, Y. Hiraoka, M. Kodera, Y. Furukawa, A. Yamashita, Japan Synchrotron Radiation Research Institute Hyogo, Japan

Abstract

Recently, there have been high demands of open data to promote data science such as material informatics. At SPring-8, we have been operating the data transfer system (SP8DR) for open data of the X-ray Absorption Fine Structure (XAFS) standard sample since 2013. However, it proved to be difficult to use during various experiments at SPring-8. To overcome these problems, we recently developed the BEamline ExperimeNTal stations oriENted data transfer system (BENTEN) for generic use in synchrotron radiation experiments. BENTEN provides an easy-to-use and unified interface with REST API for data access from both inside and outside SPring-8. BENTEN implements user authentication and can also provide restricted data access among the members of the experiment. Data registration is performed with metadata files that describe data such as experimental conditions and samples. To manage multiple metadata in the experiments, Elasticsearch was used as a metadata store. Data can be accessed flexibly via a full-text search. We launched the BENTEN system in March 2019 and provided open access to the XAFS standard sample and restricted data access in user experiments at BL14B2. We plan to use BENTEN on public experimental stations to promote data science as well as other experimental data.

INTRODUCTION

SPring-8 is a third-generation synchrotron radiation facility in Japan. Brilliant synchrotron radiation X-rays are produced from 8 GeV electron beams and are used for various experimental measurements for scientific research and industrial applications. Experimental stations are equipped for each of the 57 beamlines and a large amount of data is produced from these experiments.

Recently, there have been notable advances in data science, such as materials informatics. As data science produces knowledge from data, it is highly desirable that data be opened up to experimental data. Open data from the point of view of social responsibility are also desired, especially in the case of data obtained through public funds. In the case of other synchrotron radiation facilities, such as ESRF, the embargo period for the publication of experimental data is set at 3 years as data policy [1].

At SPring-8, we have been providing open data access for XAFS standard sample since 2013 with the experimental data transfer system called as SP8DR [2]. The amount of XAFS data is approximately 800, which corresponds to the second place in the world [3]. These XAFS data were utilized as reference for the measurements in user experiments.

There are also demands for restricted data access for remote experiments and proxy measurements at SPring-8. To satisfy these demands, SP8DR implements authentication with the SPring-8/SACLA user information account (SPring-8 ID) and provides authorized data access. Although SP8DR requires authentication, everyone can access to the open data because the SPring-8 ID account registration is open to public users.

However, there were several problems with SP8DR. For example, SP8DR required building a system for each beamline. Therefore, it was difficult to extend the system into other beamlines. It was also not easy-to-use nor flexible to manage metadata in various experiments. To overcome these problems, we recently developed the experiment data transfer system, BENTEN. In this paper, we report about BENTEN and its operation at SPring-8.

EXPERIMENTAL DATA TRANSFER SYSTEM BENTEN

We developed BENTEN as a generic software for experimental data transfer to provide data access through the Internet for synchrotron radiation experiments with an easy-to-use interface. As BENTEN covers open data access, it is recommended to follow the FAIR principle [4]. FAIR stands for Findable, Interoperable, Accessible and Reusable. Therefore, it is not enough to open data as it is. Data must be regulated for human comprehension by attaching metadata in the experiments, such as samples and measurement parameters. Machine readability is also important because the data will be used with artificial intelligence (AI), such as machine learning.

To operate the remote data access system in our facilities, we also need to have flexible data management. Therefore, we required BENTEN to comply with the following conditions:

- Easy-to-use for data transfer functions such as authentication, metadata creation, data registration, and data access.
- Metadata of raw data and derived data can be easily described and flexibly managed for measurement data in several experiments.
- Flexible data search can be performed with registered metadata items.
- The data cycle, such as creation, access to and deletion of open/closed, can be easily controlled.
- To refer dataset, each dataset is linked with persistent ID (PID), and the contact person or organization is associated.
- The data reliability in data transferring is guaranteed by the checksum verification.

† matumot@spring8.or.jp

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUBPL04

PUBLIC CLOUD-BASED REMOTE ACCESS INFRASTRUCTURE FOR NEUTRON SCATTERING EXPERIMENTS AT MLF, J-PARC

K. Moriyama, Comprehenshive Research Organization for Science and Society (CROSS), Tokai, Ibaraki, Japan
T. Nakatani, Japan Atomic Energy Agency (JAEA), Tokai, Ibaraki, Japan

Abstract

An infrastructure for remote access for supporting research workflow is essential for neutron scattering user facilities such as J-PARC MLF. Because the experimental period spans day and night, service monitoring the measurement status from outside the facility is required. Additionally, convenient way to bring a large amount of data back to user's home institution and to analyse it after experiments is required. To meet these requirements, we are developing a remote access infrastructure as a front-end for facility users based on public clouds. Recently, public clouds such as Amazon AWS and Google Cloud, have been rapidly developed, so that development and operation schemes of computer systems have changed dramatically. Various architectures provided by public clouds enable advanced systems to develop quickly and effectively. Our cloud-based infrastructure comprises services for experimental monitoring, data distribution and data analysis, using architectures such as object storage, event-driven serverless computing, and virtual desktop infrastructure (VDI) based on a microservice approach for application implementation. Facility users can access this infrastructure using just a web browser and VDI client. This paper reports the current status of this remote access infrastructure.

INTRODUCTION

The Materials and Life Science Experimental Facility (MLF) at the Japan Proton Accelerator Research Complex (J-PARC) is a neutron scattering experiment user facility with one of the world's highest intensity pulsed neutron beams in operation since 2008. Currently, the 21 installed neutron instruments are used by domestic and international users, in various research fields to conduct experiments.

Many of the instruments at MLF have introduced IROHA2 [1], which is a web-based integrated instrument control framework that controls data acquisition and sample environmental devices. IROHA2 is able to perform automatic measurement changing measurement conditions. Using this function, long-running measurement over periods of several days can be performed. Therefore, the progress of measurement can be monitored remotely via the web. However, due to security issues and system capabilities, currently, only instrument staff are allowed to monitor the measurement status in this way.

With intense neutron beams, high-precision and -resolution position-sensitive detectors are used with advanced event recording methods for data acquisition; experimental data is generated at very high rates [2]. The total amount of data generated in experiment ranges from hundreds of MB to several TB per experiment.

These data are analysed using Linux-based analysis software. After each experiment, most beamline users would normally take their data back to their home institutions in the portable storage such as hard disks. Since many facility users are unfamiliar with the Linux environment, virtual machines with analysis software installed are distributed to them.

Given this facility usage, to support research workflows of facility users, the remote access infrastructure requirements are:

- The ability to remotely monitor the progress of long-running measurements over the experiments
- To remove the necessity for users to physically take large amounts of raw data back to their home institutions, by allowing data analysis and results acquisition to be performed by one-stop, remote access.
- To allow multiple facility users quick and easy remote access from both inside and outside the country.

To satisfy these requirements, we built a remote-access infrastructure linked to the on-site systems at MLF using the Amazon Web Services (AWS) [2], one of the main public clouds. Figure 1 shows an overview of our remote access.

WHY USE A PUBLIC CLOUD?

The reality of system and infrastructure development has various limitations such as budget, human resources, and security. Given these limitations, using a public cloud as a development platform has the following advantages compared to conventional on-site systems and private clouds:

- No hardware management or installation costs are required.
- System development and operations can start small and can be scaled as need. This can optimize resource usage and thus, operating costs as well.
- Advanced services and functions provided via public clouds enable efficient system development; in particular, a proactive use of advanced AI-related services can be expected.
- The security of the service infrastructure is ensured at a higher level than that of on-premises systems.
- A managed service that simplifies system management and operation.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUBPL05

RecSyncETCD: A FAULT-TOLERANT SERVICE FOR EPICS PV CONFIGURATION DATA*

T. Ashwarya, E.T. Berryman, M.G. Konrad
Facility for Rare Isotope Beams, Michigan State University, East Lansing, USA

Abstract

Record Synchronizer (RecSync) is comprised of a client module called RecCaster and a server module called RecCeiver. Together they work to make PV (or record) metadata residing in the Input-Output Controller's (IOC) database available to client applications like ChannelFinder. Currently, the server module RecCeiver is a custom-built Python application that cannot be run as a cluster, hence it does not provide fault-tolerance. Further, the existing RecSync does not implement any authentication or security feature that controls the access to reads and writes to only verified clients and servers. In this paper, we explore an alternative to the current RecCeiver called ETCD which is an open-source off-the-shelf distributed key-value storage known for its high-availability storage and retrieval abilities. It provides a role-based authentication feature along with CA certificates-based and TLS-based security feature to make client-server communication encrypted and verified. It also provides useful features like conditional atomic transaction operations, live-watching of keys in its storage and ability to view historical changes to a key.

RECORD SYNCHRONIZER

The existing RecSync [1] consists of two modules: RecCaster which is a client application that runs as part of an IOC and RecCeiver which is a stand-alone server application written in Python using the Twisted networking library. Their aim is to make the metadata related to a PV being hosted on an IOC available to clients like ChannelFinder [2]. The PV metadata that is sent from RecCaster to RecCeiver consists of EPICS base version, a whitelisted set of environment variables, name & type of all records and any info tags associated with the records. ChannelFinder further provides RESTful APIs that various other applications use in order to read the PV metadata.

Theory of Operation

RecCaster is a client application written in C and works as an EPICS support module that spawns a thread on start-up to upload all the IOC records to a server. After spawning its thread, it waits for an announcement from RecCeiver. Once it discovers a live and ready RecCeiver, it exchanges handshake messages with the server. After the exchange of initial handshaking, greeting messages are sent between the client and server and then the uploading of the records to the server begins. Once the records are uploaded successfully to the RecCeiver, client and server exchange periodic heartbeat messages indefinitely to signal they are alive. In case the client dies and does not respond to the server, the server closes its connection to the client.

RecCeiver on a successful upload of data from RecCeiver pushes all the data to ChannelFinder. In case of a disconnected IOC, it signals to ChannelFinder to mark the PVs from a disconnected IOC as inactive. RecCeiver can be additionally configured to write the data it receives from RecCaster to a SQL database or print it to screen or logs.

RecCeiver is a standalone application that cannot be run as a distributed service on multiple nodes in a cluster. This makes RecCeiver a possible single point of failure if the node running the service goes down or gets disconnected from network. At this point, until the RecCeiver service is restored, the data available in clients like ChannelFinder will be stale and will not accurately reflect an IOC's state. Also, the existing RecSync does not provide any authentication feature that restricts the transfer of the IOC record metadata to only verified servers and clients. These shortcomings can be overcome using ETCD as a replacement for the existing Python-based RecCeiver.

ETCD

In this section, we talk about the data model, operations and useful features of the proposed alternative to the existing RecCeiver i.e. ETCD [3]. ETCD is a distributed, open-source key-value storage providing full replication of the key-value store on a cluster of servers. It is commonly used for distributed system coordination and metadata storage like Zookeeper and Consul which are two other popular alternatives. ETCD uses Raft [4] to perform cluster operations like election of the cluster leader and getting a majority quorum before the data is committed and written to the disk. Raft protocol is a distributed consensus protocol run on each member of a cluster to maintain a replicated state machine. Raft provides the leader election algorithm to elect a leader node from among the cluster nodes. A cluster-wide log called replicated log is used to keep the same state among cluster members. Leader node is responsible for writing the data to the replicated log and distributing it to follower nodes to maintain an in-sync state within the cluster. In case the leader node dies or gets disconnected, a re-election is held to choose a new leader from the remaining nodes.

Data Model

From a logical perspective, ETCD's data store is a flat-binary key space where byte string keys are lexically sorted indices. The entire key store have multiple revisions that monotonically increment over the cluster's lifetime and older revisions of a key are available for fetch and read. From a physical point of view, ETCD's data store is a persistent b+ tree that is ordered by key in lexical byte order

* Work supported by the U.S. Department of Energy Office of Science under Cooperative Agreement DE-SC0000661

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-TUBPL06

ENERGY CONSUMPTION MONITORING WITH GRAPH DATABASES AND SERVICE ORIENTED ARCHITECTURE

A. Kiourkos†, S. Infante, K. Seintaridis, CERN, Geneva, Switzerland

Abstract

CERN is not only the biggest particle physics laboratory in the world but also a major electricity consumer. In 2018 alone, CERN consumed 1.25 TWh, equivalent to 1/3 the consumption of the canton of Geneva. Reliable monitoring of this consumption is crucial, not only for obvious operational reasons but also for raising the awareness of users regarding their energy utilization. This monitoring is currently done via a web based system, developed internally at CERN that is quite popular within the community. In order to accommodate the increasing requirements, a migration is underway that utilizes the latest technologies for data modelling and processing. The architecture of the new energy monitoring system with an emphasis on the data modelling, versioning and the use of graphs to store and process the model of the electrical network for the energy calculations is presented. The algorithms that are used are also presented and a comparison with the existing system is performed in order to demonstrate the performance improvements and flexibility of the new approach. The system embraces the Service Oriented Architecture principles and it is illustrated how these have been applied in its design. The different modules and possibilities are also presented with an analysis of their strengths, weaknesses, and integration within the CERN infrastructure.

MOTIVATION

Energy Management

Energy Management has become an essential element of operations management and allows the users to plan and make decisions based on the historical data about their energy consumption [1]. The management of energy in the industry and facilities like CERN is very context specific, as it largely depends on the process. This implies that energy management solutions from other industries cannot be easily copied. It is therefore important that users are able to get accurate, reliable and easily accessible information about the energy across the site and the different accelerator installations.

Current Solution & Data Flow

To accommodate the energy management needs, a web based application was developed at CERN (WebEnergy) more than five years ago, Fig. 1. This application extracts data archived by the SCADA system that is responsible for the monitoring of the electrical network. These are energy consumption measurements that are provided by the protection relays (IEDs) installed at the high voltage level of the electrical network. These measurements are then collected by the SCADA and subsequently archived to an internal long term data storage system.

Using APIs provided by the archiving system, WebEnergy pulls the measurement data and combines it with the electrical network model. The model has been defined and is maintained by the system administrator within the system. The application uses this combined data to calculate the energy consumption at the different levels of the network and for different consumers. The calculations occur daily with help of a scheduled task and with the system containing data up to the previous day.

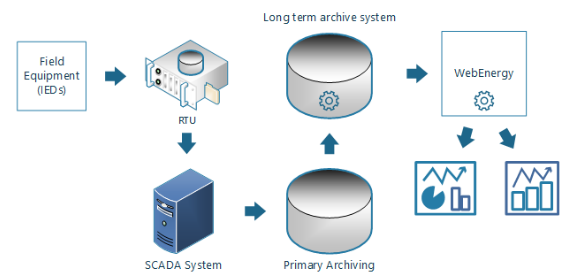


Figure 1: WebEnergy dataflow.

The users can access the previously calculated data via various dashboards where there are categorised and visualized in various charts.

Weaknesses

The application is used daily by numerous people at CERN. The results are quite accurate and have been validated over time against the energy consumption bills generated by the electricity suppliers.

Despite its success though, there are still opportunities for further enhancements and a number of additional features that will boost performance and add value for the users of the application and the organization.

The major area of improvement of the current application is the data model. Although simple to understand it is purely hierarchical, Fig. 2, allowing single parent-child relationships and most importantly: it is missing the notion of time.

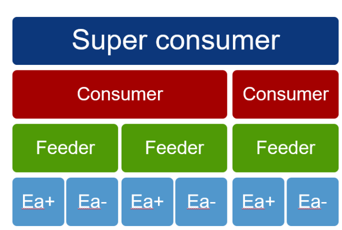


Figure 2: WebEnergy data model.

The electrical network in a facility like CERN changes continuously in order to accommodate the emerging needs. Storing only the energy consumption over time is not enough, the state of the model at any point in time is required in order to make meaningful comparisons. Because

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUBPR01

THE DISTRIBUTED OSCILLOSCOPE: A LARGE-SCALE FULLY SYNCHRONISED DATA ACQUISITION SYSTEM OVER WHITE RABBIT

D. Lampridis*, T. Gingold, M. Malczak[1], F. Vaga,
T. Włostowski, A. A. Wujek, CERN, Geneva, Switzerland
[1]also at Warsaw University of Technology, Warsaw, Poland

Abstract

A common need in large scientific experiments is the ability to monitor by means of simultaneous data acquisition across the whole installation. Data is acquired as a result of triggers which may come either from external sources, or from internal triggering of one of the acquisition nodes. However, a problem arises from the fact that once the trigger is generated, it will not arrive to the receiving nodes simultaneously, due to varying distances and environmental conditions. The Distributed Oscilloscope (DO) concept attempts to address this problem by leveraging the sub-nanosecond synchronisation and deterministic data delivery provided by White Rabbit (WR) and augmenting it with automatic discovery of acquisition nodes and complex trigger event scheduling, in order to provide the illusion of a virtual oscilloscope. This paper presents the current state of the DO, including work done on the Field-Programmable Gate Array (FPGA) and software level to enhance existing acquisition hardware, as well as a new protocol based on existing industrial standards. It also includes test results obtained from a demonstrator used to showcase the DO concept, based on two digitisers separated by a 2.5 km optical fibre.

INTRODUCTION

From the monitoring of particle accelerators to smart electrical grids and from scientific experiments performed at the bottom of the sea to astronomical observatories and meteorological stations on mountaintops, a common requirement in large-scale Test and Measurement (T&M) setups has always been the ability to remotely control, automate and synchronise the involved equipment (instruments). This is of course true even in smaller setups, such as the ones found in laboratories, but it becomes even more important when the equipment is distributed across longer distances, or placed in remote and hard-to-reach locations.

The accelerator complex of the European Organisation for Nuclear Research (CERN), which includes the Large Hadron Collider (LHC) with a circumference of 27 km, is a prime example of such an installation. CERN operators need to be able to monitor the state of the accelerators and, very often, to correlate measurements performed across the accelerators (e.g. at beam transfer lines).

This paper presents the White Rabbit Trigger Distribution (WRTD) [1] system, a new development at CERN that allows sub-nanosecond synchronisation of instruments across several kilometres of distance and distribution of triggers over White Rabbit (WR) [2, 3] in the form of network messages. It also presents the Distributed Oscilloscope (DO) [4], a proof-of-concept project for WRTD.

BACKGROUND

Historically, the need for synchronisation of instruments has been addressed by various types of T&M systems. Already in the late 1960s, the General Purpose Interface Bus (IEEE-488, GPIB) [5] was introduced to allow control and readout of up to 14 daisy-chained instruments from a computer, using a cable of up to 20 meters total length. GPIB extenders were later introduced to overcome these limitations. Long coaxial cables were (and still are being) used to synchronise the instruments and distribute triggers by means of their external trigger input/output and "sync" ports. In the 1980s, the popularity of VMEbus led to the development of the VME eXtensions for Instrumentation (VXI) [6], and a decade later, PCI eXtensions for Instrumentation (PXI) [7] was added to the list. Both VXI and PXI included multiple trigger lines on their backplanes for synchronisation between the instruments.

In more recent years, the proliferation of Ethernet-based computer networks led in 2005 to the introduction of the LAN eXtensions for Instrumentation (LXI) [8]. Where VXI and PXI also imposed the mechanical format of the instruments, LXI focused on the protocol and provided services (such as automatic discovery of attached devices), allowing for rack-mounted, bench-top, modular or any other type of form factor to be used and interconnected, as long as they have a Local Area Network (LAN) port. Furthermore, the Ethernet network allowed for long distances between the instruments and their operators.

The LXI standard is divided into the so-called "Device Specification" [9] which contains requirements for all LXI-compatible devices, and a set of optional "Extended Functions". A group of three such extended functions, Clock Synchronisation (CS) [10], Event Messaging (EM) [11] and Timestamped Data (TD) [12] provide a synchronisation layer on top of the core LXI standard, offering similar capabilities to those found on the backplanes of VXI and PXI. LXI CS is based on the IEEE 1588 Precision Time Protocol (PTP) [13] and it specifies its own PTP profile. LXI devices supporting this function have their clocks synchronised with sub-microsecond accuracy. LXI EM defines the methods for the exchange of messages directly between the instruments using multicast User Datagram Protocol (UDP) or point-to-point Transmission Control Protocol (TCP) connections. LXI TD adds timestamps to all messages, events

* dimitrios.lampridis@cern.ch

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUBPR02

A 4-CHANNEL, 7 ns-DELAY TUNING RANGE, 400 fs-STEP, 1.8 ps RMS JITTER, DELAY GENERATOR IMPLEMENTED IN A 180 nm CMOS TECHNOLOGY

F. Badets, G. Billiot, S. Bouquet, G. Regis, A. Spataro, A. Fustier, F. Lepin,B. Caillat, C. Magnier
CEA-Leti, Grenoble, France
D. Monnier-Bourdin, B. Riondet, Greenfield Technology, Bordeaux , France

Abstract

This paper discloses the silicon integration, in a 180 nm CMOS technology, of a 4-channel delay generator dedicated to timing with resolution lower than a picosecond. This integrated circuit aims at miniaturizing delay generators while fulfilling the needs of modern physics experiments.

The delay generation principle relies on the linear charge of a capacitor triggered by the input pulse. The output pulse generation occurs when the capacitor voltage exceeds a programmable threshold voltage. A calibration circuit is embedded on-chip to automatically match the delay tuning range to the period of the master clock, ranging from 5ns to 7ns. The delay value is set with the help of a 14-bit DAC which leads to a 400fs delay step. Among other features, the chip embeds a combination mode of either 2 or 4 channels to output narrow width pulses. The chip is fully compliant with LVDS, LVPECL and CML differential pulses at its input and produces LVPECL pulses output.

The chip has been fully characterized over temperature (0 to 60°C) and supply voltage (+/- 10%). The chip is compliant with pulse repetition frequencies up to 20 MHz. The measured INL is 100 LSBs and the RMS jitter is 1.8 ps. The power consumption has been measured to 350 mW for 4 active channels.

INTRODUCTION

Synchronization of large physics experiments requires accurate timing systems having resolution down to a few picoseconds. They are usually made of several multi-channel delay generators that provide different sampling rates for multi-shot or single short triggering for each instrument distributed in a large area [1][2]. Experiments using large number of instruments require large number of delay channels and thus several multi-channel delay generators. Therefore, miniaturization of delay channels is highly desirable in order to provide either low volume delay generator modules or delay generators embedding a high number of delay channels.

A few Integrated Circuits providers offer delay generator products [3] [4] [5]. Nevertheless, as it could be seen in Table 1, these solutions failed to fulfil the needs of physics experiments in terms of time step which is limited to 5ps in the best case. This paper discloses the implementation of a 4-channel delay generator in a 180 nm CMOS technology and packaged in a small 56-pin QFN package. Its small form factor as well as its overall performances in terms of delay step (< 500 fs), full scale (7ns) and jitter (1.8ps) pave the way to the design of delay generators having very large numbers of delay channels and compliant with physics experiment requirements.

Table 1: Existing Delay Generators Integrated Circuits

Parameter	[3]	[4]	[5]	Unit
Delay Step	5	10	10	ps
Full Scale	5	5.6	10	ns
RMS jitter	1	-	3	ps
Pules repetition rate	1	1.5	1.2	GHz
Temperature drift	-	10	-	ps/°C
INL	30	40	20	ps

The paper is organized as follows: a first section introduces the targeted specification. A second section details the overall architecture of the proposed delay generator and the implementation of its building blocks and functions. The third section presents the measurements results. Some conclusions end the paper.

Table 2: Delay Generator Specifications

Parameter	Value	Unit
Channels	4	
Delay Step	<1	ps
Full Scale	7	ns
RMS jitter	< 2	ps
Pulse repetition rate	< 20	MHz
Master Clock Frequency	150 – 200	MHz
Temperature drift	<4	ps/°C
INL	<1	% FS

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUBPR03

MAJOR UPGRADE OF THE HIT ACCELERATOR CONTROL SYSTEM USING PTP AND TSN TECHNOLOGY

A. Peters†, C. Schoemers, J. M. Mosthaf
Heidelberg Ionbeam Therapy Facility (HIT), Heidelberg, Germany

Abstract

Two important reasons led to the first developments for a new ACS for the HIT ion therapy accelerator complex: a) the first implementation of the ACS was done in 2003-2005. It was implemented as a proprietary solution, which works very well and reliable for more than 10 years. However, more and more components e.g. parts of the device control units (DCUs, [1]) are no longer in stock. Thus a new realization using standard SoCs or similar is necessary; b) new functionalities like multiple energy operation [2] should enhance the duty factor of the accelerator facility resulting in significantly higher patient irradiation efficiency. In cooperation with our commercial partner Eckelmann [3] we are investigating the newly available deterministic Ethernet technologies like "Time-Sensitive Networking" with several IEEE 802.1xx sub standards [4]. Early TSN implementations in embedded controller boards and switches were obtained in a test installation in autumn of 2018. The test bench was set up to study the feasibility of e.g. the required timing precision using PTP, respectively IEEE 802.1AS-Rev. The aim is to realize a "one-wire-ACS" based on Ethernet only for deterministic data transfer and message based triggers for synchronized ACS functions. Results from our TSN test bench experiences will be reported.

THE HEIDELBERG IONBEAM THERAPY FACILITY

The HIT accelerator complex is based on a linac-synchrotron system accelerating ions to energies up to 430 MeV/u corresponding to ion penetration depths of approx. 30 cm in human tissue.

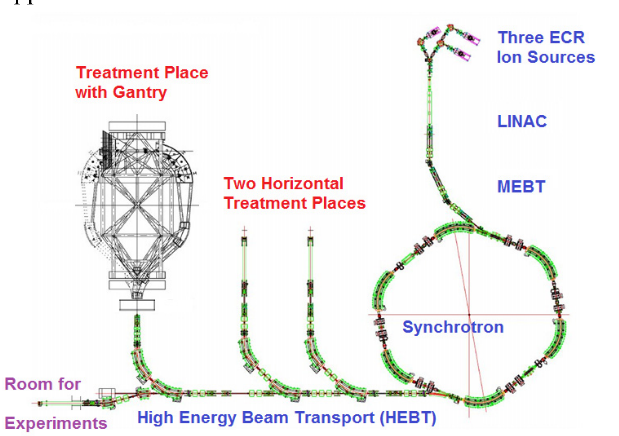


Figure 1: Schematic view of the HIT accelerator complex.

The facility – constructed from 2003-2008 – is equipped with two fixed horizontal beam lines, a rotating beam line (heavy ion gantry), all for patient treatment, and an experimental area (Fig. 1). The synchrotron is a cyclic operating device with phases of beam injection, acceleration to the desired particle energy, corresponding to the desired penetration depth (iso energy slice), beam extraction, and preparation for the following cycle. Ions are slowly extracted by the transverse knock-out extraction method with extraction times which last up to 5 s [5]. HIT uses the intensity controlled raster scanning method of pencil beams as dose delivery system [6].

Cancer therapy with carbon ion and proton beams has been carried out at HIT since 2009 (gantry part since 2012). Currently around 700 patients are irradiated per year; approximately 5700 patients have been treated in total since the beginning.

CURRENT HIT ACS AND ITS TIMING SYSTEM

The HIT ACS has the following top-down structure:

- *Presentation Layer* with GUIs on Operator PCs under Win10,
- *Coordination Layer* with different servers (OracleDB, Sequence Control, SataSupply Model) under WinServer2019 and the timing master, which feeds the real-time bus (RTB) and communicates with the Therapy Control System (TCS),
- *Communication Layer* with network, RTB and linked DCUs
- *Devices* and Subsystems for slow controls.

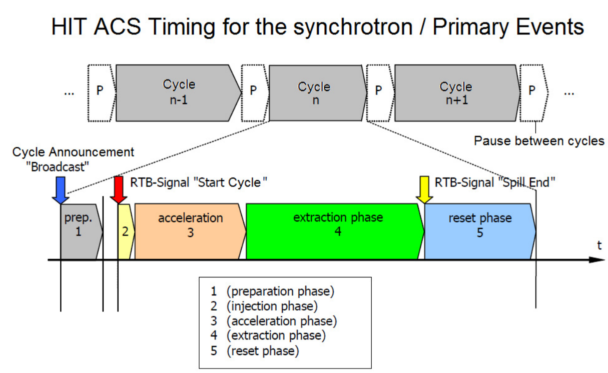


Figure 2: HIT ACS timing for the synchrotron part [7].

During treatment the TCS sends commands with the next ion beam request via CAN bus to the ACS, which contain the next accelerator settings (ion species, energy, intensity, focus) to be carried out – safety in this communication and the subsequent cycle execution is assured by several mechanisms like redundancy, checksums, etc.

† Andreas.Peters@med.uni-heidelberg.de

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUBPR04

THE FAULT DIAGNOSIS OF EVENT TIMING SYSTEM IN SuperKEKB*

D. Wang[1,†], K. Furukawa, M. Satoh, H. Kaji, H. Sugimura, KEK, Japan
Y. Iitsuka, East Japan Institute of Technology Co., Ltd, Japan
T. Kudou, S. Kusano, Mitsubishi Electric System & Service Co., Ltd, Japan
[1]also at SOKENDAI, Japan

Abstract

The new MRF event timing system is one of the most important components to maintain the reliable and stable operation of the SuperKEKB project. This system is utilized to distribute high precision level timing signals and accompanying control instructions to synchronize different subsystems and machines. Event generator (EVG) generates signals of different beam modes every 50-Hz pulse which contains several event codes while Event receivers (EVR) receives them and output signals to dedicated devices all over the installation. To certain these events are consistent during the distribution, an event fault diagnosis system is essentially needed. An EVR based event timing diagnostic system is thus developed by modifying the driver support module to provide a log system of persistent event data as well as comparing the received event codes with the beam injector pattern, detecting the event timing interval fault and notifying the results by email every day. Then, we are able to locate the fault, analyze the data, fix bugs or replace hardware and resume accelerator operation quickly.

INTRODUCTION

The SuperKEKB is an electron/positron collider upgraded from the KEKB project since 2010 at KEK, whose scientific target is to update the world highest luminosity record and discover new particle physics by Belle II experiment. The 7 GeV electrons and 4 GeV positrons are injected into different main rings (MRs) called high energy ring (HER) and low energy ring (LER), respectively. During the phase-2 operation in 2018, a newly constructed 1.1 GeV Damping Ring (DR) at the middle of injector linac (LINAC) aiming to lower the positron emittance was deployed. One of the major challenges in the phase-3 operation is the upgrade of timing system of SuperKEKB [1]. The 700-meter LINAC delivers trigger signals to five rings, SuperKEKB HER/LER, two light source ring PF and PF-AR and a positron DR. The DR timing signal must synchronize with the MR signal.

For switching the beam modes swiftly, a large quantities data like bucket selection delay time, beam mode event codes and pulse number is required to update rapidly. Two sets of timing signals should be conveyed because the condition of beam line before DR and after DR is dissimilar. Therefore we make use of three EVGs and some other modules to construct our timing delivery system. With the growth of the system complexity, it is important to diagnose the timing system and ensure the correctness of the signal delivery. Hence a fault diagnosis system is developed to assist us to monitor the timing event information.

In this paper, some efforts to improve the reliability and stability of timing system will be introduced.

* Work supported China Scholarship Council
† sdcswd@post.kek.jp

TIMING SYSTEM ARCHITECTURE

At an accelerator facility, the function of timing system is to provide synchronized trigger signal to all relevant components and devices which locates at LINAC, beam transmission line, injection and extraction system and beam monitor system. The precision of synchronization depends on the operation requirement as well as the size of the installation [2]. In our case, a jitter of less than 30 ps is required in SuperKEKB MR and 300/700 ps in PF/PF-AR.

MRF Event Timing

We introduced the event timing system produced by the MRF company to meet our requirements [3]. The products we choose for our main timing system are "Event Generator (VME-EVG-230)" and "Event Receiver (VME-EVR-230-RF)". At SuperKEKB, the event clock rate is 114.24 MHz and hence the minimal event code interval is around 9 nanosecond. Every 20 ms, depending on the beam mode, 11 or 12 events are generated and distributed to more than 60 EVRs all over the SuperKEKB.

The EVG uses optical fiber to transmit a 16-bit word by 8b/10b encoding. Inside this 16-bit word, 8-bit distributed bus running in parallel and independent of the other 8-bit allows distribution of timing signals updated with the event clock rate. Almost all 256 event codes except for some special functions can be defined by users. The EVG distributes signal by fanout Units to an array of EVRs. Each EVR is able to generate pulse with an associated delay and width to devices. It must be emphasized that the MRF timing system is well integrated with EPICS system with the help of mrfioc2 device support module [4]. Figure 1 is the scheme of the MRF timing system.

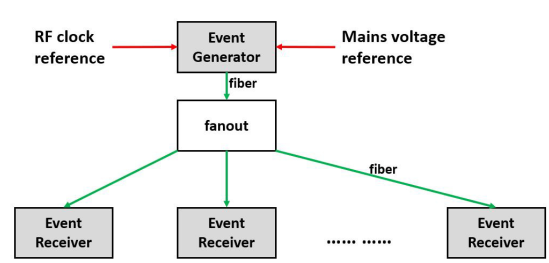


Figure 1: The MRF event system.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUBPR05

LEReC TIMING SYNCHRONIZATION WITH RHIC BEAM*

P. K. Kankiya†, J. P. Jamilkowski, M. R. Costanzo
Brookhaven National Laboratory, Upton, USA

Abstract

RHIC low energy bunched beam cooling experiment, LEReC, a 704 MHz fiber laser is modulated such that when striking a photocathode, it produces corresponding electron bunches which are accelerated and transported to overlap an ion beam bunched at 9 MHz RF frequency The need for precise timing is handled well by the existing infrastructure. A layer of software application called the timing manager has been created to track the LEReC beam concerning the RHIC beam and allow instruments to be fired in real-time units instead of bunch timing or RHIC turns. The manager also automates set-tings of different modes based on the RF frequency and maintains the timing of instrumentation with a beam. A detailed description of the bunch structure and scheme of synchronizing the RF and laser pulses will be discussed in the paper.

INTRODUCTION

In contrast to previous electron cooling systems that utilized DC beams, the Low Energy RHIC eCooling (LEReC) accelerator has recently demonstrated cooling of RHIC ion beams using bunched electron beams. During the FY2019 RHIC run, cooling was commissioned using electron beams with kinetic energy of 1.6 MeV to cool Au ions at 3.85 GeV/nucleon as well as using 2 MeV electron beams to cool Au ions at 4.6 GeV/nucleon [1].

Cooling a bunched beam of electrons is achieved by illuminating a multi-alkali photocathode, inserted into a high-voltage dc gun with an operating voltage around 400 kV. Light emitted from a fiber laser hitting the photocathode produces individual electron bunches of 40 ps full length at 704 MHz frequency, modulated at a 9 MHz macro-bunch frequency, to match the repetition rate of ion bunches in RHIC. A macro-bunch of electrons consisting of 30 individual electron bunches is timed with each individual ion bunch. The use of such macro-bunches allows the total charge of about 3 nC required for cooling, to be divided into 30 e-bunches with 0.1 nC per bunch. The LEReC beam structure consists of each group of electron bunch spaced by 1.4 ns placed on a single ion bunch, with ion bunch repetition frequency of 9 MHz [2].

The desired timing parameters for generation of bunched beam of anticipated characteristics are given in Table 1.

Generation of Triggers

As depicted in Fig. 1, the triggers for various instruments are generated by two main RHIC timing modules namely Beam-synchronous Trigger (V124) and Timing Decoder (V202) VME boards that are programmed to generate fine timing triggers for accelerator operations and also have software interfaces [3].

Table 1: Machine Timing Parameters

Name	harmonic	Frequency (Hz)
Frev	1	7.82e4
28Mhz Bsyn	360	2.82e7
704 Loopback	1944	1.52e8
704 Laser IF	1944	1.52e8
140Mhz Dig Clk	1800	1.41e8
Nominal 9Mhz	120	9.39e6
704Clk/9Mhz		75

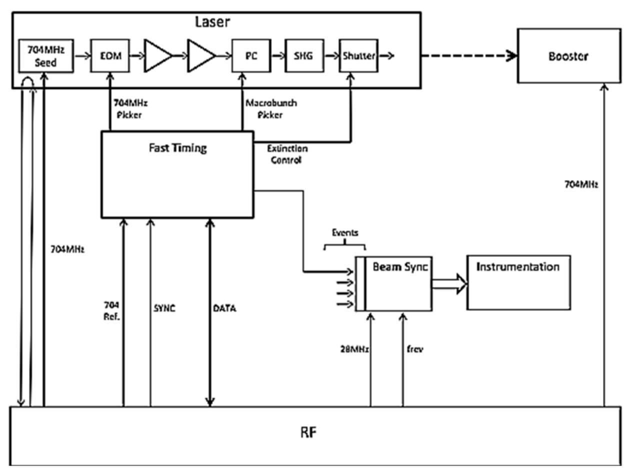


Figure 1: Trigger generation scheme for RHIC and LEReC beam phase synchronization.

The key elements of timing and synchronization are a distributed 100 MHz clock and the Update Link [4], which is a deterministic data and control link that ties systems together whether in the same rack or halfway across the facility.

Each low-level RF system creates a 400 MHz clock from the global 100 MHz clock. The phase advances every 2.5 ns is calculated from the harmonic and revolution frequency. Since T (the clock period) is the same for every system, the frequency ratios are whole numbers. The RF output comes from a lookup table for sine and cosine values the frequency ratios are whole numbers. The RF output comes from a lookup table for sine and cosine values.

When the ratio between 9 MHz and 704 MHz is not an integer, we need to change the number of “off” bunches between macro bunches.

* Work supported by Brookhaven Science Associates, LLC under Contract No. DE-SC0012704 with the U.S. Department of Energy.
† pkankiya@bnl.gov

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUBPR06

LASER MEGAJOULE TIMING SYSTEM

T. Somerlinck, N. Bazoge, T. Falgon, S. Hocquet, D. Monnier-Bourdin
CEA CESTA, Le Barp, France

Abstract

The "Laser MégaJoule" (LMJ) timing system, under development since the year 2000 and tested on the "Ligne d'Intégration Laser" (LIL), laser facility (prototype of LMJ), is now entering in his final commissioning and installation. To synchronize the laser beams on the target better than 40 ps rms, the timing system needs to produce electrical pulses with jitter lower than 5 ps rms and drift limited to 20 ps peak to peak. These requirements have been reached with the last evolution of delay generator in our distributed optical architecture.

INTRODUCTION

The LMJ facility is a high-power laser designed to deliver about 1.4 MJ of laser energy to targets for high energy density physics experiments, including fusion experiments [1]. This energy is produced by 176 laser beams gathered in quadruplets of 4 beams. Each quadruplet is equipped with an Arbitrary Waveform Generator (AWG) that generates the desired temporal pulse shape (lasting typically 3 ns). Synchronization of LMJ's 176 laser beams is crucial to compress symmetrically the millimeter-size target in order to ignite the deuterium and tritium filled capsule. The most demanding experiences need to synchronize the quadruplets to better than 40 ps rms despite the fact the quadruplet laser sources are separated within the building by several hundred meters. In addition to laser beams synchronization, the LMJ timing system is in charge to deliver, with the same or lower accuracy, two kinds of signals: fiducials for both temporally mark signals and plasma diagnostics, and triggers signals for manifold devices (sources, amplifiers, Pockels cells, diagnostics...).

The synchronization is therefore one of the most important components for shot experiment, from the laser sources to the target inside the chamber, as shown in Fig. 1.

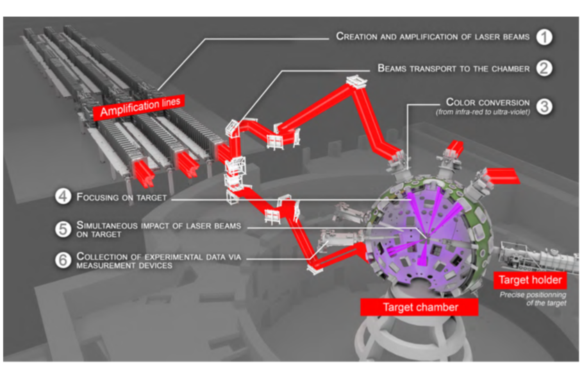


Figure 1: From laser amplification to experiment in the chamber.

The error budget calculus to reach the 40 ps rms specification has showed that the requirements for the timing systems were the following [2]:

- < 5 ps rms jitter or 8 ps rms between 2 outputs
- < 10 ps p-p drift / 24 hours
- < 20 ps p-p drift / 1 month

LMJ TIMING SYSTEM

As seen previously, the LMJ requires a lot of timing channels with different accuracies. From 2002 to 2013, three levels of timing system were defined [2] [3] [4] until new data were analyzed from:

- LIL prototype,
- First laser beams on LMJ facility,
- Measurements by Greenfield Technology (GFTy).

Finally, the second and third level timing systems were merged in only one High Precision Timing system (HPT) in complement to the Standard Precision Timing system (SPT), Fig. 2.

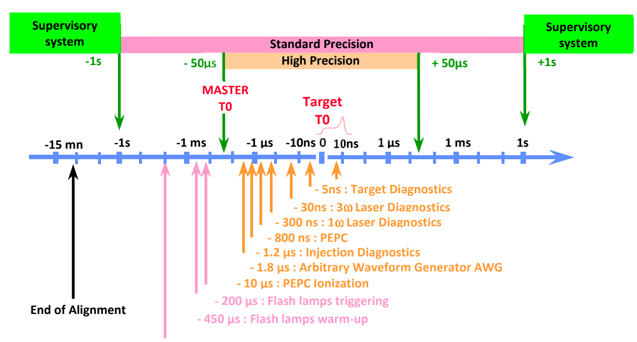


Figure 2: New LMJ synchronization time line.

Table 1 below summarizes the actual requirements of the two levels of precision:

Table 1: LMJ Timing System Requirements

Precision	Jitter rms	Temporal Drift peak to peak			Accuracy	Range
		24 h	7 days	1 month		
Standard (SPT)	<150ps	<200ps	<500ps	<1ns	<±1ns	1s
High (HPT)	<5ps	<10 ps	<20ps	<20ps	<±10ps	100µs

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUCPL01

ADDING MACHINE LEARNING TO THE ANALYSIS AND OPTIMIZATION TOOLSETS AT THE LIGHT SOURCE BESSY II

L. Vera Ramirez*, T. Mertens, R. Mueller, J. Viefhaus, G. Hartmann
Helmholtz-Zentrum Berlin, Germany

Abstract

The Helmholtz Association has initiated the implementation of the Data Management and Analysis concept across its centers in Germany. At Helmholtz-Zentrum Berlin, both the beamline and the machine (accelerator) groups have started working towards setting up the infrastructure and tools to introduce modern analysis, optimization, automation and AI techniques for improving the performance of the (large scale) user facility and its experimental setups. This paper focuses on our first steps with Machine Learning (ML) techniques over the past months at BESSY II as well as organizational topics and collaborations. The presented results correspond to two complementary scenarios. The first one is based on supervised ML models trained with real accelerator data, whose target are real-time predictions for several operational goals (beam lifetime, injection efficiency, beam loss...); some of these techniques are also used for additional tasks such as outlier detection or feature importance analysis. The second scenario includes first prototypes towards self-tuning of machine parameters in different optimization cases (booster current, injection efficiency, orbit correction...) with Deep Reinforcement Learning (RL) agents.

MOTIVATION

The integration process of ML tools with real accelerator data at BESSY II is being carried out with two main goals: modelling and prediction on the one hand and self-optimization on the other. As for today many specific prediction models for different accelerator parameters have been already constructed and analyzed - apart from the beam lifetime case presented here, different beamloss monitors along the ring as well as injection efficiency have been modeled. This is also an important preparatory step for the RL-based tuning as well as for the self-optimization of surrogate models. Besides, further significant effort is being put into beamline raytracing and the conception of *digital twins*, which can be also connected with the RL-based self-optimization. The aim of this paper is to summarize some of this application cases as a sort of *proof-of-concept* of the major possibilities opened by the incorporation of ML tools at BESSY II.

PREDICTION OF BEAM LIFETIME

We present a representative case of beam lifetime prediction restricted to a *blind* scenario: time-series-based prediction of beam lifetime *only* with context variable readbacks, i.e., omitting the previous measurements of the target variable. This scenario allows us to identify unknown correlations and patterns in the readbacks avoiding an excessive reliance on the previous target variable measurements but also to reuse the information and experience gained with the prediction models in a RL context.

The beam lifetime τ is defined through the current decay rate $\frac{1}{\tau} = -\frac{\dot{I}}{I}$, where I denotes the beam current (for a study of the beam lifetime at BESSY II see, e.g., [1]). For these experiments we approximated the *instantaneous* lifetime through a piecewise linear regression with k previous measurements of the beam current I_t (usually $k = 20$ seconds):

$$\frac{1}{\tau} \approx -\frac{1}{I_t} \frac{\sum_{i=0}^{k} (I_{t-i} - I_{t_0})\,(t - i - t_0)}{\sum_{i=0}^{k} (t - i - t_0)^2}$$

The potentially beam lifetime affecting variables used as input (185 after preprocessing, see Appendix) are:

- Gap and shift of insertion devices (elliptical) undulators affecting the dynamic aperture (21 readback variables).
- Power supply currents into quadrupoles define the linear optics (58 readback variables), into sextupoles define non linear behavior (7 variables).
- Offsets to power supplies for quadrupoles define the feed forward compensations (38 variables).
- Collisions with rest gas particles, vacuum pressures measured by getter pump current (12 variables).
- Local beam loss fractions, monitored by counters close by (49 variables).

Comparison of Methods

We have worked with the following supervised learning models:

- Extremely Randomized Trees (`ExtraTrees`, [2]).
- Support Vector Regression *approximated* with Random Fourier Features (`SVR-RFF`, [3]).
- Standard dense feed-forward Neural Networks (`DNN`, e.g., [4]).

Table 1 contains the results of these prediction models for two different test set elections: a random uniform set (20%) along the measurement period or the last 20% of the measurement period. The different hyperparameter configurations after grid search as well as further test settings can be found in the Appendix[1].

* luis.vera_ramirez@helmholtz-berlin.de

[1] Other tested algorithms (traditional Random Forests and SVR with different kernels) presented similar or worse results so we excluded them from the table for the sake of clearness.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUCPL02

PROCESSING SYSTEM DESIGN FOR IMPLEMENTING A LINEAR QUADRATIC GAUSSIAN (LQG) CONTROLLER TO OPTIMIZE THE REAL-TIME CORRECTION OF HIGH WIND-BLOWN TURBULENCE*

M. K. Kim†, S. M. Ammons, B. Hackel, L. A. Poyneer
Lawrence Livermore National Laboratory, Livermore, USA

Abstract

LLNL has developed a low latency, real-time, closed-loop, woofer-tweeter Adaptive Optics Control (AOC) system with a feedback control update rate of greater than 16 kHz. The Low-Latency Adaptive Mirror System (LLAMAS) is based on controller software previously developed for the successful Gemini Planet Imager (GPI) instrument which had an update rate of 1 kHz. By tuning the COTS operating system, tuning and upgrading the processing hardware, and adapting existing software, we have the computing power to implement a Linear-Quadratic-Gaussian (LQG) Controller in real time. The implementation of the LQG leverages hardware optimizations developed for low latency computing and the video game industry, such as fused multiply add accelerators and optimized Fast Fourier Transforms. We used the Intel Math Kernel Library (MKL) to implement the high-order LQG controller with a batch mode execution of 576 6x6 matrix multiplies. We will share our progress, lessons learned and our plans to further optimize performance by tuning high order LQG parameters.

INTRODUCTION

The development of an Adaptive Optics (AO) system to correct for high wind-blown turbulence requires a team of experts in the fields of optics, fluid dynamics, controls systems, and software. This paper focuses on the processing and software optimizations.

Increasingly, the more significant terms in adaptive optics wavefront error budgets spanning a myriad of applications are temporal wavefront errors. These are associated with an adaptive optics system' inability to keep up with ever evolving turbulence due to a deficiency in sensor update rate ("frame rate") and/or the latency in the system. The latency is the amount of time required for the wavefront sensing measurement to be completed and readout, the reconstruction computation time, and the electrical and mechanical latency in updating the deformable mirror position

The maximum frame rate is set by multiple factors, including the camera readout time, reconstruction latency, and beacon brightness, and is thus tied to the total end-to-end latency. With standard leaky integrator controllers utilizing modal gains, there is limited performance benefit to increasing the frame rate beyond a level at which the latency is 2-3 frame times. The principle strategy to reduce bandwidth and delay errors is to reduce end-to-end latency.

This manuscript is concerned with the wavefront reconstruction process in adaptive optics systems, particularly for the Low Latency Adaptive Mirror System (LLAMAS) at Lawrence Livermore National Laboratory [1]. This testbed was designed to run with minimal computational latency so as to maximize performance at high frame rates (> 16 kHz). The system utilizes Linear Quadrature Gaussian (LQG) control to maximize bandwidth at a given frame rate [2,3].

PROCESSING IMPROVEMENTS

The team focused on improving the processor-intensive portion of the AO system. Since the processing system was selected for GPI during its Preliminary Design Phase, the technology has advanced considerably. The latest servers and operating systems were researched with latency performance and determinism in mind.

Hardware Selection

The team selected the HPE ProLiant DL580 Gen 10 8SFF CTO Server. The LLAMAS server specifications and server specifications for the GPI project are listed for comparison in Table 1.

Table 1: LLAMAS vs. GPI Server Specification Comparison

	CPU	Clock (GHz)	Cores	Cache (MB)	RAM (GB)
LLAMAS	Intel Xeon Platinum 8158 (3)	3.0	3 x 12	24.75	128
GPI	Intel Xeon E7440	2.4	4 x 4	16	32

With only a 25% improvement in clock speed, one may conclude that the performance of the software would not improve by magnitudes. This metric is no longer the primary factor when comparing performance. The increase in cores provides a performance advantage for a software architecture that uses multiple execution threads to maximize the parallel processing. Even though the LLAMAS server has less sockets (3 vs. 4), the number of cores per socket is also a processing advantage if execution multiple threads

* Work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory Contract DE-AC52-07NA27344 with document release number LLNL-PROC-792238.
† kim94@llnl.gov

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUCPL03

THE LMJ TARGET DIAGNOSTICS INTEGRATION

S. Tranquille-Marques, P. Prunet, CEA/CESTA, Le Barp, France

Abstract

The French Laser Megajoule (LMJ) is, behind the US NIF, the second largest inertial fusion facility in the World. The main activity of this facility is the acquisition of several physical phenomena as neutron, gamma, X rays produced by the indirect attack of hundreds of high power laser beams on targets through measurement devices called “target diagnostics”.

More than 30 diagnostics will be installed and driven in a huge and complex integrated computer control system. All these Target Diagnostics are coming one by one, each with its own specificity and complexity. The Tango framework and the Panorama SCADA are used for command and control.

The aim of this paper is first, to introduce how Target Diagnostics are progressively integrated in the command control. We will then see how Target Diagnostics are managed to cohabit even if they are in different phases of their integration process.

Finally the paper explains how Target Diagnostics are configured and computer-driven during the shot sequence.

INTRODUCTION

Since it definitively abandoned nuclear testing, France has relied on the Simulation Program to guarantee the operational performance and safety of its nuclear deterrent weapons throughout their lifetime.

Successful simulation requires both:

- Qualified computer codes that integrate laboratory-validated physics models to simulate weapon functioning;
- Teams of qualified physicists to use these codes.

In this respect, the Megajoule Laser (Fig. 1) is used to validate the numerical codes and certify the skills of French physicists.

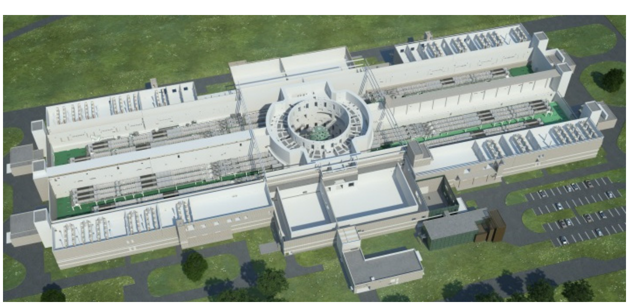


Figure 1: LMJ view.

On October 23, 2014, French Prime Minister Manuel Valls declared the facility operational after starting up the first experiment.

Target Diagnostics (TD) are a key for numerous physical data acquisition. CEA will develop more than 30 of these equipments during next twenty years (Fig. 2).

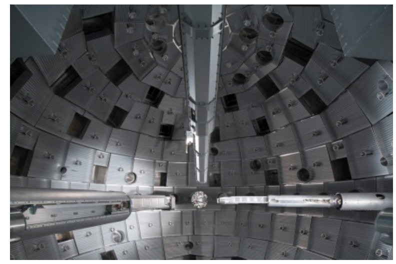


Figure 2: Target chamber view.

Each Target Diagnostic will be dedicated to one or several kind of measurements like X-ray, visible, UV or parti-cles like neutron (Fig. 3).

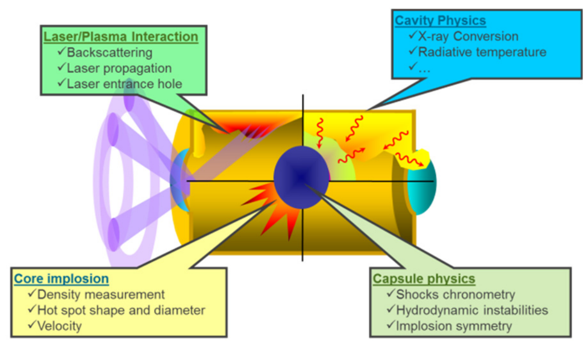


Figure 3: Target diagnostics measurements.

These Target Diagnostics are classified in 5 families (Fig. 4):

- Insertables TD
- Specific mechanical TD
- Visible/UV TD
- Neutronics TD
- Petal+ TD

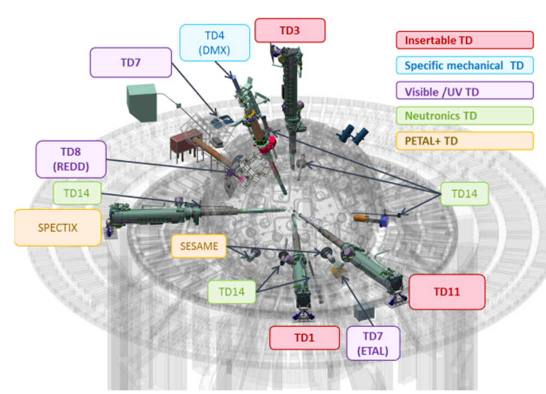


Figure 4: Target diagnostics families.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUCPL04

A MODEL-BASED SIMULATOR FOR THE LCLS ACCELERATOR

M. Gibbs*, W. Colocho, J. Shtalenkova, A. Osman,
SLAC National Accelerator Laboratory, Menlo Park, USA

Abstract

The Linac Coherent Light Source (LCLS) at the SLAC National Accelerator Laboratory is currently undergoing a major upgrade. In order to facilitate the development of new software that will be needed to operate the upgraded machine, a simulator of the LCLS electron beam, and accelerator devices that measure and manipulate the beam, has been developed. The simulator is comprised of several small "services" that simulate different types of devices, and provide an EPICS interface identical to the real control system. All of the services communicate with a central beam line model to change accelerator parameters and retrieve information about the simulated beam.

MOTIVATION

SLAC has spent the majority of 2019 with the LCLS accelerator offline while a new superconducting linear accelerator is installed upstream of the existing normal-conducting linear accelerator, and the original transport line from the linac to the hard x-ray undulator line is replaced with two new transport lines feeding two new undulator lines. This will fundamentally change the structure of LCLS from a "straight line" machine with one electron source and one beam destination to a machine with two electron sources, each of which can feed either a hard x-ray or soft x-ray undulator line. A large number of high-level software applications used for the operation of the accelerator (including applications to measure electron beam characteristics, automation tools to perform routine procedures, and tools to align and calibrate beam-line devices) need modification to support multiple beam-lines, or new hardware. The hardware installation schedule dictates that most of the software modifications need to happen before hardware is installed or connected to the EPICS control system.

An accelerator simulation system called Simulacrum was created to give software developers a way to test their applications against a simulated electron beam using an EPICS interface identical to what the real accelerator provides.

ARCHITECTURE

Simulacrum is a modular system, comprised of many independent device simulation services communicating with a single accelerator model service (see Fig 1). Each device service hosts the complement of EPICS process variables (PVs) that applications use to interact with the device type. When applications read or write to these PVs, the device service can inform the model of changes to device parameters, trigger a re-calculation of the simulated beam, and query the model for the state of the beam at a particular location.

* mgibbs@slac.stanford.edu

IMPLEMENTATION

All services are written in Python 3, and use the PyZMQ module [1] to communicate with the model service via ZeroMQ [2] messages over TCP. The services are typically all run from one computer, but the TCP-based communication provides a path to running the simulator across several computers, if the need arises.

Model Service

The model service is the only service that users *must* run. It hosts an instance of Tao, an accelerator modelling application that is part of the BMAD software toolkit for charged particle and x-ray simulations [3]. The model service takes a BMAD accelerator lattice definition as input, and instantiates an accelerator model. Because the model service is in charge of keeping the lattice definition, it is easy to simulate different accelerators, as long as their devices share the same EPICS interface. This is routinely used to switch between simulating the machine as it existed in 2018 to simulating the machine as it will exist when operations resume.

A request-response pattern is used to interact with the model: the model service sends its Tao instance a command, and a result is returned. This is the primary way other services interact with the model: they send a Tao command via ZeroMQ, the model service receives the command, runs it, and replies with the result of the command. It is also possible to open an interactive terminal to send and receive Tao commands - this is useful for debugging, and also allows expert users to configure parameters of the model that are unavailable via EPICS, like misalignment of devices and initial beam conditions.

In addition to the request-response communication, the model service has a second communication channel that operates using a publish-subscribe pattern. This channel is used to broadcast frequently-updating beam parameters (like trajectory) at 10 Hz. Any service can subscribe to these broadcasts, and update PVs (like beam position monitor signals) in real time.

Finally, the model service uses the p4p module [4] to act as a PVAccess server. It hosts a large NTTable PV that contains the beam's Twiss parameters at every element, along with the length of the element, the element's position along the beam-line, and the 6x6 transfer matrix for each element. This table is refreshed and re-published at 1 Hz.

Device Services

The device services are the clients of the model service, sending Tao commands and receiving information about the current state of the simulated devices or beam. When a device service starts, it queries the model service for a list of all the devices the service will simulate (for example, a list

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUCPL05

ESRF-DOUBLE CRYSTAL MONOCHROMATOR PROTOTYPE - CONTROL CONCEPT*

M. Brendike, G. Berruyer, H. Gonzalez, L. Ducotté, C. Guilloud, M. Perez, R. Baker,
ESRF – The European Synchrotron, Grenoble, France

Abstract

The ESRF-Double Crystal Monochromator (ESRF-DCM) has been designed and developed in-house to enable spectroscopy beamlines to exploit the full potential of the ESRF-EBS upgrade. To reach concomitant beam positioning accuracy and beam stability at nanometer scale with a reliable, robust and simple control system, a double cascaded control architecture is implemented.

The cascade is comprised of three modes: classic open loop actuation, an optimized open loop mode with error mapping, and closed loop real-time actuation. Speedgoat hardware, programmable from MATLAB/SIMULINK and running at 10 kHz loop frequency is used for the real-time mode. From the EBS startup 2020, the ESRF plans to deploy BLISS – the new BeamLine Instrumentation Support Software control system – for running experiments. An interface between Speedgoat hardware and BLISS has therefore been developed. The DCM and its control architecture have been tested in laboratory conditions.

An overview of the concept, implementation and results of the cascaded control architecture and its three modes will be presented.

INTRODUCTION

Beamlines applying x-ray experimental techniques, such as x-ray absorption spectroscopy (XAS), are among the most demanding applications for monochromators. They need to scan through x-ray wavelengths (energies) quickly, in a repeatable manner, and without disturbing the position of the x-ray beam. Such ESRF beamlines are typically equipped with vertically-deflecting, fixed exit double crystal monochromators (DCMs), which allow to scan without readjustment of the downstream optical elements, or samples. The principle of a fixed exit DCM is explained in Fig. 1. Depending on the incident angle (Bragg angle, θ) of the beam on the first crystal an energy, E, can be selected. To maintain the reflected beam at the same height for all Bragg angles, θ, the second crystal needs to be adjusted relative to the position of the first. Current DCMs at the ESRF which follow this principle, are KOHZU monochromators as described in [1]. These DCMs keep the second crystal at the correct distance by a double cam mechanism. Since this is a purely mechanical process, second crystal positioning precision is limited. Prestipino et al. [2] proposed an active feedback system, called MOCO and later MOCO2 to compensate for mechanical errors. Although this improves results, limitations in terms of actuation bandwidth and usability for low energy XAS (such as ESRF-ID21) remain. As most of the current KOHZU DCMs were purchased around 20 years ago, ageing is a major issue. In addition the ESRF-extremely brilliant source (EBS) upgrade will result in a more coherent and more intense light source [3]. The subsequent improvement in experimental techniques will further increase the demand in beam position stability and actuation bandwidth. These new requirements led to the decision of designing a new generation of DCMs in-house at the ESRF.

* **Authors contributions:** M.B. design and implementation; G.B., C.G. and M.P. implementation in BLISS and BLISS-RPC; H.G. electronics; L.D. concept of metrology and actuation; R.B. Project coordinator

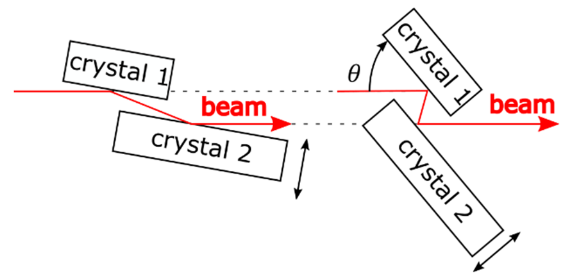


Figure 1: Principle of a fixed exit DCM; left at low Bragg angle; right at high Bragg angle.

To exploit the potential of the new ESRF-EBS, monochromators need a second crystal positioning system that is simultaneously faster and more accurate than that of KOHZU DCMs. Baker et al [4] presented the mechanical concept of the ESRF-DCM prototype in 2018. Figure 2 shows the principle of the second crystal positioning system. While KOHZU DCMs use a double cam mechanism, the ESRF-DCM is equipped with a tripod positioning stage composed of three stepper motors, for the long stroke (in-

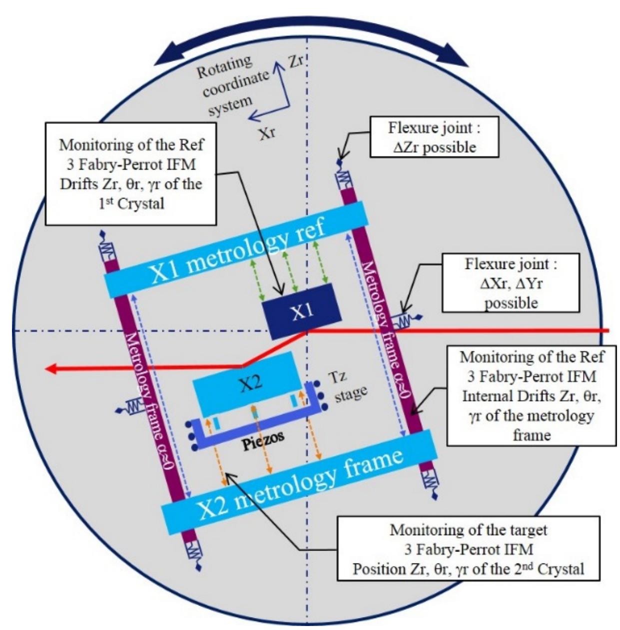


Figure 2: Online crystal metrology and second crystal actuation schema.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUCPL06

ACCELERATING MACHINE LEARNING FOR MACHINE PHYSICS (AN AMALEA-PROJECT AT KIT)

T. Boltz and W. Wang*, E. Bründermann, M. Caselle, A. Kopmann, W. Mexner, A.-S. Müller
Karlsruhe Institute of Technology, Karlsruhe, Germany

Abstract

The Innovation Pool project Amalea of the Helmholtz association of Germany will explore and provide novel cutting-edge machine learning techniques to address some of the most urgent challenges in the era of large data harvests in physics. Progress in virtually all areas of accelerator-based physics research relies on recording and analyzing enormous amounts of data. This data is produced by progressively sophisticated fast detectors alongside increasingly precise accelerator diagnostic systems. As KIT contribution to Amalea, it is planned to investigate the design of a fast and adaptive feedback system that reacts to small changes in the charge distribution of the electron bunch and establishes extensive control over the longitudinal beam dynamics. As a promising and well-motivated approach, reinforcement learning methods are considered. In a second step the algorithm will be implemented as a pilot experiment to a novel PCIe FPGA readout electronics card based on ZYNQ UltraScale+ MultiProcessor System on-Chip (MPSoC).

INTRODUCTION

With the increasing demand for compact, energy- and cost-efficient accelerator systems, in addition to tailored photon emission matched to the often extreme requirements of experiments in physics and photon science, the control systems have to cope with increasing complexity, high sensor data output rates, large data volumes as well as the desire for fast feedbacks and extensive beam control. Artificial intelligence with its subfield of machine learning including unsupervised, supervised and reinforcement learning, as well as deep learning, promises to assist in reducing the effort and complexity for operating a control system up to the point, where it may eventually control an accelerator autonomously. At the Karlsruhe Institute of Technology (KIT), since a few years, we are exploring machine learning methods for data classification, data reduction, and accelerator control informed by fast and precise sensor networks [1–4]. Since 2019, the Helmholtz Association in Germany is funding an Innovation Pool project called Amalea (Accelerating Machine Learning for Physics), which is exploring machine learning for accelerator-based physics, fast data reduction, and fast feature extraction from data, to name a few application areas. Amalea is driven by four Helmholtz centers, led by Deutsches Elektronen-Synchrotron (DESY), Helmholtz-Zentrum Berlin (HZB), Helmholtz-Zentrum Dresden-Rossendorf (HZDR) and KIT. The aim of Amalea is to investigate how novel machine learning methods, applied to the fields of particle physics, photon science and accelerator physics provide meaningful and effective use cases. At KIT and as one of the use cases contributing to the Amalea project, we explore how we can accelerate machine learning algorithms in real-time for machine physics applications and control. In this contribution, we discuss our efforts towards the design of a longitudinal feedback that acts on the RF system of the KIT storage ring KARA (Karlsruhe Research Accelerator) and aims for control of the micro-bunching instability. Driven by the interaction of short electron bunches with their own emitted coherent synchrotron radiation (CSR), this instability leads to the formation of dynamically changing microstructures within the longitudinal charge distribution of the bunch. Given its dynamic nature, a fast and adaptive feedback system is required to establish extensive control over the longitudinal beam dynamics. Reinforcement learning is a general-purpose approach to solving such problems, which has seen great success over the past decades. In [4], we illustrate how reinforcement learning can be applied to this task specifically, yielding the design of a longitudinal feedback loop. In the following, we review this idea and, in extension to [4], discuss some of the challenges in implementing this approach on a fast hardware system to meet the strict requirements regarding execution time. Therefore, KIT is developing a reinforcement learning hardware platform for the eventual implementation of the feedback design discussed below. The platform consists of two boards, the KAPTURE-2 front-end electronics that samples the pulse from the accelerator, and a high-end FPGA data acquisition board that provides high-data volume throughput that can process the data continuously. Based on which, a fast neural network inference can be deployed on FPGA for the fast inference requirement, and a lightweight training process is developed on ARM (or both on ARM side). To provide a proof of concept, the textbook *CartPole* environment is built on a ZYNQ MPSoC platform to test the performance of the reinforcement learning algorithm on hardware.

MICRO-BUNCHING INSTABILITY

Above a certain threshold current, which depends on the machine settings of the storage ring [5], the CSR self-interaction of short electron bunches leads to a dynamically changing longitudinal charge distribution and thus to fluctuating CSR emission (illustrated in Fig. 1). These fluctuations have been measured at a wide range of synchrotron light sources [6–18]. Additionally, the underlying longitudinal dynamics can be simulated by numerically solving the Vlasov-Fokker-Planck (VFP) equation [19], where the CSR

* These authors contributed equally to the presented work.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUCPL07

OPTIMAL CONTROL FOR RAPID SWITCHING OF BEAM ENERGIES FOR THE ATR LINE AT BNL*

J. P. Edelen†, N. M. Cook, RadiaSoft LLC, Boulder, USA
K. A. Brown, P. Dyer, Brookhaven National Laboratory, Upton, USA

Abstract

The Relativistic Heavy Ion Collider (RHIC) at Brookhaven National Laboratory will undergo a beam energy scan over the next several years. To execute this scan, the transfer line between the Alternating Gradient Synchrotron (AGS) and RHIC or the so-called the ATR line, must be re-tuned for each energy. Control of the ATR line has four primary constraints: match the beam trajectory into RHIC, match the transverse focusing, match the dispersion, and minimize losses. Some of these can be handled independently, for example orbit matching. However, offsets in the beam can affect the transverse beam optics, thereby coupling the dynamics. Furthermore, the introduction of vertical optics increases the possibilities for coupling between transverse planes, and the desire to make the line spin transparent further complicates matters. During this talk, we will explore three promising avenues for controlling the ATR line: model based control, on-line optimization methods, and hybrid model based and optimization methods. We will provide an overview of each method, discuss the tradeoffs between these methods, and summarize our conclusions.

INTRODUCTION

The Relativistic Heavy Ion Collider (RHIC) at Brookhaven National Laboratory will undergo a beam energy scan [1] over the next several years. To execute this scan, the transfer line between the Alternating Gradient Synchrotron (AGS) and RHIC or the so-called the ATR line [2, 3], must be retuned for each energy. This transfer line controls the orbit matching, optics matching, and dispersion matching, of the beam into RHIC. The optics are further complicated by a 1.7 m vertical drop in order to get the beam from the AGS to RHIC. In order to ensure optimum performance of RHIC during the energy scan the magnets and correctors will need to be properly set on demand. A high-level diagram of the transfer line is shown in Figure 1.

The first part of the ATR (referred to as the U-line) line starts with the fast extraction from the AGS and stops before the vertical drop from the AGS to RHIC. The U-line consists of two bends. The first bend is 4.25° consisting of two A-type dipole magnets. The second bend is an 8° bend consisting of four C-type combined function magnets (placed in a FDDF arrangement), and thirteen quadrupoles.

* This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Nuclear Physics under Award Number DE-SC0019682.

† jedelen@radiasot.net

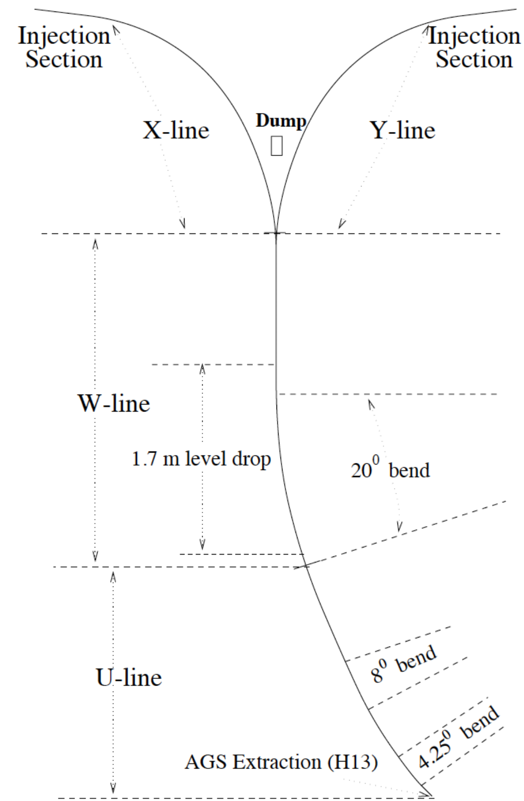


Figure 1: High level schematic of the RHIC transfer line [4].

The primary purpose of the U-line is to: 1) Match the Twiss parameters at the AGS extraction point and provide achromatic transport of the beam to the exit of the 8° bend, 2) Create a beam waist with low beta function values at the location of a thin gold foil which is placed just upstream of the quadrupole Q6 of the U-line, 3) Match the Twiss parameters of the line to the ones at the origin of the W-line, and 4) Keep the beam size small to minimize losses.

The second part of the ATR line (referred to as the W-line) introduces the vertical drop for injection into RHIC and the matching sections for the injection lines. It contains eight C-type combined function magnets that each make a of 2.5° bend, followed by six quadrupoles. The eight combined function magnets form a 20° achromatic horizontal bend placed in a (F-D) configuration. The W-Line is also responsible for lowering the beam elevation by 1.7 m. This is accomplished by two vertical dipoles referred to as pitching magnets. The first bends the beam down, and is located between the first and second combined function dipoles of the W-line. The second, which restores the beam to the horizontal level (bend-up), is located between the second and third

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUCPR01

DEVELOPING A TOOLKIT FOR ANALYSIS OF LCLS PUMP-PROBE DATA*

S. Nelson†, SLAC National Accelerator Laboratory, Menlo Park, CA, USA

Abstract

The data format and volume at LCLS requires significant computing expertise which not all user groups can provide. We will describe the path to and current status of a Python module that enables user groups to translate and reduce their data into a format that they can easily work with. The package is developed in Python and uses the standard LCLS data analysis framework. It encapsulates knowledge of the standard beam line components and adds convenient ways to reduce the data of larger detectors. Both an event-based (best for small event sizes) and a binned approach which is able to handle larger data as megapixel size detectors are simple to setup. MPI is used for fast turn around, enabling close to real time feedback necessary to make decisions of how to use the limited amount of beam time. Jupyter notebooks are provided to demonstrate some of the available options and can serve as a convenient quick start for fast turn around analysis.

ANALYSIS ENVIRONMENT AT LCLS

The requirements for the LCLS data acquisition system (DAQ) are that within each instrument, data are acquired for all devices at the beam rate of 120 Hz, tagged with the fiducial from the timing system and a UNIX timestamp. These event data are then appended to a file in eXtended tagged container (XTC) format [1]. The DAQ system is capable of reading out 5 GB/s per instrument and is described in more detail in [2].

The DAQ software was written in C++ which was also the language of the the initial analysis framework. While the learning curve to do analysis in a complex framework is standard in HEP where scientists join a collaboration and will work in a given framework for a few years up to decades, light sources are user facilities where typical users are only present for a short amount of time for beamtime that has been assigned to one of their proposals.

In order to allow groups to analyze their data without having to develop code in C++, an hdf5 [3] translation system was set up. This is accessible from the "Data Manager", the main interface to the experiment. Having the data in hdf5 format allows user groups to use a variety of programs that they are already familiar with, e.g. MATLAB, Igor or python. The translated files mirrored the structure of the original xtc files, which is not very intuitive. Each component of the data contains the values and timestamps and the user must perform timestamp matching when analyzing the data. The data had to be read sequentially in a single thread. For longer runs with mega pixel detectors, the datawas only be available for analysis a few hours after data taking.

Some of the earlier user groups had sufficient resources to develop frameworks for analysis based on the SLAC code stack and the detectors typically used. In particular for compute intensive analyses like crystallography a couple of different analysis frameworks were developed that are used to date. For other communities where the analysis was typically more less computationally challenging, this was not the case. Some groups developed analysis setups based on the LCLS analysis framework while other groups based their analysis on MATLAB code provided by beam line staff which was based on the standard translation provided by LCLS.

Hidden costs of user frameworks became visible during years of operations: the first is a technical cost for the facility when the user code needs to be adjusted for a new beam time or in rare cases, even for analysis of old data. Such changes are necessary when the LCLS analysis stack changes significantly, when common beam line components are upgraded, new physics detectors are used or a user group moves to a different experimental end station. In addition, the barriers for small, new user groups can be high if the analysis system provided for them requires more computing expertise than they have available. These groups will either not propose their experiment, feel the need to include other groups who have an LCLS-compatible analysis setup or take a long time to publish their data after the beam time assuming that their experiment allowed the LCLS-provided live analysis framework to work well enough so that the measurement plan could be followed.

REQUIREMENTS FOR GENERAL USER SUPPORT ANALYSIS PACKAGE

The requirements listed below have been taken from a few years of supporting analysis groups at XPP [4] and XCS [5] in particular, but they are common for other experiments conducted at other hutches. Crystallography experiments will typically use a different code stack:

- portable data (hdf5), small with timestamp aligned data
- no setup for standard beam line data
- simple interface to data reduction methods for larger data sources.
- option to keep full data of big detectors, but average multiple events
- data ready for analysis within a few minutes after the run has ended

The needs of staff that is supporting a significant number of user experiments during a year adds a few more requirements.

* This work was performed in support of the LCLS project at SLAC supported by the U.S. Department of Energy, Office of Science, Office of Basic Energy Sciences, under Contract No. DE-AC02-76SF00515
† † snelson@slac.stanford.edu

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUCPR02

DATA EXPLORATION AND ANALYSIS WITH Jupyter NOTEBOOKS

H. Fangohr*[1], M. Beg, M. Bergemann, V. Bondar, S. Brockhauser[2,3], C. Carinan, R. Costa, F. Dall'Antonia, C. Danilevski, J. C. E, W. Ehsan, S. G. Esenov, R. Fabbri, S. Fangohr, G. Flucke, C. Fortmann[4], D. Fulla Marsa, G. Giovanetti, D. Goeries, S. Hauf, D. G. Hickin, T. Jarosiewicz[5], E. Kamil, M. Karnevskiy, Y. Kirienko, A. Klimovskaia, T. A. Kluyver, M. Kuster, L. Le Guyader, A. Madsen, L. G. Maia, D. Mamchyk, L. Mercadier, T. Michelat, J. Möller, I. Mohacsi, A. Parenti, M. Reiser, R. Rosca, D. B. Rueck, T. Rüter, H. Santos, R. Schaffer, A. Scherz, M. Scholz, A. Silenzi, M. Spirzewski[5], J. Sztuk, J. Szuba, S. Trojanowski[5], K. Wrona, A. A. Yaroslavtsev, J. Zhu
European XFEL GmbH, Schenefeld, Germany
J. Reppin, F. Schlünzen, M. Schuh, DESY, Hamburg, Germany
E. Fernandez-del-Castillo, G. Sipos, EGI Foundation, Amserdam, Netherlands
T. H. Rod, J. R. Selknaes, J. W. Taylor, ESS, Copenhagen, Denmark
A. Campbell, A. Götz, J. Kieffer, ESRF, Grenoble, France
J. Hall, E. Pellegrini, J. F. Perrin, ILL, Grenoble, France

[1] also at University of Southampton, Southampton, United Kingdom
[2] also at University of Szeged, Szeged, Hungary
[3] also at Biological Research Center of the Hungarian Academy of Sciences, Szeged, Hungary
[4] also at Max-Planck-Inst. for Evolutionary Biology, Plön, Germany
[5] also at NCBJ, Otwock, Poland

Abstract

Jupyter notebooks are executable documents that are displayed in a web browser. The notebook elements consist of human-authored contextual elements and computer code, and computer-generated output from executing the computer code. Such outputs can include tables and plots. The notebook elements can be executed interactively, and the whole notebook can be saved, re-loaded and re-executed, or converted to read-only formats such as HTML, LaTeX and PDF. Exploiting these characteristics, Jupyter notebooks can be used to improve the effectiveness of computational and data exploration, documentation, communication, reproducibility and re-usability of scientific research results. They also serve as building blocks of remote data access and analysis as is required for facilities hosting large data sets and initiatives such as the European Open Science Cloud (EOSC). In this contribution we report from our experience of using Jupyter notebooks for data analysis at research facilities, and outline opportunities and future plans.

INTRODUCTION

Data analysis and data science studies often begin with interactive exploration, often already during the experiment. Short feedback cycles are crucial for this: the person exploring the data should be free to quickly try different data analysis options and see the results with minimal mental overhead. As the analysis progresses, the focus shifts to recording and communicating findings and how they were reached. Whether someone shares their analysis or keeps it for their own reference, they will need an explanation of what was done, and a record of the code used and the results, to establish confidence and to serve as a base for further work.

Digital notebook interfaces are an increasingly popular way to meet these needs. A notebook is a document combining free text with code which can be interactively executed, producing results inline which are saved as part of the notebook. Such interfaces have been familiar in computational mathematics software such as Mathematica and SageMath for many years. More recently, notebooks have spread to general programming and data science, in particular as Jupyter notebooks (formerly IPython notebooks) [1, 2].

Figure 1 shows an example Jupyter notebook introducing some of the capabilities. This example is available online [3], where it can also be interactively executed using the Binder service [4].

Jupyter is a notebook interface which can work with various programming languages, thanks to backends known as *kernels* which can be installed individually. Jupyter notebooks are viewed and edited through an interface in the web browser. This can be run by an individual on their own computer, but it is also relatively straightforward to provide remote access to Jupyter running on a server. Jupyter notebooks are typically used through common web browsers, but they can also be viewed and edited in a desktop application called *nteract* [5]. Very recently, a useful review on collaborative data science with Jupyter notebooks has become available [6].

In this paper, we discuss our experiences of using Jupyter notebooks at a range of research facilities and will illustrate this with use cases from European XFEL (EuXFEL) to high-

* hans.fangohr@xfel.eu

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUCPR03

OUR JOURNEY FROM JAVA TO PYQT AND WEB FOR CERN ACCELERATOR CONTROL GUIS

I. Sinkarenko, S. Zanzottera, V. Baggiolini, BE-CO-APS, CERN, Geneva, Switzerland

Abstract

For more than 15 years, operational GUIs for accelerator controls and some lab applications for equipment experts have been developed in Java, first with Swing and more recently with JavaFX. In March 2018, Oracle announced that Java GUIs were not part of their strategy anymore [1]. They will not ship JavaFX after Java 8 and there are hints that they would like to get rid of Swing as well.

This was a wakeup call for us. We took the opportunity to reconsider all technical options for developing operational GUIs. Our options ranged from sticking with JavaFX, over using the Qt framework (either using PyQt or developing our own Java Bindings to Qt), to using Web technology both in a browser and in native desktop applications.

This article explains the reasons for moving away from Java as the main GUI technology and describes the analysis and hands-on evaluations that we went through before choosing the replacement.

INTRODUCTION

The majority of operational GUI applications running in the CERN Control Centre and other accelerator control rooms have been written in Java Swing. This technology is used here since the early 2000s. The Controls Group only develops general-purpose GUIs for mission critical applications, such as the InCA/LSA settings management [2] or the Sequencer [3]. For all the other GUIs, we rely on the equipment experts and operators to develop them because they know best what these GUIs should look like, and want to flexibly adapt them. To facilitate their task, we provide a framework, which consists of an application frame with a toolbar and a logging console, a graph component (JDataViewer), and, of course, different controls-specific widgets. We also provide a comprehensive set of client APIs to interact with the control system.

In early 2016, we moved away from Swing because we saw it as a legacy technology and recommended JavaFX as the successor, after it had become an official part of the Oracle JDK. Of the 500 operational GUIs currently in use, 90% are still in Java Swing. Unfortunately, in 2018, Oracle, the company backing Java, stated in their Java Client Roadmap Update [1] that JavaFX is a “niche” technology with a “market place [that] has been eroded by the rise of mobile-first and web-first applications”, and announced that JavaFX is no longer a part of the Oracle JDK, but shall live on as an independent product, to be maintained by the open-source community. This official announcement was a wake-up call for us. We decided to completely re-evaluate our strategy and technology choices for GUI, even at the cost of not using Java – our core technology – for GUIs anymore.

CRITERIA FOR SELECTING A NEW GUI TECHNOLOGY

In our evaluation of GUI technologies, we considered the following criteria:

- Technical match: suitability for Desktop GUI development and good integration with the existing controls environment (Linux, Java, C/C++) and the APIs to the control system;
- Popularity among our current and future developers: little (additional) learning effort, attractiveness for new recruits;
- Longevity of the technology and reasonable maintenance cost medium-term to long-term.

We looked at a very broad spectrum of technology and then seriously evaluated the following options:

- Java Swing or JavaFX – continue with Java in spite of the Oracle announcement;
- Web technology – use the currently most popular GUI technology;
- Qt – use one of the most popular desktop GUI frameworks with either Java or join the exploding popularity of Python with PyQt, or adopt future-oriented QtQuick GUIs.

In our evaluation, we accepted a possible move away from Java to another high-level language, such as JavaScript/TypeScript or Python, but we have never considered moving our users or ourselves to C or C++.

Most of the content below is explained in far more detail in the Master Thesis of one of the authors [4].

ANALYSIS

This section summarizes our findings and assessment during the analysis. Please note that this evaluation was done in 2018 and refers to the situation at that moment. Also, some of the statements are based on empirical analysis and might be tainted by personal interpretations.

Java Swing or JavaFX

The technical match and integration with the controls system is (obviously) very good: as our long-standing technology, it fulfils our needs very well and has an excellent integration with the controls system. Acceptance is mixed. Those developers who know and use JavaFX typically like it very much. However, it clearly is less attractive for new recruits – very few young engineers want to learn JavaFX, let alone Swing.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUCPR04

IMPROVING USER EXPERIENCE IN COMPLEX SYSTEMS

M. J. Clarke*, G. C. Murphy, T. S. Richter, European Spallation Source, Copenhagen, Denmark
R. Nørager, Design Psychology, Copenhagen, Denmark

Abstract

Don Norman and Jakob Nielsen define User Experience (UX) as "encompassing all aspects of the end-user's interaction with the company, its services, and its products". The question is, however, is it possible to provide a significantly better UX in an inherently complex environment, such as at a neutron beamline instrument? With this in mind, we decided to ask the professionals at Design Psychology to see what might be achievable for user-facing scientific software at the European Spallation Source.

During a series of short workshops, we looked at general UX principles and how they could be applied to two of our software projects. We learned a number of useful practices and ideas, such as:

- Why UX is more than just the graphical user interface
- The value of creating user personas and mapping their workflow
- How to design for the user's "System 1"

A bad UX may make the user feel like they are fighting against the system rather than working with it. A good UX, however, will unobtrusively help them do what they need to do without fuss or bother. If done well, UX is not a zero-sum game: improvements can be made so novices and experts alike can work more efficiently.

INTRODUCTION

The European Spallation Source (ESS) [1] will be a multi-disciplinary research facility based on a next-generation world-leading neutron spallation source which will allow users from a range of scientific and engineering fields to study their materials at a level not achievable at existing neutron sources. Sweden and Denmark are the joint hosts of the ESS, with the facility itself being constructed in Lund, Sweden and the Data Management and Software Centre (DMSC) being located in Copenhagen, Denmark.

In parallel to the construction of the facility, the DMSC team have been constructing a suite of user-orientated software to assist the users in pursuing their science. This suite provides software for the whole lifetime of the experiment starting from the initial proposal submission, through performing the experiment at the facility, to analysing, storing and cataloguing the data. The software suite consists of both thin-client web applications and more traditional thick-client desktop applications; however, regardless of the technologies used, it is important that the software provides a good User Experience (UX) [2] for both expert and novice users. With this goal in mind, the DMSC asked the UX specialists at Design Psychology [3] to run a series of UX workshops. The first one was a general workshop introducing UX principles and brief examination of existing DMSC software. The second and third workshops were focussed solely on SciCat [4] and NICOS [5], which are two of the more mature applications developed at the DMSC.

The focus of this paper is on the application of UX principles to SciCat and NICOS.

About Design Psychology

Design Psychology extend the classic user centred design process by grounding it in expert knowledge of human psychology. This dual user and human centred design approach enables them to turn descriptive user insights into prescriptive design drivers. This increases the quality of UX design in projects and reduces risk. They work with companies across many business areas to help them realise the business value of improving UX. Their services cover both strategic design and design implementation. In addition, they have a UX laboratory with a diverse testing toolbox than can deliver quantified data to support design decisions.

SCICAT AND NICOS

SciCat is a catalogue for providing users with access to the scientific metadata and raw experiment data for their experiments. It covers the whole life-cycle of the experiment, from the initial proposal, through data collection and analysis to publication and beyond. The project is an open-source collaboration between the Paul Scherrer Institute (PSI) [6] in Switzerland, MAX IV [7] in Sweden and the ESS. The user interface for SciCat is web-based, built using the Angular [8] framework. The user interface is shown in Figure 1.

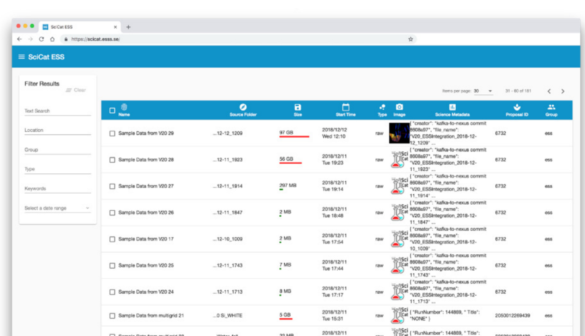


Figure 1: The SciCat user interface prior to the workshops.

NICOS is a network-based experiment control program written for neutron scattering at the Forschungs-Neutronenquelle Heinz Maier-Leibnitz (FRMII) [9] in Germany. Though originally developed for FRMII, NICOS is now being used at a number of other facilities including the ESS and SINQ (PSI). NICOS is written in Python and uses

* matt.clarke@esss.se

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUCPR05

UX FOCUSED DEVELOPMENT WORK DURING RECENT ORNL EPICS-BASED INSTRUMENT CONTROL SYSTEM UPGRADE PROJECTS*

X. Yao†, R. D. Gregory, G. S. Guyotte, S. M. Hartman, K.-U. Kasemir, C. A. Lionberger, M. R. Pearson, Oak Ridge National Laboratory, Oak Ridge, USA

Abstract

The importance of usability and easy-to-use user interfaces (UI) have been recognized across many domains. However, the user-friendliness of scientific experiment control systems often lags behind industry standards in the flourishing user experience (UX) field. Scientific control systems can certainly benefit from these new UX research methods and approaches. Recent instrument control system upgrade projects at the SNS and HFIR facilities at Oak Ridge National Laboratory demonstrate the effectiveness of UX focused development work, and further reveal the need for more utilization of such techniques coming from the UX field. The ongoing control system upgrades are targeting the key facility-level priority of higher scientific productivity, and UX is one of the important tools to help us achieve this priority. We will highlight research methods and practices, introduce our findings and deliverables, and share challenges and lessons learned in applying UX methods to scientific control systems.

INTRODUCTION

Oak Ridge National Laboratory (ORNL) is home to two world-class neutron scattering user facilities: The Spallation Neutron Source (SNS) and The High Flux Isotope Reactor (HFIR). In recent years, a series of significant software/hardware upgrade projects has progressed to improve the reliability and usability of the scientific instrument beamlines at these facilities. At the core of these upgrade projects is the application of the Experimental and Industrial Control System (EPICS) [1], which has also been used for the SNS Accelerator controls systems since the beginning of the SNS project [2]. The mature EPICS toolkit has contributed greatly to the reliable and stable operation of the SNS Accelerator and has now been applied for the many individual instrument/beamline control systems. The beamlines had experienced a variety of operational challenges under the earlier legacy data acquisition system, with both reliability and user complexity issues. Based on a careful and systematic review of existing software packages and toolkits, the decision was made in 2012 to upgrade the majority of SNS beamline control systems software to use the EPICS toolkit and Control System Studio (CS-Studio) [3][4][5]. These control system overhauls have proceeded well and have greatly improved the reliability, maintainability, and data/science throughput of the beamlines [6].

*This manuscript has been authored by UT-Battelle, LLC, under contract DE-AC05-00OR22725 with the US Department of Energy (DOE). Work supported by …This information must be within the text and
† yaox@ornl.gov

The first instance of EPICS and CS-Studio being applied to the beamline control systems was for the Imaging beamline, to demonstrate more reliable, flexible, and efficient beamline operation [7]. During the past seven years, 22 beamlines at the SNS and HFIR have been upgraded to use EPICS and CS-Studio for their beamline control systems, including 17 SNS beamlines and 5 HFIR beamlines. These upgrade projects were quite successful and have subsequently freed up many resources within our group, enabling us to focus more on improving the overall user experience (UX) and ease-of-use of the user interfaces (UIs) of our beamline control systems. The past four years saw an increased number of specific UX/UI deliverables within our group, which were welcomed by our diverse scientific user community.

Our recent UX/UI deliverables have been designed and developed by applying the methods and practices of the UX [8] and Design Thinking [9] field. These deliverables empower new and external users to effectively and efficiently set up an experiment, plan and collect data, monitor experiment progress, and make informed decisions along the way. Meanwhile, the increased reliability of the control systems has simultaneously relieved the instrument staff from constant hands-on operational support. Some fundamentals of the UX/UI methodologies and practices are described in more detail in the following sections, followed by some specific examples of their use in the development of our scientific beamline control systems.

EFFECTIVE METHODS AND PRACTICES

UX and the Design Thinking process provide a new way of thinking, seeing, and doing development work based on previous research emphases, such as Human Computer Interaction (HCI), UI, User-Centered Design (UCD), and Usability. By adopting UX methods to look at the entire experience of a user interacting with our beamline control systems, we have identified opportunities for improvement, and explored solutions beyond simple user interfaces. Our goal is to decrease our users' physical effort (such as a mouse move/click and typing), mental effort (such as remembering and thinking), and emotional effort (such as perceived task difficulty), and to help them complete their tasks with a more delightful, enjoyable, and therefore productive experience.

Similarly, the Design Thinking process has proven to likewise be effective with its well-established steps, including empathize, define, ideate, prototype, and test [10]. In practice, we have discovered that the iterations of this process are more like a "spiral staircase" than that of a linear

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUCPR06

FAST INTERACTIVE PYTHON-BASED ANALYSIS OF STREAMED IMAGES*

A. Sukhanov†, W. Fu, J. P. Jamilkowski, R. H. Olsen
Brookhaven National Laboratory, Upton, USA

Abstract

This paper reports on development of a general purpose image analysis application, tailored for beam profile monitor cameras of RHIC Collider-Accelerator complex. ImageViewer is a pure Python application, based on pyqtgraph and scipy packages. It accepts image streams from camera servers (RHIC or EPICS), local cameras or from file system. The standard analysis includes recognition of connected objects; for each object the parameters of a fitted ellipsoid (position, axes and tilt angle) are calculated using 2nd-order image moments, the parameters are corrected using 1D or 2D gaussian fit. Many other features are supported: saving, image rotation, region of interest, projections, subtraction of a reference image, multi-frame averaging, color mapping, contrast control, pixel to millimeter calibration. Playback feature allows for fast browsing and clean-up of saved images. User add-ons can be added dynamically as included modules. The existing add-ons provide turn-key solutions for moving-slit and multi-slit beam emittance measurements, hollow-beam characterization and others.

Each camera of the RHIC complex is equipped with a server (graphic-less) version of this application, providing the same analysis and publishing calculated parameters to RHIC Controls Architecture.

INTRODUCTION

The imageViewer is a general purpose image analysis application, written in pure Python, it is based on pyqtgraph [1] and scipy [2] packages. The pyqtgraph is a fast Python-based visualization package that makes use of Qt GraphicsView for primitive graphics (lines, shapes, antialiasing) and is capable of rendering 1024x1024 video at 50 fps.

Everything is controlled from a single window as shown on Fig. 1, the window is divided into adjustable panes:

- Image pane,
- Control pane with parameter tree of GUI elements (buttons, checkboxes, spinboxes, sliders, etc.),
- Message pane,
- Console pane, a Python interactive console where users can enter python expressions/statements and access local objects for debugging.

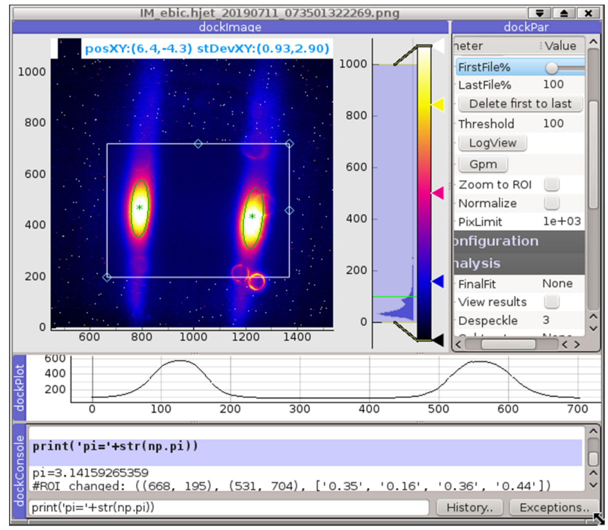


Figure 1: imageViewer window.

FEATURES

Image Delivery

The imageViewer is an event-driven application, the input back-end raises an event for the main thread to indicate arrival of new image.

The back-ends for following image sources are integrated:

- Camera server (RHIC Controls). The image is either PNG-compressed (default) or byte array.
- EPICS Area Detector, image is byte array.
- File system.
- Local USB camera.
- HTTP.

The default file format is PNG; it handles color depths higher than 8 bits. Other formats are supported as well.

Asynchronous delivery of large amounts of data from camera servers may cause buffer overloading of TCP stacks on the server or client side (when image size is large and client is slow to receive or process data). To throttle the input stream, the imageViewer asynchronously requests a small scalar parameter which is updated when a new image is ready, if imageViewer is busy processing previous images, then the request is dropped, otherwise the image array is received synchronously.

Rotation and Orientation

Rotation to an arbitrary angle and flipping can be applied to all images in the stream or interactively.

* Work supported by Brookhaven Science Associates, LLC under Contract No. DE-SC0012704 with the U.S. Department of Energy.
† sukhanov@bnl.gov

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUCPR07

HIGH-LEVEL PHYSICS CONTROLS APPLICATIONS DEVELOPMENT FOR FRIB *

T. Zhang†, D. Maxwell, K. Fukushima, M. Ikegami and P. Ostroumov
Facility for Rare Isotope Beams, Michigan State University, East Lansing, USA

Abstract

For the accelerators driven by EPICS distributed control system, controls engineers solve the problem to make the devices work, while accelerator physicists dedicate themselves to make the machine run as the physics predicted. To fill the gap between the high-level physics controls and the low-level device controls, we developed a software framework so-called `phantasy` that can help the users like accelerator physicists and operators, to work well with the machine in an object-oriented way, based on which the implementations for the physics tuning algorithms could be very efficient, understandable and maintainable. Meanwhile, the modularized UI widgets are developed to standardize the high-level GUI applications development, to greatly reuse the codebase and ease the development. The most important thing is all the development also applies to other EPICS based accelerators. In this paper, the design and implementation for both interactive Python scripting controls and high-level GUIs development will be addressed.

INTRODUCTION

The driver LINAC of Facility for Rare Isotope Beams (FRIB) can accelerate all the stable isotopes to the kinetic energy higher than 200 MeV/u, deliver the energy of up to 400 kW on the end target, which will be more than two orders advancement in the heavy ion accelerators regime [1]. To achieve such goal, reliable and sophisticated applications for machine tuning should be ready for daily operation use.

Generally, high-level physics control is about controlling the accelerator with physics algorithms. The purpose is to apply the physics solution to the machine, and to expect the physics predicted machine behavior.

From the view of controls aspect, for the EPICS [2] driven facility, the entire machine is composed of many different kinds of devices, each of them is controlled by Input-Output-Controller (IOC) [3]. While all the IOCs are distributed in the same ether network, e.g. FRIB Controls Network, the data communication among these IOCs and any other client with the network access is defined by Channel Access (CA) protocol [4].

For client application implemented with a different programming language, specific software is required to be able to speak CA 'language'. For instance, `PyEPICS` [5] is the Python interface to CA.

* Work supported by the U.S. Department of Energy Office of Science under Cooperative Agreement DE-SC0000661, the State of Michigan and Michigan State University.
† zhangt@frib.msu.edu

On the other hand, accelerator physicists care more about the physics property and behavior, e.g. the magnetic field of a dipole, the gradient of a quadrupole, the simulated beam trajectory along with the accelerator, etc.

To bridge the device control and machine tuning on accelerator facility, systematic design of the software infrastructure for high-level physics controls is required.

The software framework should be able to help physicists establish a software environment for tuning algorithms development, here is the list of key problems to be resolved:

- Device control should be simple and easy enough to understand
- Implement physics algorithm should be convenient and maintainable
- Quick developing and testing should be supported

At FRIB, Python-based software solution for the high-level physics controls has been shaping gradually during the past few years, the project is named as `phantasy`, which stands for Physics High-level Applications aNd Toolkit for Accelerator SYstem [6]. Based on `phantasy`, various physics high-level applications are developed and deployed to FRIB Controls Network for the efficient beam commissioning. The next few sections go with the details about the development.

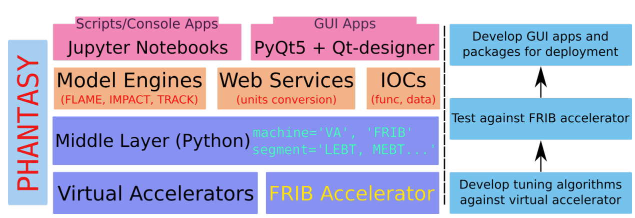


Figure 1: Development workflow of physics application with `phantasy`.

SOFTWARE FRAMEWORK OF PHANTASY

`phantasy` is designed to support quickly developing physics tuning algorithms on EPICS-based accelerator controls system. It is expected that accelerator physicists with Python knowledge could write scripts to control the accelerator by properly importing and using functionality provided by `phantasy`. Figure 1 shows the typical physics application development workflow. The virtual accelerator application which is part of the essential `phantasy` toolkit could be used with algorithm prototyping, once the algorithm is developed, the same script can test against FRIB accelerator

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUDPP01

A MONITORING SYSTEM FOR THE NEW ALICE O2 FARM

G. Vino†, D. Elia, INFN, Bari, Italy
V. Chibante Barroso, A. Wegrzynek, CERN, Geneva, Switzerland

Abstract

The ALICE Experiment has been designed to study the physics of strongly interacting matter with heavy-ion collisions at the CERN LHC. A major upgrade of the detector and computing model (O2, Offline-Online) is currently ongoing. The ALICE O2 farm will consist of almost 1000 nodes enabled to readout and process on-the-fly about 27 Tb/s of raw data.

To increase the efficiency of computing farm operations a general-purpose near real-time monitoring system has been developed: it lays on features like high-performance, high-availability, modularity, and open source. The core component (Apache Kafka) ensures high throughput, data pipelines, and fault-tolerant services. Additional monitoring functionality is based on Telegraf as metric collector, Apache Spark for complex aggregation, InfluxDB as time-series database, and Grafana as visualization tool.

A logging service based on Elasticsearch stack is also included. The designed system handles metrics coming from operating system, network, custom hardware, and in-house software. A prototype version is currently running at CERN and has been also successfully deployed by the ReCaS Datacenter at INFN Bari for both monitoring and logging.

INTRODUCTION

The ALICE Experiment

ALICE (A Large Ion Collider Experiment) [1] is a detector designed to study the physics of strongly interacting matter (the Quark–Gluon Plasma), produced in heavy-ion collisions at the CERN Large Hadron Collider (LHC). ALICE consists of a central barrel and a forward muon spectrometer, allowing for a comprehensive study of hadrons, electrons, muons and photons produced in the collisions of heavy ions. The ALICE collaboration also has an ambitious physics program for proton–proton and proton–ion collisions. After the successful Run 1 (2010-2013) and Run 2 (2015-2018) data taking periods, the LHC entered into a consolidation phase (Long Shutdown 2) and ALICE started its upgrade to fully exploit the increase in luminosity expected in Run 3. The upgrade foresees a complete replacement of the computing systems (Data Acquisition, High-Level Trigger and Offline) by a single, common O2 (Online-Offline) system.

The ALICE O2 System

The ALICE O2 computing system [2] will allow the recording of Pb–Pb collisions at a 50 kHz interaction rate.

† gioacchino.vino@cern.ch

Some detectors will be read out continuously, without physics triggers. Instead of rejecting events the O2 system will compress the data using online calibration and partial reconstruction. The first part of this process will be done in dedicated FPGA cards that receive the raw data from the detectors. The cards will perform baseline correction, zero suppression, cluster finding and inject the data into the memory of the FLP (First Level Processors) to create a sub-timeframe. Then, the data will be distributed over EPNs (Event Processing Node) for aggregation and additional compression. The O2 facility will consist of 200 FLPs and 750 EPNs. The O2 farm will receive data from the detectors at 27 Tb/s, which after FLP and EPN processing will be reduced to 720 Gb/s.

MONITORING SYSTEM OBJECTIVES

The Monitoring subsystem is part of O2 and provides comprehensive functionality in metric collection, routing, processing, storage, visualization and alarming as shown in Fig. 1.

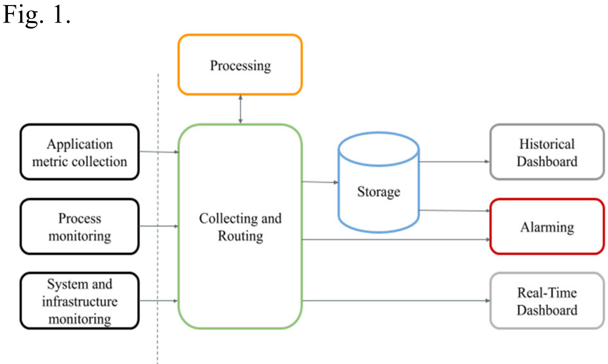


Figure 1: Functional architecture of the system.

Three classes of metrics (application, process and system/infrastructure) are collected and pushed to the Collecting and Routing backend. Metrics requiring processing are forwarded to the Processing component that injects back the processed values. Then, all the values are written into permanent storage. From that point they can be browsed and visualized in the historical record dashboard. Selected metrics are published for alarming and real-time visualization.

System and Infrastructure Monitoring

The System monitoring provides probes to various operating system metrics, for example:

- CPU
- Memory
- Network
- Storage
- Hardware status

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.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUDPP03

IMPROVEMENT OF EPICS SOFTWARE DEPLOYMENT AT NSLS-II

A. A. Derbenev†, Brookhaven National Laboratory, Upton, USA

Abstract

The NSLS-II Control System has workstations and servers standardized to the usage of Debian OS. With exceptions like RTEMS and Windows systems where software is built and delivered by hand, all hosts have EPICS software installed from an internally-hosted and externally-mirrored Debian package repository. Configured by Puppet, machines have a similar environment with EPICS base, modules, libraries, and binaries. The repository is populated from epicsdeb, a community organization on GitHub. Currently, packages are available for Debian 8 and 9 with legacy support being provided for Debian 6 and 7. Since packaging creates overhead on how quickly software updates can be available, keeping production systems on track with development is a challenging task. Software is often customized and built manually to get recent features, e.g. for AreaDetector. Another challenge is services like GPFS which underperform or do not work on Debian. Proposed improvements target keeping the production environment up to date. A detachment from the host OS is achieved by using containers, such a Docker, to provide software images. A CI/CD pipeline is created to build and distribute software updates.

SYSTEM OVERVIEW

The NSLS-II control system is built on Experimental Physics and Industrial Control System (EPICS) infrastructure with a typical controls application being created as an Input-Output Controller (IOC) [1]. In addition to IOCs which are meant to communicate with hardware and other IOCs to implement control logic and functions, the software suite includes a diverse set of higher-level tools, services, libraries, command line and graphical interface applications. Examples are Channel Access command line tools and Python interface, Archiver Appliance, Control System Studio and Phoebus, Olog, Alarm Server, MASAR. While there can be several dozen IOCs per beamline to serve hardware integration and automation needs, tools usually come one installation per workstation, and services come one per beamline.

Controls applications are typically built and run in a specific development and runtime environment. Servers which run IOCs are standardized to use Debian operating system (OS). EPICS base, modules, and tools necessary for IOC development and operation are delivered as Debian packages available from the NSLS-II repository maintained by NSLS-II Controls [2]. EPICS source code is not "debianized" by default and is converted to package format on GitHub thanks to collaboration efforts [3]. When an IOC system is configured, apt package sources are specified appropriately, and a set of default packages is installed via usage of Puppet.

With the development environment made available and any special dependencies manually installed (e.g. vendor-supplied libraries for hardware), IOC systems become ready for building and running EPICS applications. A typical IOC is manually checked out from the internal GitLab or Mercurial repository, built in-place, and registered to run in the system via the sysv-rc-softioc utility. The manage-iocs toolkit provided by the utility serves as a uniform and standard way of running production IOC instances, and provides essential features like detached console access, logging, run/stop/restart control, and status reporting. The approach to application delivery is hence manual, limited to application level, and is version control system (VCS) based for deployment and change management.

KEY CONSIDERATIONS

A multitude of software is involved in running the machine, and development of controls applications is constantly ongoing as updates become available and new controls integration and automation needs emerge. It is prudent to make sure that approaches to software delivery for NSLS-II Controls stay current with evolving technology and requirements. A well-understood and standardized solution brings many benefits from reduced costs of systems scaling and replication to ease of continuous maintainability to long-term sustainability of Controls software infrastructure.

Whatever the approach proposed, it should aim to respect the multitude of solutions, practices, and mechanisms currently employed for NSLS-II Controls applications delivery. When considering any kind of standardization, it is important to recognize that Controls environment is often shared by different developer groups, and many stakeholder parties have their interest in the approach which is to be set as standard. Service developers and maintainers, IOC developers, tool developers, beamline staff, IT, etc. contribute to the evaluation of existing and proposed solutions and make sure that critical needs are met. Several considerations were identified.

Scalability

With NSLS-II Controls spanning over accelerator systems and over two dozen beamlines, it is important for the software delivery approach to be flexible and applicable for all environments which need to be supported. The solution should resolve facility-scale software delivery needs by design, as made possible by an existing set of control system standards which make controls environment mostly uniform across beamlines. Practical example of this consideration is implication that hundreds of IOCs and other apps will need to be managed eventually.

† aderbenev@bnl.gov

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUDPP04

DATA ACQUISITION AND VIRTUALISATION OF THE CLARA CONTROLS SYSTEM*

R. F. Clarke †, G. Cox, D. M. Hancock, P. W. Heath, B. G. Martlew, A. Oates, P. H. Owens, W. Smith, J. T. G. Wilson, S. Kinder, N. Knowles, STFC Daresbury Laboratory, Warrington, UK

Abstract

STFC Daresbury Laboratory is developing a novel Free Electron Laser (FEL) test facility focused on the generation of ultra-short photon pulses of coherent light with high levels of stability and synchronisation. The first phase of the Compact Linear Accelerator for Research and Applications, CLARA was successfully used to produce electron beam used for experiments during 2018 to 2019. It is planned that CLARA will continue to be exploited for user experiments during development. The next user period will start early in 2020. Work on the EPICS Archiver Appliance, Virtual Accelerator and a mid-level interface were successfully trialled during the last run.

The last run showed new requirements that CLARA will place on the controls system. There was heavy use of data archiving, the request to duplicate the Virtual Accelerator for CLARA simulations and rapid expansion of EPICS soft IOCs for a mid-level interface. This motivated a review of the control system's current infrastructure for the future of CLARA.

Clara's current control system has been tuned for production and is very stable with most services running for years without issue. If virtualisation is to be embedded into this production system it must be stable, recover automatically in the event of power outages and be easy to diagnose.

This paper will discuss how virtualisation of certain services can harden the current control systems against failure. It will discuss how it provides flexible services for users and sandbox environments for developers. It will discuss various tests and challenges in integrating virtualisation into the current control system. Finally, it will discuss how it will act as a backbone to novel projects such as the Accelerator Physics group's machine learning program.

INTRODUCTION

The controls system is designed around a common framework provided by EPICS [1]. EPICS provides a common interface to disparate subsystems via process variables (PVs). PVs generally form the backbone to GUIs for operation of the machine in the control room. With CLARA this landscape is changing.

Due to the complexity, timing constraints and numerous subsystems, CLARA [2] will require a mid-layer level of automation between the experimental and machine level controls.

Traditionally, scripts would be developed by users on the client side to steer EPICS at this level. This is hard to maintain and expand and as these custom scripts become more numerous. They also open the possibility of race conditions if one script is trying to drive PVs in use by another. Errors like this are notoriously hard to trace.

The controls group is working in collaboration with the accelerator physics group to develop a managed layer of abstraction between high level physics and machine operations. This abstraction has been called the mid-level interface. Solutions include adopting a common naming by using an abstract device defined at the physics level with simple states and expanding that to control all the different and related subsystems attached to it. Another solution is to identify and move all critical client-side scripts into the EPICS Input Output Controllers, (IOCs). IOCs developed as part of the mid-level interface exists only to drive EPICS PVs and in this way group together IOCs that drive hardware. The IOCs developed for physics use are called interface IOCs. Interface IOCs can become complicated as they they make heavy use of state machines to drive all the operations and manage the the hardware IOCs to get them into the state expected by the physics operation.

Interface IOCs are used to automate the RF gun conditioning system and deliver a common interface to drive different YAG screens and magnet IOCs controllers over the whole CLARA. Interface IOCs are considerably more stable and easier to implement than client-side scripts.

IOCs for hardware control are currently hosted on dedicated servers. As the mid-level interface starts to accumulate interface IOCs for different use cases, and as they are not limited by hardware, their numbers could rapidly expand and consume servers and rack-space. Virtualising these software interfaces will allow the controls group to contain their proliferation to a couple of hosts and offer the accelerator physics group the flexibility of designing the mid-level interface with minimal constraints.

This trend of integration at a physics level to the machine operation will require more than simple PV data being passed between the Accelerator Physics group (AP) and the controls system. EPICS Version 7 [3] is being investigated as a solution to add structured data to PVs. Controls data will be fed to, and feed from, the AP group's data acquisition and simulations databases. The development of PVs to provide physics engineering units for magnets PVs is a case for EPICS V7. These units are calculated in EPICS from magnet fitting curves provided by AP group but depend on the state of the machine. These are set in soft IOCs monitoring real magnets. The output of these PV are used to update the AP group's database, a set of YAML configuration files [4] that are then used for simulations. Using EPICS V3 PVs for this operation shows that PVs would be better grouped into

* STFC Daresbury Laboratory
† rory.clarke@stfc.ac.uk

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEAPP01

OLD AND NEW GENERATION CONTROL SYSTEMS AT ESA

M. Pecchioli, ESA/ESOC, Darmstadt, Germany

Abstract

Traditionally Mission Control Systems for spacecraft operated at the European Space Operations Centre (ESOC) of the European Space Agency (ESA) have been developed based on large re-use of a common implementation covering the majority of the required functions, which is referred to as mission control system infrastructure. The generation currently in operations has been successfully used for all categories of missions, including many commercial ones operated outside ESOC. It is however expected that its implementation is going to face obsolescence in the coming years, thus an ambitious Project is currently on-going aiming at the development and operational adoption of a new generation. The resulting infrastructure capitalizes as much as possible on the European initiative (referred to as EGS-CC, see [1]) which is progressively developing and delivering a modern and advanced platform forming the basis for any type of monitoring and control applications for space systems.

This paper is going to provide a technical overview of the various generations of the mission control infrastructure at ESOC, highlighting the main differences from technical and usability standpoints, thus describing the main lines of long-term evolution.

INTRODUCTION

Background

The operations of space assets are conducted at the European Space Operations Centre (ESOC) via the so called Mission Control System (MCS). This system therefore plays a central role in the Operational Ground Segment, which consists of the hardware and software based systems located on ground and integrated together in order to support the necessary interaction with the space segment on the one side and with other ground segments (e.g. launcher segment, payload data processing) on the other side. The main functions of the Operational Ground Segment are shown at conceptual level in the Fig. 1 below.

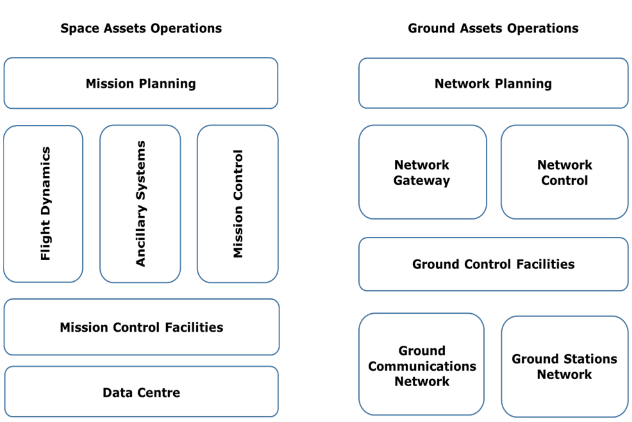


Figure 1: Operational Ground Segment (Conceptual).

It should be noted that the basic needs of the mission operators have not radically changed since the early days when spacecraft operations were first conducted. The main functions of the Mission Control systems have of course evolved but rather as a consequence of the ever increasing complexity imposed by the more and more challenging missions which need to be supported. However, the implementation of the Mission Control System infrastructure and of associated systems has been subject to much more radical evolution, primarily driven by the following factors:

- Spacecraft orbits: they dictate the frequency and the type of ground/space communication passes which can be supported. Originally only orbits enabling very long (if not continuous) visibility/contact as well as very short communication delays were supported. Nowadays the vast majority of missions rely on orbits which either provide intermittent visibility based on short passes (e.g. polar near Earth orbits) or require very long communication delays to receive and transmit the space/ground signals (e.g. deep space orbits);
- Space assets design: this is relevant to the ground systems design at several levels. The main ones affecting the mission control are: i) the protocol used to exchange telemetry (downlink) and telecommand (uplink) data; ii) the capability (generally referred to as 'monitoring and control services) supported by on-board functions which can be accessed by ground to execute mission operations; iii) volume and rates of the data exchanged with ground, in particular to download housekeeping information and payload products and iv) the level of autonomy or conversely dependence on ground operations to achieve the mission objectives;
- Operations concepts: originally mission operations strongly relied on the expertise of the ground operators and on their ability to manually execute the necessary operations. The ever increasing complexity of the mission operations as well as the necessity to minimise the associated costs during the routine phases (which heavily influence the overall costs because of the typically very long mission durations) have pushed for more modern approaches relying on a higher level of interaction (e.g. executing procedures rather than sending individual commands), on a higher level of automation (operations executed by ground applications rather than human operators) as well as a on higher level of on-board autonomy (e.g. time or event driven operations autonomously executed on-board);
- Software technologies: quite obviously the development of Mission Control Systems relies on off-the-

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEAPP02

MODERNIZATION CHALLENGES FOR THE IT INFRASTRUCTURE AT THE NATIONAL IGNITION FACILITY*

A. Casey, P. Adams, M. Christensen, E. Ghere, N. Spafford, M. Srirangapatanam, K. Tribbey, K. White, R. Vadlamani, D. Yee, Lawrence Livermore National Laboratory, USA

Abstract

As the National Ignition Facility (NIF) enters its second decade of full-scale operations, the demands on all aspects of the Information Technology (IT) infrastructure are becoming more varied, complex and critical. Cyber security is an increasing focus area for the NIF&PS IT team with the goal of securing the data center whilst providing the flexibility for developers to continue to access the sensitive areas of the controls system and the production tools. This must be done whilst supporting the interoperability of controls system elements executing on legacy bare metal hardware in an increasingly homogenized virtual environment in addition to responding to the user's requests for ever-increasing storage needs and the introduction of cloud services. While addressing these evolutionary changes, the impact to continuous 24/7 Shot Operations must also be minimized. The challenges, strategies and implementation approaches being undertaken by the NIF&PS IT team at the NIF to address the issues of infrastructure modernization will be presented.

DEFINING THE PROBLEM

With NIF routinely executing over 400 hundred experiments annually, there seems to be little issue with the IT infrastructure used to run NIF. However, ever since 2009 when NIF transitioned to a user facility, technologies have been changing at ever faster rates and the infrastructure has not been able to keep pace with them.

What was state of the art in 2009 is often considered obsolete in 2019. For example, in 2009 servers based on the Solaris 10 OS comprised most of the compute power in the Data Center but today it is difficult to find System Admins with the skillset to administer those hosts.

Secondly, NIF is first and foremost a research facility and maximizing operational availability is key to meeting the shot rate goals. However, this does conflict with the ability of the IT team to make the necessary evolutionary changes to the infrastructure in order to keep the facility operating as securely and as efficiently as possible.

REVISIT THE DESIGN

Before starting the process of updating the infrastructure, the decision was made to validate the current IT infrastructure against a set of standards that could be used to benchmark the current state and thus form the basis of an implementation gap analysis. As cyber security has become a critical part of IT operations and the risks highlighted by publicized incidents such as the Stuxnet worm and the City of Baltimore ransomware attack, the Center for Internet Security (CIS) Top 20 Controls and Resources [1] were chosen to provide the security framework.

IT Data Management

The basic CIS controls stipulate the need to actively manage all of the enterprise's IT assets through inventory control and configuration management. By analyzing the data produced by the IT tools monitoring aspects of the system, such as network traffic for example, making investments in new tools where data was not available, and then aggregating it all in a vizualizations tool, it was possible to develop a view into the infrstructure that could be then used to guide the next steps in the modernization process.

Industry Standards

Over the last 10 years, the NIF has evolved as operational needs have changed and as technologies have been introduced to improve performance and to make it easier for users – both internal and external – to do their work on a day-to-day basis.

Verifying that these changes do not break any of the standards utilized when the original infrastructure design was laid out, and that they are still applicable ten years later, was the next step in the process.

As the NIF's IT infrastructure encompasses Industrial Controls Systems (ICS) is part of the supervisory control and data acquisition (SCADA) network, a critical part of this effort was to ensure that the network design still conforms to the Purdue Enterprise Reference Architecture (Fig. 1). The goal has been to ensure that the paths from the institution network (level 5) to the controls themselves (level 3 down to level 0) are understood and its configuration managed. This addresses the concern that over time, "configuration drift" has added connection paths that are not desired.

Aside verifying design standards, effort was spent on ensuring that hardware and software was installed and configured as close to manufacturer recommended configurations as could be achieved. For example, a significant number of servers are based on the CISCO Unified Computing System (UCS) [2] platform and the installation was done in accordance with Cisco Validated Design (CVD) to ensure maximum compatibility with operating system, database and storage components.

Risk Assessments

The final part of the prcoess was to reflect on what had been learned previously. A lot of information was gathered in validating the infrastructure design and that data was then used to create a risk model for each of the CIS

* This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344, Release # LLNL-CONF-790677

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEAPP03

CONVERTING FROM NIS TO RED HAT IDENTITY MANAGEMENT*

T. S. McGuckin, R. J. Slominski,
Thomas Jefferson National Accelerator Facility, Newport News, USA

Abstract

The Jefferson Lab (JLab) accelerator controls network has transitioned to a new authentication and directory service infrastructure. The new system uses the Red Hat Identity Manager (IdM) as a single integrated front-end to the Lightweight Directory Access Protocol (LDAP) and a replacement for NIS and a stand-alone Kerberos authentication service. This system allows for integration of authentication across Unix and Windows environments and across different JLab computing environments, including across firewalled networks. The decision making process, conversion steps, issues and solutions will be discussed.

INTRODUCTION

For more than a decade the JLab Accelerator Computing Environment (ACE) network relied on a combination of Network Information System (NIS) and custom scripts to manage user accounts, groups, aliases, ssh host keys and sudo rules. This was a legacy configuration used to support access across multiple, differing and aging platforms (first HP-UX, then Sun Solaris and finally Red Hat Linux).

Several security, scalability, and interoperability limitations make NIS undesirable. NIS's password hashing only pays attention to the first eight characters (further characters are ignored). Directory lookups can suffer from scalability problems with NIS because a client query to an NIS server results in the entire database being transferred to the client upon which the client must then filter out the portion of the database (file) of interest. With LDAP directory service lookups filtering is done server-side and only the portion of data requested is returned. Integrating off-the-shelf software and network devices with NIS often requires scripting whereas integration with LDAP servers is nearly ubiquitous. Finally, NIS is end-of-life and no longer actively maintained. An updated network service solution was needed.

A first step in this upgrade process was performed in 2012 when password storage and authentication was moved to a Kerberos database. Following the success of this work a more complete directory services solution was sought.

Upgrades and/or retirements of old architectures also allowed ACE to update to a more modern/secure service without having to deal with a lot of legacy/backward compatibility. Finally, a solution was needed that could integrate many of ACE's disparate services into one management interface (Fig. 1).

* This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Nuclear Physics under contract DE-AC05-06OR23177.

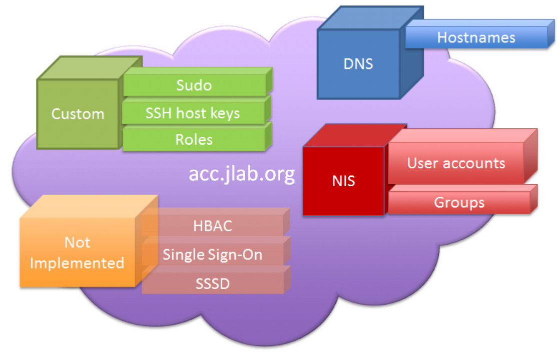


Figure 1: ACE's services spread out across multiple solutions (or not implemented) prior to upgrade.

In investigating a solution it was important to consider that JLab's ACE network (acc.jlab.org) exists as a firewalled sub-domain of the lab's larger network (jlab.org) which is maintained by the separate Computing & Networking Infrastructure (CNI) group (Fig. 2). A core requirement for the ACE network is that it be able to continue operations even if connection with the CNI network was lost. Any solution that was implemented therefore had to be able function in a stand-alone capacity and also be able to asynchronously duplicate user and host information from CNI's servers automatically.

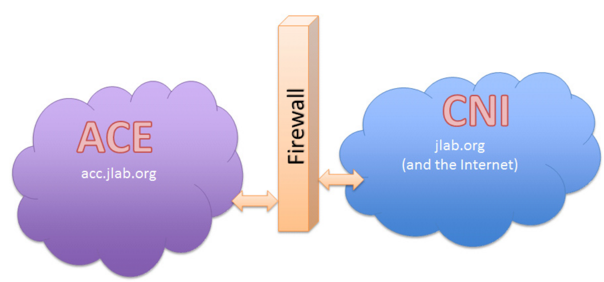


Figure 2: ACE-network relation to JLab domain.

To this end two main solutions were investigated: open source LDAP and Microsoft Active Directory. Active Directory was ultimately eliminated as an option because almost all ACE systems were Red Hat/Linux architecture with only a handful of Windows systems. Therefore, investigation began on an LDAP solution but progress was slow due to difficulties in configuration and maintenance of a bare LDAP installation. The solution that was determined to be the best fit for ACE's environment was found to be Red Hat's Identity Management software suite.

Red Hat Identity Management (IdM) is a suite of tools available as part of Red Hat Enterprise Linux (RHEL) distributions for setting up a collection of multi-master replicated clustered network domain controllers which provide services similar to Active Directory, but for all platforms, including Linux. The services provided are collectively

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEAPP04

ICS INFRASTRUCTURE DEPLOYMENT OVERVIEW AT ESS

B. Bertrand*, S. Armanet, J. Christensson, A. Curri, A. Harrisson, R. Mudingay, ESS, Lund, Sweden

Abstract

The ICS Control Infrastructure group at the European Spallation Source (ESS) is responsible for deploying many different services. We treat Infrastructure as code to deploy everything in a repeatable, reproducible and reliable way. We use three main tools to achieve that: Ansible (an IT automation tool), AWX (a GUI for Ansible) and CSEntry (a custom in-house developed web application used as Configuration Management Database). CSEntry (Control System Entry) is used to register any device with an IP address (network switch, physical machines, virtual machines). It allows us to use it as a dynamic inventory for Ansible. DHCP and DNS are automatically updated as soon as a new host is registered in CSEntry. This is done by triggering a task that calls an Ansible playbook via AWX API. Virtual machines can be created directly from CSEntry with one click, again by calling another Ansible playbook via AWX API. This playbook uses proxmox (our virtualization platform) API for the VM creation. By using Ansible groups, different proxmox clusters can be managed from the same CSEntry web application. Those tools give us an easy and flexible solution to deploy software in a reproducible way.

INTRODUCTION

The Integrated Control System Division (ICS) is an organisational unit responsible for the control systems within the European Spallation Source (ESS) facility, including control systems for accelerator, target, neutron scattering systems and conventional facilities. Within ICS, the Control System Infrastructure group is in charge to design, implement and operate the IT infrastructure needed to reliably run the Experimental Physics Industrial Control System (EPICS) eco-system. As such we have a large number of networks, physical devices and virtual machines to administer. To make this task manageable by a limited team, we try to put a lot of automation in place and treat infrastructure as code to make deployment repeatable, reproducible and reliable. We rely upon an internal GitLab [1] server to store all our code and JFrog Artifactory [2] to host binary artifacts. We do extensive use of GitLab's integrated CI/CD pipelines for continuous integration. Our deployment workflow is built on Ansible [3] (an IT automation tool), AWX [4] (a GUI for Ansible) and CSEntry [5] (a custom in-house developed web application).

ANSIBLE

Ansible is an open-source configuration management, orchestration and application deployment tool. Its main goals are simplicity and ease-of-use. It was developed by Michael DeHaan and acquired by Red Hat in 2015.

* benjamin.bertrand@esss.se

Ansible Concepts

Ansible is **agentless**. Only Python and OpenSSH are required on the **managed nodes**, the servers or network devices you want to manage. The list of nodes that can be accessed is defined in an **inventory**, which can be in different format (YAML or ini). The inventory is also used to organize the hosts in different groups as shown in Fig. 1.

```
[lbservers]
loadbalancer1

[webservers]
web1
web2
```

Figure 1: Ansible inventory.

An Ansible **playbook** is a **YAML** file describing an ordered lists of **tasks** that are mapped to a group of hosts, as illustrated in Fig. 2.

```yaml
- hosts: all
  become: true
  tasks:
    - name: install chrony
      package:
        name: chrony
        state: present

    - name: ensure chrony is running and enabled
      systemd:
        name: chronyd
        state: started
        enabled: true
```

Figure 2: Ansible playbook.

Each task performs an action using a **module**, the unit of code Ansible executes. Each module has a particular use. In Fig. 2 example, the *package* module installs chrony using the Operating System package manager (yum on CentOS) and the *systemd* module starts and enables the chronyd daemon. Those tasks are performed on the *all* group, a default group that refers to all the hosts defined in the inventory. The name of the task provides a human readable description that is displayed by Ansible when running it. As you can see, playbooks are quite easy to read even for people not familiar with Ansible.

Ansible provides a large number of modules to work with files, database, package managers and even network devices. There is probably a module for the action you want to perform. If not, you can always fallback to the *command* module or even write your own. All Ansible modules are designed to be **idempotent**, meaning that the second time a task is

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEBPP01

CONTROL SYSTEM DEVELOPMENT AND INTEGRATION AT ELI-ALPS*

L. Schrettner†, B. Bagó, B. Erdohelyi, L.J. Fulop, Z. Héjja, F. Horvath, Sz. Horváth, G. Kávai, V. Kurusa, ELI-ALPS, Szeged, Hungary

Abstract

ELI-ALPS will be the first large-scale attosecond facility accessible to the international scientific community and its user groups. Control system development has three major directions: vacuum control systems, optical control systems, as well as the integrated control, monitoring and data acquisition systems. The development of the systems has asked for different levels of integration. In certain cases low-level devices are integrated (e.g. vacuum valves), while in other cases complete systems are integrated (e.g. the Tango interface of a laser system). This heterogeneous environment is managed through the elaboration of a common and general architecture. Most of the hardware elements are connected to PLCs (direct control level), which are responsible for the low-level operation of devices, including machine protection functions, and data transfer to the supervisory control level (CLIs, GUIs). Certain hardware elements are connected to the supervisory layer (cameras), as well as the Tango interface of the laser systems. This layer handles also data acquisition with a special focus on the metadata catalogue.

INTRODUCTION

Extreme Light Infrastructure (ELI) is the first civilian large-scale high-power laser research facility to be realized with trans-European cooperation in three sites. ELI's long-term objective is to become the world's leading user facility utilizing the power of state of the art lasers for the advancement of science and applications in many areas of societal relevance [1]. The main objective of the ELI Attosecond Light Pulse Source (ELI-ALPS) pillar is the establishment of a unique attosecond facility that provides ultra-short light pulses with high repetition rates.

The typical layout of *beamline systems* at the ELI-ALPS is as follows (see Fig. 1): the laser source produces pulses in the femtosecond duration range, which is connected via a beam transport system to one or more secondary source(s). Secondary sources are designed to produce attosecond pulses from the femtosecond laser pulses by utilizing various technologies based on Gas High-Harmonic Generation (GHHG) and Solid High-Harmonic Generation (SHHG). Beamline systems may have end stations as their closing system.

Beamline systems contain a high number of controlled devices, most importantly optomechanics (translation stages, motorized mirrors, irises, etc.), cameras and detectors, and in certain sections vacuum devices (vacuum pumps, valves, gauges, etc.)

* The ELI-ALPS project (GINOP-2.3.6-15-2015-00001) is supported by the European Union and co-financed by the European Regional Development Fund..
† lajos.schrettner@eli-alps.hu

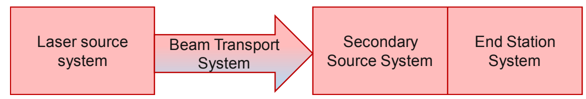


Figure 1: Research technology overview.

METHODOLOGY

Motivation

ELI-ALPS research technology in its current state of development already contains numerous devices to be controlled by the control system. This number is currently around 100, but in the final state of the facility with all the beamlines implemented, we expect more than 2000 devices total. Regarding the number of device types, the current figure is around 40, which is expected to be approximately double by the end of installation. Since the devices form a highly heterogeneous environment, their integration into one system poses challenges that can be overcome only by a unified strategy. The control system design and development has been carried out by considering similar projects [2-9] as well as industrial standards and recommendations.

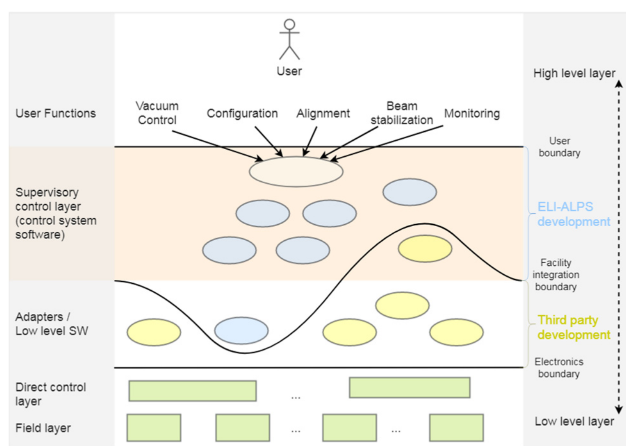


Figure 2: Control system scheme.

Control System Functions

While keeping standard control system design considerations in mind, control system development at ELI-ALPS follows a demand driven approach where focus is always given to areas where the research infrastructure reaches a certain implementation phase.

A general scheme to visually represent the major goals of control system developments are shown in Fig 2. In this scheme the low level layer consists of all controllable devices that together comprise our research infrastructure.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEBPP02

CENTRALIZED SYSTEM MANAGEMENT OF IPMI ENABLED PLATFORMS USING EPICS*

K. Vodopivec†, Oak Ridge National Laboratory, Oak Ridge, TN, USA

Abstract

The Intelligent Platform Management Interface (IPMI) is a specification for computer hardware platform management and monitoring. The interface includes features for monitoring hardware sensors, such as fan rotational speed and component temperature, inventory discovery, event propagation, and logging. Additional features are available in PICMG compliant systems, including ATCA and Micro TCA. With IPMI support implemented in the hardware, all IPMI functionality is accessible without any host operating system involvement. In fact, IPMI can even be used to control remote host power management. With its wide breadth of support across many hardware vendors and the backing of a standardization committee, it is a compelling instrumentation choice for integration into control systems for large experimental physics projects. Integrating IPMI into the EPICS system provides the benefit of centralized monitoring, archiving and alarming within the facility control system. A new project has been started to enable this capability, by creating a native EPICS device driver built on the open-source FreeIPMI library for the remote host connection interface. The driver supports automatic system component discovery for creating EPICS database templates, including detailed device information from the Field Replaceable Unit interface, as well as sensor monitoring with remote threshold management, geographical PV addressing in PICMG based platforms and PICMG front panel lights readout.

INTRODUCTION

The Intelligent Platform Management Interface (IPMI) [1] is becoming a widely adopted technology, as supported by many hardware providers including Intel, Dell, Hewlett Packard, Cisco and others. More importantly, for the applications in large scale research facilities, the PICMG AdvancedTCA Base Specification [2] builds on IPMI with functionality specific to AdvancedTCA and MicroTCA platforms. With many VME components facing obsolescence issues, an increasing number of new applications at particle accelerator and synchrotron facilities are looking for alternatives, and settling for ATCA or MTCA platforms as the new standard for hardware based solutions. One of the reasons for this decision is also the availability of built-in native support for the IPMI standard, as this automatically furnishes the application with system health monitoring. Functionality that had previously been implemented on a case by case basis, and was often overlooked, is now part of every system and therefore can be used for more thorough monitoring and control of core system functions.

The IPMI standard provides interfaces to monitor embedded sensors such as temperature, voltage, current, fan speed and others, depending on the particular component implementation. Monitoring core sensors alone provides useful benefits for detecting component failures or potentially trying to prevent them. For example, a failed fan inside the chassis will immediately restrict the air flow and can rapidly cause the temperature of the components in the air flow path to increase, in turn causing the components and/or entire system to fail. By detecting the reduced fan speed and temperature increase, the failing fan can be identified and replaced. Alternatively, the speeds of other fans in the system can be increased to maintain adequate air flow until the failing fan can be replaced. Another notable feature of IPMI is its ability to read inventory information through the Field Replaceable Unit (FRU) mechanism. With the FRU interface, individual system components – including power supplies, cooling units, processing blades or expansion boards – can be addressed and queried for information such as the manufacturer name, model name or code, manufacturing date, serial number or several others. Having complete inventory information of the system and its individual components allows for detailed cataloging and traceability. A statistical analysis of hardware failures can be carried out in order to schedule preventive replacements of critical system components. Controlling low-level system functions is also available through IPMI. For example, it is possible to control a smart power supply and toggle power to the CPU, or in the case of ATCA and MTCA, the power distribution to individual expansion slots.

The Experimental Physics and Industrial Control System (EPICS) framework, on the other hand, provides infrastructure for distributed access and management of devices. Especially where already employed as the control system of choice, adding support for IPMI-capable devices greatly expands the repertoire of systems that EPICS can talk to or manage. Along with existing EPICS tools for real time status monitoring, alarm management, archiving data points for analyzing historic behaviour and others, IPMI-compliant systems can be easily integrated into the environment of experimental physics and industrial control systems. A new project called "*epicsipmi*" has been designed to provide the bridge between IPMI interfaces and the EPICS framework, enabling easy monitoring of system health functions in new

* This manuscript has been authored by UT-Battelle, LLC under Contract No. DE-AC05-00OR22725 with the U.S. Department of Energy. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes. The Department of Energy will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan(http://energy.gov/downloads/doe-public-access-plan).

† vodopiveck@ornl.gov

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEBPP03

THE LASER MEGAJOULE FACILITY: FRONT END'S CONTROL SYSTEM

J. Langot, C. Baret, P. Fourtillan, J.F. Gleyze, D. Hamon, D. Lebeaux, A. Perrin
CEA, CESTA, Le Barp, France

Abstract

The Laser Megajoule (LMJ) is a 176-beam laser facility, located at the CEA CESTA Laboratory near Bordeaux (France). It is designed to deliver about 1.4MJ of energy to targets, for high energy density physics experiments, including fusion experiments. Six 8-beams bundles are currently operational. The Front-End is the LMJ subsystem built to deliver the laser pulse which will be amplified into the bundles. It consists of 4 laser seeders, producing the laser pulses with the expected specificities and 88 Pre-Amplifier Modules (PAM). In this paper, we introduce the architecture of the Front-End's control system which coordinate the operations of the laser seeders and the PAMs's control systems. We will discuss the ability of the laser seeders and their control systems to inject the 88 PAMs almost independently. Then we will deal with the functions that enable the expected laser performances in terms of energy, spatial and temporal shapes. Finally, the technics used to validate and optimize the operation of the software involved in the Front-End's equipment performance will be detailed.

THE LASER MEGAJOULE FACILITY FRONT END'S

The Front-End's mission on the LMJ installation is to deliver a laser beam defined in terms of time shape, spatial shape and energy to the laser bundles. In this document, we will describe the functioning of the Front-End's as well as its Control Command System (CCS).

The Front-End (FE) consists of two components: the Seeders and the Pre-Amplification Modules (PAMs). The Seeder provides the laser pulses with controlled spectral and temporal characteristics to the PAMs that guarantee the spatial shape and pre-amplify the pulse before the laser bundle injection.

For the LMJ, there are four Seeders (one per Laser Hall), each Seeder delivers the laser pulses by fiber to the PAMs of their Laser Hall (Figure 1).

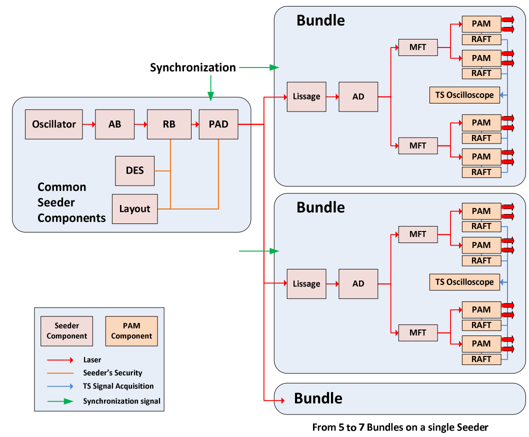


Figure 1: LMJ Front-End scheme of a Laser Hall.

THE LMJ'S SEEDER

As shown in Fig. 1, the LMJ Seeder is composed of two parts; the common part to a laser hall includes:

- The Oscillator, for the single mode 1053 nm laser pulse generation,
- The AB that performs a spectral widening to avoid Brillouin effects on the optics of the bundle,
- The RB-DES-Distri group is a functional safety device that blocks the signal if its spectral expansion is not correct,
- The PAD which Pre-amplifies and distributes the signal to the bundle bays.

Each bundle parts include:

- A smoothing module which performs a spectral widening to avoid spekel effects on the target,
- An AD module, that amplifies and distributes the signal to the MFTs,
- 1 MFT per Quadruplet (4 laser beams) that shapes the laser temporal signal. Each MFT can inject 2 PAMs.

The Seeder has an interface with the synchronization system to coordinate the operation of the LMJ's lasers.

THE LMJ'S PAM

There is one PAM for two laser beams or four per bundle. For the 22 bundles of the LMJ there are 88 PAMs.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEBPP04

P99: AN OPTICAL BEAMLINE FOR OFFLINE TECHNIQUE DEVELOPMENT AND SYSTEMS INTEGRATION FOR PROTOTYPE BEAMLINE INSTRUMENTATION

A. D. Parsons, S. Ahmed, M. Basham, D. Bond, B. Bradnick, M. Burt, T. Cobb, N. Dougan, M. Drakopoulos, F. Ferner, J. Filik, C. Forrester, L. Hudson, P. Joyce, B. Kaulich, A. Kavva, J. Kelly, J. Mudd, B. Nutter, N. O'Brien, P. Quinn, K. Ralphs, C. Reinhard, J. Shannon, M. Taylor, T. Trafford, X. Tran, E. Warrick, A. Wilson ,A. D. Winter
Diamond Light Source, Oxon, UK

Abstract

We present a beamline analogue, capable of system prototyping, integrated development and testing, specifically designed to provide a facility for full scientific testing of instrument prototypes. With an identical backend to real beamline instruments the P99 development rig has allowed increased confidence and troubleshooting ahead of final scientific commissioning. We present detail of the software and hardware components of this environment and how these have been used to develop functionality for the new operational instruments. We present several high impact examples of such integrated prototyping development including the instrumentation for DIAD (integrated Dual Imaging And Diffraction) and the J08 (Soft X-ray ptychography) beamline end station.

INTRODUCTION

Diamond Light Source is a publicly funded 3rd generation national synchrotron soon to boast 39 operational state-of-the-art user accessible instruments covering a wide range of physical and life science applications. Such instruments pose a large number of challenges from initial scientific concept, to final user experience. To get best efficiency and value for money, Diamond operates a modular approach for engineering and software systems support, usually with each custom hardware or software component coming together on the final instrument in-situ. This can be a high-risk strategy with any issues arising during the final scientific commissioning of the beamline. This is often unavoidable due to the bespoke nature of the instruments. Small prototyping rigs for instruments have been used to some success, but often do not consider the underlying infrastructure, or are focussed on only specific sections of the software/hardware stack, which have been identified as problematic. Such approaches risk ignoring the complete user/operator experiences.

P99 is a collaborative offline testing facility that aims to tackle such shortfalls by providing a space for collaboration and innovation amongst support teams and beamline staff. Such prototyping allows a degree of confidence to be established in the building blocks of otherwise complex and multifaceted projects. A laser-based analogue to a real beamline, P99 is specifically targeted at cases that either 1) target the full software stack requiring a full beamline infrastructure backend for testing, 2) require a high brightness/coherence source with high speed triggering 3) require high stability i.e. for nano-positioning use cases.

In this manuscript we introduce the design of P99 and show 3 case studies where this facility has allowed testing of multiple prototypes across multiple instruments (DIAD [1], J08 [2]), and current ongoing ptychography [3, 4] developments facilitating collaboration and prototyping across groups of the instrument as a whole well in advance of the final deployment, allowing issues to be realised and approaches refined.

P99 DEVELOPMENT RIG

The P99 development rig is identical in networking, compute and controls infrastructure to a true user facing instrument. This means that any solution developed on P99 should work equally as well on a user beamline/instrument making P99 unique compared to other such testing facilities at Diamond. This infrastructure is housed in the rack in Fig. 1 and is compliant with the architecture described in [5] and so can make full use of this work.

Figure 1: P99 optical development rig in the precision metrology laboratory, complete with main and small testing rack, opaque enclosure and beamline workstation.

P99 is situated in Diamonds precision metrology laboratory [6], with a foundation of an actively damped honeycomb optical breadboard. Inside of an acoustically damped, opaque enclosure (Fig. 1), there is a secondary breadboard, which is isolated from the actively damped breadboard by Sorbothane pads. This is similar in design

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WECPL01

STATUS OF THE CONTROL SYSTEM FOR FULLY INTEGRATED SACLA/SPring-8 ACCELERATOR COMPLEX AND NEW 3 GeV LIGHT SOURCE BEING CONSTRUCTED AT TOHOKU, JAPAN

T. Sugimoto*[1,2], N. Hosoda[1,2], M. Ishii[1,2], K. Okada[1], M. Yamaga[1,2], T. Fukui[2]
[1]Japan Synchrotron Radiation Research Institute, Hyogo, Japan
[2]RIKEN SPring-8 Center, Hyogo, Japan

Abstract

In the SPring-8 upgrade project, we plan to use the linear accelerator of SACLA as a full-energy injector to the storage ring. For the purpose of simultaneous operation of XFEL lasing and on-demand injection, we developed a new control framework that inherits the concepts of MADOCA. We plan to use the same control framework for a 3-GeV light source under construction at Tohoku, Japan. Messaging of the new control system is based on the MQTT protocol, which enables slow control and data acquisition with sub-second response time. The data acquisition framework, named MDAQ, covers both periodic polling and event-synchronizing data. To ensure scalability, we applied a key-value storage scheme, Apache Cassandra, to the logging database of the MDAQ. We also developed a new parameter database scheme, that handles operational parameter sets for XFEL lasing and on-demand top-up injection. These parameter sets are combined into 60 Hz operation patterns. For the top-up injection, we can select the operational pattern every second on an on-demand basis. In this paper, we report an overview of the new control system and the preliminary results of the integrated operation of SACLA and SPring-8.

SPring-8 UPGRADE AND 3 GeV LIGHT SOURCE PROJECTS

Synchrotron radiation (SR) is necessary in many scientific fields such as material science, biological science, non-linear optics. More than two decades have passed since the third-generation SR facilities established, such as APS, ESRF, and SPring-8. To maintain the top performance, we have begun upgrading the SPring-8 [1]. The upgrade plan aimed to improve the brilliance by 100 times with 1/20 beam emittance than the present facility. To achive the requirement, we use SACLA [2], which is an X-ray free electron laser (XFEL) facility, as a injector. In order to achieve such a combind operation of two facilities, we have to upgrade present control framework.

Another SR facility project is underway in Japan. A compact 3 GeV light source project [3] has been approved by the Japanese Government from fisical year 2019. The accelerator components are designed based on previous studies of the SPring-8 upgrade project. Under such a circumstance, we proposed the new framework developed for the SPring-8 upgrade project.

* takashi.sugimoto@spring8.or.jp

In this paper, we show the basic concepts of the control framework MADOCA, presently used at the SPring-8 accelerator complex. We also show the design overview of the new control framework, which is took over the concepts of MADOCA. The current status of the new control framework development and application not only to the SPring-8 acclerator complex but also the 3 GeV light source are shown.

MADOCA CONTROL SYSTEM AND ITS LIMITATIONS

MADOCA (Messaging and Database oriented control architecture) is a control framework developed for the SPring-8 storage ring control [4]. MACOCA is characterized as S/V/O/C style messaging, whose idea was derived from English grammar. Computer networks are used as field buses for messaging between workstation and device contorllers. By using standard TCP/IP-based communication, we can use commercial off-the-shelf network instruments, such as Ethernet. Another key component of the MADOCA is the database system. We use SYBASE, which is a relational database system, as the central database system. The SYBASE is used for both instrument logging and save/recall operation parameters.

The original MADOCA was developed to control the SPring-8 storage ring in 1997. Subsequently, the MADOCA was also applied to the SPring-8 injector accelerators. [5] The MADOCA is also applied to other accelators outside of the SPring-8 campus, for example, the HiSOR at Hiroshima University [6] and SCRIT at RIKEN RI beam factory. In 2011, the X-ray free electron laser facility SACLA commenced operation. For the control system of the SACLA, we applied the MADOCA. The beam operation of SACLA is different from SPring-8. SPring-8 is a storage ring; therefore, beam operation is based on periodic polling basis. On the other hand, SACLA is single-path beam machine with 60 Hz repetition rate; therefore, beam operation is on an event-synchronized basis. To complement the limitations of MADOCA, we have developed many control subsystems, such as event-synchronized data acquisition (DAQ) system [7], abnormal waveform recorder [8], and more. To operate two acclerator facilities SACLA and SPring-8 storage ring as one machine, we must integrate all the control sub-systems. Considering these circumstance, we started upgrading the MADOCA control framework [9].

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WECPL02

ROADMAP TO 100 Hz DAQ AT SwissFEL: EXPERIENCES AND LESSONS LEARNED

T. Celcer†, S.G. Ebner, F. Märki[1], A. Babic, L. Sala
Paul Scherrer Institut, 5232 Villigen PSI, Switzerland

Abstract

Providing a reliable and performant Data Acquisition System (DAQ) at Free Electron Lasers (FELs) is a challenging and complex task due to the inherent characteristics of a pulsed machine and consequent need of beam synchronous shot-to-shot DAQ, which enables correlation of collected data associated with each FEL pulse.

We will focus on experiences gathered during the process of moving towards 100 Hz operation at SwissFEL from the perspective of beam synchronous DAQ. Given the scarce resources and challenging deadlines, many efforts went into managing conflicting stakeholder expectations and priorities and into allocation of time for operation support and maintenance tasks on one side and time for design and development tasks on the other side. The technical challenges we encountered have shown a great importance of having proper requirements in the early phase, a well thought system design concept (which considers all subsystems in the DAQ chain), and a well-defined test framework for validation of recorded beam synchronous data

INTRODUCTION

SwissFEL Project Timeline

The SwissFEL free electron laser facility [1] is the newest accelerator at the Paul Scherrer Institute (PSI). The overall length of SwissFEL measures 720 m and consists of a 270 m long common accelerator line (injector, linac 1 and linac 2), followed by a switchyard, splitting it into two lines. The first, straight line, being part of the first construction phase, represents the third part of the linac and the hard X-Ray branch Aramis. Aramis currently serves two experimental stations, namely Alvra and Bernina. The project officially started in 2013 and was declared finished at the end of 2017.

In the currently ongoing second construction phase, a parallel, soft X-Ray branch Athos [2] will be added, delivering photons to two additional experimental stations, Maloja and Furka. The project timeline is 2018 – 2020 and first lasing is planned for December 2019. First pilot experiment in Maloja experiment station is planned for April 2020, while first pilot experiment in Furka is planned one year later, in April 2021. In addition, the project of building a third Aramis experimental station Cristallina is slowly starting. SwissFEL layout is shown in Figure 1.

DAQ Challenges at FELs

In order to better understand the challenges of a FEL DAQ system, one first needs to understand the main differences between the continuous beam facility like a synchrotron and the pulsed machine like a FEL.

At PSI, three other large user facilities, operated before SwissFEL (SLS, SINQ and SμS), are all continuous beam sources. If we take SLS as a comparison, it delivers a stable, continuous beam to 18 user beamlines (or experimental stations). Due to the beam stability and continuity, the data obtained at the experimental station at a certain point in time does not depend (in large extent) on the machine parameter. This greatly simplifies the data acquisition process, as the timing requirements are quite relaxed and there is no need to correlate the experimental data with machine data (except for some asynchronous, slowly changing parameters).

The inherent characteristic of a pulsed machine is that the data readout has to be triggered synchronous to the electron or photon beam in order to collect useful data. As the beam (consisting of individual electron bunches) properties can fluctuate considerably from bunch to bunch, the data collected at the experimental station depends significantly on physical parameters of every bunch (pulse) on its path along the accelerator line, and therefore needs to be tagged appropriately. The BS DAQ system needs to store very large data sets and provide the ability to correlate the collected experimental data associated with each FEL pulse. For a 100 Hz repetition rate, this means that data (scalars, waveforms and images) all along the accelerator line have to be reliably collected, processed, tagged and recorded inside a 10 ms time slot. Therefore, the accelerator part of the FEL facility is much more involved in the experiment than in a synchrotron, even more so as usually a single bunch train delivers photons to only a single experimental station.

In addition, the users would like live (or close to live) analysis of beam synchronous data, which further increases the overall complexity as not only beam synchronous data recording but also beam synchronous data retrieval is needed.

All this presents an important shift in required technical skills of personnel working in the Control system section. A significant increase for experienced software developers and versatile control system specialists with higher knowledge level in real-time (embedded) systems arises.

† tine.celcer@psi.ch
[1] now at Institute for Data Science, School of Engineering, University of Applied Sciences NorthWestern Switzerland, 5210 Windisch, Switzerland

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WECPL03

IMPLEMENTATION OF THE MOTION CONTROL SYSTEM FOR LCLS-II UNDULATORS

M. A. Montironi, C. Andrews, H. Bassan, K. Lauer, Y. Levashov, H.-D. Nuhn, Z. Wolf, SLAC National Accelerator Laboratory, Menlo Park, California
Ž. Oven, Cosylab, Ljubljana, Slovenia

Abstract

As part of the LCLS upgrade called LCLS-II, two new undulator lines were introduced: a soft X-Ray line (SXR) and a hard X-Ray line (HXR). Serving distinct purposes, the two undulator lines employ different undulator designs. The SXR line is composed of 21 vertical gap, horizontally polarizing undulators while the HXR line is composed of 32 undulator segments designed to operate on the horizontal axis and to produce a vertically polarized X-Ray beam. The HXR undulators will replace the LCLS ones and thus the control system was designed with the main goal of maximizing the re-utilization of existing hardware and software. For this purpose, the motion control system based on RTEMS running on VME with Animatics SmartMotors was developed as an upgrade of the LCLS design and the cam-based undulator girder positioning system has been reused. The all new SXR undulators employ a new control system design based on Aerotech motion controllers and EPICS soft IOCs (input-output controllers). This paper describes how the most challenging motion control requirements were implemented focusing on motion synchronization, K-value to gap transformation, cams kinematics and calibration, and user interaction.

INTRODUCTION

As part of the LCLS upgrade called LCLS-II, two new undulator lines were introduced: a Soft X-Ray line (SXR) and a Hard X-Ray line (HXR) [1]. The SXR line is composed of 21 vertical gap, horizontally polarizing undulators each paired with a downstream interspace assembly mounted on a separate support and a phase shifter. The HXR line is composed of 32 undulator segments designed to operate on the horizontal axis and to produce a vertically polarized X-ray beam [2, 3]. As for the soft line, each HXR undulator is paired with a phase shifter and an interspace assembly mounted on the same girder as the undulator. The drive system for the SXR undulators, an example of which is shown in Figure 1, consists of four Harmonic Drive servo motors with feedback from internal rotary encoders and two full-gap encoders attached at each end of the undulator segment. Brakes integrated with each motor allow holding the gap in position. The gap motion control is achieved for each undulator through a four axis Aerotech motion controller. The controller allows to drive the motors in a coordinated manner, supports safety features such as reacting to limit switches and ESTOP, and allows the measurement of the vacuum chamber position through linear potentiometers connected to its analog inputs.

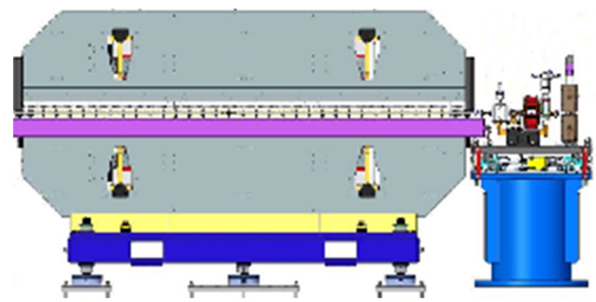


Figure 1: SXR undulator cell. Undulator and downstream interspace assembly.

Figure 2 shows a schematic representation of the EPICS IOC (input-output controller) developed to interface the controller with the accelerator distributed control system and other high-level applications (HLA).

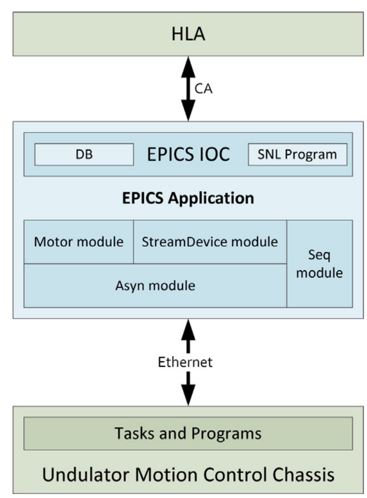


Figure 2: Structure of EPICS IOC for SXR undulators.

An SXR interspace assembly, shown downstream of the undulator in Figure 1, was designed to be inserted between consecutive undulator segments to provide a mounting surface for the vacuum assembly, Beam Position Monitors (BPM), quadrupole magnets, and a permanent magnet variable gap phase shifter. The top plate of the interspace assembly can be positioned independently of the undulator using a 5 Degrees-Of-Freedom (DOF) cam mover system driven by stepper motors to support beam based alignment and to allow repointing the undulator line. The gap of the phase shifter is controlled through a DC servomotor and an absolute linear encoder. A 6-axis Aerotech motion controller and an EPICS IOC structured similarly to the one used for the undulator are utilized for precise motion control of the interspace assembly.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WECPL05

MIGRATING TO TINY CORE LINUX IN A CONTROL SYSTEM

R. A. Washington, ISIS, Rutherford Appleton Laboratory, Didcot, U. K.

Abstract

The ISIS Accelerator Controls (IAC) group currently uses Microsoft Windows Embedded Standard 2009 (WES) as its chosen Operating System (OS) for control of front-line hardware. Upgrading to the latest version of Microsoft Windows Embedded is not possible without also upgrading hardware or changing the way that the software is delivered to the hardware platform. The memory requirements are simply too large for this to be considered a viable option. A new alternative needed to be sought; which led to Tiny Core being selected due to its frugal memory requirements and ability to run from a RAM disk. This paper describes the process of migrating from Windows Embedded Standard to Tiny Core Linux as the OS platform for IAC hardware.

A NEED TO UPGRADE

The ISIS Control System uses Vista Control Systems [1] Vsystem software as the primary user interface. The Control System [2] monitors some 29,000 values, the majority of which are acquired through custom hardware designed by the ISIS Accelerator Controls (IAC) group. The latest version of this hardware is the IAC CompactPCI Standard (CPS) system which run Microsoft Windows Embedded Standard 2009 (WES) as the embedded Operating System (OS) and platform for the CPS software handlers. The CPS systems have been very reliable but the extended support for WES ended on January 8th 2019 meaning that an upgrade to the OS was already well overdue.

A NEW PLATFORM

CPS systems are built around a CompactPCI backplane with slots for up to seven peripheral cards and a system slot containing a processor card. CPS processor cards that are currently used are the Kontron [3] CP305, CP306 and CP3004.

The natural choice for the new OS, given the previous experience of the IAC group, was to pursue the latest embedded version of Microsoft Windows, Windows 10 IoT Enterprise. However it quickly became clear that this OS would not meet the particular deployment requirements. The CPS systems are required to remotely acquire an OS image over the network and continue to run the OS as a RAM disk. This is because conventional hard disk drives, specifically the disk controllers, fail due to radiation within the area of the inner synchrotron. To this end, the OS image is downloaded via Pre-boot Execution Environment (PXE) into the on-board RAM of the processor card, where it continues to run as a RAM disk.

All this is possible in Windows 10 but the size of the Windows 10 OS image footprint (typically around 8 GB) is far greater than that of the WES image (350 MB) and therefore requires more available RAM on the processor card. For reference, the older CPS processor cards have just 1 GB of available RAM. A decision was made to look at alternative OS solutions that have a smaller footprint and therefore smaller RAM requirement.

A comparison of several Linux distributions resulted in Tiny Core (TC) [4] being selected as the new platform. TC is a minimal Linux distribution and has the added advantage of being designed to run completely from a RAM disk.

THE CPS SERVICE

The CPS Service is a software manager that handles hardware reads and writes that have been initiated by Vsystem. It provides the link between the hardware in a CPS system and the control screen operated by the end user. The flowchart for the CPS Service is illustrated in Fig. 1.

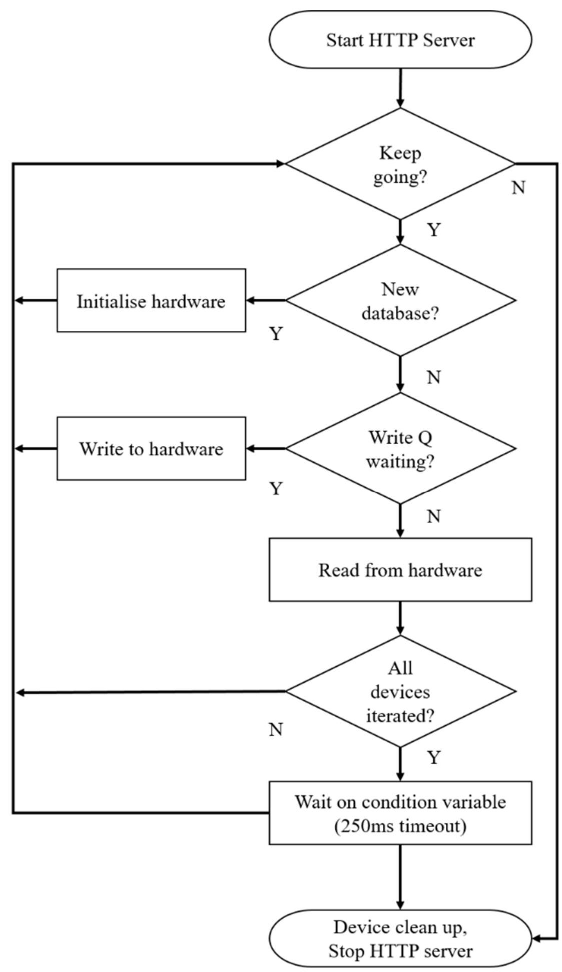


Figure 1: CPS Service flowchart.

The CPS Service code is organised into three projects: CPSService, HTTPServer and PCIDeviceClasses.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WECPR01

EPICS 7 CORE STATUS REPORT*

A. N. Johnson†, G. Shen, S. Veseli, Argonne National Laboratory, Lemont, Illinois, USA
K. Shroff, Brookhaven National Laboratory, Upton, New York, USA
T. Korhonen, European Spallation Source ERIC, Lund, Sweden
H. Junkes, Fritz Haber Institute, Berlin, Germany
M. G. Konrad, FRIB, East Lansing, Michigan, USA
R. Lange, ITER Organization, St. Paul lez Durance, France
S. M. Hartman, K. U. Kasemir, Oak Ridge National Laboratory, Oak Ridge, USA
M. A. Davidsaver, Osprey DCS LLC, Ocean City, USA
M. R. Kraimer, Osseo, USA
K. H. Kim, SLAC National Laboratory, Menlo Park, USA

Abstract

The integration of structured data and the PV Access network protocol into the EPICS toolkit has opened up many possibilities for added functionality and features, which more and more facilities are looking to leverage. At the same time the core developers also have to cope with technical debt incurred in the race to deliver working software. This paper will describe the current status of EPICS 7, and some of the work done in the last two years following the reorganization of the code-base. It will cover some of the development group's technical and process changes, and echo questions being asked about support for recent language standards that may affect support for older target platforms, and adoption of other internal standards for coding and documentation.

EPICS 7

The first version of the EPICS Control System Toolkit [1] that was officially named "EPICS 7" was released in December 2017, fulfilling a request from the EPICS community for a single downloadable software package that contained the core EPICS Base code and the extra "EPICS Version 4" C++ modules (now called "PVA modules") that support structured data and the pvAccess network protocol.

In the two years since that release, members of the development team have been adding features to the core, debugging and re-engineering the new modules, and improving the integration between the different parts of the code-base. Some code that will not be developed any further has been unbundled from the core and published separately as stand-alone modules.

Release History

The new 7.0 release series replaced the 18-month old Base-3.16 series, and prompted the closing of the 16-year old Base-3.14 series to reduce the maintenance burden on the limited number of core developers. The Base-3.15 series will continue to be maintained for sites using older hardware and operating systems that cannot support newer C++ compilers.

* Work supported by U.S. Department of Energy Office of Science, under Contract No. DE-AC02-06CH11357
† Andrew Johnson <anj@anl.gov>

Table 1 lists the versions of EPICS Base that have been released in the last two years.

Table 1: EPICS Releases from December 2017

Version	Release Date	Description
3.14.12.7	2017-12-15	Stable, bug fixes
3.14.12.8	2018-09-14	Final 3.14 release
3.15.6	2018-10-11	Stable, bug fixes
3.16.2	2018-12-12	Final 3.16 release
7.0.1.1	2017-12-15	First 7.0 release
7.0.2	2018-12-17	Modules rejoined, APIs
7.0.2.1	2019-03-20	Bug fixes
7.0.2.2	2019-04-23	Bug fixes
7.0.3	2019-07-31	Bug fixes, API changes

Source Code Repositories Each release series is maintained on its own branch of a Git [2] distributed version control system repository, but the new PVA modules each have their own separate Git repository which is integrated into the main one as a Git submodule. The master repository is on Launchpad.net [3] along with the main EPICS bug-tracker, but is also mirrored to a repository on GitHub [4] where the PVA submodules are kept.

Continuous Integration During initial development of the PVA modules they were built and tested in multiple configurations against the different release series of Base using a Jenkins [5] Continuous Integration server running on a commercial cloud-based service, and also by a Jenkins server hosted at Argonne which tested builds on different operating systems. The commercial service has now been replaced by Travis CI [6] and Appveyor [7], which are both free for use by open source software projects and integrate well with GitHub.

Release Frequency After EPICS 7.0.2 was published the developers agreed to try releasing new versions more often, generating bug-fix releases instead of publishing patch files when important fixes became available. This policy is getting some push-back from sites that prefer to apply patches to their existing installations than to import and build a completely new version.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WECPR02

BENEFITS AND DRAWBACKS OF USING RUST IN AN EXISITNG C/C++ CODEBASE*

B. S. Martins[†1], Facility for Rare Isotope Beams, East Lansing, USA
[1]also at Columbia University, New York, USA

Abstract

Mozilla has recently released a new programming language, Rust, as a safer and more modern alternative to C++. This work explores the benefits (chiefly the features provided by Rust) and drawbacks (the difficulty in integrating with a C ABI) of using Rust in an existing codebase, the EPICS framework, as a replacement for C/C++ in some of EPICS' modules.

INTRODUCTION

Mozilla's new systems programming language, Rust, promises to make entire classes of bugs detectable and preventable at compile time [1]. Its first stable version, dubbed 1.0, was released on May 2015. Rust has a heavy focus on memory safety and uses novel concepts, such as resource lifetimes and the borrow checker, to achieve that goal. These features, however, come at the cost of increased language complexity. Rust also aims to be fast and provide binary compatibility with C while providing high level constructs, making it a great candidate for replacing both C and C++ as system languages. While Rust does not have full feature parity with C yet [2], the language is rapidly evolving in this direction.

EPICS [3] is an industrial controls system framework used in several big science facilities around the world. EPICS is primarily written in C and C++ and has been incorporating contributions from several people along its three decades of existence. Therefore, its codebase contains a mixture of legacy and modern C and C++ code that presents several opportunities for improvement.

Rust's safety claims and modern features are enticing for projects like EPICS. This work investigates the use of the Rust language in the context the EPICS framework in an attempt to answer the following research questions:

1. Would Rust have prevented actual EPICS bugs?
2. Is it straightforward to translate C/C++ code into Rust?
3. Is it feasible to rewrite parts of EPICS into Rust?
4. Is it worth it to rewrite a big C/C++ project into Rust?

Question 1 will be answered by first examining EPICS' issue tracker [4] and classifying its bugs into a few categories, and then rewriting a few representative bugs in Rust to verify if its compiler would have caught them. Questions 2, 3 and 4 will be answered by evaluating the manual reimplementation of a single EPICS base component, iocsh, into Rust.

RUST'S MEMORY SAFETY FEATURES

While Rust has many modern programming language features, such as first-class functions, closures, algebraic types, async/await, etc., its strength lies in its ownership system, which can be divided into three concepts: **ownership**, **borrowing** and **lifetimes**. These concepts play a fundamental role in ensuring memory safety.

Ownership

Rust's ownership mechanism ensures, at compile time, that all values in a Rust program have an owner. Typically, the owner of a value is the variable the value was first assigned to. When that first value is assigned to a second variable, it is said that the value is *moved*, and the first variable loses *ownership* to the second variable. After a value is *moved*, the first variable cannot be used anymore to reference it. For example, in Listing 1, the variable `a` is the first *owner* of the vector containing the values 1, 2 and 3. Then, on the following line, the vector is *moved* to the variable `b`, which means `a` no longer *owns* the vector; trying to access `a` again would violate Rust's *ownership* constraints, so the compiler prevents it from happening.

Listing 1: Ownership example

```
1    fn main() {
2      let a = vec![1,2,3];
3      let b = a;
4      println!("{:?}", a);
5    }
```

As shown in Listing 2, the compiler emits helpful messages: it tells where the value was first moved (at line 3 when being assigned to `b`) and hints that the particular type does not implement the `Copy` trait. The Copy trait indicates that, rather than being *moved*, the value can be instead *copied* to the destination. All primitive types implement the `Copy` trait. If non-`Copy` values had to be moved back and forth between owner variables Rust would be a very impractical language. Therefore, there is a mechanism for taking references to a value, which is called borrowing.

* This material is based upon work supported by the U.S. Department of Energy Office of Science under Cooperative Agreement DE-SC0000661, the State of Michigan and Michigan State University.
† martins@frib.msu.edu
[1] Category of interest for this study

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WECPR03

STATUS OF THE KARABO CONTROL AND DATA PROCESSING FRAMEWORK

G. Flucke*, N. Al-Qudami, M. Beg, M. Bergemann, V. Bondar, D. Boukhelef, S. Brockhauser[1], C. Carinan, R. Costa, F. Dall'Antonia, C. Danilevski, W. Ehsan, S. G. Esenov, R. Fabbri, H. Fangohr[2], D. Fulla Marsa, G. Giovanetti, D. Goeries, S. Hauf, D. G. Hickin, E. Kamil, Y. Kirienko, A. Klimovskaia, T. A. Kluyver, D. Mamchyk, T. Michelat, I. Mohacsi, A. Muennich, A. Parenti, R. Rosca, D. B. Rück, H. Santos, R. Schaffer, A. Silenzi, K. Wrona, C. Youngman, J. Zhu
European XFEL GmbH, Schenefeld, Germany
[1]also at Biological Research Center of the Hungarian Academy of Sciences, Szeged, Hungary and at University of Szeged, Szeged, Hungary
[2] also at University of Southampton, Southampton, United Kingdom

Abstract

To achieve a tight integration of instrument control and (online) data analysis, the European XFEL decided in 2011 to develop Karabo, a custom control and data processing system. Karabo provides control via event-driven communication. Signal/slot and request/reply patterns are implemented via a central message broker. Data pipelines for e.g. scientific workflows or detector calibration are implemented as direct TCP/IP connections. The core elements of Karabo are self-describing devices written in C++ or Python. They represent hardware, orchestrate other devices, or provide system services like data logging and configuration storage. To operate Karabo, a Python command line interface and a generic GUI written in PyQt are provided. Control and data widgets compose Karabo scenes that are provided by devices or are manually customized and stored together with device configurations in a central database. Since 2016, Karabo is used to commission and operate the currently three photon beam lines and six scientific instruments at the European XFEL. This contribution summarizes the status of Karabo, highlights achievements and lessons learned, and gives an outlook for future directions.

INTRODUCTION

The European X-ray Free Electron Laser (EuXFEL) facility provides hard and soft X-ray beams via three photon beamlines to six instruments. Up to 27,000 photon pulses per second are arranged into 10 Hz trains of pulses at 4.5 MHz. High-repetition-rate, large-area 2D imaging detectors capable of detecting images of scattered photons produced by a single XFEL photon pulse create very high data rates. Detector data needs to be calibrated on-the-fly with low latency to provide feedback to the experiment control.

In view of these requirements it was decided that a new distributed control system, Karabo, with integrated data acquisition and workflow capabilities should be designed and developed. Hence, Karabo has been developed since early 2012 [1] and has been in use since September 2017 to enable scientific user experiments at the EuXFEL [2].

KARABO IN A NUTSHELL

Karabo is designed to provide supervisory control and data acquisition for the European XFEL. Hardware devices, system services, and control procedures are represented by Karabo software devices of which many can run within the same server process, distributed among various control hosts. Devices are self-describing their properties, commands, configuration possibilities, and their availability depending on the *state* of the device. This description can be expanded at run-time, e.g. according to discovered hardware details. Devices can expose that they have specific *capabilities* like providing scenes or macros (see section on user interfaces) or *interfaces*. An interface, e.g. as a motor or a camera, defines a set of commands and properties.

Control communication is routed via a central broker. Currently, Karabo uses the the Java Messaging Service (JMS) broker [3] that can be clustered. Large data from detectors is transported via data pipelines implemented as direct TCP connections. A graphical and a command line interface provide flexible ways to interact with a Karabo system that is defined by a specific communication topic on the broker. Temporary procedures can be implemented as macros that run centrally on dedicated macro servers. Karabo's graphical user interface (GUI) connects into the system via a TCP connection to a gui server device.

An illustration of a Karabo system is shown in Fig. 1.

KARABO COMMUNICATION

A unique id identifies each object in a Karabo installation. Uniqueness is ensured when registering to the installation. Any message is routed via the broker according to this id. The header of a message contains the name of a *slot*, i.e. a method of the object that has been registered to be callable remotely.

Two broker communication patterns are implemented: In the *request/reply* case a (remote) slot is called and its success or failure of execution received. Up to four slot arguments

* gero.flucke@xfel.eu

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-WECPR04

AUTOMATED TESTING AND VALIDATION OF CONTROL PARAMETERS*

P. Kankiya†, J. P. Jamilkowski, A. Sukhanov
Brookhaven National Laboratory, Upton, USA

Abstract

The BNL CA-D controls environment has recently been adopting modern programming languages such as Python. A new framework has been created to instantiate setting and measurement parameters in Python as an alternative to C++ and Java process-variable-like objects. With the help of automated testing tools such as pyTest and Coverage, a test suite is generated and executed before the release of Python-based accelerator device objects (ADO) to assure quality as well as compatibility. This suite allows developers to add custom tests, repeat failed tests, create random inputs, and log failures.

INTRODUCTION

A particle accelerator machine is a collection of large number of hardware equipment that form the basic layer of control system software. The control access to these instruments at collider accelerator department of Brookhaven National Laboratory is provided by a logical device object known as ADO (Accelerator Device Object) [1]. ADO is a C++ container type class which provides a software view of a collection of collider control points known as parameters. These parameters are the basis of supervisory control , data acquisition and monitoring. Parameters are software entities derived from a base class with several properties that constitute the metadata of each instantiated control and measurement object. These objects when added to the container class represent the controls framework. By assuring that creation of these parameters is of a certain standard of quality- will help to prevent crashes in operational time and hence down time of beam.

Large part of testing of this framework is exercised manually at the time of creation of ADOs. The steps to test the properties associated with each parameter are repeatable and therefore should be automated. Automation of testing software provides benefits such as repeatability, reliability, and report generation. Several open source tools are rich in features that can make the unit testing of ADOs seamless. One such tool in consideration is pyTest [2].

ROADMAP OF UNIT TESTING OF ADOS

The testing phase is broken down into five steps that should be carried out to before release of ADO software as a best practice. These steps are depicted in Fig. 1.

Phase 1 requires preparation of test data in general unit testing terms this phase is referred to as set up phase. Once the test data is available and expected format it should run through all the respective test suites, which is phase 2. This phase concludes the quality assurance phase testing. This step will assure ADOs are compatible with most of the control's framework in place.

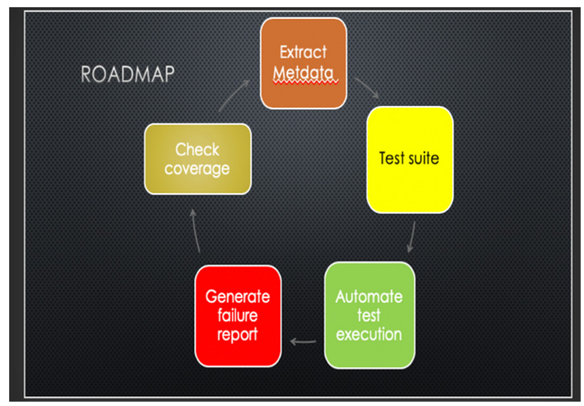
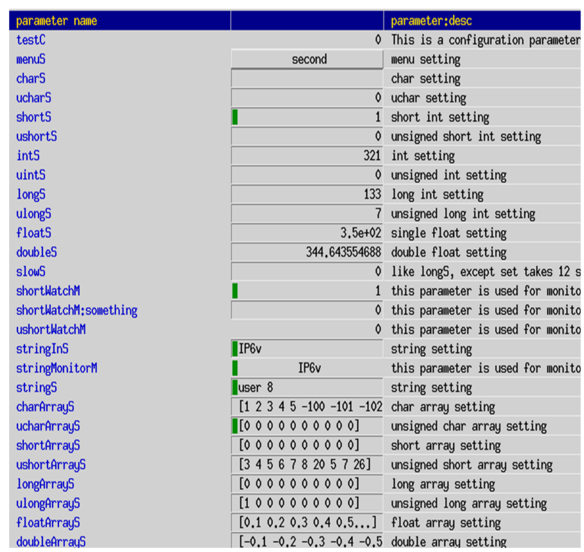


Figure 1. A general guideline to maximise coverage of tests to follow for creating unit tests for ADO parameters.



parameter name		parameter:desc
testC	0	This is a configuration parameter
menuS	second	menu setting
charS		char setting
ucharS	0	uchar setting
shortS	1	short int setting
ushortS	0	unsigned short int setting
intS	321	int setting
uintS	0	unsigned int setting
longS	133	long int setting
ulongS	7	unsigned long int setting
floatS	3.5e+02	single float setting
doubleS	344.643554688	double float setting
slowS	0	like longS, except set takes 12 s
shortWatchM	1	this parameter is used for monito
shortWatchM:something	0	this parameter is used for monito
ushortWatchM	0	this parameter is used for monito
stringInS	IP6v	string setting
stringMonitorM	IP6v	this parameter is used for monito
stringS	user 8	string setting
charArrayS	[1 2 3 4 5 -100 -101 -102	char array setting
ucharArrayS	[0 0 0 0 0 0 0 0 0 0]	unsigned char array setting
shortArrayS	[0 0 0 0 0 0 0 0 0 0]	short array setting
ushortArrayS	[3 4 5 6 7 8 20 5 7 26]	unsigned short array setting
longArrayS	[0 0 0 0 0 0 0 0 0 0]	long array setting
ulongArrayS	[1 0 0 0 0 0 0 0 0 0]	unsigned long array setting
floatArrayS	[0.1 0.2 0.3 0.4 0.5...]	float array setting
doubleArrayS	[-0.1 -0.2 -0.3 -0.4 -0.5	double array setting

Figure 2: A generic ADO called simple containing a large number of parameters.

System Under Test

The snapshot in Fig. 2 shows the variety of parameters under test ADO. This ado called simple ADO is used to perform testing upon. Parameters here vary from being configuration data , numeric normative types, strings to arrays of such basic data types.

---

*Work supported by Brookhaven Science Associates, LLC under Contract No. DE-SC0012704 with the U.S. Department of Energy.
†pkankiya@bnl.gov

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WECPR05

PULSED MAGNET CONTROL SYSTEM USING COTS PXIE DEVICES AND LABVIEW

Y. Enomoto*, K. Furukawa, T. Natsui, M. Satoh, KEK, Tsukuba, Japan
H. Saotome, Kanto Information Service, Tsukuba, Japan

Abstract

About one hundred channel of pulsed magnet power supply control system was installed in 2017 in KEK electron positron LINAC to realize pulse-to-pulse control of output current every 20 ms. The control system totally consist of commercially available devices, namely a computer (Windows 8.1), a PXIe crate and several PXIe boards such as ADC, DAC communication and timing. The software is written by LabVIEW. EPICS channel access protocol is used to communicate with OPI over standard Ethernet network. Depending on the destination of the beam, there are ten beam mode. The software is able to keep parameters for each mode independently, which makes it possible for us to operate one LINAC as if it were ten virtual LINACs. During two years of operation, there were no significant problem. Although the Windows is not a real-time OS, dropping rate of the trigger coming every 20 ms is less than ppm. Rebooting the computer or software is only a few times in a year.

INTRODUCTION

The KEK injector linac [1] has delivered electrons and positrons for particle physics and photon science experiments for more than 30 years. Figure 1 shows electron and positron accelerator complex in the KEK Tsukuba site. There are four storage rings, i.e. two rings for light source, PF and PF-AR, and two rings for electron positron collider, SuperKEKB [2] HER and LER. In addition to them, positron damping ring have been in operation since February 2018. All of the rings require full energy injection, 2.5 GeV for PF, 6.5 GeV for PF-AR, 4 GeV for SuperKEKB LER and 7 GeV for SuperKEKB HER as shown in the Fig. 1.

In 2009, simultaneous injection to three rings but PF-AR were realized [3] with common DC magnet settings. However, in the SuperKEKB era from 2016, the previous scheme has no longer valid due to strict requirement for injection beam to the SuperKEKB rings. Table 1 compares several required injection parameter for the KEKB rings and SuperKEKB rings. One of the big difference is that the beam life time in the SuperKEKB rings expected to be as short as 6 minutes that was 150 or 200 minutes in the KEKB rings. Usually, it takes at least a few minutes to load the parameters for PF-AR ring, injection into the ring and reload the parameters for SuperKEKB rings. It is not acceptable for the SuperKEKB rings to stop injection for such a long time, otherwise most of the particles in the SuperKEKB rings have been lost. Another difference is the emittance requirement to the SuperKEKB rings. To avoid emittance growth in the LINAC, optics and orbit setting should be optimized for both of the rings which require different injection energy.

To satisfy these requirements, about one hundred pulsed magnets were installed in 2017 and 2018 Using these magnets, magnetic field can be changed shot by shot in 20 ms and was optimized for each destination.

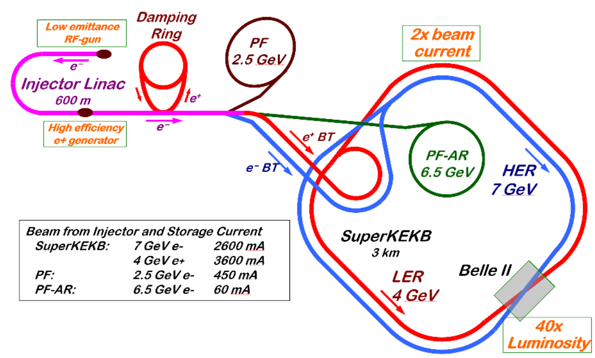


Figure 1: A schematic view of the electron and positron accelerator complex in the KEK.

Table 1: Required Injection Parameters

Stage	KEKB		SuperKEKB	
Beam	e^-	e^+	e^-	e^+
Energy (GeV)	8.0	3.5	7.0	4.0
Life time (min.)	200	150	6	6
Emittance (μ m)	310	1400	40/20 (H/V)	100/15 (H/V)
Bunch charge (nC)	1	1	4	4
Energy spread (%)	0.13	0.13	0.07	0.16

In this paper, the control system and software are mainly described. Details of the hardware such as magnet, power supply circuit are found in the reference [4].

OVERVIEW OF THE SYSTEM

Concept and Overview

There are many requirements for the power supply to truly realize the required flexible operation. In addition, since the required number of the power supply is not a few, unit cost,

* yoshinori.enomoto@kek.jp

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEDPL01

IN-PLACE TECHNOLOGY REPLACEMENT OF A 24x7 OPERATIONAL FACILITY: KEY LESSONS LEARNED AND SUCCESS STRATEGIES FROM THE NIF CONTROL SYSTEM MODERNIZATION

M. Fedorov, G. Brunton, C. Estes, B. Fishler, M. Flegel, A. P. Ludwigsen, M. Paul, S. Townsend
Lawrence Livermore National Laboratory, Livermore, USA

Abstract

The National Ignition Facility (NIF) is the world's largest laser system for Inertial Confinement Fusion (ICF) and High Energy Density (HED) experiments. Design of the NIF control system started in the 1990s, incorporating established hardware and software technologies of that era. The architecture of the control system has stood the test of time, successfully scaling up to a full 192 laser beam configuration in 2009, and then transitioning to 24x7 operations and sustaining 400 shots annually since 2016. The control system has grown with NIF to add new major capabilities, such as cryogenic layering, a petawatt-class laser, 3D neutron imaging and others. In parallel, with scaling up and efficiency optimizations, the software had to adapt to changes dictated by the fast-paced computer industry. Some of our originally chosen technologies have become obsolete and replaced by new programming languages, frameworks and paradigms. In this paper, we discuss how the NIF control system has leveraged the strengths of its distributed, crossplatform architecture to successfully modernize "in-place" computing platforms and programming languages without impacting the demanding experiment schedule.

INTRODUCTION

Large experimental physics facilities embody significant investment of societal resources and they are expected to last, gainfully generating scientific knowledge for 20-40 years. Computer industry moves much faster, periodically forcing facilities' control systems into major modernizations to address technology obsolescence, cybersecurity and paradigm shifts in programming technologies.

Continuous pursuit of the experimenters' goals at these facilities may also mean that there will never be an extended downtime for a comprehensive "full rewrite" of the control system. The alternative is an incremental "in-place" upgrade in parallel with scientific operations, stretching the modernization over numerous small windows (2-4 hours) in the facility schedule. By overlapping with other maintenance activities, the "in-place" approach does help to minimize the overall downtime budget. However, each of these numerous upgrades carries a risk of an unexpected behavior change or a system incompatibility. While the planned downtime budget can be frugally negotiated, facilities have zero tolerance for unplanned downtime since it directly impacts the quantity and quality of scientific output.

We explain how we have approached the "in-place" upgrade by carefully inspecting the "pillars" supporting our control system architecture. Some of these pillars had to be removed and replaced, while others stayed and served as pivots which helped our team to propel NIF Integrated Computer Control System (ICCS) towards modern technologies.

To address the unplanned downtime risk, we have expanded our automated testing by adding focused verifications of the fidelity of the software migrations, assuring that new behaviors, timings and exceptions match those of the legacy software. Finally, we have leveraged the data-driven aspect of our architecture to develop a fast and reliable "conversion-reversion" process which assures that we can always undo a migration upgrade and return the facility to normal operations in a predictable time.

EVOLUTION OF ICCS TECHNOLOGIES

Early Days

The NIF control system was designed at the end of the 1990s. Reliability and scalability were the primary concerns for hardware and software architects, which resulted in selection of proven, well established technologies with solid industry support. For the low-level, hardware-facing Front-End-Processors (FEPs) the NIF team selected VME-bus, Motorola PowerPC diskless crates running VxWorks RTOS, Fig.1.

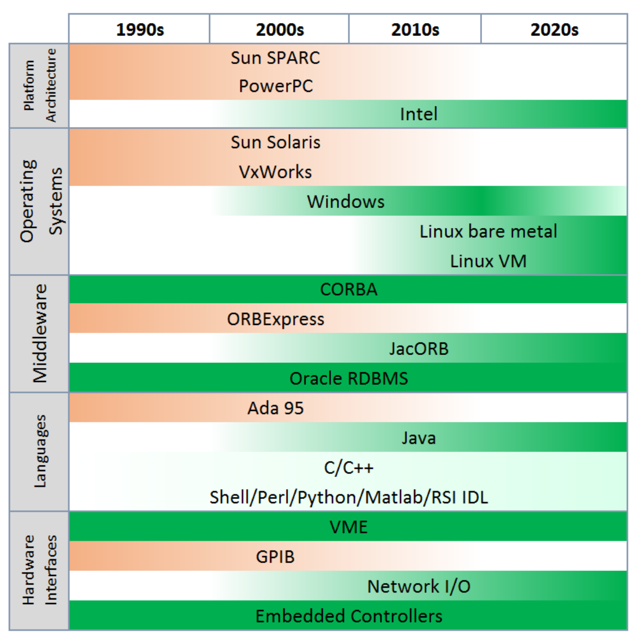


Figure 1: Evolution of ICCS technologies: fading red color indicates legacy technologies on their way to obsolescence. Deepening green illustrates growth of modern alternatives. Solid green highlights enduring pillars of our architecture.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEDPL02

AliECS: A NEW EXPERIMENT CONTROL SYSTEM FOR THE ALICE EXPERIMENT

T. Mrnjavac*, K. Alexopoulos†, V. Chibante Barroso‡, G. Raduta§, CERN, Geneva, Switzerland

Abstract

The ALICE Experiment at CERN LHC (Large Hadron Collider) is undertaking during Long Shutdown 2 in 2019-2020 a major upgrade, which includes a new computing system called O² (Online-Offline). To ensure the efficient operation of the upgraded experiment along with its newly designed computing system, a reliable, high performance and automated experiment control system is being developed with the goal of managing all O² synchronous processing software, and of handling the data taking activity by interacting with the detectors, the trigger system and the LHC. The ALICE Experiment Control System (AliECS) is a distributed system based on state of the art cluster management and microservices which have recently emerged in the distributed computing ecosystem. Such technologies will allow the ALICE Collaboration to benefit from a vibrant and innovating open source community. This communication illustrates the AliECS architecture. It provides an in-depth overview of the system's components, features and design elements, as well as its performance. It also reports on the experience with AliECS as part of ALICE Run 3 detector commissioning setups.

INTRODUCTION

The O^2 Computing System

The ALICE experiment [1] is undergoing a major upgrade [2] which is being deployed during LHC's Long Shutdown 2 (2019-2020) in preparation for LHC Run 3. The new and upgraded detectors result in a significantly increased data rate, and in order for the data processing to keep up, a new computing system called O^2 [3] is being designed, developed and deployed.

In its production stage, the O^2 computing system will consist of 100,000s of processes, deployed over roughly 1000 heterogeneous nodes, fulfilling roles including data readout, processing, storage and auxiliary services. The system will read out 27 Tb/s of raw data and record 800 Gb/s of reconstructed data.

The data driven components of the O^2 computing system will run on two main typologies of computing nodes: FLPs (First Level Processors) and EPNs (Event Processing Nodes). Each FLP is fitted with CRU (Common Readout Unit) [4] or C-RORC (Common Readout Receiver Card) [5] hardware, depending on the detector. These PCI-Express cards are capable of two way communication with detector front end electronics.

* e-mail: teo.m@cern.ch
† e-mail: kostas.alexopoulos@cern.ch
‡ e-mail: vmcb@cern.ch
§ e-mail: george.raduta@cern.ch

The O^2 computing system will be capable of two kinds of data-driven workflows: synchronous operation, intended to be *synchronous* with detector readout, and asynchronous operation, which will take place at any time regardless of detector or beam conditions. Most nodes are expected to run dozens of processes of different kinds, including long running services, WLCG-like (Worldwide LHC Computing Grid) environments for asynchronous processing, and data-driven process workflows. Synchronous workflows operate on data coming from detector data links, thus they must run in the O^2 facility at LHC Point 2. Asynchronous workflows do not have this constraint, they can therefore run at any time on WLCG nodes, or on O^2 facility resources when they are not needed for synchronous operation.

The O^2/FLP Computing Cluster

O^2 is being developed as a complete solution for the data processing needs of the ALICE experiment during Run 3, but the O^2 compute system is split up in two separate computing clusters due to significant differences in requirements between FLPs and EPNs. This division yields the O^2/FLP computing cluster and the O^2/EPN computing cluster, both deployed at LHC Point 2.

The fundamental difference between these two main kinds of nodes stems from the fact that FLPs have direct fiber links to detector front end electronics, making them permanently bound to a specific detector or detector component. Different FLPs may also have a variable number of CRU or C-RORC cards, and different system specifications. FLPs are not interchangeable, thus the O^2/FLP cluster is inevitably a heterogeneous environment. On the other hand, EPNs do not have direct links to detector front end electronics, and they are largely interchangeable, with the purpose of hosting scalable processing workflows which can be replicated on as many EPNs as needed, depending on the required workload.

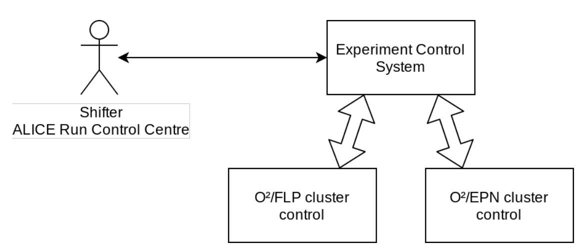


Figure 1: O^2/FLP and O^2/EPN cluster control with respect to the ALICE Run Control Centre.

Each of the two computing clusters has its own specialized control system. Ultimately, the O^2 system as a whole will be controlled via a single user interface, an ECS (Experiment Control System) solution in the ALICE Run Control Centre (see Fig. 1).

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEDPL04

CONSOLIDATION AND REDESIGN OF CERN INDUSTRIAL CONTROLS FRAMEWORKS

P. Golonka†, F. Varela-Rodriguez, CERN, Geneva, Switzerland

Abstract

The Industrial Controls Frameworks, JCOP [1, 2] and UNICOS [3, 4], have been employed to develop hundreds of critical controls applications in multiple domains, like the LHC Experiments Detector Control System, accelerator complex (cryogenics, powering, interlocks) and technical infrastructure, leading to an unprecedented level of homogeneity. These frameworks, used by around a thousand of developers worldwide, will now undergo a major consolidation and re-engineering effort to prepare them for the new challenges of the next 20 years in the High Luminosity (HL) LHC era, as well as to streamline their maintenance.

The paper presents the challenges that will be faced during this project due to the breadth of technological stack and large code-base contributed over two decades by numerous authors. Delivery of innovation induced by evolution of technologies and re-factoring of the ageing code must be done in a way that ensures backward-compatibility for existing systems. The vision and the current state of the frameworks is discussed, alongside the main deliverables planned in the medium term. Lessons learnt, optimizations of processes to make best use of the available resources and efforts towards open-source licensing of the frameworks are also presented.

INTRODUCTION

The development of the two CERN Industrial Controls Frameworks (JCOP and UNICOS) started in early 2000's. With the approach of the LHC era, the CERN Joint Controls Project [5] was set up in 1998 to streamline the development efforts for the controls of the LHC experiments and avoid duplicated efforts. The recommendation of the project, approved by the experiments, was to select a common set of industrial COTS components (at the hardware, middleware and software layer), with centralized effort to validate, enrich, enhance, and provide support. These components should be used as standardized and trusted building blocks. Key to the success of JCOP has been the strong partnership established with the commercial providers. At that time, high and low voltage power supplies and VME crate controllers were chosen with maintenance contract signed with vendors, as well as the OPC Data Access (DA) as the preferred middleware and PVSS SCADA, later rebranded to WinCC Open Architecture (OA) [6] as supervisory system. This careful selection/validation process involved around 10 man-years of the central team, yet it spared similar scattered efforts from each of the LHC experiment's controls team.

† Piotr.Golonka@CERN.CH

At the same time another initiative was started to develop the control system for cryogenics of the LHC [7], adopting the formal approach for process decomposition, modelling (IEC61512-1) and engineering that employed industrial components.

The project, called UNICOS, chose to standardize on Siemens and Schneider Programmable Logic Controllers (PLC) as well as the supervisory system and soon converged to use the same SCADA as selected by the JCOP project in 2001.

Both Frameworks are complementary in their approach and employ common components in their stack of technologies. Whereas the JCOP Framework provides guidelines and tools to implement flexible and complex control systems that comprise a large variety of equipment, like the detector control system of the LHC experiments, the UNICOS Framework defines a strict engineering process and provides tools to develop PLC-based control systems and allowing to generate parts through templates. Component-based approach used in both allow to compose functionality as required by the control system and extend it as necessary. UNICOS employs numerous components delivered by the JCOP Framework as shown in Fig. 1.

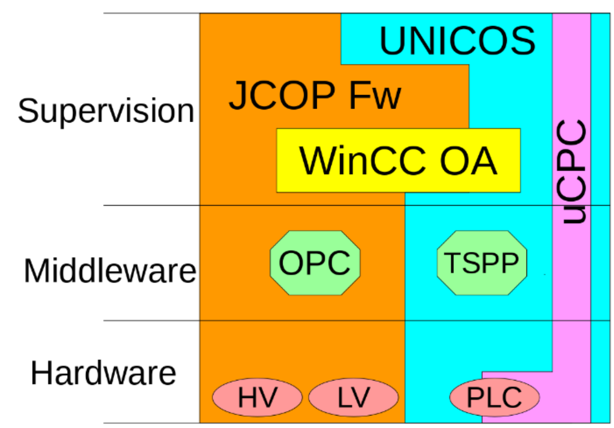


Figure 1: JCOP and UNICOS frameworks relation.

The JCOP framework is developed as a collaboration between the LHC Experiments and CERN industrial controls group, where each of the partners develops, maintains and supports generic components that are widely used across CERN. The UNICOS project is entirely developed by the industrial controls group and is provided to many groups at CERN to engineer their control systems.

The first controls applications at CERN based on the Industrial Controls Frameworks entered production in 2001. Today, the Industrial Controls Frameworks are in use in more than 600 critical controls applications to ensure the smooth running of the accelerators, detectors and technical infrastructure contributing to unprecedented performance of the research facilities. The flexibility of the frameworks

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEDPR01

CUMBIA: GRAPHICAL LIBRARIES AND FORMULA PLUGIN TO COMBINE AND DISPLAY DATA FROM TANGO, EPICS AND MORE

G. Strangolino, Elettra-Sincrotrone Trieste S.C.p.A., Basovizza, Trieste, Italy

Abstract

Cumbia libraries offer the next generation core (C++) and graphical (Qt) software to write complete and lightweight applications that provide a unified user interface, regardless of the underlying engine (Tango, EPICS, WebSocket, ...) With the new formula plugin, results can be manipulated and combined by JavaScript functions and displayed in the appropriate widget. Qt has a deep JavaScript integration that allows efficient introduction of program logic into the application. Using the Qt + QML technologies, apps can be designed for the desktop and mobile devices. Switching between the two targets is an immediate operation. A WebSocket based service has been used to test Qt + QML mobile applications on portable devices. It makes it possible to connect to Tango and EPICS without their installation. A new tool called *la-cumparsita* lets non-programmers use the Qt designer to realize complete applications ready to communicate with the control system in use: Tango, EPICS or any other abstraction framework (e.g. WebSocket). These apps seamlessly integrate with the desktop. Most demanding users can integrate JavaScript functions and use them as data sources for the GUI elements.

STRUCTURE OF THE FRAMEWORK

The *cumbia* library is made up of several modules. The core and the engine specific ones are written in pure C++, while those providing graphical elements employ the *Qt* framework [1]. Figure 1 outlines the relationship between the main modules that compose the software.

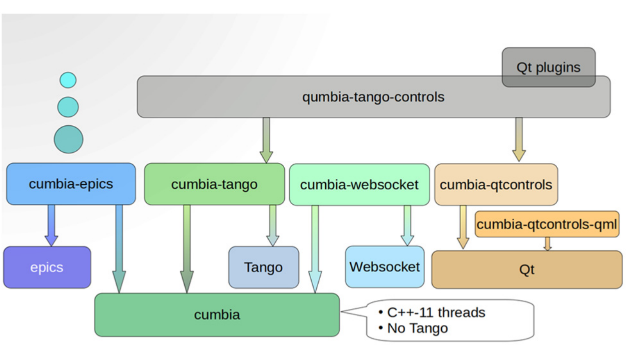


Figure 1: Relationship between *cumbia* main modules.

The next paragraphs describe each component in more detail.

MODULES

Cumbia Base Module

Cumbia is a component that offers a carefree approach to multi thread application design and implementation. The user writes *activities* and decides when their instances are started and to which thread they belong. A *token* is used to register an *activity* and those with identical tokens are run in the same thread. Work is done inside the *init*, *execute* and *exit* methods. The library guarantees that they are always called in the activity thread. From within *init*, *execute* and *exit*, computed results can be forwarded to the main execution thread, where they can be used to update a graphical interface. Data is exchanged by means of a dedicated key/value bundle, named *CuData*.

Cumbia-tango

Cumbia-tango integrates *cumbia* with the Tango [2] control system framework, providing specialised *activities* to read, write attributes and impart commands. Readings are accomplished through either a poller or the Tango event system, for those attributes suitably configured. Write operations are always executed in an asynchronous thread and the result is delivered later in the main thread. *Cumbia activities* are employed by the module to setup the connection, access the database, subscribe to events or carry out periodic readings. Progress and result events are delivered to the main thread from the background *activity*. As stated in the previous section, activities identified by the same *token* belong to the same thread. Here, the *token* is the Tango device name. Applications that connect to the Tango control system will typically instantiate a *CumbiaTango* object that defines which kind of threads will be used (e.g. *Qt*'s for graphical interfaces) and thereafter parametrizes each reader or writer. Several modern design patterns have been exploited to provide a flexible and scalable architecture. Singletons have been completely replaced by *service providers* in order to offer services For graphical applications. The component provides helpful classes that can be used from outside an activity to access devices, fetch database properties or interpret exceptions raised from within the engine. Aside from these utilities, one would not normally employ this module directly. *Cumbia-qtcontrols* and *qumbia-tango-controls* is where to look for when the integration between the control system and the user interface is the objective.

Cumbia-epics

Cumbia-epics integrates the Experimental Physics and Industrial Control System (EPICS) [3] control system with *cumbia*. The interaction with the lower level *cumbia* base component and the interface offered to clients is equivalent to the *cumbia-tango*'s. Configuration, monitor and *put* operations are currently implemented. Data is exchanged through the same aforementioned key/value structure (*CuData*). Differences between the EPICS and Tango engines are concealed and utmost effort has been taken to unify the representation of the results. For example, Tango *Max value* database attribute property and EPICS *upper_disp_limit* from *dbr_ctrl* data are both stored into the

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEDPR02

BENEFITS OF LOW CODE DEVELOPMENT ENVIRONMENTS ON LARGE SCALE CONTROL SYSTEMS

B. Lefort, V. Costa, CERN, Geneva, Switzerland

Abstract

The rapid evolution of science and of scientific projects usually implies high levels of mobility among researchers, engineers and applied scientists. In parallel, software development has been getting easier as computing technology has evolved. A direct consequence of these two paradigms is the proliferation of custom, sometimes small, ad-hoc software. This software, usually quickly developed and with low code hygiene, later becomes a burden to the organization, especially if the original developer and responsible departs. Based on this experience many organizations are now successfully adopting low-code application development. Inspector is a low-code development platform to design control interfaces. It features a visual interface composer, a visual programming language and supports small integrated scripts in Python. More than 600 Inspector applications are actively used at CERN. We explain how developers with little experience of writing software can create applications that they could not otherwise explicitly code for themselves. Finally, we demonstrate how Inspector offers enhanced security, higher productivity and maintenance relief by delegating the core software development and maintainability to high skill developers and IT members.

INTRODUCTION

The number of PhDs and postdocs in science has grown substantially. A report in Nature reported a jump by 150% in the number of postdocs between 2000 and 2012 (see Fig. 1) and the growth shows no sign of slowing since then. [1]

PhD graduates and Postdocs confront a dwindling number of academic jobs. As an example, only 15% of PhD graduates can attain academic positions in the USA [2]. Many of them go the entrepreneurial route and become involved in start-ups, research labs or commercial R&D centres.

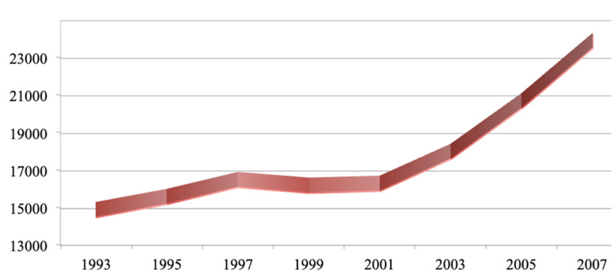


Figure 1: Annual number of science and engineering doctorates graduating from US universities.

The typical maximum funding period for doctoral students is four years. There is no set length for a postdoctoral researcher as it depends on a number of factors such as the university, country of research, PI, or funding. That being said, most positions are two to three years even it they can be extended up to 6 years.

Over the past 15 years, the number of public workers on short-term contracts has increased. In France they now represent 35% of university staff and 27% at research institutions. Contracts tend to be very short. 80% of them are lasting less than 2 years [3].

Consequently, Academia is facing a high employment turnover rate combined with short-term contracts. In regards to software engineering, such conditions make quality and productivity competing objectives. Code quality is naturally seen as less important when compared to fast/cheap development, as such, software projects are showing signs of poor code quality, undetected vulnerabilities and low code readability – contrary to the fundamentals of software maintainability.

Considering the trends, we will show how a low-code application, such as Inspector, a Rapid Application Development (RAD) framework, gives access to application development for people with little experience of writing software and how it reduces the number of Lines Of Code (LOC) that has to be maintained when the person who developed and maintains it departs.

To boost the productivity, Inspector proposes a separation between the UI and the software technology, essentially allowing the creation of zero code GUI in order to diminish the cost of developing and maintaining such applications. Inspector itself is built on proven technologies (such as Java) and takes care of critical software aspects such as multithreading, data communication and application reliability. Final application developers don't need to worry about these aspects; they can create their entire application using the GUI and tiny scripts for complex operations. This concept also introduces indirect benefits, for instance all Inspector applications are uniform. They'll look and behave the same way. This helps create a sense of familiarity and control that helps users going seamlessly from one application to another without the need to learn new ropes to get around.

LINES OF CODE AND CODE QUALITY

Capers Jones, an American specialist in software engineering methodologies, has compared many methodologies (RUP, XP, Agile, Waterfall, etc.) and programming languages over thousands of projects and has determined that programmers write between 325 and 750 production lines of code (LOC) per month. [4, 5] These numbers apply to software scientists working in teams that follow a strict development cycle where specification is followed by development and precede unit & integration tests.

Short-term contracts workers in a lab environment are typically result driven. User stories such as "develop a

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEDPR03

SYNOPTIC GUIS IN NSRC SOLARIS FOR BEAMLINES AND ACCELERATORS VISUALIZATION AND CONTROL

M.K. Fałowski†, T.R. Noga, N. Olszowska, M. Zajac
National Synchrotron Radiation Centre, SOLARIS, Krakow, Poland

Abstract

High demand from scientists and operators to create new, clear and intuitive SCADA graphical interfaces for new beamlines and replace or supplement existing beamlines' and accelerators' graphical user interfaces is a challenging task. This is not only time consuming but very often requirements from users vary, change quickly and even sometimes they are mutually exclusive. To meet this challenge and provide clear, scalable and ergonomic graphical user interfaces, SOLARIS chose 'Taurus' and 'svgsynoptic2' to create synoptic applications which allow to visualize and control beamlines and accelerators with ease. In addition, it was decided to use identical scheme of visualization and control for synoptic applications on all beamlines, so scientists can get used to it, even if they carry out research on different beamlines. This paper presents the overall architecture and functionality of the applications.

INTRODUCTION

One of the most important roles of the control systems is to show the actual state of controlled process. Visualisation of the state of the process is crucial for operators to properly control it and react in any case of event or emergency occurrence during the process. This task is achieved with SCADA systems. One of the elements of SCADA are synoptic views, which are graphical representation of the process. They should provide as much information about control system as possible but also preserve readability, be reliable and intuitive.

In NSRC SOLARIS there are two accelerators, linear and synchrotron, two fully operational beamlines, one during commissioning and next four under construction. Each system contains hundreds of devices and signals. For now, the main task for SOLARIS is to deliver stable beam for scientists with interruption and breaks as short as possible. To achieve this goal it is critical to monitor accelerators and beamlines and in case of any accidents fast recognition and repair of the fault.

PREVIOUS SOLUTIONS

Since the start of operation, applications to represent beamlines and accelerators in SOLARIS were prepared with JDraw synoptic and Qt GUIs developed with Taurus. They had a lot of limitations and were not very intuitive. Moreover, they could not represent entire subsystems in one view and had to be divided into tabs or segments.

Example of an application is showed in Fig. 1. It is used to visualize segments of the beamline or the storage ring. This application parses configuration file and generates Taurus widgets. It has many disadvantages, such as very complicated and hard to modify code (e.g. adding new type of device or adapting to newer version of dependency libraries was very time consuming), switching between segments was taking a lot of time, resulting in not showing any panel for short period, and configuration files had custom and hard to read format. In addition, non-graphical representation required operators to learn positions of the elements to quickly recognise state of the subsystems. Last but not least, application used custom and unintuitive colours for defining states (e.g. orange colour represented closed state, where in Tango it represents alarm state). Next example is the application responsible for displaying interlocks shown in Fig. 2. It had many tabs so searching for an interlock was difficult and sometimes caused oversight of it by operators. Last example is JDraw application of LINAC showed in Fig. 3. Due to static view, LINAC segments had to be separated into many tabs.

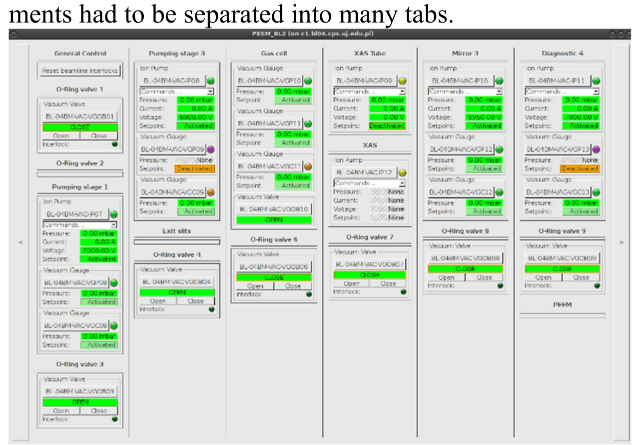


Figure 1: Application representing one of the segments of the beamline. It contains states of the devices, such as valves and thermocouples, and allows to control them.

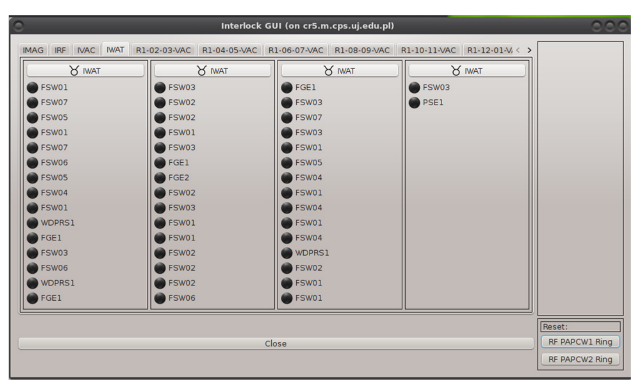


Figure 2: Application responsible for displaying interlocks in the accelerators.

† michal.falowski@uj.edu.pl

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEDPR04

THE WEB AS THE PRIMARY CONTROL SYSTEM USER INTERFACE*

R. Neswold†, B. Harrison‡, Fermilab, Batavia, USA

Abstract

Fermilab's Control System uses a proprietary application framework written decades ago. Considered state-of-the-art at one time, the control system now lacks many features we expect from a modern interface and needs to be updated. Our investigation of Web browsers and JavaScript revealed a powerful, rich, and state-of-the-art development environment. We discuss JavaScript frameworks, JavaScript language features, and packaging tools. We also discuss issues we need to resolve before we are confident this can become our primary application platform.

INTRODUCTION

We set out to reimagine accelerator control applications. Exploring modern development tools and current best practices help us move away from existing, aging dependencies while improving the users' experience.

Fermilab's parameter page application is broadly considered to be the workhorse of the control room. The parameter page allows operators to freely request live data from any device in the control system. Key features include the ability to manipulate devices, query the control system for meta-information about a given device, plot the data over time, and provide textual context for the set of devices. The parameter page application also allows users to save their set of queries and notes for easy retrieval and, therefore, has become a simple way for machine experts to create basic applications. Many other applications are structured views of data that allow the user to read and manipulate data in a predefined, restricted, way.

We found these features in common with modern dashboard applications. Dashboards offer a series of configurable panels that can be added to a view and saved for later. Views are for a specific task or related data. Dashboards in the browser allow for easy shared access to saved views. Browsers also provide many accessibility features that would require lots of effort to implement in traditional applications. We can consider JavaScript applications so readily because we have an existing client library that allows for streaming accelerator data from the control system to JavaScript via WebSockets.

Identifying the dashboard's component-like structure led us to investigate web application frameworks that support self-contained and reusable code. We aim to provide operators with a blank canvas and the ability to intuitively add new panels with standard components. They then save this view and get a unique endpoint that they can return to in the future. We hope that this component-based design encourages developers to reuse and modify code rather than reimplement very similar features over many applications. In our investigation, we looked at framework maturity, adoption, and documentation. While others like Angular, VueJS, and Java fulfilled the requirements, ReactJS's ubiquity, quality documentation, and ease of use made it stand shoulders above.

* This manuscript has been authored by Fermi Research Alliance, LLC under Contract No. DE-AC02-07CH11359 with the U.S. Department of Energy, Office of Science, Office of High Energy Physics.
† neswold@fnal.gov
‡ beau@fnal.gov

INVESTIGATING REACT

React[1] is a JavaScript library used to create "components" along with an engine that efficiently renders changes in the DOM. A component is a JavaScript module that renders HTML elements and manages the state associated with them. Each component is self-contained; the outside state is provided when the component is created, but from that point on, the component updates its own, internal state. Since components are insulated from external effects, they can easily be combined to make more complicated components. React-based applications are nothing more than a series of nested components.

The React team provides a command-line tool to help set up a new React project called `create-react-app`. This tool creates a directory tree containing initial, sample JavaScript source along with the necessary configuration files to build your application. It also sets up an area used for creating unit tests for your project. As your application grows, tests should be added to make sure previously working features still work. The build environment includes another powerful feature where, after successfully building the project, a web server is launched to run your application. The server listens on the localhost address and opens a tab in the default browser on your desktop displaying your application.

React projects typically use an extension to JavaScript called JSX[2] which makes the rendering code much easier to understand. JSX allows you to use HTML-style tags in your JavaScript code, rather than the explicit function calls it takes to create the elements. Files using JSX notation have the file extension `.jsx`. When building the application, files with this extension are processed by converting any JSX notation into calls to `createElement()` rendering a `.js` file. The resulting file has the normal `.js` extension and can be loaded by the browser.

Aside from the component hierarchy, applications also require some logic to manage global state, interface to 3rd party libraries, and even to pass state between components. This logic can get complicated, and due to JavaScript's dynamic typing, simple mistakes aren't necessarily caught until the code path is executed, resulting in run-time errors. Fortunately, we found that we could use Microsoft's TypeScript language with our React projects, which eliminated a whole class of bugs in our code and helped speed up our development.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEMPL001

AN APPLICATION OF MACHINE LEARNING FOR THE ANALYSIS OF TEMPERATURE RISE ON THE PRODUCTION TARGET IN HADRON EXPERIMENTAL FACILITY AT J-PARC

K. Agari†, H. Akiyama, Y. Morino, Y. Sato, A. Toyoda, KEK, Ibaraki, Japan

Abstract

Hadron Experimental Facility (HEF) is designed to handle intense slow-extraction proton beam from 30-GeV Main Ring (MR) of Japan Proton Accelerator Research Complex (J-PARC). A total amount of 3.6E19 protons in the 2018 run were irradiated on the production target in HEF. In order to evaluate soundness of the production target, we have analysed long-term variation of temperature rise on the target, which can be affected by beam conditions. Predicted temperature rise measured with thermocouples mounted on the target was calculated from the existing training data of beam intensity, spill length (duration of beam extraction in 5.2 second accelerator cycle), and beam position on the target, using a linear regression analysis with a machine learning library, scikit-learn. As a result, predicted temperature rise shows good agreement with the measured one. We have also examined whether the present method of the predicted temperature rise from the existing training data can be applied to the new data in the future runs. The present paper reports the current status of the measurement system of temperature rise on the target with machine learning in detail.

INTRODUCTION

Hadron Experimental Facility [1] (HEF) at Japan Particle Accelerator Research Complex (J-PARC), shown in Figure 1, is designed to handle intense slow-extraction proton beam from 30-GeV Main Ring (MR). The period of beam extraction from MR to HEF is 2 seconds and the operation cycle is 5.2 seconds.

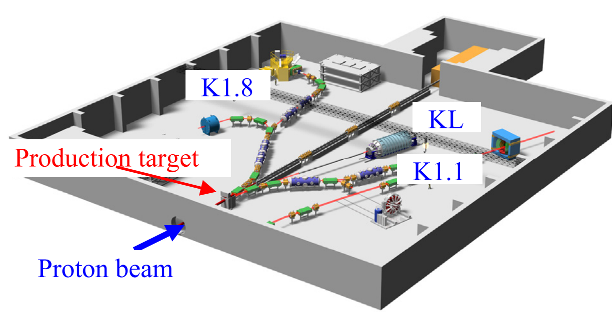


Figure 1: An illustration of the HEF.

#agari@post.kek.jp

Production Target

The production target [2], currently using at the HEF, is made of a gold and a copper block with coolant stainless pipes, as shown in the Figure 2. Gold is chosen for high density, high thermal conductivity, and good chemical stability. The current target is designed to be capable for up to 50-kW proton beams. The dimension of gold target is $15^W \times 6^H \times 66^L$ [mm]. The gold target is divided into 6 blocks in order to reduce thermal stress by beam irradiation. A double-headed design enables to switch the one target to the other quickly and remotely on demand. The water coolant pipes, embedded in the copper block, are made of stainless steel to avoid erosion and corrosion. The gold target, the copper block, and the coolant pipes are bonded with a Hot Isostatic Pressing (HIP) process. The production target is enclosed with an airtight chamber filled with circulating helium gas. The beam entry and exit are covered with beam windows.

In order to monitor temperature rise of the target, six thermocouples, named as TC1 to TC6, are mounted on top of each gold block along the beam direction. We also monitor temperatures of the copper block, the water and helium gas pipes, and the edge of the beam windows.

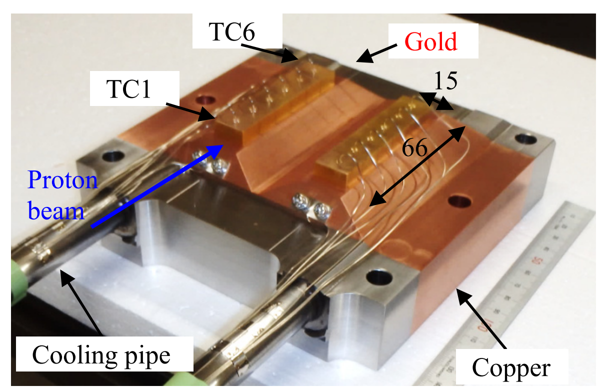


Figure 2: A photograph of production target.

During continuous 50 kW beam operation, temperature on the target rise close to 342K, the allowable limit estimated by a FEM stress analysis. In the point of soundness of the target, unexpected temperature rise by cumulative damage on the target may occur during a normal beam operation. Thus, it is important for damage control of the target to distinguish an unexpected temperature rise by damage from a normal drift due to fluctuation of beam conditions.

Recently, a number of machine-learning methods has been widely used in a variety of fields, and many easy-to-

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEMPL002

PROJECT Nheengatu: EPICS SUPPORT FOR CompactRIO FPGA AND LabVIEW-RT

D. Alnajjar*, G. de S. Fedel, J. R. Piton,
Brazilian Synchrotron Light Laboratory, Campinas, Brazil

Abstract

A novel solution for integrating EPICS with Compact RIO (CRIO), the real-time embedded industrial controllers by National Instruments (NI), is proposed under the name Nheengatu (NHE). The CRIO controller, which is equipped with a processor running a real-time version of Linux (Linux-RT) and a Xilinx Kintex FPGA, is extremely powerful for control systems since it can be used to program real-time complex data processing and fine control tasks on both the Linux-RT and the FPGA. The proposed solution enables the control and monitoring of all tasks running on Linux-RT and the FPGA through EPICS. The devised solution is not limited to any type of CRIO module, setup or combination of modules. Its architecture can be abstracted into four groups: FPGA and LabVIEW-RT interface blocks, the Nheengatu library, Device Support and IOC. The Nheengatu library, device support and IOC are generic - they are compiled only once and can be deployed on all CRIOs available. Consequently, a setup-specific configuration file is provided to the IOC upon instantiation. The configuration file contains all data for the devised architecture to configure the FPGA and to enable communication between EPICS and the FPGA/LabVIEW-RT interface blocks.

INTRODUCTION

Experimental Physics and Industrial Control System (EPICS) [1] is an open-source set of libraries that helps building distributed soft real-time control systems for scientific instruments, like the ones that are used in the LNLS UVX beamlines [2], and those to be used in SIRIUS [3]. EPICS has been used to control LNLS experimental stations' equipment. Consequently, using EPICS to control CompactRIO (CRIO) [4] devices, the real-time embedded industrial controllers by National Instruments (NI), would be highly favourable.

The CRIO chassis contains a Xilinx FPGA and an Intel processor running an NI version of Linux. It also contains several slots where NI C-series modules containing peripherals can be plugged in. The peripherals can be accessed by the FPGA (hardware) and the LabVIEW-RT (Linux real-time operating system). Both the FPGA and LabVIEW-RT can be programmed using a visual programming language (LabVIEW) [5]. In that development environment, the user develops a software (VI) where building blocks can be connected to generate the desired circuitry to be implemented on LabVIEW-RT or the FPGA. Whether the LabVIEW-RT or the FPGA or both are chosen, it depends on the system's requirements. The FPGA can be used to perform high-speed processes and high-speed fine-control. LabVIEW-RT is slower than FPGA; however, it provides the ease and power of programming much more complex task limited by the processor speed and memory.

EPICS runs on Linux or Windows, so hypothetically it should run on the Linux-RT; however, extra libraries need to be developed to allow access of variables available on the FPGA or LabVIEW-RT. Several options developed by the EPICS community/industry were proposed and are being used. NI EPICS was developed by National Instruments [6], which is good for simple demands; however, it is limited in its functionalities. Consequently, IRIO [7] and CA LAB [8] were developed. IRIO developed an EPICS device driver to send and receive data to and from FPGA peripherals; however, all peripherals connected to the CRIO platform need to be supported by the library, and the library needs to be recompiled with every new setup. CA Lab uses libraries developed on top of LabVIEW-RT along with EPICS native libraries. All variables need to pass through LabVIEW-RT whether they belong to LabVIEW-RT or not, and it is difficult to integrate the rest of EPICS support software with this infrastructure (i.e. synApps [9]). For our use in SIRIUS, it would be highly advantageous to our development/debug cycle that the proposed solution can provide flexibility in several aspects, and they are:

- Allow EPICS to read variables generated on the FPGA VI, obtained from FPGA peripherals or generated on LabVIEW-RT VI running on CRIO.
- Allow EPICS to write to variables on the FPGA VI, that can be used internally or connected to a peripheral, or LabVIEW-RT VIs running on CRIO.
- Ease of integration with synApps or any other software intended to support the common requirements of particle accelerator beamlines.
- The deploy process should be as simple as possible. It is favourable that no compilation is necessary at all during the deploy process.

Given the previous requirements, there was a need to rethink a new solution that satisfies all these constraints and, consequently, Nheengatu was born.

* dawood.alnajjar@lnls.br

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEMPL004

INCEPTION OF A LEARNING ORGANIZATION TO IMPROVE SOLEIL'S OPERATION

A. Buteau, G. Abeillé, X. Delétoille, J.-F. Lamarre, T. Marion, L. S. Nadolski
Synchrotron SOLEIL, Gif-sur-Yvette, France

Abstract

High quality of service is a key mission of SOLEIL since 2007. Historically operation processes and information systems have been defined mostly on the fly by the different teams all along the Synchrotron's journey. Some major outcomes are a limited cross-teams collaboration and a slow learning organization. Consequently, we are currently implementing a holistic approach with common operational processes upon a shared information system.

Our first process is "incident management"; an incident is an unplanned disruption or degradation of service. We have tackled incident management for IT in 2015, then for the accelerators since January 2018. We are starting to extend it to beamlines since beginning 2019. As a follow-up, we will address the "problem management" process (*a problem is the cause of one or more incidents*) and the creation of a knowledge database for the operation.

By implementing those processes, the culture of continuous improvement is slowly spreading, in particular by driving blameless incident and problem analysis.

This paper will present the journey we have been through including our results, improvements and difficulties of implementing this new way of thinking.

SOLEIL OPERATION CONTEXT

The operation reliability is strategic for SOLEIL as it delivers around 6500 hours of beam time per year included 5019 hours for users [1]. SOLEIL permanent staff is composed of 350 people with:

- 70 dedicated to Accelerators operation
- 180 dedicated to 29 Beamlines.
- 70 for transversal services for facility management, ultra-high vacuum, electronics, computing…

Historically, no global governance for operations had been set up, in particular regarding incident management or preventive maintenance plans. All methods and tools have been defined "on-the-fly" by technical teams with overlaps:

- Some teams that manages hardware use a CMMS (Computerized Maintenance Management System), Maintimedia [2]
- Others use Jira [3] mainly as a software bug tracking.
- The operational teams (Accelerators operators and Experiment Hall coordinators) also use a tool called ELOG [4] to track some incidents.

The major drawbacks of such an organisation were:

- A limited cross-team collaboration as the information was logged in each team specific tool
- Incident response was focused on the technical topics instead of the service restoration, underestimating the impact on the operation of SOLEIL
- Difficulties to improve the global services, due to repetitive incidents

Our team (Controls) is daily involved in the operations of the Accelerators and the Beamlines. By considering all previously quoted issues, we have decided to define and drive the implementation of new operation processes, first for our team, and then for the whole organisation. The long terms objectives of such an approach are:

- Reinforces cross-team and collaboration, e.g. between Accelerators, Beamlines, and technical teams.
- Share the operation knowledge and add efficiency
- Provide a living, up-to-date and easy-to-use information system to support these processes.

ROLLOUT OPERATION PROCESSES FOR IT

In 2015 SOLEIL IT division decided to adopt ITIL [5] to enhance all IT services (*from office IT, Controls, IT Infrastructure*) as detail in the following paper [6]. ITIL was first used in the Controls group and after a few months helped gradually to improve the service delivered to Accelerators and Beamlines teams. "Incident Management" was considered as the first transversal process to enhance but some I.T teams were already implementing other processes such as "Problem Management", "Change Management", etc.

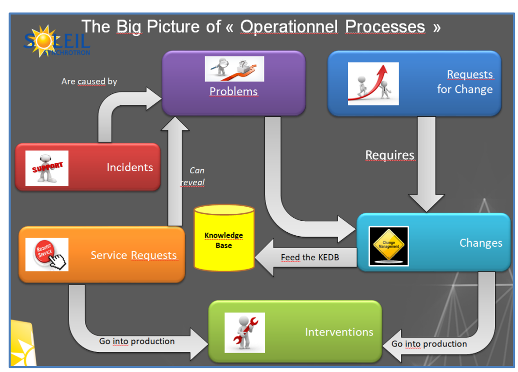


Figure 1: ITIL Operational Processes.

On the technical side, all the processes have been implemented with Jira [3] since 2015 (see Fig. 1). Jira has been linked to the SOLEIL CMMS tool, Maintimedia [2] in 2017:

- Jira is the main user portal and the defects are cascaded to Maintimedia.
- the progression of Maintimedia issues can be directly checked into Jira;

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-WEMPL005

A TECHNOLOGY DOWNSELECTION FOR SKA USER INTERFACE GENERATOR

M. Canzari*, INAF - Osservatorio Astronomico d'Abruzzo, Teramo, Italy
V. Alberti, INAF - Osservatorio Astronomico di Trieste, Trieste, Italy
P. Klaassen, M. Nicol, S. Williams, UK Research and Innovation, Edimburgh, UK
H. Ribeiro, FCUP (CICGE) Centro de Investigação em Ciências Geo-Espaciais, Porto, Portugal
S. Valame, Persistent Systems, Pune, India
V. Hardion, F. Bolmsten, H. Petri, Max-IV Institute, Lund, Sweden

Abstract

The Square Kilometre Array (SKA) project is an international collaboration aimed to design and build the world's largest radio telescope, composed of thousands of antennae and related support systems, with over a square kilometre of collecting area. In order to ensure the proper and uninterrupted operation of SKA, the role of the operator at the control room is crucial and the User Interface is the main tool that the operator uses to control and monitor the telescope. During the current bridging phase, a user interface generator has been prototyping. It aims to provide a tool for UI developer to create an own engineeristic user interface compliant with SKA User Interface Design Principle and operator and stakeholder needs. A technology downselection has been made in order to evaluate different web-solution based on TANGO.

INTRODUCTION

Square Kilometre Array (SKA) [1] is a project which aspires to build the biggest radio telescope in the world. It is composed of two arrays of radio-telescope: SKA1-LOW, consisting of about 200,000 dipole antennas that operate in the frequency range ranging from 50 to 350 MHz in Australia; SKA1-MID, composed of 197 dishes, covering frequencies from 350 MHz to 14.7 GHz, in South Africa. SKA General Headquarter is located in the UK. In order to ensure proper and uninterrupted operation, the experience and the skills of a human operator play a central role in controlling and monitoring a complex facility like SKA. Graphical User Interface (GUI) is the main tool the permits to the user to carry out such system control operations. Currently, the SKA project is in the bridging phase, the period between the Design Phase and the Construction, scheduled for the last quarter of 2020. The aim of this phase is to mitigate the risks raised during the Critical Design Phase and to develop early prototypes to adopt during the Construction phase. Considering the importance of the user's role and user interfaces, a downselection of the GUI technology has been necessary for this period.

USER INTERFACE

A large radio telescope like SKA is composed of several like electronic, mechanic, computer hardware and software components, which work in coordination with each other to carry out a high number of complex operations. The role of the operator is crucial during the monitor and control of the facility, in order to maximize the observing time, minimize the overhead and operational cost. GUI is the only tool available to the operator in performing such monitor and control operation and in collecting information useful for decision making. Considering the importance of the operator, SKA adopted the User-Centered Design (UCD) [2] Approach during the Design Phase. UCD is a process in designing and developing User Interfaces focused on the users, that has the aim to develop UIs that satisfy consumer's skills and expectations. The UCD process is summarized in the following diagram (Fig. 1).

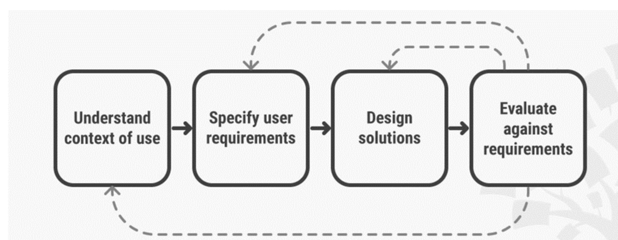


Figure 1: User Centered Design process diagram.

The outcome of the analysis, the result of a series of interviews with SKA-precursor control room operators, can be summed up in a list of requirements [3] to take into account during the technological downselection:

- scalability
- integrated tools
- extendability
- completeness

In addition, during the Bridging Phase, SKA stakeholders required other functional and non-functional requirements. In particular:

- full compatibility with Tango Controls [4], the control framework adopted by SKA for the whole telescope
- adoption of open-source software, in order to share with the Tango community problems and solutions
- web-based software, in order to exploit the accessibility and deployability features of a web application

Starting from the requirements defined during the Design and Bridging phase, a SAFe [5] team has been formed in order to choose and develop a user interface generator to

* matteo.canzari@inaf.it

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-WEMPL006

THE MINISCULE ELT CONTROL SOFTWARE: DESIGN, ARCHITECTURE AND HW INTEGRATION

C. Diaz Cano, N. Kornweibel, R.Abuter, J.Sagatowski, H.Tischer, T.R.Grudzien, European Organization for Astronomical Research in the Southern Hemisphere (ESO), Garching bei Muenchen, Germany

Abstract

This paper presents the development of the MELT (Mini ELT) Control System, to be used for testing and validating key functionalities of the Extremely Large Telescope (ELT) during AIV/commissioning and operation phase. MELT is an optical test bench with a turbulence generator, whose main objective is to deploy and validate the Central Control System (CCS) and the Wavefront control strategies. The subsystems under control are: a segmented primary mirror, a secondary mirror on a hexapod, an adaptive fourth mirror, a fast tip/tilt mirror, phasing sensor, a light source, a Wavefront sensor, a IR camera, together with their control interfaces that emulate the ELT conditions. The CCS integration layer, the Core Integration Infrastructure (CII), will be deployed to MELT for their verification and testing strategy, producing feedback to their requirements and design.

This paper describes the Control SW distributed architecture, communication patterns, user interfaces and SW infrastructure. The control algorithms are being developed separately and will be integrated into the control loop via MATLAB script API.

INTRODUCTION

MELT is a table-top emulator of the ELT (see Fig. 1), the European Extremely Large Telescope, the next generation Telescope developed by ESO [1]. It will be used for testing and validating key functionalities of the ELT, during the periods of system verification, wavefront control commissioning, through the handover to science, up to regular diagnostic, monitoring, or validation tasks during operations.

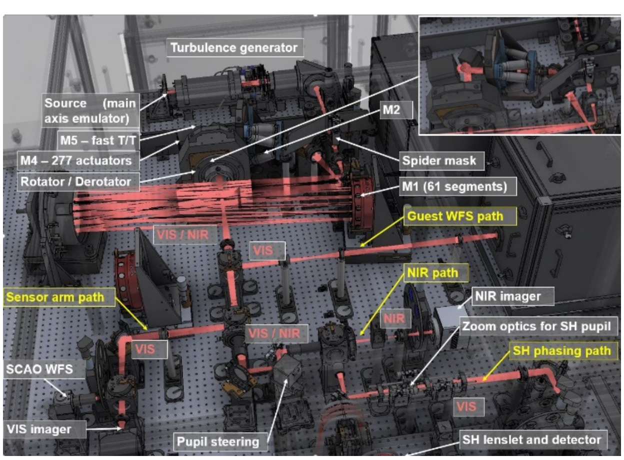


Figure 1: MELT optical test bench layout.

Another expected outcome of MELT would be to produce and validate requirements for the phasing and diagnostic station (PDS) of the ELT.

The MELT Control System (CS) Architecture follows the principles of the ELT Control Software and its Common Development Standards. Basically, the system is divided into hierarchical layers, i.e. into individual control systems associated with Telescope subsystems, collectively termed Local Control Systems, and the system that integrates these, termed the Central Control System. There are several products that have already been integrated within the bench: The network infrastructure (physical and data link layer interfaces); the messaging protocols through Core Integration Infrastructure (CII) middleware abstraction layer (MAL); the Instrument Control Framework (IFW); and the ELT Development Environment. The overall Software counts more than 550 files and 65K LOC, split in different programming languages, e.g.: C++/C (35K), Java (27K) and Python (11K).

SYSTEM DESCRIPTION

General Layout

MELT has been used as a precursor to the definition of user requirements, functional analysis, and define the most relevant functions. The CS block diagram (Fig. 2) describe the components functions throughout the optical path.

- Source: Laser driven incoherent white light in the wavelength of 500-1700nm, though a 25um multimode fiber.
- M1 active segmented mirror: consisting of 61 segments, each driven by 3 piezos to control piston, tip, and tilt with a free mechanical stroke of 15 um for wavefront control.
- M2 hexapod: hexapod is a compact 6DOF parallel kinematics system for the positioning and adjustment of precision elements with a resolution of 50 nm
- M4 Deformable mirror: ALPAO 277 actuator deformable mirror with a clear aperture of 24.5 mm, based on electro-magnetic actuators.
- Sensor arm: Fast tip/tilt (M5) and VIS imager, SCAO SH WFS 256x256 pixel with 207 um lenslets, 16 x 16 subapertures on a 3.3 x 3.3 mm pupil.
- IR Path: Before entering the IR path, the beam passes by the pupil stabilization tip/tilt mirror, with a fast full

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEMPL007

EPICS CONTROLLED WIRELESS SENSORS

M. Rolland[1], K. J. Gofron†, Brookhaven National Laboratory, Upton, USA
[1]also at Stony Brook University, Stony Brook, USA

Abstract

For sensor application, wireless technologies are more portable and convenient than their wired counterparts. This is especially true in scientific user facilities, where many environmental factors must be recorded at many locations simultaneously during data collection. Using wireless technologies, scientists can often reduce cost while maximizing the number of sensors without compromising sensor quality. To this end, we have developed EPICS controllers for both Bluetooth Low Energy (BLE) sensors and Zigbee sensors. For BLE, we chose the Nordic Thingy:52 for its low cost, high battery life, and impressive range of sensors. The controllers we developed combine EPICS base functions, the Bluetooth generic attribute data structure library, and multithreading techniques to enable real-time broadcast of the Thingy's 20+ sensors' live values. Because BLE is limited in range, we also developed a controller for an XBee sensor which, through the Zigbee mesh protocol, can expand its range through each node added into the network. With these controllers, NSLS-II scientists have access to a whole new class of sensors which are both easier to deploy and cheaper than their wired predecessors.

INTRODUCTION

At the beamlines of Brookhaven National Laboratory's National Synchrotron Light Source II, scientists have countless environmental parameters that they want to monitor and control. Using wired sensors is a popular solution, but requires fiddling with wiring, ports, converters, and typically several protocols. Wireless sensors would remove all the monotonous tinkering while still providing the accurate sensing that scientists require. Thus, we set out to create Experimental Physics and Industrial Control System (EPICS) [1] input-output controllers (IOCs) that scientists at the Synchrotron may use to interface with wireless sensors that are supremely simple to deploy. Along with the IOCs, we developed intuitive screens for the Control Systems Studio [2] client to help users visualize data and interact with the IOC software. Note that our goal is not to maximize wireless range, but to create a wireless sensing system that is durable, portable, cost-effective, long-lasting and simple to deploy. The primary purpose of such a system is to initially replace wired sensors in an individual beamline hutch, and thus does not require extensive long-range communication.

ZIGBEE: DIGI XBEE L/T/H

The first wireless sensor we considered was the XBee L/T/H sensor [3], a sensor produced by Digi which utilizes the Zigbee protocol and senses light, temperature, and humidity. Figure 1 shows the sensor as well as the peripherals we used to interface with the sensor. We were interested in a Zigbee sensor due to the protocol's low power, high battery life, and mesh networking support.

Figure 1: An XBee L/T/H sensor, Digi Wi-Fi Gateway, and XBee XStick.

Mesh networking is a local network topology in which every node dynamically and non-hierarchically communicates with other nodes to efficiently route data. The communication is forwarded through nodes to reach far away nodes. In effect, mesh networking allows for an extensible network with virtually unlimited range given enough nodes.

The first step in creating the controller for the sensor was to bridge the gap between the host device running the IOC and the sensor. With the Digi XBee sensors, we found ourselves locked into Digi's ecosystem; they used a proprietary Zigbee protocol that forced us to use their gateway hardware to communicate with the sensors. For our development we used an XBee Zigbee Wi-Fi Gateway [4], which we could connect to through Ethernet and configure our sensors. By default, the Gateway simply publishes sensor readings to a paid cloud service provided by Digi; this was unsuitable for our needs, as we only wanted to store and broadcast the readings locally. We found that the Gateway runs Python scripts with a proprietary package created by Digi to allow reading of the sensors, so we developed a script which runs a TCP server on the Gateway that receives simple commands and responds with sensor readings. We found that the easiest way to develop an IOC to interface with this server would be by utilizing StreamDevice, an IOC structure which uses simple protocols to read and write streams of bytes to a socket through EPICS. With the server script set to run indefinitely on the Gateway and using a custom XBee protocol file for our input-output controller, we had successfully created an EPICS system to read, and broadcast the readings from our sensors.

† kgofron@bnl.gov

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEMPL008

THE MAX IV WAY OF AGILE PROJECT MANAGEMENT FOR THE CONTROL SYSTEM

V. H. Hardion, M. Lindberg, D. P. Spruce
MAX IV Laboratory, Lund University, Lund, Sweden

Abstract

Projects management of synchrotron is both complicated and complex. Building scientific facilities are resource consuming although largely made out of standard and well known components. The industrial approach of project management resolves this complication by requiring analysis and planning to facilitate the execution of tasks. The complexity comes by all the research making unique the accelerators, the beamlines and its usage. Known unknown requires experiments which evolve continuously causing the development path to be naturally iterative. Agile project management has come a long way since its definition in 2001. Nowadays this method is ubiquitous in the software development industry following different implementation like Scrum or XP and started to evolve at a bigger scale (i.e Scaled Agile) applied within an entire organization. The versatility of the Agile method has been applied to a Scientific technical development program such as the MAX IV Laboratory control system. This article describes the experience of 7 years of Agile project management and the use of Lean Management principles to develop and maintain the control system.

MAX IV WAY

MAX IV Laboratory is the first 4th generation synchrotron which is based on an innovative multibend achromat design carrying high expectations in terms of brilliance, stability and coherence of the X-Ray beam. This new laboratory inaugurated in 2016 is based in Lund, Sweden and has been built upon the foundations of Max-Lab, a facility which had operated 3 previous generations of accelerators until 2015. Max-Lab began in the 1980's with the first accelerator, MAX I, constructed by the team members themselves [1]. The build-up was handled by a very small staff and on a very limited budget. In spite of that the MAX I facility could be taken into operation in 1985. The small staff also implied that all personnel had to take a large responsibility [2].

The Control and IT Support group (KITS) were the first to introduce Agile methodology at the MAX IV Laboratory. Its main responsibility is to provide Software, Electronics and IT to build and develop the MAX IV Laboratory.

COMPLEX AND COMPLICATED

Synchrotrons are complex systems in the sense that they are made up of many interacting components, from the different subsystems with their corresponding experts, to the varied flora of scientific experiments. The interactions between the components and the constantly changing scientific requirements result in many unknowns. Getting each separate part to work well is a complicated problem which can be solved, but the general complexity of the facility can only be managed by a willingness to try, learn and adapt.

AGILE

Agile [3] is a movement that describes an iterative approach to project management and development, with a focus on evolving requirements and self-organizing teams. This methodology is based on 4 principles which define the "The Agile Manifesto" Fig. 1.

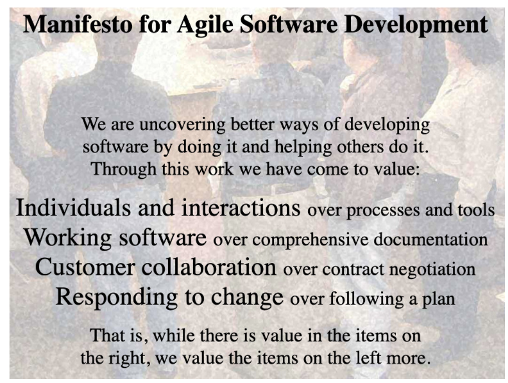


Figure 1: The 4 principles of Agile.

Several implementations of the methodology are based on Agile such as SCRUM or XP. SCRUM is a framework that describes a set of rules and methods that enable a team to collaborate on complex problems in an iterative and agile way. Scrum is more focused on the project management while eXtreme Programming (XP) emphasises Agile through the techniques of development. Most of them are known outside of this framework such as Continuous Integration [4], Pair programming, Unit Testing and Code Review. These are often cherry-picked in different methodologies than Agile.

Lean Management

Lean management [5] in an organization aims to deliver better value to the customer by systematic, iterative and continuous improvement of the workflow and processes involved. Kanban is a framework that allows management and visibility of the workflow making it possible to adapt and collaborate in a lean and agile way.

PRINCIPLES APPLIED AT MAX IV

The following section explains the organisation of the Control Group in terms of project management and development techniques seen through the narrow prism of Agile. Each of the principles applied are related to the way the old facility Max-Lab was organised and how these principles

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEMPL009

TRACKING APS-U PRODUCTION COMPONENTS WITH THE COMPONENT DATABASE AND eTraveler APPLICATIONS*

D. P. Jarosz†, S. Veseli, N. D. Arnold, J. Carwardine, G. Shen, N. Schwarz, G. Decker
Argonne National Laboratory, Argonne, USA
D. Liu, Osprey DCS LLC, Ocean City, USA

Abstract

The installation of the APS-U has a short schedule of one year, making it imperative to be well prepared before the installation process begins. The Component Database (CDB) has been designed to help in documenting and tracking all the components for APS-U. Two new major domains, Machine Design domain and **M**easurement and **A**nalysis **Arc**hive (MAARC) domain, have been added to CDB to further its ability in exhaustively documenting components. The Machine Design domain will help define the purpose of all the components in the APS-U design and the MAARC domain allows association of components with collected data. The CDB and a traveler application from FRIB have been integrated to help with documenting various processes performed, such as inspections and maintenance. Working groups have been formed to define appropriate work flow processes for receiving components, using the tools to document receiving inspection and QA requirements. The applications are under constant development to perform as expected by the working groups. Over some time, especially after production procurement began, the CDB has seen more and more usage in order to aid in preparation for the APS-U installation.

INTRODUCTION

The APS Upgrade has begun to receive production components in preparation for APS-U installation in 2022. The Component Database (CDB) is a tool that is actively being developed to track components through procurement, receipt, inspection, preliminary testing, and installation. A tightly coupled companion application is the eTraveler, a tool that mimics the paper travelers historically used for tracking components. As engineers and technicians begin using these tools for production hardware, numerous feature requests have been implemented to improve usability and efficiency. Previous versions of CDB [1,2] were instrumental in gaining acceptance of its use. The APS-U QA Policy relies on these tools for tracking inspections and logging discrepancies.

NEW DOMAINS IN CDB

Utilizing the generic schema that supports the Component Catalog and Component Inventory, two new domains have been added, the Machine Design and the Measurement and **M**easurement and **A**nalysis **Arc**hive (MAARC). These domains extend the use of CDB from a simple inventory system to a tool that allows engineers to specify and track the set of components needed to create a "machine". The MAARC domain supports the archiving of test data indexed and referenced to a particular component or to portion of the machine design.

Machine Design

The Machine Design domain allows users to specify a hierarchy of components to be installed to fulfill a particular function in the overall project. For example, Figure 1 shows how a hierarchical machine design can depict both electrical equipment (the contends of a rack) and accelerator equipment (components in the tunnel). Having a common mechanism to capture both electrical and mechanical components to be installed will allow for shared work processes for the assembly and installation of all APS-U components. It is also the basis for other relationships between any components, such as cables, control flow, or power distribution.

Life cycle of a Machine Design Item A machine design item is essentially a "reserved space" or "placeholder" for some type of component (from the CDB Catalog) to be installed at a given location. For example, the VME crate in Figure 1 can be entered into the machine design as "S27:VME1" housed in "S27 RTFB DAQ RACK" before the exact model of the VME crate is determined. Once the model for the VME chassis is known, it will be added into the CDB Catalog and then assigned to "S27:VME1", which will then indicate "S27:VME1 will be a Tracewell Model #XYZ VME crate". Likewise, S01A:Q1 can represent the first quadrupole in Sector 01A even before the exact characteristics of that component are known and described in the Catalog.

In addition to the machine design item holding the intended component type (from the Catalog), one can also specify which component instance (e.g.,the serial number) of that type of component is installed in that location at the current time. Since the CDB records any changes to these

* The submitted manuscript has been created by UChicago Argonne, LLC, Operator of Argonne National Laboratory (?Argonne?). Argonne, a U.S. Department of Energy Office of Science laboratory, is operated under Contract No. DE-AC02-06CH11357. The U.S. Government retains for itself, and others acting on its behalf, a paid-up nonexclusive, irrevocable worldwide license in said article to reproduce, prepare derivative works, distribute copies to the public, and perform publicly and display publicly, by or on behalf of the Government. The Department of Energy will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan. http://energy.gov/downloads/doe-public-access-plan
† djarosz@aps.anl.gov

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEMPR001

DATA ANALYSIS INFRASTRUCTURE FOR DIAMOND LIGHT SOURCE MACROMOLECULAR & CHEMICAL CRYSTALLOGRAPHY AND BEYOND

M. Gerstel†, A. W. Ashton, R. J. Gildea, K. E. Levik, G. Winter
Diamond Light Source, Oxfordshire, United Kingdom

Abstract

The Diamond Light Source data analysis infrastructure, Zocalo, is built on a messaging framework. Analysis tasks are processed by a scalable pool of workers running on cluster nodes. Results can be written to a common file system, sent to another worker for further downstream processing and/or streamed to a LIMS (Laboratory Information Management System). Zocalo allows increased parallelization of computationally expensive tasks and makes the use of computational resources more efficient. The infrastructure is low-latency, fault-tolerant, and allows for highly dynamic data processing. Moving away from static workflows expressed in shell scripts we can easily re-trigger processing tasks in the event that an issue is found. It allows users to re-run tasks with additional input and ensures that automatically and manually triggered processing results are treated equally. Zocalo was originally conceived to cope with the additional demand on infrastructure by the introduction of Eiger detectors with up to 18 Mpixels and running at up to 560 Hz framerate on single crystal diffraction beamlines. We are now adapting Zocalo to manage processing tasks for ptychography, tomography, cryo-EM, and serial crystallography workloads.

INTRODUCTION

Data collected at single crystal diffraction beamlines are processed automatically at Diamond Light Source (DLS) [1]. These experiments generally involve the generation of a substantial amount of data. For a typical data collection at a macromolecular (MX) beamline with a DECTRIS Pilatus 6M detector a crystal is rotated through 360° while being exposed to X-rays. At each oscillation step of 0.1° an image is read out from the detector, resulting in 3,600 6 MB images (21 GB). Depending on the beamline parameters and equipment these data may be collected in 36 seconds (100 images/s) to 2.4 minutes (25 images/s). A different type of preparatory experiment is a grid scan (Fig. 1), for which usually fewer than 1,000 still images are obtained within 1-2 minutes across a sample area to locate diffracting material. Since users may often wait for initial analysis results before deciding on how to proceed with their experiment, the time to process the experimental data is critical to the overall facility efficiency. Technological advances work against the requirement to provide speedy feedback to the experimenters: DLS recently installed DECTRIS Eiger2 XE 16M detectors on beamlines I03 and I04 and an Eiger 2 X 4M on beamline VMXi. These considerably increase both the size of individual images as well as achievable frame rates. While the data processing infrastructure at DLS has provided reliable service in the past, these technical developments as well as the launch of two new MX beamlines, VMXi and VMXm, required a major overhaul to ensure smooth data processing for the future.

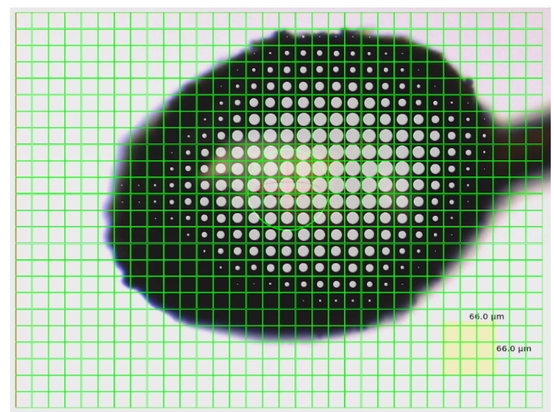


Figure 1: Results of grid scan per-image analysis overlayed with optical image of sample. Circles indicate presence of diffracting material.

To achieve this we implemented a distributed infrastructure called ‘Zocalo’. Zocalo is built on a messaging framework where fine-grained tasks are submitted to a queue, picked up and processed by a flexible number of specialist services, which run on high-performance cluster nodes. Services can be slotted together to form a larger processing pipeline, the results of each step can be stored on the file system, sent to a downstream processing service, or both. This provides many advantages over the previous analysis system, including low-latency data processing, self-monitoring, automatic resource allocation, fault-tolerance and more efficient use of computational resources.

There are many elements to full automated data processing of diffraction data, such as strategy calculations, per-image analysis, data reduction, experimental phasing, molecular replacement, and difference map calculation. In this paper two tasks will be highlighted to demonstrate how they are implemented in the new processing infrastructure: the per-image analysis, which is central to grid scans, and the initial data reduction for data collections. Both are currently run for every macromolecular (MX) and chemical crystallography (CX) data collection at DLS, and they pose different and representative challenges in data processing.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEMPR003

EXPLORING EMBEDDED SYSTEMS' DEDICATED CORES FOR REAL-TIME APPLICATIONS

P. H. Nallin†, R. C. Ito, J. G. R. S. Franco, A. R. D. Rodrigues
Brazilian Synchrotron Light Laboratory, Campinas, Brazil

Abstract

Developments and research in high technology leads to powerful and sophisticated machines which are highly important for many scientific fields. Considering real-time applications, however, these systems tend to become non-deterministic and users may find themselves inside a not completely controllable environment. Exploring open-hardware single board computers with a system-on-a-chip which usually runs an operational system on their main processor(s) and also have real-time units is a good alternative. These real-time units are designed as a microcontroller embedded on the chip where a firmware is loaded, runs concomitantly and exchanges data with the main system. As a result, it is possible to achieve performance increase, high temporal resolution and low latency and jitter, features that are widely desired for controls and critical data acquisition systems. This system architecture allows moving real-time data into high level servers, such as Redis (Remote Dictionary Server) and EPICS, easily. This paper introduces and shows uses of Beaglebone Black, an inexpensive single-board computer, its Programmable Real-Time Units (PRUs) and data sharing with Redis data structure.

INTRODUCTION

Designing a new 3 GeV and ultra-low emittance synchrotron machine, brings along many technological challenges. In the case of controls systems, it impacts on controlling and monitoring a large variety of equipment, including the ones that are very accurate and/or critical to light generating. As a solution for both deterministic and general controls, it has been chosen to use an open hardware single board computer, Beaglebone Black [1], which comes with two embedded Programmable Real-Time Units (PRUs). Running on an embedded linux environment, applications have fast hardware access and are integrated to Controls System network. Sirius, the future fourth-generation Brazilian Light Source, comes to the final phase of systems installations for machine engineering. Among them, there are some designed by Controls Group using the Beaglebone Blacks and Programmable Real-Time Units.

USE OF BEAGLEBONE BLACK IN BRAZILIAN LIGHT SOURCES

UVX Facility

Regarding controls system upgrades in UVX facility, the pioneer Brazilian light source, it is remarkable when first CPUs (Z80 and eZ80) were replaced by a commercial fanless single board computer, Advantech PCM-4153F, in 2010, increasing reliability and ease of maintenance.

In the beginning of 2016, the first Beaglebone Black single board computers were introduced to UVX facility controls system, in order to replace older CPU generations [2], either because of electronic components unavailability or outdated equipment. Already intended to be used in Sirius accelerator, it has also been a great bench test for Beaglebone's embedded system.

Since then, running a Debian Linux distribution, it has been detected only one issue concerning hardware/software operation, a possible freezing after a warm reboot, which was corrected in a newer kernel version.

Sirius

Chosen to be the main distributed core in Sirius controls systems, as shown in Fig. 1, there will be more than 350 units running full-time in Sirius applications, such as vacuum system monitoring, power supplies, pulsed power electronics control and temperature acquisition. The choice was made considering its performance, low cost and open hardware project.

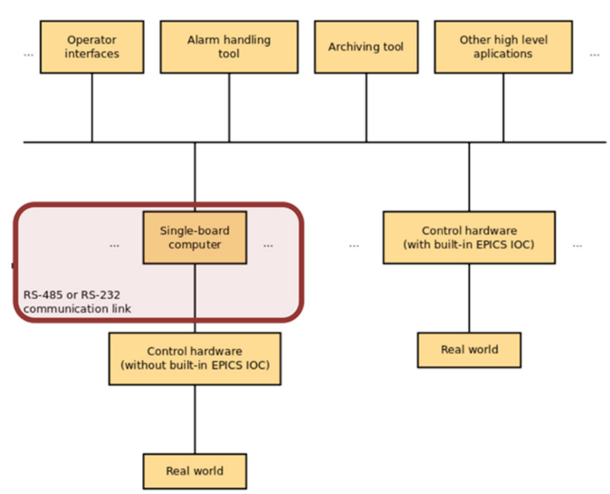


Figure 1: Sirius Controls System simple layout, considering distributed cores.

However, some of applications have requirements to be performed in real-time systems for several reasons. High amount of data transfer, synchronized operations, low jitter and great time accuracy and determinism.

PROGRAMMABLE REAL TIME UNITS

Beaglebone Black is based on a System on a Chip (SoC) AM335x family, designed and manufactured by Texas Instruments. As subsystems, it has two independent real-time

† patricia.nallin@lnls.br

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEMPR004

WHY SHOULD YOU INVEST IN ASSET MANAGEMENT? A FIRE AND GAS USE CASE

H. Nissen, S. Grau, CERN, Geneva, Switzerland

Abstract

At present, the CERN Fire and Gas detection systems involve about 23000 assets and their number is increasing rapidly at the same time as the number of equipped installations grows. These assets cover a wide spectrum of technologies, manufacturers, models, parameters, and ages, reflecting the 60 years of CERN history. The use of strict rules and data structures in the declaration of the assets can make a big impact on the overall system maintainability and therefore on the global reliability of the installation. Organized asset data facilitates the creation of powerful reports that help asset owners and management address material obsolescence and end-of-life concerns with a global perspective.

Historically, preventive maintenance has been used to assure the correct function of the installations. With modern supervision systems, a lot of data is collected and can be used to move from preventive maintenance towards data-driven maintenance (predictive). Moreover, it optimizes maintenance cost and increases system availability while maintaining reliability. A prerequisite of this move is a coherence on the assets defined in the asset management system and in the supervision system.

CERN BE-ICS-AS SERVICE

The CERN *Alarm System* service is responsible for the installation, the maintenance, and the renewal of safety alarm systems at CERN. This includes fire and gas detection systems (Flammable Gas, Oxygen Deficiency, Toxic Gas), emergency telephones (Red Telephones) and the alarm transmission systems to CERN main control rooms (SCR-CERN Fire Brigade, CCC-CERN Control Centre and XCR-Experiment Control Rooms). There are currently 9,687 automatic smoke detectors, 783 automatic gas detectors, 2115 manual break-the-glass devices and 413 emergency telephones installed all over CERN sites, and covering from office buildings to accelerator complexes and experimental halls.

ASSET MANAGEMENT

According to the asset management standard ISO 55000, an asset is defined as: *An item, thing or entity that has potential or actual value to an organisation [1]*

This very general definition can cover many types of assets e.g. Physical, Financial, Human and Intangible assets. In this paper, we will concentrate on the physical assets that are physical objects installed to fulfil a purpose and therefore have a value to the organisation.

Asset Management History

Asset management concepts have existed as long as there have been assets to maintain such as the first early tools used by farmers. A common definition of asset management is: *The practice of managing the entire life cycle (design, construction, commissioning, operating, maintaining, repairing, modifying, replacing and decommissioning/disposal) [1]*.

In the beginning, this asset management was an informal system where the asset information was kept by persons involved in the life of the assets. As the number and complexity of assets grew, a more formal way of dealing with the assets was necessary. This led to the first paper-based assets management where important information about the assets was noted down on paper. This included a central list of assets with primary characteristic and local logbooks stored with the assets to note interventions and problems.

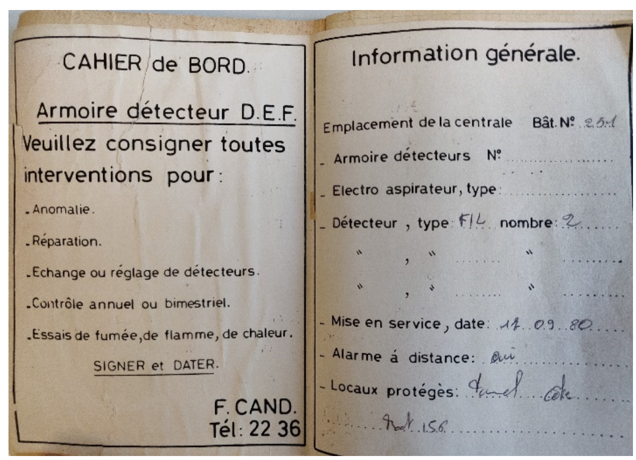


Figure 1: Example of local maintenance logbook.

This paper-based system (Fig. 1) was used for the CERN fire and gas installations from the start of CERN until the introduction of the first computer-based asset management system in the mid-eighties. The information from the former paper-based system was then transferred to a computer system that made the information more generally available to the asset managers. This facilitated the creation of work orders to keep trace of intervention on assets. In general terms, the abbreviation CMMS (Computer Maintenance Management Systems) is used for all the early computer maintenance systems that were very focused on the maintenance aspects of the assets. Since then the CMMS software has steadily been upgraded to reflect the changing need of the organisation and today the software used at CERN is called INFOR[3] Enterprise Asset Management. The main difference between a CMMS and an EAM (Enterprise Asset Management) is that traditionally the CMMS software was exclusively focused on the asset maintenance in the

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEMPR005

THE ARRAY CONTROL AND DATA ACQUISITION SYSTEM OF THE CHERENKOV TELESCOPE ARRAY

I. Oya*, E. Antolini, M. Füßling, J. A. Hinton, A. Mitchell, S. Schlenstedt
CTAO gGmbH, Heidelberg, Germany
L. Baroncelli, A. Bulgarelli, V. Conforti, N. Parmiggiani, INAF - O.A.S. Bologna, Italy
J. Borkowski, CAMK, Torun, Poland
A. Carosi, J. Jacquemier, G. Maurin
LAPP, Univ. Grenoble Alpes, Univ. Savoie Mont Blanc, CNRS, France
J. Colomé, IEEC-CSIC, Spain
C. Hoischen, Universität Potsdam, Germany
D. Lebout, Université Paris-Sud, CNRS, INRIA, Université Paris-Saclay, Orsay, France
E. Lyard, R. Walter, University of Geneva, Switzerland
D. Melkumyan, K. Mosshammer, I. Sadeh, T. Schmidt, P. Wegner, DESY, Zeuthen, Germany
U. Schwanke, Humboldt-Universität zu Berlin, Germany
J. Schwarz, INAF - Osservatorio Astronomico di Brera, Italy
G. Tosti, University of Perugia, Italy
for the CTA Observatory†

Abstract

The Cherenkov Telescope Array (CTA) project is the initiative to build the next-generation gamma-ray observatory. With more than 100 telescopes planned to be deployed at two sites, CTA is one of the largest astronomical facilities under construction. The Array Control and Data Acquisition (ACADA) system will be the central element of on-site CTA Observatory operations. The mission of the ACADA system is to manage and optimize the telescope array operations at each of the CTA sites. To that end, ACADA will provide all necessary means for the efficient execution of observations, and for the handling of the several Gb/s of data generated by each individual CTA telescope. The ACADA system will contain a real-time analysis pipeline, dedicated to the automatic generation of science alert candidates as data are being acquired. These science alerts, together with external alerts arriving from other scientific installations, will permit ACADA to modify ongoing observations at sub-minute timescales in order to study high-impact scientific transient phenomena. This contribution describes the challenges, architecture, design principles, and development status of the ACADA system.

INTRODUCTION

The Cherenkov Telescope Array (CTA) is an initiative to build the next-generation atmospheric Cherenkov gamma-ray observatory [1]. The CTA Observatory (CTAO) will consist of two facilities, one in the Southern (close to the Paranal Observatory, Chile) and the other in the Northern Hemisphere (at the Roque de Los Muchachos Observatory, La Palma, Spain). The two sites will contain dozens of telescopes of different sizes – Large, Medium, and Small-Sized Telescopes (LSTs, MSTs, SSTs) –, constituting one of the largest astronomical installations under development.

The observations of the CTA installation will be coordinated by the Array Control and Data Acquisition (ACADA) System. Two major software systems [2] of CTA are the Data Processing and Preservation Systems (DPPS), which will be hosted in a set of offsite data centres, and the Science User Support System (SUSS), which will be deployed in the CTA Science Data Management Centre in Zeuthen, Germany. SUSS will, among other functions, make processed data and science tools available to end users, as well as provide the interface to create and submit CTA observation proposals.

This contribution describes the ACADA system. After introducing the mission of ACADA and its main requirements, we present an overview of the ACADA system architecture, followed by the most significant design principles of the system. We close with a summary of the development status of the ACADA system.

ACADA SYSTEM MISSION

The ACADA system comprises all software responsible for the control and data acquisition of telescopes and the additional devices responsible for array calibration and environment monitoring at each of the CTA sites; it plays the role of the supervisory control and data acquisition (SCADA) system. ACADA is also responsible for the efficient execution of pre-scheduled observations and those triggered by science alerts – which allows CTA to respond on sub-minute timescales and observe interesting transient phenomena such as gamma-ray bursts [3]. These science alerts can be triggered externally by other scientific installations, or internally within ACADA thanks to a dedicated

* igor.oya@cta-observatory.org
† http://www.cta-observatory.org

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEMPR006

APPLICATION DEVELOPMENT IN THE FACE OF EVOLVING WEB TECHNOLOGIES AT THE NATIONAL IGNITION FACILITY

E. Pernice, C. Albiston, R. Beeler, E. Chou, C. Fry, M. Shor, J. Spears,
D. Speck, A. Thakur, S. West
Lawrence Livermore National Laboratory, Livermore, USA

Abstract

The past decade has seen great advances in web technology, making the browser the de-facto platform for many user applications. Advances in JavaScript, and innovations such as TypeScript, have enabled developers to build large scale applications for the web without sacrificing code maintainability. However, this rapid growth has also been accompanied by turbulence. AngularJS arrived and saw widespread adoption only to be supplanted by Angular 2+ a few years later; meanwhile other JavaScript-based languages and developer tools have proliferated. At the National Ignition Facility (NIF), the Shot Setup Tool (SST) is a large web-based tool for configuring experiments on the NIF that is being developed to replace a legacy Java Swing application. We will present our experience in building SST during this turbulent time, including how we have leveraged TypeScript to greatly enhance code readability and maintainability in a multi-developer team, and our current effort to incrementally migrate from AngularJS to React.

INTRODUCTION

The Campaign Management Tool (CMT) is an application that is used to configure experiments at the NIF. Originally developed as a commissioning tool for the NIF laser, CMT was subsequently put to work as the production experiment editor for NIF experimental operations. It remained under constant development for 15 years supporting the ever-expanding stable of NIF target diagnostics and the ongoing refinement of the NIF laser. However, CMT's architecture was not optimal for the development focus of the program, and it also carried a very steep learning curve for software developers. In 2014 CMT was identified as a bottleneck for shot operations as the NIF sought a dramatic increase in its experiment shot rate. A project was undertaken to address CMT usability and maintainability concerns. These concerns included an outdated technology stack as well as several architectural features that inhibited maintainability. As a result of this effort, we decided a new application was needed to meet programmatic needs [1].

It was critical to select a technology stack with long term sustainability and which followed current computing trends. Since the initial development of CMT, a highly interactive desktop application, web browser-based applications had increased significantly in capability and ubiquity. Model-View-Controller (MVC) frameworks, such as AngularJS and Backbone, allowed much better client-side rendering and supplanted server-side rendering, such as JavaServer Pages (JSP), as the standard for interactive web applications. In order to maintain parity with CMT functionality we decided to build SST as a client rendered web application, on a RESTful back end. In 2015 we began development on SST.

INITIAL TECHNOLOGY STACK

For our initial release of SST, we adopted a handful of front-end technologies that would help us achieve our early goals but that would also give us flexibility to adapt to changes in the ecosystem or evolving requirements.

TypeScript

Typescript [2] was a new and somewhat unproven technology at the time we adopted it. TypeScript overcame some of our largest concerns about moving from a large, Java-heavy application to a large JavaScript-heavy one because it is a superset of JavaScript that adds many features to improve code maintainability and scalability. TypeScript code is transpiled to JavaScript as a build step, which is then deployed with the application.

Compile-time error checking In a vanilla JavaScript application, many types of errors that would have been compile time errors in Java are run-time errors and will only be raised when the offending code is executed. This was a concern for us because a typo in a variable or method name could cause a bug that might not be found until after deployment. Even troubleshooting in a development environment would be a headache. TypeScript addressed this problem directly and effectively by failing during our build cycle when validity errors such as these were detected.

Self-documenting code A type-safe language like Java intrinsically provides certain self-documenting features. Consider a method that takes a ShoppingCart as a parameter and returns a list of Items. In JavaScript, clarity must be achieved by apt variable names or comments, which are less explicit and less reliably correct. TypeScript allows us to provide type annotations, which are checked for correctness during transpilation.

Advanced language features TypeScript provides backward compatible use of features from future JavaScript releases such as classes, decorators, and lambdas. This allowed us to support many browsers. It also provides features exclusive to TypeScript, like interfaces and private members. As a team with mainly Java expertise, the addition of familiar object-oriented constructs was welcome.

IDE support Compared to Java, IDE support is very limited for JavaScript. TypeScript enables excellent support for IDE functions like auto-completion, refactoring automation, and searching for method or variable references.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEMPR008

WEB EXTENSIBLE DISPLAY MANAGER 2*

R. J. Slominski†, T. L. Larrieu, JLab, Newport News, USA

Abstract

The Web Extensible Display Manager (WEDM) was first deployed at Jefferson Lab (JLab) in 2016 with the goal of rendering Extensible Display Manager (EDM) control screens on the web for the benefit of accessibility, and with version 2 our aim is to provide a more general purpose display toolkit by freeing ourselves from the constraints of the EDM dependency. Over the last few years WEDM has been extensively used at JLab for 24/7 information kiosks, on-call monitoring, and by remote users and staff. The software has also been deployed to Oak Ridge National Laboratory, and has become more robust as many bug fixes and contributions have been added. However, adoption and utility of the software as a general purpose control system display manager is limited by EDM, which is no longer actively maintained. A new toolkit can be built on modern frameworks, fully embrace web conventions and standards, and support multiple control system data sources. This new version is a result of a technology review and selection, and introduces a web inspired display file format, a web based display builder, new widgets, and a data interface intended to support pluggable data.

INTRODUCTION

There has been an explosion of interest in control system displays on the web in the last few years. It has increasingly become a user expectation for displays to be available on a variety of devices including smart phones, and often the web is the best method to deliver this experience. Our initial version of Web Extensible Display Manager (WEDM) relied on the ageing Extensible Display Manager (EDM) to provide a screen builder tool and editable displays while we added a read-only web runtime [1]. The primary objective of version 2 is to break the dependency on EDM and provide an independent web based display builder. The second version of WEDM is named Puddysticks, and is shown in Fig. 1. During development we studied and embraced modern software frameworks, adopted web paradigms, and implemented a data source agnostic interface to widgets. We report our status as we work towards a web based display manager that serves all devices, whether they be in the control room or in your pocket.

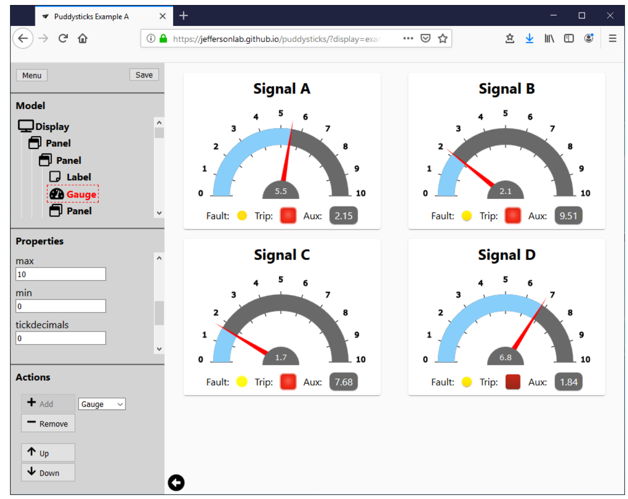


Figure 1: "Puddysticks" prototype.

A Case for Zero Code

Control systems have a long history of display managers: Graphical User Interface (GUI) builder tools that allow non-programmers to create control screens without writing any code. A display manager provides a consistent and familiar workflow to users, and the reusable framework saves software developers from having to spend time on each new project requiring displays.

A Case for Mobile

According to the Pew Research Center 76% of people in advanced economies had a smart phone in 2018 [2]. Using mobile devices with control systems is an opportunity for improved user experience. Remote monitoring of control systems from desktops may not even be an option for many as users drop home broadband in favor of cellular Internet and discontinue personal ownership of desktops. Besides ensuring staff are always connected to the control system wherever they are, mobile devices also provide cheap and readily available access to numerous additional user interaction mechanisms such as touch gestures (haptics) and voice commands. Readily available built-in cameras and GPS may also provide new opportunities. The best user experience might be for operators to adjust control system settings via touch or voice, instead of mouse and keyboard.

A Case for the Web

The web is a well-established standardized cross-platform way to provide interactive content on nearly all devices, including mobile devices. Web applications can work well on devices of varying screen sizes, but do not do so implicitly, and those that do are often referred to as having a responsive design. Going a step further, Progressive Web Applications (PWAs) are web applications that follow a set of best practices making them

* Authored by Jefferson Science Associates, LLC under US DOE Contract № DE-AC05-06OR23177
† ryans@jlab.org

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEMPR009

DEVELOPMENT OF EVENT RECEIVER ON Zynq-7000 EVALUATION BOARD

H. Sugimura*, High Energy Accelerator Research Organization (KEK), Tsukuba, Japan

Abstract

The SuperKEKB accelerator uses “Event Generator” and “Event Receiver”, an event timing module developed by Micro Research Finland company. It is suitable for pulse-by-pulse control because Injector Linac is generated various parameters of timings for the multi rings in each pulse. I tried to develop a new event receiver module by using FPGA (Zynq) evaluation board so that the specification can be changed flexibly according to the nature of the accelerator. I focus on the convenience of Zynq-chip such as serial data optical transfer (GTX) and embedded processor system. Finally, I aim for developing standalone event receiver module, and composite module that are integrated with BPM and RF system.

INTRODUCTION

KEK accelerator facility in Tsukuba, Japan is managed five rings (PF, PF-AR, SuperKEKB (LER, HER, DR) [1]). The first two rings are synchrotron radiation facility, and others are collider rings for particle physics. These rings are handled electron and positron beam, and are injected from one linac simultaneously at the repetition of 50 Hz. The timing system of our accelerator facility is required various parameter to fire at pulsed magnets such as Kicker and Septum magnets. And also, RF system and pulsed magnets (dipole, quadrupole and steering ...) at linac are required to identify which particle and which energy will be accelerated because the linac is needed to consider the five rings. So, we introduced event timing system to deliver some timings and identification codes simultaneously [2]. We focused on the function of event code delivery made by Micro Research Finland Oy(MRF) [3], and introduced “Event Generator(EVG)” and “Event Receiver(EVR)” as main component of a timing station. Furthermore, very flexible modules, EVG and EVR (EVE and EVO series), were developed by SINAP which conforms to MRF modules. The advantage of the SINAP timing module have a fine delay function. The resolution is 5ps. SINAP EVE and EVO are used sub-timing station at LER and HER in SuperKEKB.

MRF timing system is assigned the first 8-bit is event code and the last 8-bit is data buffer mode or distributed bus bit mode alternatively in a frame. The 16-bit of the event frame is transferred in synchronization with 114.24 MHz of RF clock which is obtained by dividing S-band 2856 MHz by 25. So, the data size is 114.24×20 Mbyte/s (16-bit data is transferred with 8b10b conversion). We integrated beam gate control to event timing system by using the distributed bus bit region [4], and some kind of shot information by using data buffer region.

* hitoshi.sugimura@kek.jp

The commissioning of SuperKEKB is going well and there is no critical error in event timing system.

EVENT RECEIVER ON Zynq EVALUATION BOARD

Purpose

Just as additional function has been added in event timing module in SINAP, we might update to suit our accelerator specification. At such time, we need to update firmware easily and quickly. In this situation, it will be important to develop the circuit coding of FPGA ourselves to keep stable accelerator operation. From another point of view, there is an advantage that development costs can be reduced by making it by ourselves. In particular, the difference becomes large when a lot of mass production is performed.

Requirement for the Event Receiver

Since the event code is transmitted in synchronization with the RF clock of 114.24 MHz, including 8b10b conversion, a communication speed of 2.5 Gb/s or higher is required, and a low jitter of less than 20 ps is also required. Therefore, in consideration of cost, we decided to use GTX embedded in Xilinx FPGA. The GTX is included in Kintex-7 in Xilinx 7-series FPGAs, and is also used in MRF 300 series. The SINAP event timing modules are also using GTX transceiver in Virtex-6.

Open Source Event Receiver

MRF released the source code of the FPGA including GTX configuration so that the MRF module’s user could make event receivers by themselves. The released code uses Zynq7000 to describe the circuit. GTX is included in Kintex-7, and Zynq7000 also have a part of equivalent to Kintex-7 in programmable logic part (only for Z7030 and Z7040 series), so GTX can be used. The merit of Zynq is that it can be controlled by the ARM core. This can be a standalone module without going through the bus control, and also can be run in EPICS IOC in Zynq. Since the released code was developed using Avnet’s picozed [5], we decided to purchase a similar board and proceed with development based on open source code.

Figure 1 shows the picture of picozed. It is a dark green substrate in the center of the figure, and is used by inserting it into a carrier card, which is a vermilion substrate. The carrier card model number is AES-PZCC-FMC-V2-G, and picozed model number is AES-Z7PZ-7Z030-SOM-I-G.

GTX Set Up

We constructed GTX configuration with IP core tool of Vivado design suite. At first, I setup using Vivado version of

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEMPR010

ANOMALY DETECTION FOR CERN BEAM TRANSFER INSTALLATIONS USING MACHINE LEARNING

T. Dewitte, W. Meert, E. Van Wolputte*, KU Leuven, Leuven, Belgium
P. Van Trappen, CERN, Geneva, Switzerland

Abstract

Reliability, availability and maintainability determine whether or not a large-scale accelerator system can be operated in a sustainable, cost-effective manner. Beam transfer equipment (e.g. kicker magnets) has potentially significant impact on the global performance of a machine complex. Identifying root causes of malfunctions is currently tedious, and will become infeasible in future systems due to increasing complexity. Machine Learning could automate this process. For this purpose a collaboration between CERN and KU Leuven was established.

We present an anomaly detection pipeline which includes preprocessing, detection, postprocessing and evaluation. Merging data of different, asynchronous sources is one of the main challenges. Currently, Gaussian Mixture Models and Isolation Forests are used as unsupervised detectors. To validate, we compare to manual e-logbook entries, which constitute a noisy ground truth. A grid search allows for hyper-parameter optimization across the entire pipeline. Lastly, we incorporate expert knowledge by means of semi-supervised clustering with `COBRAS`.

INTRODUCTION

The CERN Large Hadron Collider (LHC) is the world's largest and most powerful particle accelerator [1]. Maintaining a machine of this size in an efficient and sustainable manner is highly non-trivial. Currently, detection of problematic situations (anomalies) during operation mostly happens manually; experts analyze the data, identify anomalies and track down root causes. In the case of equipment failure, this manual process results in the loss of precious machine time. Furthermore, this manual approach requires strong knowledge of the complex relations between LHC components and does not scale to future – larger – accelerators.

We present a proof-of-concept, scalable solution for automatic anomaly detection in the LHC. Concretely, Machine Learning (ML) methods model normal behaviour of its injection kicker magnets, based on historical data. Afterwards, this learned model flags unexpected behaviour in unseen data. We situate our work, describe the data, explain the design of the application and finally report on our experiments, including one on incorporating expert feedback into the ML system.

* Supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme, grant agreement No. 694980 "SYNTH: Synthesising Inductive Data Models"

INJECTION KICKER MAGNETS

Within the CERN accelerator complex, particle beams are extracted from one accelerator and injected into another by means of beam transfer equipment. At CERN it is the responsibility of the Accelerator Beam Transfer (TE/ABT) group to design, install and maintain this equipment. Especially critical for LHC operation are the Injection Kicker Magnets (MKI) [2], a set of two times four magnets that inject beams from the SPS into the LHC.

An MKI installation consists of four magnets, which are named A, B, C and D. Each pair of magnets (A-B and C-D) is powered by one generator, with a single high-voltage resonant charging power supply (RCPS) to charge two pulse forming networks (PFNs); thus one PFN per magnet. The LHC requires two injection installations (named MKI2 and MKI8), one for each counter-rotating beam. This brings the total of magnets to eight.

DATA

Typically, ML algorithms assume clean datasets of tabular form: each row represents a single observation, each column represents a feature. However, the raw data collected by CERN does not obey this ideal form; distilling sensible feature vectors from such a complex mixture of (mostly) asynchronous data sources constitutes a key challenge. This complexity stems mostly from the fact that the data sources were never set up with downstream ML applications in mind. Concretely, the dataset contains time series data (continuous), IPOC data (event data), state data (categorical), controller data (numerical), general LHC data (numerical), and logbook data (categorical and text).

Continuous, Time Series Data

First, we have continuous, time series data, e.g. pressure and temperature measurements of the MKI installation. The temperature measurements occur at two points, where particles enter the magnet (*upstream*) and where they exit the magnet (*downstream*). This continuous data is sampled at a fixed frequency of 2 Hz. To increase logging efficiency, values are often only stored upon change, or after a certain threshold duration without any change [3]. This threshold is configured by the CERN user and ranges from 15 minutes to one day. For each magnet, five such continuous variables are measured. Given four magnets (A, B, C, D) per system (MKI2, MKI8), this yields 40 continuous variables in total.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA002

LCLS-II CRYOMODULE AND CRYOGENIC DISTRIBUTION CONTROL*

D.T. Robinson†, A. Benwell, C. Bianchini, D. Fairley, S. Hoobler, K. Mattison, J. Nelson, A. Ratti
SLAC National Accelerator Laboratory, Menlo Park, USA
A. Martinez, J. Kaluzny, Fermi National Accelerator Laboratory, Batavia, USA
L. Farrish, J. Gubeli, C. Hovater, K. Jordan, W. Moore
Thomas Jefferson National Accelerator Facility, Newport News, USA

Abstract

The new superconducting Linear Coherent Light Source (LCLS-II) at the SLAC National Accelerator Laboratory will be an upgrade to LCLS, the world's first hard X-ray free-electron laser. LCLS-II is in an advanced stage of construction with equipment for both cryogenic plants (CryoPlants) as well as more than half of the 37 cryomodules onsite. Thomas Jefferson Lab (JLab) is a partner lab responsible for building half of the LCLS-II cryomodules. The Low Energy Recirculation Facility (LERF) at Jefferson Lab (JLab) was used to stage and test LCLS-II cryomodules before shipping them to SLAC. The testing was done by setting up two cryomodules at a time, cryocontrols instrumentation racks, Programmable Logic Controllers (PLC) controls and Experimental Physics and Industrial Control System (EPICS) Input/Output Controllers (IOCs) at LERF with the intention of developing cryogenic controls for LCLS-II. The cryogenic controls developed at LERF would then be replicated for controlling all 37 cryomodules via an EPCIS user interface. This paper discusses the cryogenic controls developed at LERF for implementation in the LCLS-II project.

INTRODUCTION

The LCLS-II project provided funding for all the instrumentation and controls software [1]. This includes RF amplifiers, Low Level RF (LLRF) control equipment and Cryogenic Controls racks with PLCs. EPICS is used for supervisory control of the cryogenic systems while the control logic was developed in Allen Bradley PLCs. Field instrumentation is connected as Distributed I/O communicating over an EtherNet/IP based Device Level Ring (DLR). The subsystem controls included electric heaters, pressure transducers, Resistive Temperature Detectors (RTDs) and liquid level monitors. Cryogenic valves are controlled using Profibus Process Automation (PA) and Profibus Decentralized Peripheral (DP) communication protocols [2]. The test facility included 16 four kW solid state amplifiers (SSA) and new wave guides were installed to connect the two cryomodules. Low Level RF chassis were installed for controls and interlocks, along with vacuum controls. Figure 1 shows the LERF cryomodules in the vault and cryocontrols racks in the gallery.

The Central Helium Liquefier (CHL) at JLab provides 2K Helium for the CEBAF accelerator. A cryogenic-connection to CHL was designed and installed to supply helium to the new LERF cryomodule test facility. This connection to one of the CHL cryoplants allows the CEBAF Linac and LERF to be cooled down from 4K to 2K without interrupting the CEBAF operations. The first two cryomodules installed and tested at the LERF were J1.3-12 and J1.3-5. Cryomodule J1.3-12 was installed at SLAC while J1.3-5 was reconditioned to remedy cryogenic performance issues identified during initial testing.

Figure 1: LCLS-II cryomodule in the LERF vault and cryogenic controls instrumentation racks in the gallery.

CONTROLS ARCHITECTURE

The final cryogenic control system architecture to be implemented for LCLS-II was developed in collaboration with input and feedback from partner labs. The cryogenic plant supplies helium through upstream and downstream distribution boxes. The upstream distribution box supplies 17 cryomodules with helium while the downstream distribution box supplies 20 cryomodules. The LCLS-II cryomodule control system will be designed using off-the-shelf hardware and will closely mirror the mechanical design. Redundant PLC processing will be done in centralized locations using ControlLogix 1756-L83E processors. The cryomodule PLCs will communicate to EPICS and the cryogenic plant PLCs. Figure 2 illustrates the redundant architecture.

The controls architecture at the LERF replicated the proposed LCLS-II design by using two of the controls rack intended for LCLS-II with all the same instrumentation and

* Authored by SLAC National Accelerator Laboratory under U.S. DOE Contract No. DE-AC02-76SF00515. The U.S. Government retains a non-exclusive, paid-up, irrevocable, worldwide license to publish or reproduce this manuscript for U.S. Government purposes.
† dayner@slac.stanford.edu

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA010

CONTROL SYSTEMS DESIGN FOR LCLS-II FAST WIRE SCANNERS AT SLAC NATIONAL ACCELERATOR LABORATORY*

N. Balakrishnan†, L. Sapozhnikov, H. Bassan, J. Bong, J. Olsen, K. Lauer, P. Krejcik, M. Campell
SLAC National Accelerator Laboratory, Menlo Park, USA

Abstract

One of the primary diagnostic tools for beam emittance measurement at the Linac Coherent Light Source II (LCLS-II), an upgrade of the SLAC National Accelerator Laboratory's Linac Coherent Light Source (LCLS) facility, is the wire scanners. LCLS-II's new Fast Wire Scanner (FWS) is based on a similar mechanical design of linear servo motor with position feedback from an incremental encoder as that for LCLS. With a high repetition rate of up to 1MHz from the superconducting accelerator of LCLS-II, it is no longer sufficient to use point-to-point EPICS-controlled moves from wire to wire, as continued exposure will damage the wires. The system needs to perform on-the-fly scans, with a single position versus time profile calculated in advance and executed in a single coordinated motion by Aerotech Ensemble motion controller. The new fast wire scanner control system has several advantages over LCLS fast wire scanner controls with the capability to program safety features directly on the drive and integrate machine protection checks on an FPGA. This paper will focus on the software architecture and implementation for LCLS-II Fast Wire Scanners.

INTRODUCTION

For measuring the electron beam profile, wire scanners (Fig. 1) are the primary tool used at the LCLS. The fast wire scanner system is comprised of a linear motor stage with an incremental linear encoder for closed loop position feedback [1]. The movable stage which has limits switches at the end of travel of the stage (Fig. 2) holds a wire card. Beam loss monitor readings obtained during a scan of the wire card through multiple bunches of the beam, helps provide the cross section of the beam. With the correlation of the wire positions from the encoder feedback and the beam loss monitor readings beam sizes, beam emittance, energy spread, or bunch length can be determined [2].

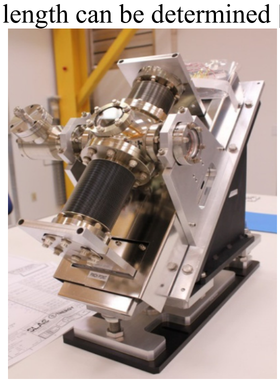


Figure 1: Wire Scanner at SLAC [1].

* Work supported by U.S. Department of Energy under contract number DE- AC02-76SF00515

† namrata@SLAC.Stanford.EDU

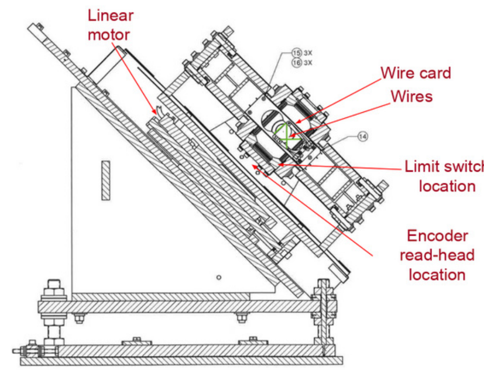


Figure 2: Wire scanner schematics showing the motor, encoder and limit switches.

CONTROL SYSTEM

Architecture Overview

The wire scanner controls electronics are housed in support buildings with long haul cables running to the hardware in the tunnel [2]. The servo motion control and PID (Proportional, Integral and Derivative) feedback is done with an Aerotech Ensemble CP20 Motion Control drive. The controller chassis enclosure supports two channels of wire scanners with independent Ethernet interfaces to the EPICS control system. The beam loss monitor signal and the external position encoder readings are recorded through a SLAC-standard common platform ATCA crate, housing a carrier card with a Xilinx FPGA. Readings are acquired beam synchronously with the inclusion of the central timing system through the crate [3]. The FPGA on the ATCA crate performs checks on the speed of the wire scanner and relays the information to Machine Protection System to prevent destruction of the wires during continuous-wave beam operations [2]. A block diagram of the system design is presented in Fig. 3

Aerotech Motion Controller

The 3U Aerotech controller chassis includes control for 2 servo motor axes by way of 2 individual Ensemble CP20 Motion Control drives. The Ensemble CP20 offers extensive tuning tools for an extended PID delivering nearly optimal performance, observable using the built-in Digital Scope [4]. Each CP20 axis supports up to 20 Amps peak (10 Amps continuous) at 160VDC, satisfying the hefty requirements of the wire scanner system, while also providing high-precision control with kilohertz-level servo tasks and high current-resolution output.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA011

SCALING AGILE FOR THE SQUARE KILOMETRE ARRAY

M. Bartolini*, N. Rees, M. Miccolis, J. Santander-Vela, L.R.S. Brederode,
M. Deegan, SKA Organisation, Macclesfield, United Kingdom

Abstract

The Square Kilometre Array (SKA) Organisation (SKAO) is responsible for the design and construction of the first phase (SKA1) of its vision: designing, construction, and operating telescopes with an equivalent collecting area to one square kilometre. The SKA1 project is finishing its pre-construction phase in December 2019. A bridging phase was kicked-off before construction commences during which lean-agile processes, structures, and practices are being prototyped. By the end of the bridging phase we plan to have pivoted from a document based, earned value, stage gated set of processes arranged around pre-construction consortia to a code based, value flow driven, lean-agile set of processes unified around the Scaled Agile Framework. During the bridging process we have onboarded more than 10 agile development teams. In this paper we describe the processes, the main technical and cultural challenges, and the preliminary results of adopting a lean-agile culture within the SKA Organisation.

PROJECT CONTEXT

The Square Kilometre Array Phase 1 (SKA1) project [1] is approaching a system-level Critical Design Review (CDR) in early December 2019. The design phase has been conducted after partitioning the SKA system on different subsystems or *elements*, and different consortia have developed a design for different subsystems of the telescope. Most elements have now passed their element-level CDR, increasing the confidence in the overall maturity of the design, but the process also highlighted some system-level issues [2]. A bridging phase has now been initiated, covering the period between element-level CDRs and the start of the construction phase. In this phase those issues are being tackled by adopting a system level approach, as the pre-construction consortia dissolved after their CDR.

The Need for an Agile System Approach

As indicated, the element-level design reviews highlighted several system-level issues, with emphasis on the lack of a single system-level perspective. Some inconsistencies have been identified in the interfaces between subsystems, there are some interfaces with not clearly demarcated responsibilities, and there are inconsistencies due to conflicting assumptions between the parties of an interface.

Compounding the issue, the design phase, and specially the way it has been targeted and delivered, has been strongly document-based. There have been several modelling attempts in order to provide a single source of truth for designs, but the final deliverable has always been a document. This is specially problematic for software components, as software architectures need validation by means of prototyping, with particular attention to their data exchange interfaces.

Upon reflection, it appears that the principles stated in the Agile Manifesto [3] well describe this situation:

- **Individuals and interactions** are needed in order to take a system approach and bridge the communication between different elements; currently that communication is mediated by formally issued documents such as Interface Control Documents (ICDs), but they remember difficult to interpret for many software engineers.
- **Working software** is needed in order to validate many assumptions in the design via evolutionary prototyping. This highlights the need for an integrated approach to software development.
- **Customer collaboration** is reflected in the necessity of a major interaction with the users of the telescope, starting from the preliminary design phase. This interaction is needed to drive the software development in an iterative fashion.
- **Responding to change** is essential for a project that still has many unknowns. Scientific projects are a complex endeavour and the SKA project is a complex system. In complex systems, the interactions between different parts resolve and identify patterns that cannot be traced, or even disappear, when we examine the parts in isolation. The complexity will evolve also in time with the increased understanding of the system itself.

Technical Implementation

The Agile principles do not live in isolation, and in recent years a set of technical best practices have emerged in the software world [4], trying to leverage agility through software craftsmanship in order to increase the quality of the software developed. Chief amongst these are:

Continuous Delivery [5] is the ability to get changes of all types – including new features, configuration changes, bug fixes and experiments —- into production, or into the hands of users, safely and quickly in a sustainable way. Addressing the development of a complex system incrementally and with a cohesive approach, greatly benefits from continuous integration and deployment of the different software components.

DevOps [6] is an essential practice that eliminates bottlenecks in the software development life-cycle, thereby making continuous delivery possible. DevOps promotes a process where teams can make decisions about their products by being responsible for their entire life cycle. This is essential

* m.bartolini@skatelescope.org

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA012

A GENERAL MULTIPLE-INPUT MULTIPLE-OUTPUT FEEDBACK DEVICE IN TANGO FOR THE MAX IV ACCELERATORS

P. Bell*, M. Sjöström, M. Lindberg, V. Martos, V. Hardion, MAX IV Laboratory, Lund, Sweden

Abstract

A general multiple-input multiple-output feedback device has been implemented in TANGO for various applications in the MAX IV accelerator system. The device has a configurable, weightable list of sensors and actuators, response matrix inversion (via SVD), gain and frequency regulation, takes account of the validity of the sensor readings and limits of actuator settings, and may respond to external interlocks. In the storage rings, it performs the slow orbit feedback using the 10 Hz data stream from the Libera Brilliance Plus Beam Position Measurement electronics, reading 400 (72) beam positions in the large (small) ring as sensor inputs. The position readings are received as TANGO events and a corrector-to-beam-position response matrix calculation outputs the corrector magnet settings. In the linac, the device is used for the trajectory correction, again with sensor input data sent as TANGO events, in this case from the Single Pass Beam Position Measurement electronics. The device is also used for tune feedback in the storage rings, making use of its own polling thread to read the sensors. Future developments will see a dedicated slow orbit feedback device derived from the general implementation in order to integrate the hardware-based fast orbit feedback, while the general device is also seeing new applications at the beamlines.

INTRODUCTION

The accelerator complex at the MAX IV laboratory consists of a 3 GeV, 250 m long full energy linac, two storage rings of 1.5 GeV and 3 GeV and a Short Pulse Facility. During 2019 eight beamlines are receiving users and the 3 GeV ring is now well proven for regular delivery to users at 250 mA stored current.

The MAX IV control system has a three-tier architecture, with specific hardware handling the real-time tasks and TANGO [1] representing the middle tier as the primary control system. For the client layer, physicists and operators can interact with TANGO via its Python and MATLAB bindings or through the SARDANA [2] layer which brings a macro server and standardised Graphical User Interfaces based on TAURUS [3]. In the control of the accelerator the MATLAB layer is used extensively, taking advantage of the rich "accelerator toolkit" developed in the wider accelerator physics community [4].

During early operation of the machine several use cases for feedback systems implemented in the TANGO layer were identified. Being able to take advantage of the TANGO event system, such devices are more performant than equivalent applications implemented in the MATLAB client layer. A TANGO device for the slow orbit feedback (SOFB) in the storage rings was first to be developed, but at an early stage this was generalised into the current TANGO Feedback Device such that it can be configured for any multiple input multiple output (MIMO) application. The first section of this report describes the design and main features of the general TANGO Feedback Device and the second section details some of its current applications at MAX IV.

TANGO FEEDBACK DEVICE DESIGN AND FEATURES

A Generic Device

The TANGO Feedback Device, written in Python, is completely generic and configurable for both MIMO and SISO systems. It currently implements an (adjustable) proportional gain term only, as this was found to be sufficient for the initial SOFB application. Interaction with the device is through the attributes and commands shown in Fig. 1, which also illustrates the state logic of the device. Two writeable spectrum attributes *SensorNames* and *ActuatorNames* provide proxies to TANGO attributes that act in those roles, i.e. provide the input signals to and writeable outputs from the feedback calculation. These may be weighted. The *SensorReferenceValues*, i.e. target values, must also be provided and the *ResponseMatrix*, **R**, which gives the change in the sensor response for unit change in the actuator settings.

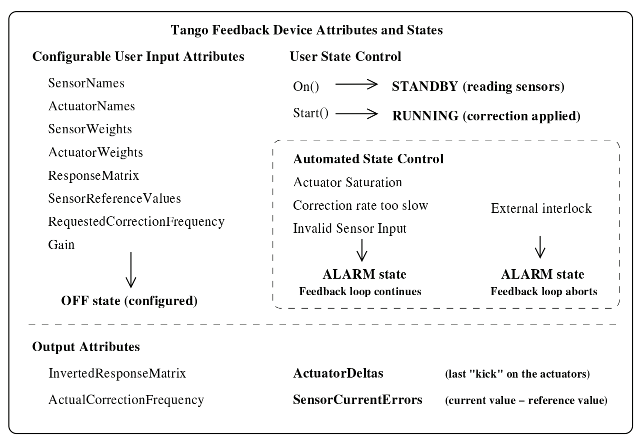


Figure 1: Commands, selected attributes and states of the generic TANGO Feedback Device. The output attributes *ActuatorDeltas* (the change in actuator settings) and *SensorCurrentErrors* (the different between the sensor values and the target) are updated for each iteration of the feedback. They are spectrum attributes but will have length 1 in case of a SISO application (one sensor, one actuator).

After configuring the above attributes the feedback device will be in OFF state. Moving to STANDBY state starts

* paul.bell@maxiv.lu.se

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-WEPHA013

PROGRAMMABLE LOGIC CONTROLLER SYSTEMS FOR SPIRAL2

C. Berthe*, F. Bucaille, G. Delavallée, G. Duteil, C. Hocini, J. R. Roze, A. Trudel, Q. Tura
GANIL, Caen, France
R. Touzery, CEA-IRFU, Gif-sur-Yvette, France
P. Graheling, CNRS-IPHC, Strasbourg, France

Abstract

PLC provides a large part of the SPIRAL 2 project's commands. The SPIRAL2 project is based on a multi-beam driver in order to allow both ISOL and low-energy in-flight techniques to produce Radioactive Ion Beams (RIB). A superconducting light/heavy-ion linac with an acceleration potential of about 40 MV capable of accelerating 5 mA deuterons up to 40 MeV and 1 mA heavy ions up to 14.5 MeV/u is used to bombard both thick and thin targets. The PLCs provide vacuum control, access control, part of the machine protection system, control of the cryogenic distribution system, cooling controls, control of RF amplifiers, they are associated with the safety control system. The standards used are presented as well as the general synoptic of the PLC control system. The details of the major systems are presented, the Cryo distribution, the machine protection system, a safety system.

INTRODUCTION

Officially approved in May 2005, the GANIL SPIRAL2 radioactive ion beam facility was launched in July 2005, with the participation of French laboratories (CEA, CNRS) and international partners [1]. In 2008, the decision was taken to build the SPIRAL2 complex in two phases: A first one including the accelerator, the Neutron-base research area (NFS) and the Super Separator Spectrometer (S3), and a second one including the RIB production process and building, and the low energy RIB experimental hall called DESIR [2].

In October 2013, due to budget restrictions, the RIB production part was postponed, and DESIR was planned a a continuation of the first phase. The first phase SPIRAL2 facility is now built, the accelerator is installed [4]. The French safety authority agreement is now validated and the accelerator is under testing with the aim of obtaining the first beam in 2019 [1].

After being pre accelerated by a RFQ, the primary stable beams (deuterons, protons, light and heavy ions) accelerated by the Linac will range from a few 10 µA to 5 mA in intensities, and from 0,75 A.MeV up to 14,5A.MeV for heavy ions, 20 A.MeV for deuterons and 33 MeV for protons in energies. PLC provides a large part of the SPIRAL 2 project's commands.

THE FUNCTIONS PERFORMED BY THE PLCS

The details of each type of function are indicated with its specificities and main characteristics. The selected architectures are also described for each system. The general overview shows the PLC control system and its organization (Fig. 1). Each color includes the type of process processed by the control system as well as each type of main function processed.

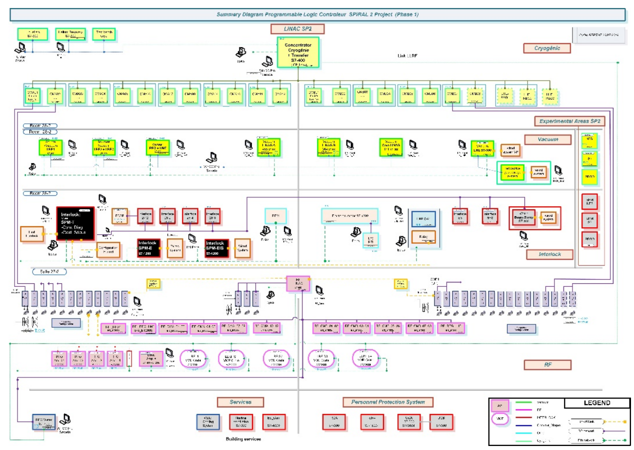


Figure 1: Summary Diagram of the system.

CRYOGENICS

The cryogenics process itself includes several systems: the liquefier, the helium recovery system, the helium transfer line, the cryomodules (19). The liquefier control system has been provided by the subcontractor with a supervision under "PC Vue". This system is linked to two other systems, the helium recovery system, which is used to collect and store evaporated helium in a flexible tank. This system uses a pressure storage and helium purification system to remove pollution after use. The link with the accelerator is made with the helium transfer line control system itself in connection with each PLC (ET 200 CPU) controlling the cryomodules. These systems provide temperature measurements, level measurements and helium level and pressure controls in the cryomodule [3]. A system coordination and centralized interface management is performed by a concentrator PLC that performs these functions as well as the management of the helium transfer line. This control system is completed by a motor control function for controlling the frequency tuning of the RF cavities. Brushless motors were chosen because of their performance, adaptability and ease of integration into PLCs. Win CC Pro supervision was used as the interface to this system. It makes it possible to monitor the state of the system, to effluent controls and to see the state of the sequences of the control graphs and their transitions evolve directly. The amount of data is

* cyrille.berthe@ganil.fr

ISBN: 978-3-95450-209-7   ISSN: 2226-0358   doi:10.18429/JACoW-ICALEPCS2019-WEPHA014

EPICS ARCHIVER APPLIANCE - INSTALLATION AND USE AT BESSY/HZB*

T. Birke†, Helmholtz-Zentrum Berlin für Materialien und Energie GmbH, Berlin, Germany

Abstract

After 2 years of tests and development, the EPICS Archiver Appliance went into operation at HZB/BESSY in April 2018. After running for a year as an optional new archiver, the EPICS Archiver Appliance switched places with the old Channel Archiver and is now the central productive archiver in currently three installations at HZB.

To provide a smooth transition from the Channel Archiver to the EPICS Archiver Appliance for end users, some frontends like, e.g., the ArchiveViewer and other applications needed some modifications to be fully usable. New retrieval frontends are also provided and will replace the ArchiveViewer in the future. In addition, the versatile retrieval API rapidly improved the development of Python applications for analysis and optimization. Experiences with installation, configuration, maintenance and use of the EPICS Archiver Appliance will be shared in this paper.

INTRODUCTION

History

Since the very beginning of user operation at BESSY-II in 1998, the Channel Archiver was the newest technology to archive control system data in an EPICS [1] based control system environment. The Channel Archiver came with a few basic coponents:

ArchiveEngine The backend application to monitor a configured set of process variables and create archiver datafiles on a filesystem as well as an associated index file.

ArchiveDataServer an XMLRPC-server, that provides a web based API to let clients retrieve data from the archives.

ArchiveDaemon daemon to control starting and stopping a set of ArchiveEngines (see Fig. 1).

ArchiveDataTool, ArchiveManager, ArchiveExport, ... a set of commandline tools to manage archived data directly on the archive server.

ArchiveViewer an easy to use Java client application to search for archived PVs and retrieve, plot and correlate data graphically (see Fig. 2).

This setup has been a reliable workhorse for many years and users gladly accepted the ease of use of the ArchiveViewer to access data from anywhere in the world, started using JavaWebStart from the central BESSY/HZB Archiver Web page. In addition, generic tools

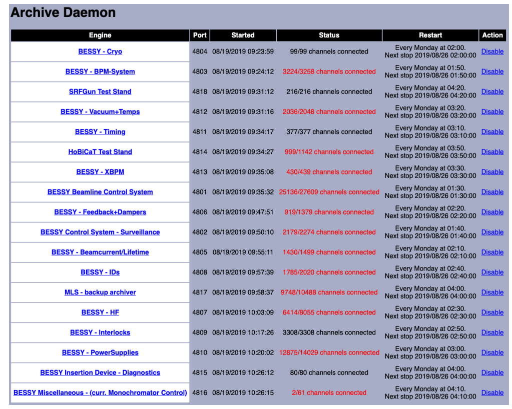


Figure 1: ArchiveDaemon with 18 ArchiveEngines configured.

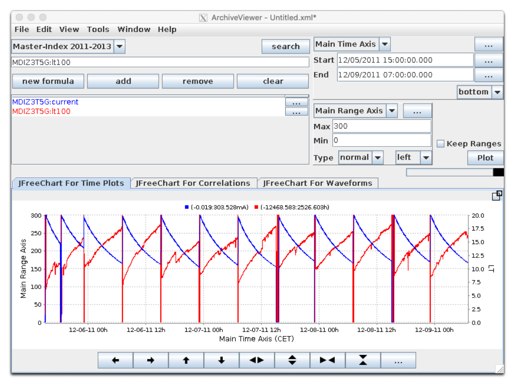


Figure 2: Original version of ArchiveViewer developed by S. Chevtsov especially for Channel Archiver.

have been developed to provide mostly pre-configured views (shift-handover, vacuum history...).

With the amount of data being archived growing over the years (see Fig. 3) managing the size-limited archive index files of 30 TB of data became a serious issue for administrators as well as for users.

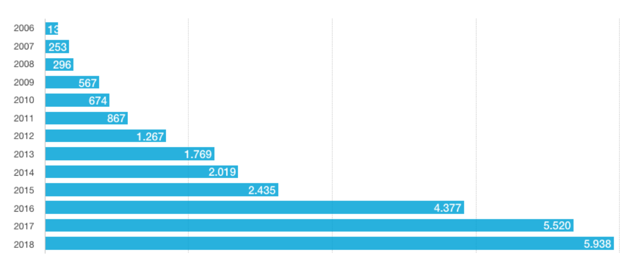


Figure 3: Growth of consumed storage by Channel Archiver data from 2006 until 2018 in GB/year.

* Work funded by BMBF and Land Berlin
† thomas.birke@helmholtz-berlin.de

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA016

A/D AND D/A PROCESSING UNIT FOR REAL TIME CONTROL OF SUSPENDED MASSES IN ADVANCED VIRGO INTERFEROMETER

M. Bitossi[†,1], A. Gennai, D. Passuello, National Institute for Nuclear Physics, INFN, Pisa, Italy
on behalf of the VIRGO Collaboration
[1]also at European Gravitational Observatory (EGO), Cascina (PI), Italy

Abstract

Advanced VIRGO is the project to upgrade the VIRGO interferometric detector of gravitational waves and has been completed in 2017, allowing VIRGO to join with LIGO in an observation state and the creation of the first network of three gravitational wave detectors. The goal of Advanced VIRGO is the detection of gravitational waves. A major upgrade consisted of the design of a new control electronics of the seismic isolation systems called Super-Attenuators. We present a new compact A/D-D/A conversion and processing unit used in the Advanced VIRGO control electronics upgrade. The unit consists of an analog to digital conversion stage that samples the input signals. One Altera Cyclone IV GX FPGA that collects, pre-processes and sends data to a Texas Instruments TMS320C6678 DSP using a PCI Express GEN2 link with 2 lanes at 2.5 Gbps. The DSP processes data and sends outputs to the FPGA for feeding the digital to analog conversion stage. The unit is equipped with six low noise and distortion 4 MSPS - 24-Bit A/D converters and six low noise and distortion 24-Bit D/A converters with sampling rates up to 640 kHz in Direct Stream Digital (DSD) mode and 384 kHz in PCM mode. Two FPGA transceivers are used for PCI Express link while a third one is used for a custom data transmission optical link. The TMS320C6678 is a multi-core fixed and floating point architecture DSP, with eight cores running at 1.25 GHz. The unit has a dedicated input to interface directly to GPS timing signals. It is pluggable into a custom MicroTCA backplane. Using multiple units, sharing the same MicroTCA backplane and communicating via 4 Serial RapidIO lanes at 5 Gbps, it allows composing a flexible and modular system with a large number of channels suitable for the Advanced VIRGO control electronics.

INTRODUCTION

Modern Astrophysics detectors, such as LIGO [1] and VIRGO [2], have been upgraded [3][4] to improve their sensitivity with the aim of increasing the number of observable galaxies (and thus the detection rate) by three orders of magnitude. The goal of Advanced VIRGO is the observation of gravitational waves, discovered with the first LIGO-VIRGO detection of a binary black holes merger, opening a new observation window on the universe [5]. One of the major upgrades concerned the control electronics of the Super Attenuators (SA). SAs are complex mechanical structures used to insulate optical elements from seismic noise. The control electronics is used to manage sensors and actuators, namely accelerometers, displacement sensors, stepping motors and magnet-coil actuators placed in the Super Attenuators [6]. We present a flexible and easily expandable A/D-D/A conversion and processing unit based on a powerful DSP and one FPGA that controls up to six high-performance 24-bit A/D and six 24-bit D/A converters and process data, used in the Advanced VIRGO control electronics upgrade. Up to 16 units are used to control a single SA, hosted into two custom MicroTCA crates, for 10 SAs for the entire Advanced VIRGO upgrade.

ARCHITECTURE

A photo of the unit is shown in Figure 1. The FPGA (Altera EP4C30GXCF23C7N) is responsible for interfacing A/D-D/A converters with the DSP (Texas Instruments TMS320C6678) through a PCI Express link x2 at 2.5Gbps. The FPGA takes care to set up D/A and A/D converters, distributing a start conversion signal so that all devices begin the conversion process at the same time; in addition, it takes care of receiving and transmitting GPS timing. Concerning the D/A converters, the digital data are sent using DSD and PCM data format. A sampling frequency up to 640 kHz and 384 kHz can be achieved using DSD and PCM data format respectively. Six parallel D/A channels are simultaneous managed. The FPGA also reads the digitized data coming from the A/D converters. All the A/D channels are read in parallel and then digitized signals are transmitted to FPGA using LVDS differential channels, with a maximum throughput of 4 MSPS per channel. Three PLL distributors take care of distributing all necessary clocks to the devices: PCI Express reference clock, A/D and D/A clock and general clock for DSP and FPGA. The DSP takes care of processing data, manages local controls and it interfaces through the MicroTCA backplane using Serial RapidIO protocols for communicating with other units for properly controlling a SA. Up to 16 units are necessary to equip and control an entire SA, for a total of 150 units for 10 SAs [6].

SUPERATTENUATOR (SA)

Superattenuator (SA), shown in Figure 2, is complex mechanical structure used for isolating the VIRGO interferometer from the seismic noise. SA is a chain of cascaded mechanical filters used for suspending and optical benches. SA is a low pass filter in all degrees of freedom, capable of attenuating the ground motion by a factor 10^{15} at 10 Hz. Up to ten SAs are used for isolating the VIRGO interferometer from the seismic noise [6].

†email address: Massimiliano.bitossi@pi.infn.it

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA017

INTEGRATION OF WIRELESS MOBILE EQUIPMENT IN SUPERVISORY APPLICATION

S. Blanchard*, R. Ferreira, P. Gomes, G. Pigny, A. Rocha
CERN, Geneva, Switzerland

Abstract

In CERN accelerators, pumping group stations and bake-out control cabinets are temporarily installed close to vacuum systems for their commissioning.

The quality of the beam vacuum during operation depends greatly on the quality of the commissioning. Therefore, the integration of mobile equipment in the vacuum supervisory application is primordial. When connected to the control system, they appear automatically integrated in the synoptic. Mobile equipment are granted with the same level of remote control, diagnostics and data logging as fixed equipment.

The wireless connection and the communication protocol with the supervisory application offer a flexible and reliable solution with high level of integrity.

INTRODUCTION

The vacuum systems of the CERN's accelerators reach high (below 1.10^{-6} mbar) and ultra-high (below 1.10^{-9} mbar) vacuum. The vacuum systems of accelerators are divided into sectors, delimited by vacuum valves or windows. Vacuum sectors reduce and even stop the propagation of a sudden pressure increase and allow independent venting and interventions.

Only few sectors have pumping group stations [1] permanently installed; however, all the others do not have any. For these sectors, pumping group stations and optionally bake-out cabinets [2] are temporarily installed during interventions, commissioning and re-commissioning of the vacuum systems.

Pumping group stations are required to achieve high vacuum prior to the start of permanently installed pumps, such as ion pumps or Non-Evaporable Getter (NEG) pumps. A typical pumping group station is composed of a primary and a turbo-molecular pump, a gauge, several valves and a PLC that drives the process and manages the interlocks.

When ultra-high vacuum is required, a bake-out cycle is mandatory to decrease the outgassing rate of the vacuum vessel and to activate the NEG thin-film coatings, if present. Bake-out control integrates a Proportional-Integral-Derivative (PID) regulation, interlocks and troubleshooting management.

At the end of the intervention, mobile equipment (i.e. pumping group stations and bake-out cabinets) are removed from the accelerator facility.

Both mobile pumping group stations and mobile bake-out cabinets are locally driven by a PLC. This makes the integration of these equipment in the vacuum controls framework possible without the need of extra interface systems, more difficult to maintain.

A typical intervention on a vacuum sector requiring a bake-out cycle may take several days. Access to the LHC tunnel is not easy nor fast: more than 45 minutes are needed to reach some tunnel areas. In these cases, remote control is mandatory.

The remote control of mobile equipment is not something new. The same bake-out cabinet and a previous version of pumping group PLCs are remotely controlled using dedicated Profibus networks installed in the LHC tunnel. This remote control is used since the beginning of the LHC. However, it has shown some limitations: wired Profibus networks require a lot of maintenance, each connection/disconnection requires the intervention of a fieldbus expert as it may break the communication of other equipment already connected on the same network. This makes connection management difficult. A software consolidation has already been done to improve the reliability, but the architecture has several limitations. There were no alive counters from the pumping group or the bake-out cabinet PLCs, making some communication interrupts difficult to detect. Data was only archived locally and not in CERN central archiving system.

The new wireless mobile equipment control relies on alive counters to guarantee the system integrity. Alive counters are read in the local PLC of pumping group stations and bake-out cabinets. Pumping group stations and bake-out cabinets are managed in a very similar way as “fixed” equipment and so get all of their features, including the central archiving (lhcLogging). With the new wireless system, data logging of mobile equipment are now permanently archived.

SCOPE

Mobile equipment concerned are pumping groups and bake-out cabinets. The vacuum systems of the CERN facilities require more than 250 mobile pumping groups and more than 120 bake-out cabinets. Around 80 mobile pumping groups and 70 bake-out cabinets are hardware compatible with the wireless communication system. In addition, not all of them will be connected at a same time. For instance, for the LHC application, not more than 40 mobile equipment will be connected simultaneously.

CONCEPT

The first idea was to create data points on-the-fly and on demand in the supervisory application when mobile equipment are effectively connected. Even if this solution is technically possible, it shows two major limitations: creating, configuring and deleting data points during operation

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA018

TESTING SOLUTIONS FOR SIEMENS PLC PROGRAMS BASED ON PLCSIM ADVANCED

Gy. Sallai*, D. Darvas, E. Blanco, CERN, Geneva, Switzerland

Abstract

Testing Programmable Logic Controllers (PLCs) is challenging, partially due to the lack of dedicated support for testing. Isolating a part of the PLC program, feeding it with test inputs and checking the test outputs often require manual work and physical hardware. The Siemens PLCSIM Advanced tool is a simulator solution for new generation Siemens PLCs and provides a rich application programming interface (API). This work presents a testing workflow for PLC programs built upon the capabilities of the PLCSIM Advanced API and the TIA Portal Openness API. Our tool takes a test case described in an intuitive but powerful tabular format, which is then executed on the full PLC program or a selected part of it. Outputs are captured automatically via the simulator API. Experience with this workflow shows that it offers an automated and scalable solution for PLC program testing and is applicable for multiple levels of testing without the need of using a physical hardware.

INTRODUCTION

Programmable Logic Controllers (PLC) are widely used devices for process control and automation. As their error-free operation is crucial, thorough verification and testing is required to gain confidence in their correct operation. While a plethora of tools are available for testing other languages and platforms, testing support for PLC programs is rather limited. As such, PLC program testing is often restricted to higher levels, with the need for a physical hardware and some tedious manual configuration steps.

This means that there is a need for improvement in PLC program testing. Testing should be easier (with easy configuration and less manual effort), more accessible (by reducing the need for dedicated testing hardware) and should not require the modification of existing source code.

Our solution, built upon Siemens PLCSIM Advanced, offers a scalable, accessible and user-friendly workflow for testing Siemens PLC programs. Using a simulator allows us to perform testing without having hardware in the loop, and by using the API offered by the simulator, the whole process can be automated. We describe a test table format, which is easy to understand and allows developers to conveniently define test cases. In addition, we introduce an automated process to isolate certain parts of the PLC program. This isolation allows us to perform lower-level (such as unit or integration) testing on PLC programs. As the whole process is automated, it may easily be used in conjunction with continuous integration solutions.

* Corresponding author. E-mail: gyula.sallai@alumni.cern

PLC PROGRAM TESTING

Testing is the process of verifying that a system is fit for its purpose and satisfies its specified requirements. For the remainder of the paper, we define three levels of program testing, relevant to our use cases.[1]

- *Unit testing* aims to test small, individual components of a system. For PLC programs, a practical unit-level component is a single function or function block. Unit testing a component requires the isolation of said component from its dependencies and dependant blocks.
- *Integration testing* targets the interaction of some units of the system, i.e. the interaction of different program blocks.
- *System testing* is performed on the whole application, checking whether the system satisfies its functional requirements.

In the domain of PLC programming, testing is usually done at the system testing level. The most commonly used testing procedures are *factory acceptance testing* (FAT) and *site acceptance testing* (SAT), as defined by the IEC 62381 standard [1]. FAT aims to verify system correctness by scrutinizing the production software on a (usually) dedicated testing hardware in laboratory conditions. SAT is done on the premises of the final installation, verifying that the software and the hardware equipment work together as intended.

Most testing workflows for PLC programs require physical hardware. However, testing with the involvement of physical hardware poses several challenges.

- It is hard to automate: hardware cannot be acquired, assigned, and configured on-demand.
- Feeding the inputs and capturing the outputs often requires manual effort, e.g. supplying inputs through the supervision system.
- Outputs cannot be extracted precisely – the latency induced by I/O communication makes it difficult to extract values at an arbitrary point of the program.
- For lower level (unit and integration) testing, precision and traceability often demands to check the behaviour of the program within really small time periods or at a given program location. However, there is no support in PLC hardware for such use cases.

[1] Unless indicated otherwise, this paper follows the terminology defined by the International Software Testing Qualifications Board (ISTQB). See `https://glossary.istqb.org` for further information.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA019

MONARC: SUPERVISING THE ARCHIVING INFRASTRUCTURE OF CERN CONTROL SYSTEMS

J-C. Tournier*, E. Blanco Vinuela† , CERN, Geneva, Switzerland

Abstract

The CERN industrial control systems, using WinCC OA as SCADA (Supervisory Control and Data Acquisition), share a common history data archiving system relying on an Oracle infrastructure. It consists of 2 clusters of two nodes for a total of more than 250 schemas. Due to the large number of schemas and of the shared nature of the infrastructure, three basic needs arose: (1) monitor, i.e. get the inventory of all DB nodes and schemas along with their configurations such as the type of partitioning and their retention period; (2) control, i.e. parametrise each schema individually; and (3) supervise, i.e. have an overview of the health of the infrastructure and be notified of misbehaving schemas or database node.

In this publication, we are presenting a way to monitor, control and supervise the data archiving system based on a classical SCADA system. The paper is organized in three parts: the first part presents the main functionalities of the application, while the second part digs into its architecture and implementation. The third part presents a set of use cases demonstrating the benefit of using the application.

INTRODUCTION

The CERN team in charge of the Industrial Controls systems, in collaboration with equipment groups, has developed and maintains around 250+ controls applications whose domain range from cold and warm magnets protection, to cryogenics, cooling and ventilation or electrical network supervision systems. These applications, based on WinCC OA [1] and the CERN standard frameworks JCOP [2] and UNICOS [3], archive their values in a centralized Oracle infrastructure. This infrastructure is split in two main parts with one cluster of two nodes dedicated to the applications belonging to the LHC magnets protection domain (referred as QPSR in 1) and another set of two nodes dedicated to the other applications (referred as SCADAR in Figure 1). To improve the availability in case of failure of the SCADAR database, the infrastructure is duplicated to a different datacenter. Each WinCC OA application has a dedicated database schema hosted by one of the two clusters holding the archive values of the application but also some metadata related to how the schema should be managed. Figure 1 shows an overview of the Oracle database infrastructure for the CERN industrial control applications based on WinCC OA.

* tournier@cern.ch
† enrique.blanco@cern.ch

CHALLENGES

While having two Oracle database clusters dedicated to all 250+ WinCC OA industrial control applications allows to have a robust and scalable archiving solution, it poses a number of challenges in terms of inventory, configuration, monitor and technical expertise:

- **Configuration** Once created for an application, each schema needs to be configured. The two main configuration parameters are the retention period, i.e. for how long the archived data should be kept, and the partitioning policy, i.e. how the tables storing the different archived values are partitioned to maximize query performance. The configuration of a schema may evolve over time depending on the user requirements, but also on the application characteristics, i.e. the number of signals to be archived and their frequency.

- **Monitor** All schemas are sharing a common resource, i.e. one of the database cluster, and the behaviour of one schema can penalize the performance experienced by the users of other schemas. It is therefore important to supervise each schema in order to be alerted when some queries are taking too much time or when a schema archives too much data compared to what was initially expected.

- **Inventory** To each WinCC OA application is associated a database schema with its own configuration such as the schema version, the data retention period or the archived data partitioning policy. Having an exhaustive list of schemas with their configuration is not necessarily trivial as applications have been developed and deployed over the years and may have been retired by the users without informing all interested parties.

- **Technical Expertise** While the Oracle database clusters themselves are managed by the CERN central database team, the schemas are under the responsibility of the team in charge of the industrial control systems whose main technical expertise is in building control systems but not necessarily in database technology.

All the challenges presented in this section can be tackled by a set of custom SQL queries which need to be applied to 250+ schemas taking into account the two different clusters as well as each node of the cluster. There is therefore the need for a high level tool to manage, configure and monitor the database schemas without having to deal with complex SQL queries.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA020

PUSHING THE LIMITS OF TANGO ARCHIVING SYSTEM USING PostgreSQL AND TIME SERIES DATABASES

R. Bourtembourg, S. James, J.L. Pons, P. Verdier, ESRF, Grenoble, France
G. Cuni, S. Rubio-Manrique, ALBA-CELLS Synchrotron, Cerdanyola del Vallès, Spain
G.A. Fatkin, A.I. Senchenko, V. Sitnov
BINP SB RAS and NSU, Novosibirsk, Russia
L. Pivetta, C. Scafuri, G. Scalamera, G. Strangolino, L. Zambon
Elettra-Sincrotrone Trieste S.C.p.A., Basovizza, Italy
M. Di Carlo, INAF - OAAB, Teramo, Italy

Abstract

The Tango HDB++ project is a high performance event-driven archiving system which stores data with micro-second resolution timestamps, using archivers written in C++. HDB++ supports MySQL/MariaDB and Apache Cassandra back-ends and has been recently extended to support PostgreSQL and TimescaleDB [1], a time-series PostgreSQL extension. The PostgreSQL back-end has enabled efficient multi-dimensional data storage in a relational database. Time series databases are ideal for archiving and can take advantage of the fact that data inserted do not change. TimescaleDB has pushed the performance of HDB++ to new limits. The paper will present the benchmarking tools that have been developed to compare the performance of different back-ends and the extension of HDB++ to support TimescaleDB for insertion and extraction. A comparison of the different supported back-ends will be presented.

INTRODUCTION

The HDB++ Tango archiving system [1] relies on the Tango archive events feature to collect Tango attributes values coming from one or several Tango Control Systems and then store these values in the Database back-end of your choice. The following back-ends are currently supported: MySQL/MariaDB, Cassandra, PostgreSQL, TimescaleDB and Elasticsearch. This list could be easily extended, thanks to the layered design of the HDB++ architecture.

HDB++ DESIGN

The HDB++ Tango archiving system relies on two main components:

- The EventSubscriber Tango device server which subscribes to Tango archive events for a list of Tango attributes and store the received events in a database
- The ConfigurationManager Tango device server which simplifies the archiving configuration and management

An abstraction library, named *libhdb++* decouples the interface to the database back-end from the implementation. To be able to store data to a specific database back-end, the EventSubscriber and the ConfigurationManager Tango devices dynamically load a C++ library implementing the methods from the libhdb++ abstract layer. The libhdb++ back-end library is selected via a Tango device property. This allows to use the same tools to configure and manage the archiving system with all the supported Database back-end. The archiving part of the HDB++ design is presented in Fig. 1.

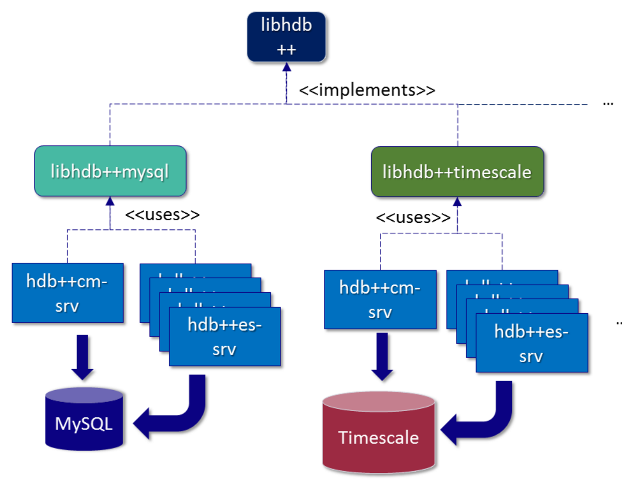


Figure 1: HDB++ Tango devices design.

Tools to help configuring the system, retrieving or viewing the archive data are also available.

SUPPORTED BACK-ENDS

MySQL/MariaDB

MySQL is a well known, widely adopted SQL database engine. After the acquisition by Oracle, in 2010, a complete open-source fork, named MariaDB, became available. MySQL and MariaDB are almost inter operable, even if some differences in the supported data types and database engines require a careful approach.

HDB++ at Elettra and FERMI The HDB++ MySQL back-end has been in production at Elettra and FERMI since 2015. Both accelerators share the same architecture: two nodes, configured in high-availability, are in charge of running all the virtual machines hosting the control system

[1] https://timescale.com

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA021

FREE-ELECTRON LASER OPTIMIZATION WITH REINFORCEMENT LEARNING

N. Bruchon*, G. Fenu, F. A. Pellegrino, E. Salvato, University of Trieste, Trieste, Italy
G. Gaio, M. Lonza, Elettra Sincrotrone Trieste, Trieste, Italy

Abstract

Reinforcement Learning (RL) is one of the most promising techniques in Machine Learning because of its modest computational requirements with respect to other algorithms. RL uses an agent that takes actions within its environment to maximize a reward related to the goal it is designed to achieve. We have recently used RL as a model-free approach to improve the performance of the FERMI Free Electron Laser. A number of machine parameters are adjusted to find the optimum FEL output in terms of intensity and spectral quality. In particular we focus on the problem of the alignment of the seed laser with the electron beam, initially using a simplified model and then applying the developed algorithm on the real machine. This paper reports the results obtained and discusses pros and cons of this approach with plans for future applications.

INTRODUCTION

Free-Electron Lasers (FELs) are complex systems that require continuous effort by experts in order to maintain the high performance that users demand. For seeded FELs, such as FERMI, there are parameters related to the alignment of the seed laser with the electron beam in addition to the large number of electron beam parameters (many dozens at FERMI) [1–4]. Perhaps the most critical parameter is the spatial-temporal overlap between the electron and laser beams in the modulator undulator, the main source of the FEL instability.

The existing feedback systems [5] are able to maintain a steady FEL intensity by controlling the trajectories of the two beams on a shot-to-shot basis. To ease the procedure the electron beam trajectory is kept steady while the position of the seed laser is varied.

Currently, the maintenance of the optimal superimposition of the two beams is carried out by an automatic procedure that is based on the correlation between FEL intensity and the natural jitter in the seed laser parameters [6]. However, this approach cannot be successful if there is insufficient natural jitter: in this case the introduction of artificial noise can help to find the optimal overlap, but FEL performance is affected by the injected noise.

Typical model-free approaches [7] applied in FEL optimization have some intrinsic limitations: they require the availability of the objective function gradient, they are very sensitive to hyper-parameters, and they do not learn from previous experiences. An interesting option to overcome these limitations is given by Machine Learning algorithms, although these approaches can be extremely time consuming.

Nowadays, different optimization techniques are adopted in various FEL facilities [8]. OCELOT [9] has been developed since 2011 at European XFEL and is currently used at the Stanford Linear Accelerator Center (SLAC) [10, 11] and at Deutsches Elektronen-SYnchrotron (DESY) [12]. [13–15] adopt neural networks to model and control particle accelerators. In addition, [16] presents a proof-of-principle of a model-free approach applied at CERN using Deep Q-Learning.

In this paper we present preliminary results obtained using a simple RL algorithm, Q-Learning with Linear Function Approximation, on FERMI. The experiments have been carried out on two different systems.

ENVIRONMENTS

In this work, two different tasks have been considered. Both of them concern the trajectory control of a laser. The first task uses the service laser in the Electro-Optical Sampling station [17–19], while in the second task the seed laser of the undulator modulator of FERMI Free-Electron Laser is used.

In the EOS station the laser movement system is a standard optical alignment scheme, as shown in Fig. 1. It is composed of two planar tip-tilt mirrors [20], each axis of which is driven by a piezo-motor (horizontal and vertical motors), and two virtual screens based on Charged-Coupled Devices (CCDs) [21]. The position of the laser on the two screens is adjusted by moving the tip-tilts. The goal of the task is to align the laser such that it passes through a pair of predefined Regions of Interest (ROI), that must contain a certain fraction of the laser spot to successfully end an episode of the task. The performance of the agent on the task is measured online by the computing the product of the intensity measured in each of the the ROIs.

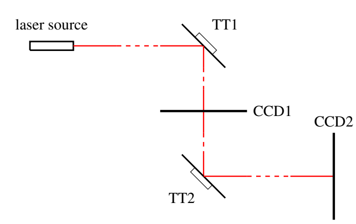


Figure 1: Experimental setup of the EOS laser alignment task. TT1 and TT2 are the tip-tilt mirrors while CCD1 and CCD2 are the virtual screen CCDs.

A simplified representation of the second task, alignment between the seed laser and electron beam at FERMI is shown

* niky.bruchon@phd.units.it

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA023

CO-SIMULATION OF HDL USING PYTHON AND MATLAB OVER Tcl TCP/IP SOCKET IN XILINX VIVADO AND MODELSIM TOOLS

Ł. Butkowski*, B. Dursun, C. Gumus, M. K. Karakurt
Deutsches Elektronen-Synchrotron DESY, Hamburg, Germany

Abstract

This paper presents the solution, which helps in the simulation and verification of the implementation of the Digital Signal Processing (DSP) algorithms written in hardware description language (HDL). Many vendor tools such as Xilinx ISE/Vivado or Mentor Graphics ModelSim are using Tcl as an application programming interface. The main idea of the co-simulation is to use the Tcl TCP/IP socket, which is Tcl build in feature, as the interface to the simulation tool. Over this interface the simulation is driven by the external tool. The stimulus vectors as well as the model and verification are implemented in Python or MATLAB and the data with simulator is exchanged over dedicated protocol. The tool, which was called cosimtcp, was developed in Deutsches Elektronen-Synchrotron (DESY). The tool is a set of scripts that provide a set of functions. This tool has been successfully used to verify many DSP algorithms implemented in the FPGA chips of the Low Level Radio Frequency (LLRF) and synchronization systems of the European X-Ray Free Electron Laser (E-XFEL) accelerator. Cosimtcp is an open source available tool.

INTRODUCTION

The correct operation of the Low Level Radio Frequency (LLRF) [1] and synchronization systems of the European X-Ray Free Electron Laser (E-XFEL) [2] accelerator requires a huge amount of Digital Signal Processing (DSP) calculations in real time. This task is mainly handled by FPGAs. In the high level design stage, the first step is to determine the exact algorithms that has to be implemented in FPGAs so the machine can operate. This work is executed with the help of tools such as MATLAB with Simulink or Python. Solutions are simulated and the exact formula is derived. Next the implementation process starts in which the code written in hardware description language (HDL) is developed. During this process the following problem always appear: how to verify functionality of written HDL code.

There are a few ways we can solve this problem. There are available methodologies and libraries which can help go trough this process. It is very common to write the testbench using HDL first. Next phase is to write the model of the component under test in order to go through the verification process. However, writing the DSP algorithm model in HDL is really difficult and time consuming. Additionally there is risk of introducing new errors in the model, which fail our verification. Most wanted in the verification process would be to use directly the same tools as the one in the high level design in a co-simulation process. This can improve significantly verification [3].

There are already available methods of HDL co-simulation with the use of high level programming languages. One of the way to use high level programming languages like C in a verification process is to use Direct Programming Language Interface (DPI) of the System Verilog [4] or Foreign Language Interface (FLI) [5] of the simulation tool. The simulation is fast but this method is not so straight forward and easy to use. It is platform depended or tool depended and requires recompilation of the code every change. One of the another way is to use Programming Language Interface (PLI) like the file input/output interface using the VHDL textio library [6]. This method has limitation in interactive simulations and reuseability.

We came with and another idea. Almost all of the HDL simulator tools support Tcl script language as an application programming interface, enabling control of the simulation using TCL commands. It also means that all the default Tcl features are available in the tool. The idea proposed was to use these features as the communication layer between HDL simulation tool and an external tool. The Tcl socket function has been found as a perfect candidate for this role. It is a build in command which opens a TCP network connection. The communication between simulation tool and high level language is done over TCP/IP socket. This is a good separation between these two. The solution gives a good balance between reuseability, easy to use and simulation speed. The TCP/IP socket protocol is well-know, widely used and has a build-in support in many tools.

The solution we came up with has been called cosimtcp [7]. It is a set of script in the form of libraries, which can be easily added to current or new simulation flow. Currently the solution is used with Xilinx Vivado, ModelSim, Matlab and Python tools.

In the following chapters the steps that has to be done to run simulation using cosimtcp libraries will be presented. Also the examples of the usage will be given. In the end prons and cons of this solution will be discussed.

CO-SIMULATION ARCHITECTURE

The block diagram of the co-simulation is presented in Figure 1. The simulation is divided into two main blocks. The server side: responsible for the HDL simulation of Unit Under Test (UUT) and the client side: responsible for the generation of stimulus and verification. Between them the data are exchanged using a dedicated protocol over a socket connection. There is one protocol used among all the tools.

* lukasz.butkowski@desy.de

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA025

INITIAL IMPLEMENTATION OF A MACHINE LEARNING SYSTEM FOR SRF CAVITY FAULT CLASSIFICATION AT CEBAF

A. Carpenter†, T. Powers, Y. Roblin, A. Solopova, C. Tennant
Jefferson Lab, Newport News, USA
L.Vidyaratne, K. Iftekharuddin, Old Dominion University, Norfolk, USA

Abstract

The Continuous Electron Beam Accelerator Facility (CEBAF) at Jefferson Laboratory is a high power Continuous Wave (CW) electron accelerator. It uses a mixture of of SRF cryomodules: older, lower energy C20/C50 modules and newer, higher energy C100 modules. The cryomodules are arrayed in two anti-parallel linear accelerators. Accurately classifying the type of cavity faults is essential to maintaining and improving accelerator performance. Each C100 cryomodule contains eight 7-cell cavities. When a cavity fault occurs within a cryomodule, all eight cavities generate 17 waveforms each containing 8192 points. This data is exported from the control system and saved for review. Analysis of these waveforms is time intensive and requires a subject matter expert (SME). SMEs examine the data from each event and label it according to one of several known cavity fault types. Multiple machine learning models have been developed on this labeled dataset with sufficient performance to warrant the creation of a limited machine learning software system for use by accelerator operations staff. This paper discusses the transition from model development to implementation of a prototype system.

INTRODUCTION

Jefferson Lab's Continuous Electron Beam Accelerator Facility (CEBAF) is a high power Continuous Wave (CW electron accelerator. It utilizes two styles of SRF modules, older, lower gradient C20/C50 modules and newer higher, gradient C100 modules. In 2013, the upgrade from 6 to 12 GeV was completed. This upgrade included the installation of 11 C100-style 100 MV cryomodules and associated RF systems (see Fig. 1) [1]. RF faults are routinely the largest contributor to lost beam time. Since the primary mechanism for reducing the occurrence of RF faults is to reduce cavity gradient, RF faults also adversely impact CEBAF's energy reach. Accurately identifying which cavity faulted allows operators to lower the gradient only on targeted cavities rather than the entire module. However, this process is complicated due to strong mechanical coupling between cavities within a C100 cryomodule whereby a fault in one cavity may precipitate faults in other nearby cavities [2]. Additionally, identifying the cause of the fault may allow engineering staff to determine remediation methods in place of gradient reduction. An effort has been underway to produce an automated machine learning system capable of identifying the location and cause of RF faults within the C100 cryomodules.

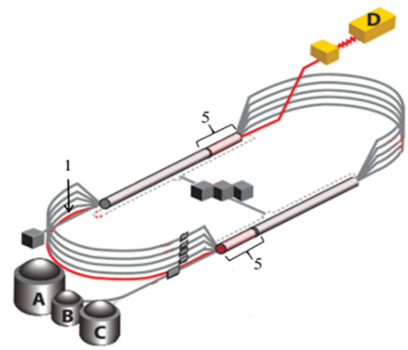


Figure 1: Schematic of the CEBAF accelerator showing the locations of the 11 C100 cryomodules for which RF fault data is recorded and analyzed.

MACHINE LEARNING SYSTEM OVERVIEW

Our machine learning system is comprised of four components: data generation, data storage, the machine learning model and its analysis, and the presentation of results. Creation of this system has required considerable effort related to the development of each component. The low-level RF hardware and EPICS control system was modified to generate synchronized diagnostic waveform data for each cryomodule. Waveform harvester software was written to collect and store the waveform data for each RF fault. A system expert then manually analyzed the stored waveform data to identify and label the cause and cavity location of the fault. This labeled dataset allows for the use of supervised machine learning techniques in developing a set of models for identification of fault location and cause. We turned the models into a software product capable of analyzing waveform data from a fault in real-time. Finally, creating additional software for displaying and aggregating the results of this analysis for CEBAF operations and engineering staff is an outstanding task we plan to tackle in the future.

Each component of the system acts as the foundation for the next, making it difficult to simply "write some software" that perform the desired analysis. Fortunately, each component provides immediate benefit making each incremental development worthwhile in its own right. For example, capturing and storing diagnostic information allows RF system experts to gain insight into the cause and location of individual faults. Accumulating and analyzing a large body of data allows operations and maintenance staff

* Work supported by Jefferson Science Associates, LLC under U.S. DOE Contract № DE-AC05-06OR23177
† adamc@jlab.org

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA026

INTEGRATING COTS EQUIPMENT IN THE CERN ACCELERATOR DOMAIN

O. Ø. Andreassen†, C. Charrondière, K. Develle, A. Rijllart, R. E. Rossel, J. Steen, J. Tagg, T. Zilliox, CERN, Geneva, Switzerland

Abstract

Successful integration of industrial equipment in the CERN accelerator complex relies mainly on 3 key components. The first part is the Controls Middleware (CMW). That provides a common communication infrastructure for the accelerator controls at CERN. The second part is timing. To orchestrate and align electronic and electrical equipment across the 27 km Large Hadron Collider (LHC) at sub nanosecond precision, an elaborate timing scheme is needed. Every component has to be configured and aligned within nanoseconds and then trigger in perfect harmony with each other. The third and last bit is configuration management. The COTS devices have to be kept up to date, remotely managed and compatible with each other at all times. This is done through a combination of networked Pre-eXecution Environment (PXE) mounting network accessible storages on the front ends, where operating systems and packages can be maintained across systems. In this article we demonstrate how COTS based National Instruments (NI) PCI eXtensions for Instrumentation (PXI) and cRIO systems have been integrated in the CERN accelerator domain for measurement and monitoring systems.

INTRODUCTION

In any control system of medium to large size it can be challenging to integrate new equipment. There might be custom rules, non-standard implementations or simply a vast amount of information and knowledge to overcome before one any new component can be added to the infrastructure. The CERN accelerator control system is no exception. Currently (2019) there are ~150.000 devices and about 5.000 unique device classes in production to control the accelerators. In addition, the timing accuracy of the LHC equipment are typically in the nanosecond or microsecond range, and data is acquired with speeds ranging from GHz (LHC Beam screens) to one sample pr. month (CAST) [1-3].

In addition to distance and time, there are several networks, databases, protocols and environmental considerations to take.

Motivation

The LabVIEW™ based Rapid Application Development Environment (RADE) was conceived as a result of an increasing need to quickly prototype and release control, analysis and test-bench tools in the CERN accelerator domain. The framework was initially targeting client type projects, doing analytical or commissioning type applications, but as the interest to use PXI and cRIO based test for not critical measurement applications for the accelerators complex grew, a need for full system integration became apparent. As a result, we started looking into the possibility to port the CERN middleware and timing system to the RADE framework [4].

MIDDLEWARE

The Controls MiddleWare (CMW) project was launched close to twenty years ago (Fig. 1). Its main goal was to unify middleware solutions used to operate the CERN accelerator complex. Initially the equipment access library "Remote Device Access" (RDA), was based on CORBA, however the ever growing demands from the run-time environment revealed shortcomings of the system and during the previous long shutdown of the LHC (2013-14) CORBA was replaced with ZeroMQ [5, 6].

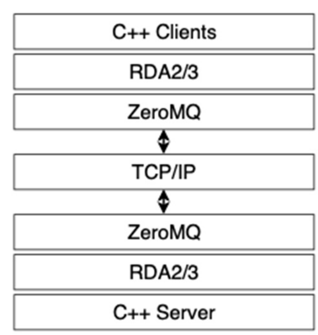


Figure 1: CMW architecture.

The CMW Device Model was originally rolled out in the PS and SPS control systems and is the same for the LHC. The model is based on named devices with properties and data fields within the property. Each device belongs to a Device Class and it is the Device Class that defines the properties, which can be used to access the device. By invoking get, set or subscribe on the device with the property name, the value of this property can be read or changed [7].

CMW RADE Integration

The CMW stack is integrated into RADE by using the built in "Call Library Function Node" in LabVIEW. A wrapper library around the RDA stack creates an instance that is kept in a factory pattern singleton, acting as a reference between subsequent calls in LabVIEW. The LabVIEW RADE CMW interface has been designed with ease of use and performance in mind, leveraging the standard "Open, Use Close" paradigm encouraged by the programming language (Fig. 2).

† odd.oyvind.andreassen@cern.ch

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA027

EVALUATION OF TIMING AND SYNCHRONIZATION TECHNIQUES ON NI CompactRIO PLATFORMS

O. O. Andreassen, C. Charrondiere, K. Develle, R. Rossel, T. Zilliox CERN, Geneva, Switzerland

Abstract

Proper clock synchronization between systems is key to successfully integrate data acquisition and control systems in the CERN accelerator domain. This applies to everything from simple diagnostics- to mission critical systems. The internal clock of a National Instruments based CompactRIO (NI-cRIO) system has an accuracy of 40 ppm at 25 □. In addition, the NI-cRIO onboard FPGA has its own 40 MHz clock with an accuracy of 100 ppm. By default, the NI-cRIO FPGA clock is not synchronized with the controller. For short measurements, this drift is usually negligible, but for continuous data acquisition systems running 24/7, the accumulated error has to be compensated. In this article we show how to correct time drift using protocols such as Network Time Protocol (NTP), Precision Time Protocol (PTP) and White Rabbit (WR) in combination with NI's FPGA TimeKeeper library.

BACKGROUND

Precise timing is essential to successfully operate a large-scale control system. At CERN, the accelerators are synchronized to nanosecond accuracy with the in-house developed, General Machine Timing (GMT) system. GMT electronic boards has been developed for several bus systems such as VME and cPCI, however at its deployment (2004), the cRIO platform was not mature enough and therefore not considered at the time. However, during the last 15 years, the cRIO platform has grown considerably. Its FPGA based backend, multitude of measurement modules and rapid development cycle has made it a mature platform which has become increasingly popular for test and monitoring systems at CERN. This has led us to start evaluating what is needed to fully integrate the platform in the CERN accelerator domain [1].

The successor of the GMT system, White Rabbit (WR), features improved timing accuracy and solves most of the GMT's shortcomings, and is actively being developed at CERN. An IEEE standardisation committee has been set up to incorporate it into a new IEEE-1588 standard. This and the availability of the WR design on the Open Hardware (OH) portal has made it possible to integrate the CERN timing on the cRIO platform at the hardware level [2].

CHALLENGE

A cRIO system consists of a Real Time (RT) controller with a standard Intel or ARM based processor and a user-programmable FPGA that is populated with one or more conditioned I/O modules. These modules provide direct sensor connectivity and specialty functions [3].

In cRIO systems, there are up to three timed components: the FPGA, the real-time processor, and hardware timed IO modules such as the GPS or WR module. Each component has a different clock that needs to be synchronized [3].

A typical RT controller such as the NI cRIO-9035 has a real-time drift of ±40 ppm at 25 °C and the FPGA has a drift of ±100 ppm, which will cause time drifts up to several seconds per day if not synchronized [4, 5].

Choosing synchronization technologies depends on synchronization accuracy requirements, distance between nodes, platform support, technology availability, and more.

In general, there are two types of synchronization: time or signal based

- Time-based: All components have a common time reference. Events, triggers, and clocks can be generated based on this time.
- Signal-based: Clocks and triggers are physically connected between systems.

In this paper we will focus on Time based synchronization and how White Rabbit can be used to accurately align clocks on both cRIO controllers and FPGA's.

WHITE RABBIT

White Rabbit is the name of a collaborative project aimed at developing a fully deterministic Ethernet-based network for general purpose data transfer and sub-nanosecond accuracy time synchronization. The hardware designs as well as the source code are publicly available.

White Rabbit provides sub-nanosecond synchronization accuracy, which formerly required dedicated hard-wired timing systems, with the flexibility and modularity of real-time Ethernet networks. A White Rabbit network may be used solely to provide timing and synchronization to a distributed electronic system, or be used to provide both timing and real-time data transfer [2].

RT CONTROLLER TIME

The cRIO RT controller clock can be synchronized similar to regular computers, using either a software or hardware-based timing system such as Simple Network Time Protocol (SNTP), Precision Time Protocol (PTP), Global Positioning System (GPS) and White Rabbit (WR).

(S)NTP

The Network Time Protocol (NTP) is a networking protocol for clock synchronization between computer systems over packet-switched, variable-latency data networks (Figure 1). In operation since before 1985, NTP is one of the oldest Internet protocols in current use and has an accuracy of ~1 ms [6].

The Simple Network Time Protocol (SNTP) is a less complex implementation of NTP, using the same protocol

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA028

POWER SUPPLY CONTROLLER FOR FUTURE ACCELERATOR FACILITIES AT BINP

P. B. Cheblakov*, S. E. Karnaev, A. V. Gerasev, D. V. Senkov
Budker Institute of Nuclear Physics, Novosibirsk, Russia

Abstract

A design of a new power supply controller was initiated in BINP for upgrade of existing accelerator facilities and for demands of future projects. Any accelerator facility includes a set of diverse power supplies which controllers have different specifications: number and precision of DAC/ADC channels, speed and algorithm of operation. Therefore, the main idea is to elaborate a controller, which consists of common digital part including an interface with a control system and specialized analog frontend that fits to power supplies requirements. The digital part provides easy integration to control system by means of some standard network protocol and performing some data processing and analysis. Ethernet is used for communication with controllers, MQTT is under consideration as a high-level transport protocol in some cases and EPICS IOC was tested to be embedded into controller. The initial prototype of controller is developed and deployed at VEPP-3 storage ring. The status of the work and future plans are presented in the paper.

INTRODUCTION

Modern circular and linear electron accelerators, which are part of accelerator complexes, are used for synchrotron radiation generation or for physical experiments. They operate with high-density beams of charged particles, which are moving in strong magnetic fields and are very spatially restricted. The smallest and even momentary fields' fluctuations from the prescribed values cause particles death. Therefore, experiments with such beams need positioning of their trajectory or equilibrium orbit with high precision up to parts of micron. These conditions lead to following requirements for magnetic system control: relative stability precision and control of main magnetic elements should be 0.001 % and better, continuous measurement with frequency about 1 kHz of power supply main parameters is necessary at circular accelerators.

To solve mentioned problems a control system based on non-trivial power supply controllers of magnetic system elements is needed. Such controllers should synchronously process and measure specified output parameters with required precision. In addition, this control system should provide transmission and on-line processing of data streams from hundreds of devices for detecting abnormalities in power supplies operation. This paper considers a perspective approach for power supplies development for such control system.

* P.B.Cheblakov@inp.nsk.su

THE HISTORY

Evolution of accelerator control systems at BINP has the following history.

The first automation systems (in 1970s) based on computers and on dedicated in-house developed digital-analog electronics connected to them. Computers and electronics were concentrated in certain places. At that time, electronics did not have built-in processors and was connected to computers through in-house designed serial communication links with cascade connection. Analog signals between digital-to-analog electronics and controlled equipment propagated over long cupper cables, causing problems from noises, attenuation, and signal dissipation.

The next step of the BINP's accelerator controls development [1] was related with the use of modular crate electronics systems and, first of all, CAMAC. Creation of intelligent crate controller in BINP, in fact – a home-developed computer, as well as development of wide range of electronic modules for CAMAC satisfied all the requirements of control systems on accelerator complexes both in BINP and in the other accelerator centres of the USSR for many years [2]. While providing a relatively high data rate to the computer, this approach to control system design had the same disadvantage: electronic devices were located in crates in certain places; analog signals from the equipment were transmitted through long cable links.

Further (in 1990-2000s), both structural and functional evolution of control systems in BINP was based on the use of processors in analog-to-digital modules, as well as on using of serial communication links, and, first of all, CAN (Controller Area Network) [3]. A typical scheme of controllers of that period is shown on Fig. 1.

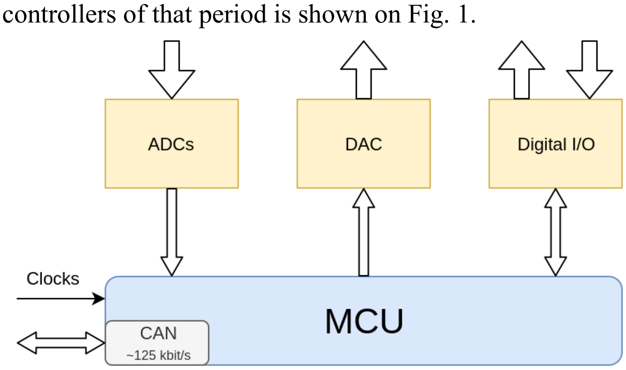


Figure 1: Power supply controller architecture based on MCU (Microcontroller Unit) and CAN.

This approach provided a possibility to develop distributed systems with control electronics located in close proximity or inside controlled devices [4] in order to reduce the lengths of analog signal wires. The use of

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA033

CONSTRUCTION AND IMPLEMENTATION OF CONTROL AND DAQ SYSTEM OF MICRO CRYSTALLOGRAPHY (MX) BEAMLINE VIA SERVER VIRTUALIZATION

H. J. Choi, H. S. Kim, S. W. Kim, W. W. Lee
Pohang Accelerator Laboratory, Pohang, South Korea

Abstract

The project aimed to implement a beamline control and data collection system through a server virtualization system, and was applied to the 5C beamline of the 3rd generation beamline of Pohang Accelerator Laboratory (PAL). The 5C beamline is currently under construction for the FBDD beamline with the goal of building a fully automated beamline. Therefore, the project was started to operate stably and efficiently various systems to be applied to the beamline.

The control system was implemented using EPICS software tools and MxDC/MxLive software for data acquisition and storage. The control and data collection system of this beamline is integrated using XCP-ng [1] (XenServer Based), and it is in operation. With the integrated server virtualization system, network organization / simplification and data send/receive between systems are more stabilized. The overall size of the system has been significantly reduced, making maintenance easier.

INTRODUCTION

The Beamline is operated by various single systems connected in various ways. The complexity of a system is the number of components, the various relationships between them, and the change in components and relationships. The opposite of complexity is simplicity. If a simple system consists of the components and relationships that are necessary for its purpose, a complex system is highly diverse, interdependent, and uncertain, making it difficult to predict any situation, process, or outcome. As a result, complexity often leads to unintended consequences or difficulties in resolving the situation.

As the system of the Beamline becomes more complicated, the inefficiency of the work due to the trouble of maintenance and management due to the failure or the transition to the new environment, and the cost burden for the maintenance of the system are generated. Server virtualization considered that it is appropriate to apply the virtualization system to the beamline in that it can reduce the management burden and increase the stability.

Virtual is described in a dictionary meaning as 'a false phenomenon that appears to be subjectively real but does not exist objectively'. Virtualization seems to exist from the user's point of view, but it can be interpreted as a phenomenon that exists from the point of view of the system as if it is running inside the system. As we will see, virtualization of a system, which will be explained in the future, refers to a way to make another system run inside the system. Commonly known virtualization methods include VirtualBox or VMware. Inadequate Hosted virtualization is another way of operating a guest OS on a machine with host OS, such as MS Windows and Linux, so the guest OS is affected by the performance of the host OS, resulting in significant performance degradation. In order to compensate for such performance degradation, the beamline integrated system was implemented and implemented through hypervisor virtualization (Para virtualization), a kind of server virtualization.

BACKGROUND

Computing speed and performance/environment are steadily improving, and the amount of resources and technologies available to one server equipment is rapidly expanding and evolving. This means that a system that used to operate in multiple computing environments can be supported by one server device for availability and scalability. However, even if the amount and technology of resources provided by one server equipment is expanded and developed, it is useless unless it is effectively utilized.

Server virtualization is a solution that is evaluated to be able to run various servers in one physical system to support the amount of resource and data operation of each server most effectively. Most existing beamline control and data acquisition systems are built using a variety of PC, workstation and server equipment. This means that it is difficult to manage because it is composed of complicated communication networks and different systems, and it is not easy to reconstruct the system due to equipment failure, install / upgrade the operating system, and trouble. To compensate for these shortcomings, the advantages of server virtualization have been applied to beamlines to operate systems operating in different environments.

OBJECTIVES

Server virtualization integrates several operating systems for fully automated beamline implementation into a single server machine. By systematically separating internal (private of virtualization system) or external (publicization of virtualization system) network, data send/receive is operated smoothly and stably. In order to ensure the stability of communication between virtual machines and virtual machines, an internal network is constructed and operated. Build a virtual machine that acts as a gateway, and let other virtual machines communicate internally and externally. External to internal connections are made through the virtual machine for the gateway. Build a single virtual machine operating environment, and system cloning ensures that other virtual machines maintain the same oper-

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA034

SOFTWARE TOOLS FOR HARDWARE ELLIPTICAL CAVITY SIMULATOR MANAGEMENT AND CONFIGURATION

K. Klys, W. Cichalewski *, TUL, DMCS, Łódź, Poland

Abstract

The European Spallation Source (ESS) is currently in the middle of its construction phase. This facility linear accelerator consists of different sections. Superconducting part of this linac will be equipped with spokes and elliptical cavities (like M-Beta and H-Beta types). Various ESS linac components will be delivered by different in-kind partners from Europe. In order to provide a reliable development and evaluation platform hardware-based electronic cavity simulator have been built. This solution is especially useful for Low Level Radio Frequency (LLRF) systems development and integration in case of limited access to real superconducting structures. This contribution presents software tools developed for efficient cavity simulator parameters configuration and management. Solutions based on Python and EPICS framework are presented. Tool adaptation to ESS proposed E3 framework and experience from cavity simulator operation are also discussed.

INTRODUCTION

The ESS will be the most powerful neutron source on earth. In this facility, the spallation will be used to produce free neutrons. The tungsten target will be hit with protons with kinetic energy of 2.5 GeV. This leads to generation of neutrons' pulses. The superconducting part, which accelerates particles, will contain 120 cavities (84 H-Beta, 36 M-Beta) and 120 klystrons, operating at 704.42 MHz [1]. For each cavity the electrical field with appropriate gradient and frequency must be delivered. Its parameters (amplitude and phase) are controlled with LLRF control system [2].

The cavity simulator is a device which emulates the behaviour of superconducting High and Medium Beta cavities and klystrons basing on signals received from LLRF control system. It was created to minimize the risk of tests and measurements conducted on real facilities using LLRF system under development. In order to provide reliable results of simulations, the cavity simulator reflects phenomena like Lorentz force detuning, piezo compensation, beam loading π-modes, mechanical modes, amplifier non-linearity and others .

The device is composed of high performance Field-Programmable Gate Array (FPGA) evaluation board with data converters, Digital to Analog/Analog to Digital Converters (DAC/ADC) modules and specially designed Radio Frequency (RF) front-end [3]. The simulator can be controlled remotely with commands sent via Ethernet network. Those messages are realised using Standard Commands for Programmable Instruments (SCPI) syntax.

* wcichal@dmcs.pl

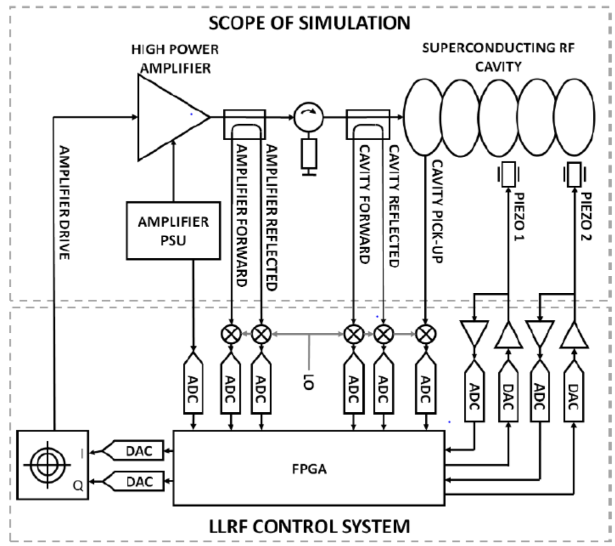


Figure 1: The cavity simulator area of simulation.

Figure 1 represents the scope of the cavity simulator's simulation and devices whose functioning is considered.

HARDWARE CAVITY SIMULATOR STRUCTURE AND COMMUNICATION

To simulate all mentioned effects and to cover the whole scope of cavities simulation, RF circuit has been designed (see Figure 2). The simulator's hardware is responsible for generating RF and base-band signals. The clock and refer-ence signals are there produced as well. The structure of hardware can be divided into several parts [4]:

- Data Conversion module,
- Down-conversion module,
- Piezo module,
- Reference Generation module,
- Local Oscillator (LO) Generation module,
- Power Supply module.

First module is used to convert different data types. It digitizes analog RF signals basing on down-conversion scheme. The generation of RF outputs is realised with vector modulator circuits. Down-conversion area decreases the frequency range of RF inputs to suitable level for ADCs. It makes use of active mixer circuit. Piezo part protects electronics against high voltage from piezo driver, lowering it 100 times. The module also emulates piezo element, functioning in 2 K

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA035

FIRMWARE LAYER IMPLEMENTATION OF THE nBLM AND icBLM SYSTEMS FOR ESS PROJECT

W. Jalmuzna, G. Jabłoński, R. Kiełbik, W. Cichalewski *, TUL, DMCS, Łódź, Poland
I. Dolenc Kittelmann, F. S. Alves, H. Carling, S. Farina, K. Rosengren, T. J. Shea
ESS ERIC, Lund, Sweden

Abstract

Both ionization chamber Beam Loss Monitor (icBLM) and neutron Beam Loss Monitor (nBLM) systems are fundamental components of European Spallation Source (ESS) accelerator safety systems. Main responsibility of this system is instantaneous and reliable detection of accelerated proton beam loss that exceeds predefined safety threshold. Nowadays DMCS (as an in-kind partner to ESS) is responsible for beam loss detection algorithm implementation, evaluation and deployment in firmware. As a hardware platform for mentioned systems MTCA.4 based form factor electronic components have been chosen (delivered by IOXOS). This contribution focuses on both cases (nBLM and icBLM) firmware realisation presentation. Proposed and developed firmware structure and functional blocks that fulfils specified by ESS requirements are described. Additionally, some aspects of the system FPGA circuit resource usage and achieved performance is being discussed.

INTRODUCTION

The ESS is a material science facility, which is currently being built in Lund, Sweden and will provide neutron beams for neutron-based research [1]. The neutron production will be based on bombardment of a tungsten target with a proton beam of 5 MW average power. A linear accelerator (linac) [2] will accelerate protons up to 2 GeV and transport them towards the target through a sequence of a normal conducting (NC) and superconducting (SC) accelerating structures (Fig. 1).

As in case of all future high-power accelerators, ESS linac operation will be limited by beam losses if machine activation is to be kept low enough for hands-on maintenance. Moreover, loss of even a small fraction of intense ESS beam can result in a significant increase of irradiation levels, ultimately leading to damage to the linac equipment. BLM systems are designed to detect showers of secondary particles produced by lost beam particles interacting with the accelerator equipment. By providing information about beam loss levels, BLM systems play an important role in machine fine tuning as well as machine protection from beam-induced damage by detecting unacceptably high beam losses and promptly inhibiting beam production.

The ESS BLM consist of two types of systems, differing in detector technology [3]. The icBLM system is based on 266 ionisation chambers [4–6] as detectors, located almost exclusively throughout the SC parts of the linac. These detectors are simple and well established for beam loss monitoring purposes. Conversely, the nBLM system consists of 82 neutron detectors, specially designed to primarily cover the lower energy part of the ESS linac and thus complement the icBLM system.

Both systems are equipped with the read-out hardware supporting the real-time FPGA-based data processing.

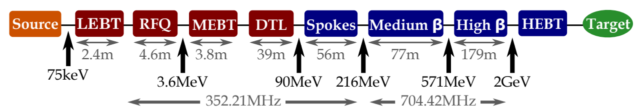


Figure 1: The ESS linac layout. Red colour represents the NC and blue the SC parts of the linac. [2]

HARDWARE AND FIRMWARE STRUCTURE

The platform standard selected for BLM systems is uTCA. This is quite complex standard with rich supporting infrastructure, which can accommodate wide range of extension modules (called AMCs) to implement all needed function. The main platform for BLM implementation is IFC1410 AMC module, which is 2 HPC FMC carrier. It provides ultimate computation power in terms of FPGA resources and is equipped by additional PPC processor to implement control system integration.

The application specific (icBLM and nBLM) data acquisition is performed by additional FMC modules (250 MHz AD3111 for nBLM and FMC-Pico-1M4 1 MHz for icBLM).

The general structure of the FPGA implementation is enforced by TOSCA framework, which provides access channels to PCIe and DDR3 resources. It also defines interfaces between FMC modules and user logic. Its simplified diagram is presented in Fig. 2. The parts implementing BLM algorithms are marked with green.

FIRMWARE IMPLEMENTATION

The firmware implementations of nBLM and icBLM share common components, related mainly to mass data transfer, that are not provided by the TOSCA framework. Both systems need to provide large quantities of various types of data to the CPU. These data are buffered in two DDR3 memory blocks. The total bandwidth (read + write) of each of these memory blocks is 2 GB/s, when the memory controller is operating at 275 MHz. The data are logically

* wcichal@dmcs.pl

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA037

STATUS OF THE CLARA CONTROL SYSTEM

W. Smith†, R. F. Clarke, M. D. Hancock, P. W. Heath, J. T. G. Wilson, S. Kinder, P. H. Owens, A. Oates, G. Cox, B. G. Martlew, N. J. Knowles, STFC Daresbury Laboratory, Warrington, UK

Abstract

CLARA (Compact Linear Accelerator for Research and Applications) is a test facility for Free Electron Laser (FEL) research and other applications at STFC's Daresbury Laboratory [1]. The control system for CLARA is a distributed control system based upon the EPICS [2] software framework. The control system builds on experience gained from previous EPICS based facilities at Daresbury including ALICE (formerly ERLP) [3] and VELA [4].

This paper presents the current status of the CLARA control system, experiences during beam exploitation and developments and future plans for the next phases of the facility.

INTRODUCTION

The build of the CLARA facility is currently staged to run across 3 main phases. Phase 1, the CLARA Front-End, Phase 2 the main accelerator and Phase 3 the FEL.

The installation and commissioning of Phase 1 comprising of the photo-injector, RF gun and the first linac was completed during 2018. Subsequent machine development and beam exploitation took place during late 2018/early 2019. Phase 2, the main accelerator is now being installed along with design and development of a Full Energy Beam Exploitation (FEBE) line for 250MeV beam experiments. Phase 3 is currently on hold awaiting funding.

The control system for CLARA is an evolution of the control system developed for VELA. It retains the use of the EPICS software toolkit and Input/ Output controllers (IOCs) running the Linux operating system. In fact, with the deployment of CLARA Phase 1, and due to VELA and CLARA sharing the same common infrastructure, the control system of VELA has effectively been absorbed into CLARA and can now be considered as a single system.

This paper gives an overview of the current status of the major sub-systems of the control system. Experience gained during machine development and exploitation of Phase 1 along with future plans for satisfying the requirements of Phase 2 is discussed.

MOTION CONTROL

Operational experience of the motion control system during Phase 1 has proven the Beckhoff EtherCAT based system to be reliable and robust. Additional experience has also been gained with the Beckhoff closed loop stepper motor control; this has resulted in smoother operation of axes that had previously been problematic.

For Phase 2 the motion control system which comprises Beckhoff TwinCAT 3, CX5020 embedded PCs and EtherCAT modular I/O terminals, has been extended to control the 5-axis Variable Bunch Compressor.[1] This includes co-ordinated movement of collimating jaws at narrow gap, and logic for collision avoidance during homing.

Additionally, for Phases 2 and 3 of CLARA the Beckhoff system is being developed to control an 8-axis de-chirper module. This will use absolute-encoder feedback for precise positioning of quartz plates, both parallel to, and at known angles to the electron beam.

RF CONTROL

The CLARA photo-injector gun shares the RF (Radio Frequency) infrastructure from the VELA photo-injector with an RF switch controllable through EPICS. The switch transfers power between the existing VELA RF gun (10Hz) and the new high repetition rate RF gun of CLARA (400Hz).

The low-level RF (LLRF) is provided by Libera, a system from Instrumentation Technologies [5]. These systems implement an EPICS IOC on board. One Libera system is shared between the 2 RF guns of CLARA and VELA and a second is dedicated to linac 1. Our experience with these systems during operation of Phase 1 has been positive, with Instrumentation Technologies supporting us with development of features to enable the pulse-to-pulse acquisition and processing of RF data through the control system at 100Hz.

The ScandiNova RF modulator for the VELA and CLARA guns is integrated into the control system via the manufacturer's proprietary ASCII protocol over TCP/IP using StreamDevice and Asyn EPICS support modules.

High-power RF for linac 1 is provided by a Diversified Technologies modulator which provides a Modbus TCP interface via its internal Beckhoff EtherCAT control system. It is integrated into the control system via this interface using Asyn & Modbus EPICS support modules. Further higher powered modulators for the 3 linacs of Phase 2 will be provided by Diversified Technologies and integrated with the control system via the same method.

Unmanned Conditioning

To facilitate faster RF conditioning a system was developed, in close cooperation with RF scientists and the accelerator physics group, to allow safe unmanned operation of the CLARA RF systems [6]. This was implemented using the EPICS Sequencer to monitor all the relevant safety and machine interlocks for failures, log the failures when they occurred, and then to reset the RF systems to bring them back online. This was improved by checking the error log to ensure that certain failures were only allowed to happen a certain number of times in a given time period before the

† william.smith@stfc.ac.uk

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA038

EXTENDING TANGO CONTROL SYSTEM WITH KEPLER WORKFLOW, PRESENTED ON AN X-RAY CRYSTALLOGRAPHIC APPLICATION

S. Brockhauser[1], K. Csankó†, V. Bugris, Zs. Filákovics,
Biological Research Center, Szeged, Hungary
P. Ács, V. Hanyecz, ELI-ALPS, Szeged, Hungary
[1]also at University of Szeged, 6720 Szeged, Hungary, and European XFEL, Schenefeld, Germany

Abstract

Nowadays there is a growing need for user friendly workflow editors in all fields of scientific research. A special interest group is present at big physics research facilities where instrumentation is mostly controlled by a robust and reliable low level control software solution. Different types of specific experiments using predetermined automated protocols and on-line data processing with real-time feedback require a more flexible and abstract high level control system[1]. Beside flexibility and dynamism, easy usability is also required for researchers collaborating from several different fields. Tentatively, to test the ease and flexible usability, the Kepler workflow-engine was integrated with TANGO[2]. It enables researchers to automate and document experiment protocols without any programming skill. The X-ray crystallography laboratory at the Biological Research Center of Hungarian Academy of Science (BRC) has implemented an example crystallographic workflow to test the integrated system. This development was performed in cooperation with ELI-ALPS.

INTRODUCTION

Control systems are vital elements of the highly complex experiments performed on large scale experimental physics facilities like synchrotrons, neutron sources or laser-facilities. During an experiment, hundreds and thousands of widely displaced devices need to be controlled simultaneously which can't be carried out only with one control device like a single computer, but it can be handled by several computers in a network. Distributed control systems uses network nodes to connect and interface several individual systems that are responsible only for a segment of the entire experimental setup. Significant development of high-level distributed control systems happened in recent decades following the evolution of available IT solutions. The development of increasingly demanding experimental methods also contributes greatly to this and have challenged the control, acquisition and processing solutions by exponentially increasing amount of data produced (by faster and bigger detectors). This created a claim of automation [3, 4]. On the basis of these developments various control systems were born in the past, such as TANGO [5], EPICS [6], or TINE [7], just to mention a few. Despite these systems are mostly under the editorship of experiment facilities, these developments have begun to flourish also in smaller laboratories in consequence of the achieved wide-range of supported instrumentation, easy configurability, and user-friendliness. One of the most popular systems is TANGO used in many research facilities like ESRF, ALBA, Soleil, DESY, ELI-ALPS [2] and others.

TANGO is a CORBA [8] based, object oriented distributed control system being developed in C++, java and Python by a collaboration headed by ESRF in France. The fundamental unit of this control system is the "device", which is a remote object (usually a device server directly connected to the controlled hardware) registered in a database. Each device has commands, attributes and states to control its instrument and also properties for configuration. The TANGO framework facilitates the implementation of control systems at each level (see Fig. 1). Both low and high-level control environment can be developed by TANGO. Although it is flexible and semi-automated, it still requires some programming skills. It has excellent ability to direct a wide variety of devices and also offers a graphical interface; but it is not offering a dynamic interface for combining complex experiments into workflows. To overcome this problem an easy to use tool needed to be integrated with TANGO which provides a common graphical interface for instrumentation and data processing services to support the requirements of the experiments and to provide a platform for the scientists where they can easily alter the experiments with minimum programming skills and without the need to learn all details of the instrumentation.

Recently, a new initiative has appeared which enables the integration of TANGO to various process management tools [1, 9, 10, 11, 12]. This integration has several advantages because the benefits of both TANGO and the management software can be utilized. On the TANGO side which is a well-supported open source control system, there are numerous former works [11] implementing reliable low level systems and abstracting the control of various hardware facilities. With an integration keeping the modularity of TANGO, the supported hardware list can be fully maintained and experiment protocols can be adopted for different hardware sets without the need of real changes.

A generic architecture of a TANGO-based control system is presented on Fig. 1. The low level system hides the specific and more detailed environment of the facility (like type of the applied devices) while the high level system provides a standard, easy to understand tool for designing and performing experiments.

† csanko.krisztian@brc.hu.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA040

IRFU EPICS ENVIRONMENT

J.F. Denis, A. Gaget, F. Gohier, F. Gougnaud, T.J. Joannem, Y. Lussignol
IRFU, CEA, Université Paris-Saclay, Gif-sur-Yvette, France

Abstract

The 3 years collaboration with ESS* at Lund (Sweden) has given us the opportunity to use new COTS hardware and new tools. Based on that experience, we have developed the IEE (IRFU** EPICS Environment) by retaining relevant and scalable ESS solutions. This platform centralized several functionalities, fully installed by scripting, on a server that is running on a virtual machine. The functionalities are an EPICS environment and the root file system with the kernel for each embedded systems. In order to provide homogeneous EPICS modules between all collaborators, a template was designed and used as containers for new developments. Furthermore, a development and a production workflow is also proposed and strongly recommended. Due to the current responsibility of CEA IRFU to provide an EPICS platform for SARAF** at Tel Aviv (Israel), IEE was chosen as the standard platform for the whole accelerator. This paper will present the new standard IRFU EPICS Environment based on MTCA and virtual machines.

INTRODUCTION

Based on our experiences of the standard platform conception for SPIRAL2 (France) and IFMIF[1] (Japan) project, and contributing to the ESS[2] (Sweden), CEA decided to combine state of the art technologies as ESS and our expertise acquired on IFMIF and SPIRAL2[3] to create the new standard platform "IEE": Irfu EPICS Environment. The main idea of this platform is to propose a generic methodology for future projects in keeping an EPICS environment centralized on a server (IEE server) shared it with all the clients fully parametrized and automatically installed. A light and standalone version of the IEE server can also be provided for development on standalone PCs. This platform is used for the SARAF[3] (Israel) project.

HARDWARE CONTEXT

Hardware

The current hardware standard for our EPICS projects are the following (see Table 1.), based on mTCA.The standardized crates are the NATIVE-R2/NATIVE-R5 from NAT with the MCH-PHYS80 and the Intel CPU COMex3.

Fast (Up to 250 MS/s) and semi-fast (Up to 5 KS/s) acquisitions for beam diagnostics and RF signals acquisition are assured respectively by the FMC boards IOxOS ADC_3110/3111 and ADC_3117. These boards are plugged on a IOxOS IFC1410 motherboard which integrates an internal FPGA with a customizable area for user specific functionalities.

The timing system is based on MRF boards, the mTCA-EVM300 for the Event master and the mTCA-EVM300U for the Event receiver with delay compensation.

For remote I/Os, our choice remained the cheap and convenient Beckhoff EtherCAT modules with communication based on Modbus/Tcp.

This hardware has been integrated in IEE, but more hardware can be easily added.

Table 1: Current Hardware Used With IEE

Designation	Reference	Tender
mTCA Crate	NATIVE-R2/ NATIVE-R5	NAT
MCH	MCH-PHYS80	NAT
CPU Intel	RTM COMex3	NAT
CPU carried board	IFC1410	IOxOS
Fast Acquisition (8ch., 250 MS/s)	ADC311	IOxOS
Slow Acquisition (20 ch., up to 5 MS/s)	ADC3117	IOxOS
Timing System Master	mTCA-EVG300	MRF
Timing System Receiver	mTCA-EVR300U	MRF

IEE SOFTWARE DESCRIPTION

EPICS

The "EPICS Software Distribution" is a selection of "Base", "IOC support modules" and "extensions" exists on official EPICS websites. The currently used EPICS Base is R3.15.4. The IOC support modules are categorized as either Software Support or Hardware Support. The extensions are host tools and Channel Access clients such as Control System Studio (CSS), Archive Appliance, CA libraries (Java, Python…).

The development environment fully relies on the standard "EPICS Build Facility". EPICS software can be divided into multiple <top> areas. An example of a <top> area is the EPICS Base itself. Each <top> may be maintained separately. A <top> directory structure essentially contains the build configuration files in a configure folder and a Makefile. The GUIs are made using CSS BOY.

The Irfu EPICS Environment provides a standard development model that all the developers involved in our control system software team have to follow. These

*ESS, https://europeanspallationsource.se/
**IRFU, https://irfu.cea.fr/en/
***SARAF, http://soreq.gov.il/mmg/eng/Pages/SARAF-Facility.aspx

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA041

THE CMS ECAL CONTROL AND SAFETY SYSTEMS UPGRADES DURING THE CERN LHC LONG SHUTDOWN 2*

D. Di Calafiori†, G. Dissertori, R. Jiménez Estupiñán, S. Zelepoukine[1], W. Lustermann,
ETH Zurich, Institute for Particle Physics and Astrophysics, Zürich, Switzerland
A. Tsirou, CERN, Geneva, Switzerland
P. G. Verdini, INFN Sezione di Pisa,
Università di Pisa, Scuola Normale Superiore di Pisa, Pisa, Italy
[1]also at University of Wisconsin-Madison, USA

Abstract

The electromagnetic calorimeter (ECAL) is one of the sub-detectors of the Compact Muon Solenoid (CMS) experiment, a general-purpose particle detector at the CERN Large Hadron Collider (LHC). The CMS ECAL detector control system (DCS) and the CMS ECAL safety system (ESS) have supported the detector operations and ensured the detector's integrity since the CMS commissioning phase, more than 10 years ago. Over this long period, several changes to both systems were necessary to keep them in-line with current hardware technologies and the evolution of software platforms. The acquired experience of long-term running of both systems led to the need of major modifications to the original design and implementation methods. Such interventions to either systems, which require mid- to long-term validation, result in a considerable amount of downtime and therefore can only be performed during long shutdowns of the experiment. This paper discusses the software and hardware upgrades to be carried out during the LHC long shutdown 2 (LS2), with emphasis on the evaluation of design choices concerning custom and standard industrial hardware.

INTRODUCTION

The electromagnetic calorimeter (ECAL) is one of the sub-detectors of the Compact Muon Solenoid (CMS) [1] experiment, a general-purpose particle detector at the CERN Large Hadron Collider (LHC) [2]. The CMS ECAL is composed by three partitions: barrel (EB), endcaps (EE) and preshower (ES). The sub-detector control and safety systems were designed according to the individual requirements of each partition.

The EB and EE partitions have similar hardware and software requirements for the control and monitoring of the powering systems, monitoring of the on- and off-detector environment, and protective/safety actions. Therefore, an integrated Detector Control System (DCS) [3-4] and the so-called CMS ECAL EB/EE Safety System (ESS) [3-4] were implemented for these partitions.

The ES DCS was implemented separately and integrated with the EB/EE only afterwards at the supervisory layer, while the ES safety system is based on the CMS tracker safety system [5] architecture and has no integration with the ESS. The CMS ECAL ES safety system is not discussed in this paper.

To keep the control and safety systems in-line with the current hardware technologies, the evolution of software platforms and new sub-detector operational requirements, several modifications to their original design have been applied over the past 10 years. Furthermore, the experience acquired through the sub-detector's daily operation also allowed the identification of improvements and extensions to these systems.

Prior to any modification to the control and safety systems, a full analysis of their impact to the sub-detector operation is performed. Minor modifications, such as improvements to user interfaces, are transparent and can be applied at any time. Modifications requiring short-term validation, that might disturb or interrupt the sub-detector operation, are applied during the LHC technical stops (TS), which normally last for one or two weeks. Minor modifications to the core of either systems or requiring mid-term validation are schedule for the LHC year-end technical stops (YETS), which normally last for a few months. Major modifications requiring long-term validation can only be carried out during the LHC long shutdown (LS) periods, which last for a few years.

This paper provides short descriptions of the CMS ECAL DCS and ESS with references to previous publications where further details can be found. The main focus of this paper will be on the major upgrades which are being carried out during the LHC long shutdown 2 (LS2).

THE CMS ECAL CONTROL SYSTEM

Software

For the high-level process supervisory management, the supervisory control and data acquisition (SCADA) architecture was implemented with the Siemens WinCC Open Architecture (WinCC OA) [6] control system toolkit. To add specific functionalities, CERN Joint Controls Project (JCOP) [7] and CMS DCS [8] frameworks are extensively used. The control software runs in two redundant sets of three servers, partitioned in individual components per sub-system or service. The integration of these components into a single application is realised with an additional component called the CMS ECAL Supervisor, which uses a finite state machine (FSM) mechanism to summarize the sub-systems status per sub-detector partition and to issue/propagate commands to the hardware. In addition, the

† diogo.di.calafiori@cern.ch
* on behalf of the CMS Collaboration

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA042

COMMISSIONING OF THE 352 MHz TRANSVERSE FEEDBACK SYSTEM AT THE ADVANCED PHOTON SOURCE*

N. DiMonte#, C.-Y. Yao, Argonne National Laboratory, Lemont, U.S.A.

Abstract

With the success and reliability of the transverse feedback (TFB) system installed at the Advanced Photon Source (APS), an upgraded version to this system was commissioned in 2019. The previous system operated at a third of the storage-ring bunch capacity, or 432 of the available 1296 bunches. This upgrade samples all 1296 bunches which allowed corrections to be made on any selected bunch in a single storage-ring turn. To facilitate this upgrade the development of a new analog I/O board capable of 352 MHz operation was necessary. This paper discusses some of the challenges associated in processing one bunch out of 1296 bunches and how flexible the system can be in processing all 1296 bunches. We will also report on the performance of this system.

INTRODUCTION

The Advanced Photon Source (APS) experiences beam instabilities in both the transverse and longitudinal planes. In the past, the P0 Feedback (Transverse Feedback) system, in its initial version, has corrected transverse instabilities in a bunch pattern that has up to 24 bunches. This is accomplished by using a pick-up stripline, drive stripline, four drive amplifiers, a 3-tap comb filter for front-end signal conditioning, and an Altera PCIe Stratix II GX FPGA-based development board coupled with a Coldfire CPU. The Coldfire CPU uses EPICS [1] with RTEMS [2] for all the remote monitoring and control. However, the P0 Feedback was unable to perform during a 324-bunch pattern. This paper discusses the addition of the 352MHz TFB system, its performance, added features and modifications, and possible future plans.

SYSTEM CONFIGURATION

The current system consists of a pick-up stripline, a front-end signal-processing unit, an Altera PCIe Stratix II GX FPGA-based development board (P0 Feedback), a Terasic TR4 Stratix IV commercial board (352MHz TFB), drive amplifiers, and a driver stripline. This system has been described in detail [3, 4]. Figure 1 shows a block diagram of the current system without the remote DAC hardware. At the core of the P0 Feedback system contains an FPGA processor that utilizes 864 32-tap FIR filters running at 117.3 MHz. The new Transverse Feedback system utilizes 2592 32-tap FIR filters running at 352MHz. The algorithm used in the filter is based on the least square fitting method to determine filter coefficients [5].

*Work supported by U.S. Department of Energy, Office of Science, Office of Basic Energy Sciences, under Contract No. DE-AC02-06CH11357.
#Email: npd@aps.anl.gov

The diagrams do not show both TFB systems in parallel with each other, but figure 2 shows how the new TFB has been integrated into figure 1 using splitter/combiner for both the inputs and outputs.

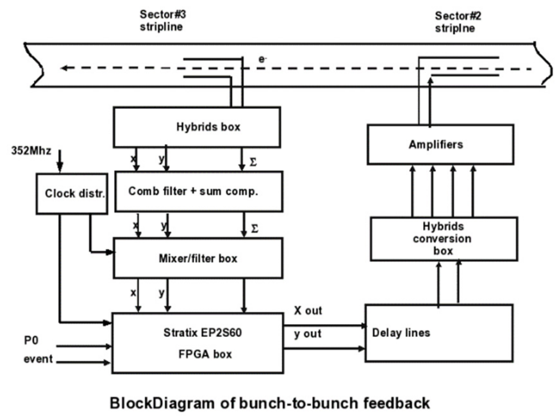


Figure 1: Block diagram of the feedback system.

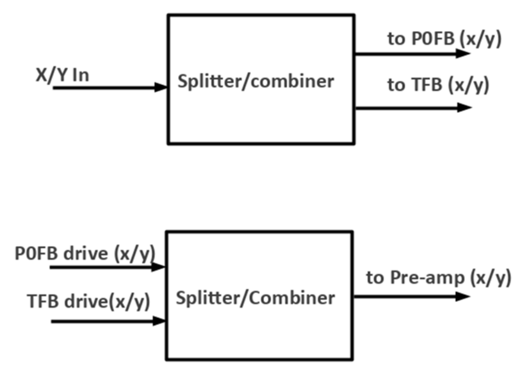


Figure 2: Block diagram of the integration of the new TFB system.

The pickup and drive striplines are located in different locations of the storage ring, which is a major issue for the X-channel since the distance between pickup and drive is seven sectors, or about 188 m, apart. A remote DAC linked to the main transverse feedback system via high-speed fiber optic cable utilizing a real-time data transfer protocol. Figure 3 shows the addition of the transceiver in the main transverse feedback chassis used to connect the remote DAC chassis.

PREVIOUS SYSTEM AND LIMITATIONS

The old PO Feedback FPGA system described above has been very successful for the Advanced Photon Source as it has been detailed in a previous paper [6]. This system is limited to 324 or 432 buckets depending on the operating frequency of 88 or 117.3 MHz when configured for a

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA044

UPGRADE OF THE BUNCH LENGTH AND BUNCH CHARGE CONTROL SYSTEMS FOR THE NEW SLAC FREE ELECTRON LASER*

M. P. Donadio†, A. S. Fisher, L. Sapozhnikov, SLAC, Menlo Park, USA

Abstract

In 2019 SLAC is building a new linear accelerator based on superconducting niobium cavities. The first one, now called the copper linac, could generate 120 electron bunches per second. The new one, called the superconducting linac, will generate 1 million per second, bringing some challenges to many devices along with the accelerator. Most sensors and actuators in the copper linac are interfaced through a VME-based Platform with its control running in software, with RTEMS as OS. This is feasible for 120 Hz, but not for 1 MHz. The new control hardware is ATCA-based Platform, that has carrier boards with FPGA connected to servers running Embedded real-time Linux OS, forming the High-Performance System (HPS). Instead of having all the new architecture installed at the accelerator and tested on the go, SLAC used the strategy of testing the systems in the copper linac, to have them ready to use in the superconducting linac in what was called the Mission Readiness Program. The Bunch Length System and the Bunch Charge System are examples of devices of this program. Both systems were tested in the copper linac at 120 Hz, with excellent results. The next step is to test them at the superconducting linac, at 1 MHz.

INTRODUCTION

LCLS has two bunch length monitors (BLEN) based on pyroelectric detectors [1] and 13 bunch charge monitors based on Toroids [2], the two types of devices that are in the scope of this upgrade. Beyond these devices, LCLS uses gap diode to measure the bunch length [3] and, to measure the bunch charge, Faraday cups [2] and BPMs [4].

Pyroelectric Detector Bunch Length Structure

LCLS has two bunch compressors (BC). Edge radiation (ER) emitted at the exiting edge of the last bending magnet of each bunch compressor is extracted from the beamline by a mirror at 45 degrees to the beam (see Fig. 1). A hole in the center of the mirror allows the electrons to pass but also emits diffraction radiation (DR) (see Fig. 2). The ER is reflected, split, filtered, and focused by an off-axis parabolic mirror (OAP) onto two pyroelectric detectors [1]. The DR, which is emitted much closer to the OAP, does not image onto the pyroelectric detectors. This light is spread out and produces a little signal.

By using specific optical filters it is possible to measure the difference in the intensity of the signals at the pyroelectric detector output for different lengths of the bunch. Comparing signal intensity for different bunches it is possible to know that one bunch is longer or shorter relative to the other. Calculating the absolute length is only possible when the system is calibrated in advance using invasive methods, like a transverse deflecting cavity. [1]

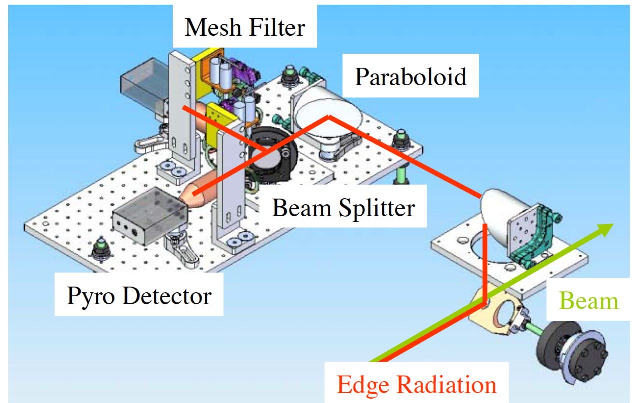


Figure 1: Optical structure of the bunch length system showing how the ER is reflected, split, and filtered before getting to the pyroelectric detectors. Reproduced from [1] with permission from the authors.

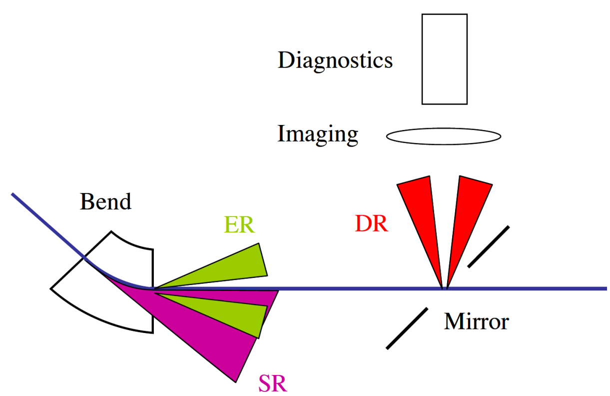


Figure 2: ER from bending magnet generating DR through a hole mirror. Reproduced from [1] with permission from the authors.

Toroid Bunch Charge Monitor (BCM) Structure

A toroid is placed around the accelerator beamline so that the electron beam passes through its hole, inducing a current in the turns. This current is conditioned and digitized. By measuring the peak of the conditioned signal, the system can calculate the bunch charge. For an explanation of the theory, please refer to [5].

The Mission Readiness Program

The goal of the SLAC Mission Readiness Program is to implement improvements in the present to guarantee that the laboratory will comply with its mission in the future and

* Work supported by US DOE contract DE-AC02-76SF00515
† marcio@slac.stanford.edu

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA045

DATA ACQUISITION STRATEGY AND DEVELOPMENTS AT MAX IV

M. Eguiraun*, A. Amjad, P. J. Bell, A. Dupre, D. A. Erb, N. A. Håkansson,
V. H. Hardion, A. Milan-Otero, J. F. J. Murari, E. Rosendahl
MAX IV Laboratory, Lund, Sweden

Abstract

The experimental capabilities at the MAX IV synchrotron consists of 17 beamlines at full capacity. Each beamline puts different requirements on the control system in terms of data acquisition, high performance, data volume, pre-processing needs, and fast experiment feedback and online visualization. Therefore, high demands are put on the data management systems, and the reliability and performance of these systems has a big impact on the overall success of the facility. At MAX IV we have started the DataStaMP (Data Storage and Management Project) with the aim of providing a unified and reliable solution for all data sources in our facility. This work presents the control system aspects of the project. It is initially aimed at providing data management solution for a selected number of detectors and beamlines. It is developed in a modular and scalable architecture and combines several programming languages and frameworks. All the software runs in a dedicated cluster and communicates with the experimental stations through high performance networks, using gRPC to talk to the control system and ZMQ for retrieving the data stream.

DATASTAMP PROJECT

MAX IV has been funded to improve the data management services offered to users, notably storage, without which it would be much more difficult to provide such services. In order to create these services, a number of areas need be given more resources than they are today – where currently there is only an ambition and no available time or money. The overall mission is to improve the value of the data generated at MAX IV in terms of the benefit to research and according to the vision of the European Open Science Commons. For these reasons MAX IV decided to unify all the related development efforts in a single project: DataStaMP [1].

Currently the control system drives all the equipment at MAX IV to run the accelerator and beamline devices. Essentially, highly complex instruments are made operable by combining information from thousands of sensors and actuators into a comprehensible scheme that users and operators can operate. In order to achieve this, a high level of automation is required to reduce the large numbers of manual steps into procedures that are automatically driven by the software. The MAX IV IT team is able to deliver a functional, integrated Accelerator and Experimental Operation to a level where today's experiments can be performed.

The increased data rates and volume coming from modern detectors impose a need for new data acquisition strategies.

* mikel.eguiraun@maxiv.lu.se

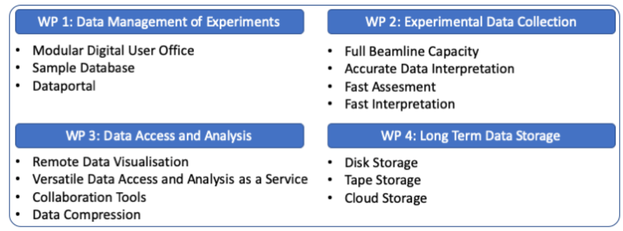


Figure 1: DataStaMP work packages, covering controls, IT infrastructure and scientific computing.

Moreover, the scale and responsibilities of the MAX IV control team suggest a unified approach for all the detectors in the facility. Therefore, this project is about a distributed (clustered) data acquisition platform for MAX IV, where all data from detectors is gathered by a multi-agent streamer ingestor. It also includes data analysis, (pre)processing and file saving.

Several implementation prototypes are being made for faster data acquisition streamed to online computing infrastructure. However, this is a moving target and plans are being made for generating data at ever higher rates in the future. The control system will need to be enhanced, particularly to improve the data acquisition but also to extract meaningful metadata and performance metrics. The researchers will analyse and interpret those in order to determine the success of the acquisition and be able to continue with the experiment. An essential factor in this is to be able to record metadata automatically while data gathering is happening. Another challenge is the variety of sensors, i.e. detectors, whose level of integration differs. This means that currently, data from all the beamlines and accelerator are not saved in the preferred standard format.

Figure 1 shows the main work packages into which the DataStaMP project has been split. Four main areas have been identified, each one covering one particular subject. *Data Management* covers all aspects related to user centred operations (user office, various databases such as sample and metadata). *Experimental Data Collection* includes acquisition, transportation and basic processing. The data analysis, scientific computing tools and services is covered by the *Data Access and Analysis*. Finally, the *Data Storage* is in charge of all the disk and network infrastructure.

The present work presents the current developments regarding data acquisition for several types of detectors. Needless to say, all mentioned topics must work together in order to provide the user first with a successful data acquisition strategy, and second, with valuable experimental data.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA046

ETHERCAT OPEN SOURCE SOLUTION AT ESS

J. Etxeberria*, J. H. Lee, A. Sandstrom, ESS, Lund, Sweden

Abstract

The European Spallation Source (ESS) [1] is a research facility being built in Lund, Sweden. The Integrated Control System (ICS) division at ESS is responsible for defining and providing a control system for all the ESS facility. ICS decided to establish open-source EtherCAT systems for mid-performance data acquisition and motion control for accelerator applications. For instance, EtherCAT will be used when the I/O system needs to be beam-synchronous; it needs to acquire signals in the kHz range; or needs to be spread across locations that are far from each other and would need cumbersome cabling, but still, belong to one system.

Following the ICS guideline, Motion Control and Automation Group developed EtherCAT Motion Control (ECMC) which is based on EtherLab open-source master. This solution was focused on Motion Control applications, but finally, data acquisition systems will be integrated into EPICS using the same approach. In this paper, we will present the ECMC solution and analyze its features showing some real applications at ESS.

INTRODUCTION

ECMC is based on EtherCAT (Ethernet for Control Automation Technology) developed by Beckhoff [2]. EtherCAT is a real-time Ethernet-based open Fieldbus that relies on conventional Ethernet frames to communicate with multiple devices in a synchronized away. Like many other Fieldbus applications, EtherCAT is based in one master/n-slave mode. EtherCAT Master relies on standard Ethernet hardware communication with the bus, so any generic network interface card (NIC, 100 MB/s Full duplex) is sufficient. Using the open-source EtherCAT Master makes a cost-effective and flexible configuration of the EtherCAT system architecture at ESS, meaning that a typical EPICS input-output controller (IOC) can be executed within an industrial PC or MTCA.4 CPU as an EtherCAT master.

In the slave side dedicated HW (EtherCAT Slave Controller ESC) provides communication on the fly with standard CAT5 connection in line, star or ring topologies. Hundred of manufacturers coordinated in EtherCAT Technology Group [3] provide slave diversity (drives, I/O, sensors and robots).

Existent Etherlab IgH EtherCAT Master [4] open source solution and EPICS applications in other facilities such as Diamond Light Source (DLS) [5] and Paul Scherrer Institute (PSI) [6] did not fulfill ESS requirements, control of motion and mid range general IO within a single system. Thus, we develop a functional open-source motion control and mid-range general IO control framework integrated into ESS EPICS Environment [7].

* julenetxeberria.malkorra@esss.se

ECMC ETHERCAT MOTION CONTROLLER

ECMC is a open-source motion control and mid-range Generic IO controller module integrated into the ESS EPICS Environment (E3). E3 is a full software environment for deploying EPICS IOCs which contains the correct EPICS base, device support module, etc. The EPICS IOC and the ECMC work together in the same Linux based CPU providing a compact solution to control motion control systems within mid-range data acquisition systems (Fig. 1).

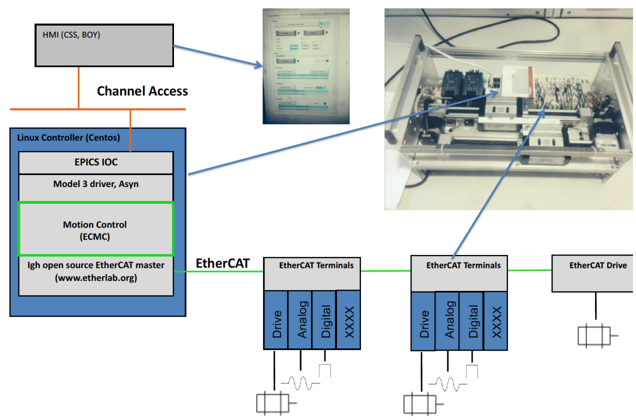


Figure 1: ECMC overview.

The ECMC communicates and configures the EtherCAT terminals through the EtherCAT protocol thanks to the tools provided by the open-source IgH EtherCAT Master. EtherLab is an open-source toolkit for rapid real time code generation under Linux. It works as a Real Time kernel module loaded within the open-source operating system Linux to communicate with peripherals devices as EtherCAT slaves through dedicated Ethernet ports. Since it is integrated into Linux kernel it has realtime characteristics, anyhow, to meet real time performance PREEMPT Real Time patch should be used, but it is not mandatory. The master provides command line tools, providing an easy way to display the status of the master and the slaves, moreover, it displays available Process Data Objects PDO and Service Data Objects SDO.

ECMC Architecture

The architecture of the ECMC is available in Fig. 2. Two main parts are described in it: AsynPortDriver and ECMC Memory.

The communication between EPICS records and ECMC is performed using AsynPortDriver [8].The different interfaces of the AsynPortDriver have been implemented allowing transfer of data of both scalar and array types in an efficient way. Typical data that is transferred over these interfaces are EtherCAT process data and other data of the configured

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA047

CABLE DATABASE AT ESS

R. Fernandes†, S. Regnell, J. Persson, S. Gysin, European Spallation Source (ESS), Lund, Sweden
S. Sah, M. Salmič, Cosylab, Ljubljana, Slovenia
L. Johansson, OTIF, Malmö, Sweden

Abstract

When completed, the European Spallation Source (ESS) will have around half a million of installed cables to power and control both the machine and end-stations instruments. To keep track of all these cables throughout the different phases of ESS, an application called Cable Database was developed at the Integrated Control System (ICS) Division. It provides a web-based graphical interface where authorized users may perform CRUD operations in cables, as well as batch imports (through well-defined EXCEL files) to substantially shortened the time needed to deal with massive amounts of cables at once. Besides cables, the Cable Database manages cable types, connectors, manufacturers and routing points, thus fully handling the information that surrounds cables. Additionally, it provides a programmatic interface through RESTful services that other ICS applications (e.g. CCDB) may consume to successfully perform their domain specific businesses.

The present paper introduces the Cable Database and describes its features, architecture and technology stack, data concepts (or entities) and interfaces. Finally, it enumerates development directions that could be pursued to further improve this application.

INTRODUCTION

The European Spallation Source (ESS) is an international neutron research facility currently under construction and expected to operate in 2023. Based in Sweden, ESS is planned to be the most powerful neutron source in the World where a multitude of scientific experiments will take place in its 22 (foreseen) end-stations instruments. For this to happen, both the machine (i.e. accelerator) and end-station instruments are powered and controlled by hundreds of thousands of cables of different types, lengths, purposes and manufacturers. Due to this overwhelming number, an application called Cable Database [1] was developed to help manage the information of (controls) cables that the divisions of the ESS Machine Directorate (and, eventually, its in-kind collaborators across Europe) are responsible for.

The Cable Database allows users to create, read, update and delete (i.e. CRUD operations) cables both through a web-based graphical interface and EXCEL files, ideal for punctual and batch cases respectively. To complement cables-related information, the application also manages the information of cable types, connectors, manufacturers and routing points (users may perform the aforementioned operations on these as well). Moreover, routing points can be associated to cables, thus effectively forming routing paths that track where cables pass (physically speaking) and provide fundamental assistance when installing, testing and maintaining these during the construction, commissioning and operation phases at ESS.

In addition, the Cable Database allows browsing cables through personalized views (e.g. display only the system, owners and installation date fields of cables), as well as filtering cables through flexible queries based on boolean expressions (e.g. filter all cables of type A owned by person B or C and with a length between D and E). Both views and queries are saved and associated to a particular user, which he/she may apply afterwards. These features, amongst others, should guarantee a high degree of satisfaction when interacting with the application (i.e. UX) and increase users' productivity.

DESCRIPTION

The Cable Database has been under development since late 2013, and a first version was publicly released late 2014. During its development phase and first years in production, the application required around 1.5 FTE. Nowadays, it only requires 0.5 FTE mainly for maintenance (i.e. implementation of minor functionalities and bug fixes), training and supporting users. Table 1 summarizes the most important metrics about the Cable Database.

To date, about 10 (major) versions of the Cable Database have been released for production, the latest (version 2.5.6) in September 2019. It currently stores approximately 35000 cables, 290 cable types, 240 connectors, 50 manufacturers and 50 routing points. The amount of data, stored in an open-source RDBMS [2], totals more than 1.2 GB.

Furthermore, the Cable Database is part of an international collaboration called DISCS [3]. This collaboration is composed of several research facilities – ESS being one of them – with the aim of developing databases, services and applications that any (experimental physics) facility can easily configure, use and extend for its commissioning, operation and maintenance.

Table 1: Metrics about the Cable Database

Description	Value
Tables (persistence tier)	22
Constraints (persistence tier)	20
Indexes (persistence tier)	23
Lines of code (persistence tier)	0
Classes in Java (business tier)	188
Lines of code (business tier)	20579
RESTful interfaces (business tier)	13
Web pages (presentation tier)	6
Dialogs (presentation tier)	23
Lines of code (presentation tier)	5304

† ricardo.fernandes@esss.se

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA048

MANAGEMENT OF IOCs AT ESS

R. Fernandes†, S. Gysin, T. Korhonen, J. Persson, S. Regnell, ESS, Lund, Sweden
M. Pavleski, S. Sah, Cosylab, Ljubljana, Slovenia

Abstract

The European Spallation Source (ESS) is a neutron research facility based in Sweden that will be in operation in 2023. It is expected to have around 1500 IOCs controlling both the machine and end-station instruments. To manage the IOCs, an application called IOC Factory was developed at ESS. It provides a consistent and centralized approach on how IOCs are configured, generated, browsed and audited. The configuration allows users to select EPICS module versions of interest, and set EPICS environment variables and macros for IOCs. The generation automatically creates IOCs according to configurations. Browsing retrieves information on when, how and why IOCs were generated and by whom. Finally, auditing tracks changes of generated IOCs deployed locally.

To achieve these functionalities, the IOC Factory relies on two other applications: the Controls Configuration Database (CCDB) and the ESS EPICS Environment (E3). The first stores information about IOCs, devices controlled by these, and required EPICS modules and snippets, while the second stores snippets needed to generate IOCs (st.cmd files). Combined, these applications enable ESS to successfully manage IOCs with minimum effort.

INTRODUCTION

The Integrated Control System (ICS) Division at ESS is mandated to deliver a system to control both its machine (i.e. accelerator) and end-station instruments. To create the system, or more precisely (distributed) control system, an open-source framework called EPICS [1] was chosen. With worldwide usage and acceptance, EPICS allows the creation of Input/Output Controllers (IOCs) that high-level software applications (e.g. CS-Studio, Archiver Appliance) may consume (i.e. connect to) to tackle domain specific businesses (e.g. OPI designing, signals archiving).

Typically, an IOC is an executable (i.e. process) that utilizes resources from EPICS modules to interface (logical or physical) devices and exposes their input/output signals as Process Variables (PVs). Eventually, an IOC may also implement logic to control these devices.

A PV is a named piece of data, usually associated with devices to represent input and output signals (e.g. status, setpoint). It has a set of attributes (i.e. fields) that integrators configure according to the specificities of the domain to solve. A PV can be read, written or monitored by applications and tools using the Channel Access (CA) library.

The (distributed) control system being built by ICS will be composed of hundreds of IOCs interfacing and controlling a multitude of devices. To develop and maintain this amount of IOCs is a challenging task, which – without proper automation – puts a heavy burden on integrators.

To alleviate this burden, the IOC Factory was developed in recent years at ICS. This application not only allows authenticated and authorized users to execute well-defined functions – configuration, generation, browsing and auditing of IOCs – but also promotes a formal, standardized workflow to manage IOCs from a high-level perspective.

DESCRIPTION

The IOC Factory has been under development since mid-2015, and a first version was publicly released early 2016. During its development phase and first years in production, the application required around one FTE. Currently, it requires less than 0.5 FTE mainly for maintenance (i.e. implementation of minor functionalities and bug fixes), training and supporting users.

To date, several versions of the IOC Factory have been released for production, the latest (version 1.2.20) in August 2019. It currently manages (i.e. stores) around 260 configurations from 50 different IOCs. These configurations (and other information) are stored in an open-source RDBMS and total approximately 10 MB. Table 1 summarizes the most important metrics about the IOC Factory.

Table 1: Metrics about the IOC Factory

Description	Value
Tables (persistence tier)	10
Constraints (persistence tier)	12
Indexes (persistence tier)	14
Lines of code (persistence tier)	0
Classes in Java (business tier)	169
Lines of code (business tier)	10801
Web pages (presentation tier)	7
Dialogs (presentation tier)	22
Lines of code (presentation tier)	2598

The IOC Factory has been developed in the context of the DISCS collaboration [2], with the aim of becoming a useful tool for more sites than only ESS. This (international) collaboration is composed of several research facilities with the aim of developing databases, services and applications that any facility can easily configure, use and extend for its commissioning, operation and maintenance.

Dependencies

To implement the aforementioned functionalities and, consequently, manage IOCs in an efficient manner, the IOC Factory relies on two other applications actively developed at ICS: the Controls Configuration Database (CCDB) [3] and the ESS EPICS Environment (E3) [4].

† ricardo.fernandes@esss.se

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA049

CERN NEUTRINO CRYOGENIC CONTROL SYSTEM TECHNOLOGY: FROM THE WA105 TEST FACILITY TO THE NP04 AND NP02 PLATFORMS

Roberto Orlandi, Czeslaw Fluder and Marco Pezzetti
CERN, CH-1211 Genève 23, Switzerland

Abstract

The CERN Neutrino Platform is CERN's undertaking to foster fundamental research in neutrino physics at particle accelerators worldwide. Different projects were undertaken and realized at CERN for this purpose. First of this series is the 35 tons liquid argon cryostat based test facility named WA105, succeeded by the 800 tons liquid argon each, designated as NP04 and NP02. The cryogenic control system of these experiments was entirely designed and constructed by CERN to operate 365 days a year in a safe way through all the different phases aimed to cool down and fill the cryostat until reaching nominal stable conditions. This paper describes the process control system design methodology, the off line validation and the operational commissioning including fault scenario handling. A systematic usage of advanced informatics tools, such as CERN/UNICOS tools, Git[1] and Jenkins[2], used to ensure a smooth and systematic software development of the process, is presented. Finally, particular attention is given to the adoption of the CERN cryogenic technical standard solutions to enhance reliability, safety and flexibility of the system working 24 hours at day.

INTRODUCTION

WA105, NP02, and NP04 are three different experiments belonging to the Neutrino Platform. WA105 and NP04 have been already commissioned during the last two years, while NP02 has been operational since June 2019.

The cryogenic process of these experiments is supported by a control system entirely designed and developed at CERN to drive the cryogenic components through the different phases of the process.

The development of such system relies on an efficient and tested methodology to build a control infrastructure that respects the initial needs of the process, as much as it allows to integrate at any moment new requirements, coming from the process evolution. This is possible thanks to the technologies that will be described in the next chapters of this paper and to a balanced computation of software and hardware spare objects in the phase of design.

The building of the control system starts from three main inputs:

- Piping and Instrumentation Diagram (P&ID): graphical representation of the full cryogenic system, showing all the objects and their connections
- Part list: detailed description of all the objects. It specifies characteristics such as the range or the type of the electrical connections
- Process logic specifications: describes the entire process and the logic to be implemented by each object

Based on these inputs, process and electrical analysis are performed to gather all the information needed to build the electrical infrastructure and the software systems that include PLC and SCADA development, as shown in Fig. 1.

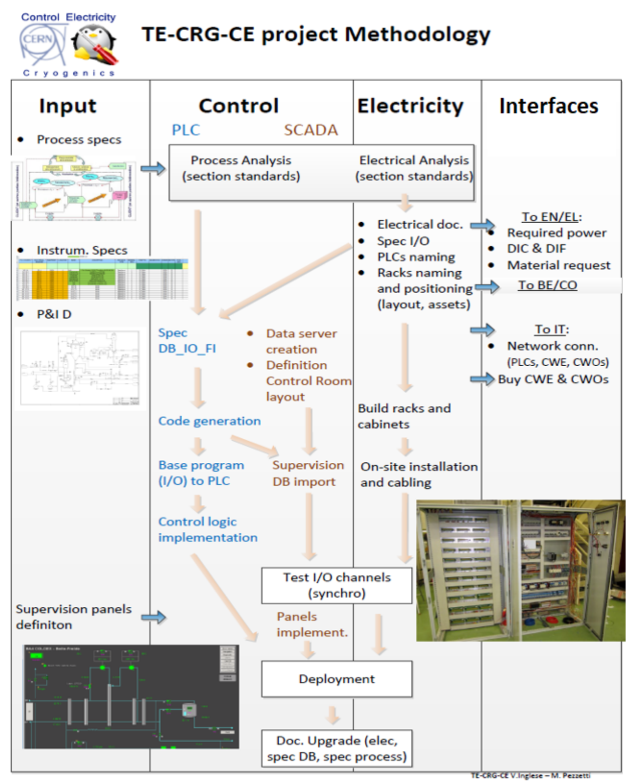


Figure 1: Cryogenics controls methodology.

CHALLENGES AND DEVELOPMENT TECHNOLOGIES

While WA105 was operational for few months, NP02 and NP04 will have a lifetime of eight years, during which the cryogenic process shall keep the cryostat in steady-state with 800 tons of liquid argon. During this time, the control system shall ensure operability, protection, maintainability, and availability 24 hours a day, 365 days a year.

Based on the LHC cryogenic system operational experience [3] and on the availability studies including root failures causes global analysis, made on other cryogenic plants at CERN, such as the LHC Atlas argon and NA62 Krypton calorimeters [4], the control architecture was designed to

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-WEPHA050

STATUS OF THE PROCESS CONTROL SYSTEMS UPGRADE FOR THE CRYOGENIC INSTALLATIONS OF THE LHC BASED ATLAS AND CMS DETECTORS

C. Fluder*, M. Pezzetti, A. Tovar-Gonzalez, CERN, Geneva, Switzerland
K. Mastyna, P. Peksa, T. Wolak, AGH University of Science and Technology, Kraków, Poland

Abstract

The ATLAS and CMS cryogenic control systems have been operational for more than a decade. Over this period, the number of PLCs faults increased due to equipment ageing, leading to systems failures. Maintenance of the systems started to be problematic due to the unavailability of some PLC hardware components, which had become obsolete. This led to a review of the hardware architecture and its upgrade to the latest technology, ensuring a longer equipment life cycle and facilitating the implementation of modifications to the process logic.

The change of the hardware provided an opportunity to upgrade the process control applications using the most recent CERN frameworks and commercial engineering software, improving the in-house software production methods and tools. Integration of all software production tasks and technologies using the Continuous Integration practice allows us to prepare and implement more robust software while reducing the required time and effort.

The publication presents the current status of the project, the strategy for hardware migration, enhanced software production methodology as well as the experience already gained from the first implementations.

INTRODUCTION

The CMS and ATLAS experiments installed in the Large Hadron Collider at CERN, have both been equipped with cryogenic installations. For the CMS experiment this installation cools a large superconducting solenoid magnet, while for the ATLAS experiment the installations are used for cooling of three superconducting toroid magnets, a superconducting central solenoid magnet and three liquid argon calorimeters.

The cryogenic system for cooling down the solenoid magnet of CMS consists of one helium refrigeration plant specified to cool down respectively the magnet's cold mass, the shield and the current leads. The cryogenic system to cool down the superconducting magnets of ATLAS is composed of two independent helium refrigeration plants (the Main Refrigerator and the Shield Refrigerator) and two Proximity Cryogenic Systems (PCS), one for each magnet system (the toroids and the solenoid). The cryogenic system of the ATLAS liquid argon calorimeters consists of the barrel and two end-cap detectors housed in three independent cryostats, filled with liquid argon. To maintain the calorimeters at the nominal temperature a complex cryogenic infrastructure and Nitrogen Refrigerating plant are required.

CONTROLS SYSTEM ARCHITECTURE

To automatize the control process of the ATLAS and CMS cryogenic installations three Distributed Control Systems (DCS) have been designed and implemented: CMS magnet, ATLAS magnets and ATLAS calorimeters. For each DCS the visualization and operation of cryogenic processes is ensured through Supervisory Control And Data Acquisition (SCADA) based on the WINCC OA® applications. Typically, one SCADA data server is connected to several autonomous Programmable Logic Controllers (PLCs), based on Schneider Quantum® technology (see Fig.1). In total 16 PLCs have been distributed throughout the cryogenic installations. Within one DCS, the PLCs share process data to coordinate the overall cryogenic process. Using the S908 Remote IO (RIO) network, each PLC is controlling I/O that are physically located in different areas: surface hall or technical or experimental caverns. In total 71 Remote IO DROPs made up of 669 I/O cards, with 7856 analog and digital RIO channels have been deployed. In addition, for some systems, the Profibus DP field-bus has been used to connect to SIEMENS S7® PLCs protecting turbines and to electronics supervising vacuum pressure gauges.

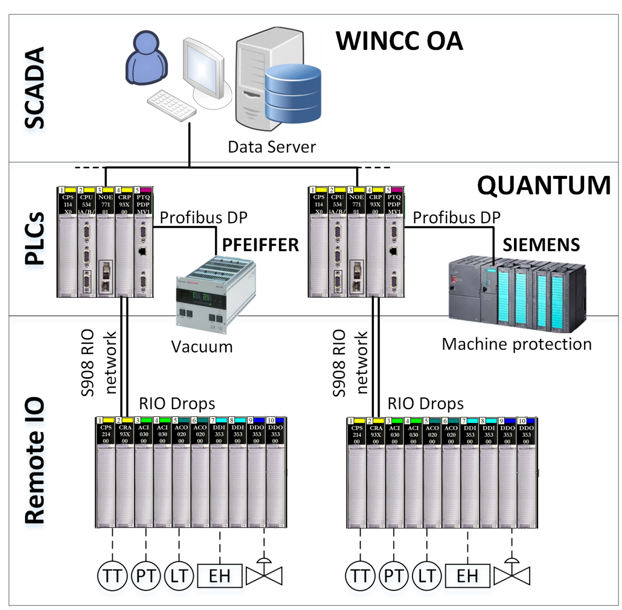


Figure 1: Typical control system architecture.

* czeslaw.fluder@cern.ch

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA052

ENGINEERING SUPPORT ACTIVITIES AT ELI-ALPS THROUGH A SYSTEMS ENGINEERING PERSPECTIVE*

L.J. Fulop†, F. Horvath, I. Kiss, A. Makai, L. Schrettner, ELI-ALPS, Szeged, Hungary

Abstract

ELI-ALPS will be the first large-scale attosecond facility accessible to the international scientific community and its user groups. The core business of ELI-ALPS is to generate attosecond pulses and provide these to the prospective users. In order to reach this ultimate goal, one key support area, the engineering development of complex systems as well as the engineering custom design service, has been systematically elaborated based on the standards, recent results, trends and best practices of systems engineering. It covers the boundaries towards all related support areas, from building operation and maintenance, to the custom manufacturing provided by the workshops, with the intention to make the model as well as the daily work as comprehensive and consistent as possible. Different tools have been evaluated and applied through the years, however, a key lessons learned is that some of the most important tools are teamwork, personal communication and constructive conflicts.

INTRODUCTION

Extreme Light Infrastructure (ELI) is the first civilian large-scale high-power laser research facility to be realized with trans-European cooperation in three sites. ELI's long-term objective is to become the world's leading user facility utilizing the power of state of the art lasers for the advancement of science and applications in many areas of societal relevance [1]. The main objective of the ELI Attosecond Light Pulse Source (ELI-ALPS) pillar is the establishment of a unique attosecond facility that provides ultrashort light pulses with high repetition rates.

The typical layout of *beamline systems* at the ELI-ALPS is as follows (see Fig. 1): the laser source produces pulses in the femtosecond duration range, which is connected via a beam transport system to one or more secondary source(s). Secondary sources are designed to produce attosecond pulses from the femtosecond laser pulses by utilizing various technologies based on Gas High-Harmonic Generation (GHHG) and Solid High-Harmonic Generation (SHHG). Beamline systems may have end stations as their closing system.

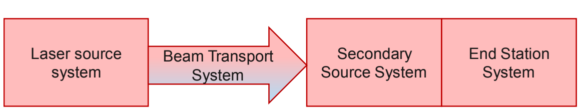


Figure 1: General model of a beamline system-of-systems from a laser until an end-station.

* The ELI-ALPS project (GINOP-2.3.6-15-2015-00001) is supported by the European Union and co-financed by the European Regional Development Fund.
† lajos.fulop@eli-alps.hu

Engineering services are really at the heart of the ELI-ALPS research facility: it is heavily connected towards the research technology as the major internal customer; however, it also has strong connections towards all other areas. In order to satisfy all expectations originating from this key role, the elaboration and organization of engineering support activities heavily considered the best practices and standards of systems engineering.

The next section summarizes the relevant best practices, standards and key works related to the scope of the paper, this way giving a sound foundation for the latter sections. Afterwards, first a big high-level picture elaborates the background and context of engineering services and then the organization of these engineering services is described.

RELATED WORK

Reference [2] lists the relevant (>30) systems engineering standards. Each focuses on specific aspects of systems engineering, especially on processes and lifecycles, as well as on vocabulary and risk management. The guide also gives a comparison as well as guidelines how to choose and how to apply these standards. A popular standard is ISO/IEC/IEE 15288 [3]0.

The openSE systems engineering framework has been developed as a common effort of research infrastructures, academic institutions and industrial partners [4]. It investigates and discusses the specificities of scientific projects regarding systems engineering and project management, with a dedicated focus on radiation safety. The framework provides an adapted and optimized systems engineering approach (in terms of lifecycle, roles and responsibilities, processes and deliverables). The framework emphasizes that scientific projects are their own prototypes, a one-of-a-kind system with an extremely complex nature and, because these reasons, the functional requirements are evolving whilst the project progresses. The authors also described that the final users, the scientific scholars taking an active part in the development effort and most of the time, they also lead projects, as it is the case also for ELI-ALPS regarding all the research technology equipment.

System integration is defined as the composition of implemented system elements into a product or a service for final validation, use and/or production (meanwhile checking the interfaces of the integrated elements) [2]. However, the systems engineering literature also refers system integration several times in a broader context, i.e. the simultaneous design and development of the systems and elements, virtually integrating designs in the early phases. System integration occurs on different levels: within a discipline, in multiple disciplines and on a socio-technical level [2]. Reference [5] introduces the W-model with the concept of continuous, virtual, cross-discipline integration and verification as early as possible.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA056

TANGO CONTROLS BENCHMARKING SUITE*

M. Liszcz, P.P. Goryl, S2Innovation, Kraków, Poland

Abstract

Tango Controls is a client-server framework used to build distributed control systems. It is applied at small installations with few clients and servers as well as at large laboratories running hundreds of servers talking to thousands of devices with hundreds of concurrent client applications. A Tango Controls benchmarking suite has been developed. It allows testing of several features of Tango Controls for efficiency. The tool can be used to check the impact of new developments in the framework as well as the impact of specific network-server and deployment architecture implemented at a facility. The tool will be presented along with some benchmark results.

INTRODUCTION

Tango Controls [1] can be used at both small and very large laboratories and scientific facilities. The efficiency, performance and resource utilization are important qualities of any control system as they allow for building large and complex installations which scale up to thousands of devices and hundreds of clients. Deploying Tango at large scale requires solutions for efficient monitoring and evaluation. Ability to assess the performance, detect regressions and identify bottlenecks is important for developers who work directly on the Tango Kernel as well as for users and for administrators who deploy Tango at their institutes. The developers need immediate feedback on whether the changes they made to the source code impact the performance. The administrators need to know how their hardware and network infrastructure affect the overall efficiency of the system. The users might be interested if the performance of Tango improves after an upgrade to a new version.

To address the need for efficiency monitoring in automated and controlled way, the Tango Controls Benchmarking Suite was developed.

BENCHMARKING SUITE

The Benchmarking Suite is a set of tools that facilitate the process of evaluating how Tango efficiency is impacted by qualities like the number of connected clients or the number of devices hosted in a device server. The suite consists of: a set of benchmark scripts, three device servers called *target device servers* and a benchmark runner script. A detailed description of each component is provided later in this section. A diagram showing all the components and relations between them is is depicted in Fig. 1

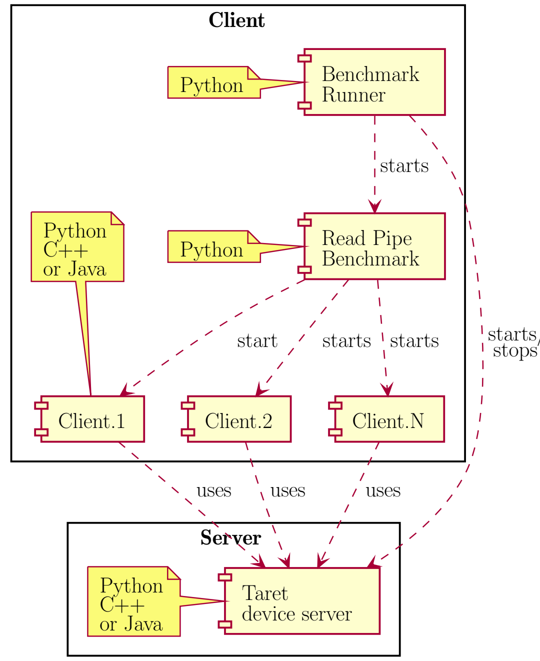


Figure 1: Components of the Benchmarking Suite.

The Benchmarking Suite was designed to be extensible. This allows users to easily write their own benchmarks and share them with others to compare the results.

Source code of the Benchmarking Suite is available online [2] under the GPLv3 license.

Benchmark Scripts

A set of benchmark scripts, written in Python, is provided with the Benchmarking Suite. These scripts implement various test scenarios. There are two kinds of tests: performance tests which execute some operation in a loop during a given period of time and efficiency tests for evaluating other aspects Tango. Following performance tests are implemented:

- Attribute Read—counts reads from an attribute,
- Attribute Write—counts writes to an attribute,
- Command—counts command invocations,
- Event—counts event subscriptions,
- Event Push—counts events received at client side,
- Pipe Read—counts read from a pipe,
- Pipe Write—counts writes to a pipe.

The tests listed above can be configured with parameters like test period, number of iterations or number of parallel operations. The actual work is performed by the client processes. The users can switch between clients implemented in Python, C++ and Java. The benchmark scripts can produce reports in CSV and reStructuredText formats.

* Work supported by the Tango Controls Collaboration

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA057

BUILDING A DATA ANALYSIS AS A SERVICE PORTAL*

A. Campbell†, A. Götz, European Synchrotron Radiation Facility, Grenoble, France
D. Salvat, A. Camps, ALBA Synchrotron, Cerdanyola del Vallès, Spain
G. Kourousias, I. Andrian, Elettra Sincrotrone Trieste, Trieste, Italy
M. van Daalen, S. Egli, Paul Scherrer Institut, 5232 Villigen PSI, Switzerland
on behalf of the CALIPSOplus JRA2 project

Abstract

As more and more scientific data is stored at photon sources as part of implementing FAIR data policies there is a growing need to provide services to access to view, reduce and analyse the data remotely. The CALIPSOplus project, in which all photon sources in Europe are involved in, has recognised this need and created a prototype portal for Data Analysis as a Service. This paper will present the technology choices, the architecture of the blueprint, the prototype services and the objectives of the production version planned in the medium term. The paper will cover the challenges of building a data analysis portal from scratch which covers the needs of multiple sites, each with their own data catalogue, local computing infrastructure and different workflows. User authentication and management are essential to creating a useful but sustainable service.

INTRODUCTION

Photon sources are hitting a data analysis wall with the huge increase in data volumes, new techniques and new user communities. Users are limited by the difficulty in exporting huge data from the source to their home labs and by the lack of easy access to data analysis programs. At the same time science is becoming more open by sharing the data, methods and publications - the so-called **O**pen Science movement. Making the data used in publications easily available enables others to reproduce the analysis and findings. This has been one of motivations for photon sources to adopt open data policies. Another motivation has been the need to alleviate the data analysis bottleneck by implementing modern data management so that data analysis services can be built on top of data catalogues. Lack of adequate data management limits the data services which can be be offered to users. In order to address these issues it was decided to include data management as a special topic in H2020 project CALIPSOplus [1].

CALIPSOplus is a project of all synchrotrons and free electron lasers in Europe and the Middle East to contribute to the completion of the European Research Area [2] and to tackle the new challenges arising from the commitment of the EU to Open science, Open innovation and Open to the world. The project aims at pushing integration with respect to user access, support, instrumentation, data analysis and data management forward. The goal is to harmonise data management and data analysis services of the different facilities. It was therefore agreed to start a Joint Research Activity (JRA) to prototype data analysis services on top of the data portals which are part of the data policies being implemented at photon sources in Europe. This paper describes the CALIPSOplus prototype data analysis portal.

* This project has received funding from the European Union's Horizon 2020 Research and Innovation programme under Grant Agreement No 730872.
† contact aidan.campbell@esrf.fr

DATA ANALYSIS AS A SERVICE

The *as a Service* (aaS) family of services include the well known *Software as a Service* (SaaS) and *Data as a Service* (DaaS) services. In order to address the data analysis bottleneck problem we propose to bring the data analysis software to the data and provide *Data Analysis as a Service* (DAAS). Data analysis can include providing data reduction services to calibrate and reduce the data to smaller volumes or different dimension e.g. from raw detector counts to physical units. The goal of DAAS is to leave data at the source and provide users with remote access to data analysis programs and algorithms. The DAAS prototype is aimed at identifying and implementing a number of prototype services for a selected set of use cases. The experience gained and feedback from users will help move towards a production ready service for data analysis in the future.

USE CASES

A survey was performed among the different CALIPSOplus members to compile a list with reusable scientific analysis software use case candidates at the different facilities. Feasible candidates for use cases were the applications for online analysis of scientific data, which are at the "end" of a data analysis chain, and therefore producing results which are directly useful for publications. However, any other application considered re-usable from other sites could be nominated. "Application" could also mean libraries, toolboxes or components from which full applications could be more easily derived. The following software packages were selected as use cases, see deliverable [3] and the software catalogue [4] for the details about the software and required dependencies:

- **CrystFEL** (DESY) : CrystFEL is a suite of programs for processing diffraction data acquired serially in a snapshot manner, such as when using the technique of Serial Femtosecond Crystallography (SFX) with a free-electron laser source.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA058

STATE OF THE TANGO CONTROLS KERNEL DEVELOPMENT IN 2019

A. Götz, R. Bourtembourg, J-M.Chaize, P.Verdier, ESRF, Grenoble, France
T. Coutinho, J. Moldes, ALBA-CELLS Synchrotron, Cerdanyola del Vallès, Spain
L. Pivetta, Elettra-Sincrotrone Trieste S.C.p.A., Basovizza, Italy
I. Khokhriakov, O. Merkulova, IK company, Moscow, Russia
S. Gara, NEXEYA Systems, La Couronne, France
P. P. Goryl, M. Liszcz, S2Innovation, Kraków, Poland
A. F. Joubert, SARAO, Cape Town, South Africa
G. Abeille, SOLEIL, Gif-sur-Yvette, France
G. Mant, STFC, Daresbury, Warrington, Cheshire, United Kingdom
T. Braun, Byte Physics, Berlin, Germany
V. Hardion, MAXIV Sweden, Lund, Sweden
M. Bartolini, SKA Organisation, Macclesfield, United Kingdom

Abstract

This paper will present the state of of kernel developments in the Tango Controls toolkit and community since the previous ICALEPCS 2017. It will describe what changes have been made over the last 2 years to the Long Term Support (LTS) version, how GitHub has been used to provide Continuous Integration (CI) for all platforms, and prepare the latest source code release. It will present how docker containers are supported, how they are being used for CI and for building digital twins. It will describe the outcome of the kernel code camp(s). Finally it will present how Tango is preparing the next version - V10. The paper will explain why new and old installations can continue profiting from Tango Controls while maintaining their investment. In other words in Tango "the more things change the better the core concepts become".

INTRODUCTION

Tango Controls is a software toolkit for building object oriented control systems. It has been adopted at a large number of sites around the world either as the main toolkit for their control system or for a sub-system or commercially acquired systems. A growing number of commercial products and control systems are now based on Tango Controls. A few commercial companies offer paying support for anyone needing help in integrating Tango Controls into their system.

With this wide user base it is important that Tango Controls continues to remain an active project and offer a clear roadmap for the future. Tango Controls has regularly given an update at the ICALEPCS series of conferences of the state of the kernel development. This paper follows in this tradition. It will provide an update of what has been achieved over the last two years as well as provide a summary of what is planned next. Readers are encouraged to refer to following papers about kernel development presented at this conference for more details on specific topics: MOPHA051, MOPHA050, WEPHA020, WEPHA056, WESH3003, and other papers on using Tango.

The main objectives of kernel developments for Tango Controls since 2017 has been to continue consolidating the continuous integration for all major platforms, maintain the Long Term Support version V9, i.e., bug fixing and add features which are strongly needed by the community and do not break compatibility with the, improve the web development platform, continue improve the documentation and maintain the website. This paper summarises these developments.

COLLABORATION

The Tango Controls collaboration is composed of several partners sites who have committed -by signing a Collaboration Contract- to financing the development and the animation of the community. The Tango Controls collaboration is growing year after year. In 2017, the largest astronomical project in the world, the Square Kilometre Array (SKA), joined the collaboration with 2 members: SKA Organisation (Manchester, United Kingdom) and SARAO (Cape Town, South Africa). In 2019, the Extreme Light Infrastructure (ELI) institute in Czech Republic joined the collaboration, to make up 11 partners financing the core development. Each partner nominates a representative on the steering committee to vote and follow-up the budget, define the strategy and the milestones.

The ESRF hosts the collaboration body and is in charge of executing the development program voted by the steering committee. The collaboration budget allowed the collaboration to boost the software development, maintenance and animation effort by sub-contracting certain tasks to commercial companies.

KERNEL DEVELOPMENT

C++ Core Library

The stable (Long-Term Support) branch of the Tango C++ Kernel is steadily evolving. The development has accelerated since the move from Sourceforge to GitHub [1] and integration of various services like Travis for continuous

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA060

FUTURE ACQUISITION ARCHITECTURE INVESTIGATIONS AT DIAMOND

K. Ralphs, J. Handford
Diamond Light Source, Didcot, UK

Abstract

At Diamond we are reviewing the current stack of in-house Software Applications that are used to control our beamline experiments and analyse the data produced by them. We intend to use this process of analysis and investigation to formulate proposals for a revised architecture to address the issues with the existing architecture, making use of the opportunities presented by modern technologies and methods, where appropriate. In doing so we hope to design a more flexible and maintainable system which addresses technical debt and functional limitations that have built up over the lifetime of our current software. This will allow us to go on to implement a powerful acquisition and analysis system to be used with the new facilities of Diamond II [1].

THE EXISTING DIAMOND SOFTWARE ENVIRONMENT

Diamond currently has a well-established stack of applications developed over many years which provides users with Data Acquisition and Analysis functionality from the Controls Hardware interface right up to the live and offline post processing and visualisation of experimental data. For organisational and technological reasons, the Controls, Acquisition and Analysis layers tend to have been developed by different teams, in some cases leading to hard boundaries of technological and operational knowledge. This often results in the need to find the expert from another team to progress the diagnosis of problems or obtain the knowledge of how to access useful information from a different layer of the stack.

Also, primarily in the Acquisition layer, the software has grown up organically over a period of 15 years leading to bad structure, un-needed redundancy, dead code and other forms of technical debt making it brittle, difficult to maintain and hard to develop. With the advent of the Diamond II redevelopment, we want to take a step back to analyse the software we have at the moment with a view to designing a new consistent platform architecture to address these and other problems and to take advantage of current industry best practises and technologies. In doing this the intention is to repackage the existing proven functionality in a more flexible structure behind a common platform API whilst revising some existing implementations and adding new capabilities and features along the way. This should allow us to migrate to a stable consistent platform that is easier to maintain and support and offers more flexibility to cross the old boundaries to get to the information required by the user.

CONCEPTUAL FUTURE ARCHITECTURE

This initiative is at a very early stage, though we do have a high level conceptual design on which we are focusing our technology examinations (see Fig 1).

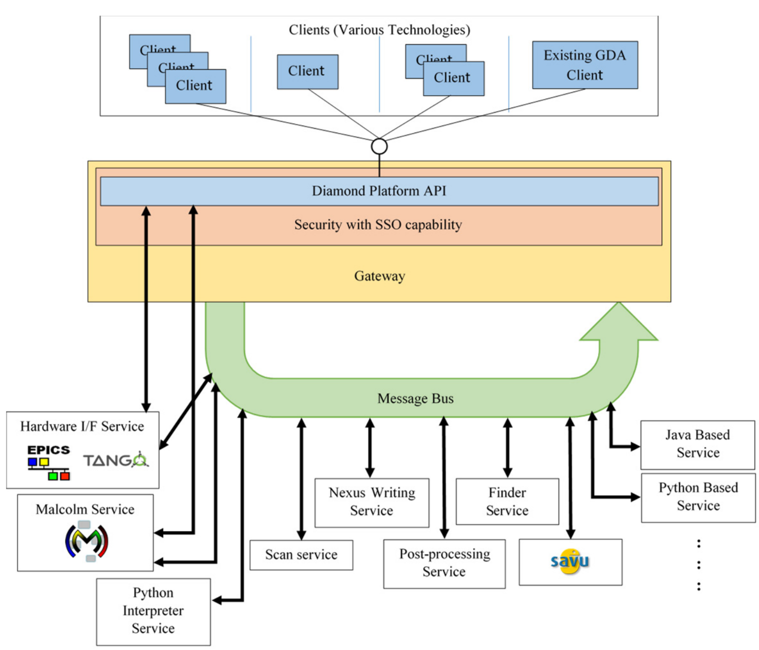


Figure 1: High Level Design.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA063

PRECISION INSERTION DEVICE CONTROL AND SIMULTANEOUS MONOCHROMATOR FLY SCANNING FOR NSLS-II

J. Sinsheimer*, P. Louis Cappadoro, T. Corwin, J. Escallier, D. Harder, D. A. Hidas, A. Hunt, M. Musardo, J. Rank, C. Rhein, T. Tanabe, I. Waluyo
Brookhaven National Laboratory, Upton, USA

Abstract

Beginning in January of 2019, 8 of the 10 In-Vacuum Undulators installed in the NSLS-II storage ring underwent in-house in-situ control system upgrades allowing for control of the magnetic gap during motion down to the 50 nm level with an in-position accuracy of nearly 5 nm. Direct linking of Insertion Devices and beamline monochromators is achieved via a fiber interface allowing precise, simultaneous, nonlinear motion of both devices and providing a fast hardware trigger for real-time accurate insertion device and monochromator fly scanning. This presentation will discuss use case scenarios at light source facilities and detail the precision achieved for simultaneous motion. Particular attention is given to the precision at which undulator energy harmonic peaks can be tracked and the variation of the peak flux in motion.

INTRODUCTION

Between January and April of 2019, eight of the 10 In-Vacuum Undulators (IVUs) installed in the NSLS-II storage ring underwent in-house in-situ control system development and replacement. The motivation for this was to correct the underperforming and unreliable operation of the vendor supplied systems, speed up step-scanning, and lay the proper groundwork for Insertion Device (ID) and monochromator synchronization for fast fly scanning of photon energy while maintaining peak photon flux. Step scanning speeds were improved by a factor of nearly five and all eight IVUs are currently running with extremely high reliability. The first real-time synchronization of an ID and monochromator has been achieved with one of the out-of-vacuum Elliptically Polarizing Undulators (EPUs) for the In situ and Operando Soft X-ray Spectroscopy (IOS) beamline with another IVU and EPU to follow shortly.

IVU CONTROL SYSTEMS UPGRADE

The eight IVUs that underwent a complete software overhaul are three 2.8 [m] long 23 [mm] period (IVU23), three 1.5 [m] long 21 [mm] period (IVU21), and two 3 [m] long 20 [mm] period (IVU20) devices. The simplest with regard to the control system among them are the two IVU20s which have one motor to control the magnetic gap and one for device elevation. This type device will not be discussed in detail, but can be thought of as the single-axis (gap) analog of the more complicated systems discussed herein. The remaining six IVUs (IVU21s and IVU23s) each have four

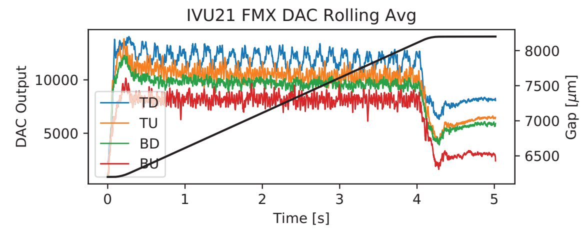


Figure 1: DAC output rolling average showing output during motion for each motor with notable oscillatory behavior of the TD output possibly indicating component misalignment or wear as an example of diagnostic plots used at NSLSII.

motor axes which we denote TU, TD, BU, and BD (where: T - Top, B - Bottom, U - Upstream, D - Downstream). The simple linear transformation shown in Eq. (1) gives the useful coordinates of gap, elevation, taper, and tilt of the device. This transformation and its inverse are used to form the forward and inverse kinematics for device motion. The IVU21 devices have an additional jack elevation motor axis which is used for alignment and not used in normal operation which is of little interest here and not discussed further.

The DeltaTau Brick Controller is used for motion control. Renishaw 1 [nm] linear encoders are used for position feedback on the four girder gap axes while the rear mounted motor rotary encoders are used for velocity feedback. Proportional Integral Derivative (PID) tuning is performed separately for top-girder and bottom-girder drives. The DeltaTau drives external servo amplifiers with a ± 10 [V] analog signal (here referred to as DAC output). It is noted here that from this output one may glean insight into mechanical misalignment or wearing, possibly preventing damage or failure if monitored occasionally. An example of this this is shown in Fig. 1. It is also noted that in these systems the linear encoders are not mounted in line with the drive shafts and may suffer from a dual feedback cantilever resonance effect. This effect has been witnessed in less massive devices [1].

$$\begin{bmatrix} gap \\ elevation \\ taper \\ tilt \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{4} & \frac{1}{4} & -\frac{1}{4} & -\frac{1}{4} \\ 1 & -1 & 1 & -1 \\ \frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} TU \\ TD \\ BU \\ BD \end{bmatrix} \quad (1)$$

IVU CONTROL PERFORMANCE

It is highly desirable to have well behaved ID gap movement during operations, not only as a prerequisite to simultaneous ID-monochromator scanning, but also to minimize

* jsinsheimer@bnl.gov

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA065

UPGRADED BEAM INSTRUMENTATION DAQ FOR GSI AND FAIR: OVERVIEW AND FIRST EXPERIENCES

T. Hoffmann, H. Bräuning
GSI Helmholtzzentrum für Schwerionenforschung, Darmstadt, Germany

Abstract

As construction of the FAIR accelerator complex progresses, the existing heavy ion synchrotron SIS18, the storage ring ESR and the high energy beam transfer lines HEBT have been upgraded to the future control system. Within this upgrade the beam instrumentation (BI) data acquisition (DAQ) systems have been heavily modernized too. These are now integrated into the control system with its White Rabbit based timing system, data supply (i.e. ion species, energy, etc.) and services like archiving. Dedicated clients running in the main control room allow visualization and correlation of the data and status of the BI devices. The DAQ hardware has been upgraded using new state-of-the-art components. With a trend to slowly phase out VME based systems, solutions based on standard Industrial PC for few channels as well as on the new µTCA standard for many channels have been successfully implemented. This contribution will give an overview over the upgraded BI-DAQ systems like current transformers and counter applications for ionization chambers, scintillators and more. It will also present first experiences during beam operation with the new control system, which started summer last year.

INTRODUCTION

While upgrading the control system of the GSI accelerators SIS18 and ESR within the last years, it was also necessary to modernize important BI-DAQ systems in parallel. This included the complete renewal of the data acquisition electronics, connection to the new White Rabbit based timing system and software integration. The goal was to design the DAQ in such a way that the solutions will be directly transferrable to the future FAIR accelerators.

The control system upgrade was a challenging issue as the established procedures for timing and software integration have significantly changed. Collaborations with CERN lead to a new, complete and useful control system infrastructure. Besides the FESA environment and the LSA settings management, the development and integration of the White Rabbit based timing system helped to achieve higher accuracy, better performance and high flexibility in the readout of BI systems.

CONTROL SYSTEM

The decision for FESA (Front-end Software Architecture) [1] as the lowest tier of the GSI and FAIR accelerator control system opened the way for the BI group to contribute directly with programming and integration of BI software into the control system. FESA is based on the classic client-server architecture, with the executable FESA classes acting as servers on the frontend computers (FEC). The FECs are operated with CentOS 7 Linux via PXEboot. Supported FECs are Intel based µTCA CPUs, Industrial PCs and in rare cases VME controllers. In addition, Siemens PLCs for slow controls are supported by FESA via the CERN made SILECS [2] interface tool. User access to all DAQ systems is provided by JavaFX graphical user interfaces, which subscribe to the relevant properties published by the FESA classes.

TIMING

The new White Rabbit (WR) based timing system plays the central role in the renovation of the BI DAQ systems. A detailed description can be found under [3]. The WR timing system is based on the IEEE 1588 2008 (precision time protocol PTP) standard and a network infrastructure similar required for Gigabit Ethernet. A timing master (TM) coordinates and synchronizes all FAIR Timing Receiver Nodes (FTRN) distributed in the field in the one-nanosecond range. The TM sends out event messages, which are then executed on time in the FEC. Each message consists of the exact time it is to be executed as well the timing group, event id, beam process id, sequence id and other information. The timing group indicates the part of the accelerator complex, for which the message is valid. While for example the SIS18 is a single timing group the HEBT is divided into several groups with each beamline segment between two switching magnets typically having its own group id. Events like beam extraction may be played in the timing group of the ring but not in the timing groups of the HEBT. Beam processes constitute the smallest unit in the operation of the accelerator, i.e. injection, acceleration, etc. Each beam process has its own id. The full accelerator cycle is given by a sequence of beam processes and has a corresponding sequence id. However, for a given beam from the source to the target, the number of beam processes and the beam process and/or sequence ids are usually different from timing group to timing group. This has to be taken into account in all BI applications which correlate the data of different devices along the beam path.

INTENSITY MEASUREMENTS

AC Current Transformer

The injection of the ion beam from the linear accelerator UNILAC into SIS18 is monitored by two AC current transformers (ACCT). Figure 1 shows the measured current for a single injection into SIS18 in the early phase of commissioning after a long shutdown. The upper graph displays the data from the ACCT in the transfer channel. It shows the approximately 40µs long macro-pulse from the UNILAC and its rather flat intensity distribution. The lower

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-WEPHA067

CONTROL SYSTEM DEVELOPMENTS AND MACHINE MODEL BENCHMARK FOR THE GSI FRAGMENT SEPARATOR FRS*

J. P. Hucka†, J. Enders, Technische Universität Darmstadt, Darmstadt, Germany
S. Pietri, H. Weick, D. Ondreka, J. Fitzek, B.R. Schlei
GSI Helmholtzzentrum für Schwerionenforschung, Darmstadt, Germany

Abstract

At the GSI Helmholtzzentrum für Schwerionenforschung and at the future FAIR facility the LHC Software Architecture LSA wil be used for a new control system for accelerators, beam transfers and storage rings. In addition, the fragment separator FRS and - at a later stage - also the superconducting fragment separator Super-FRS at FAIR will be controlled within this framework. In fragment separators of in-flight facilities, the interaction of the beam with matter in the beamline and the beam's associated energy loss needs to be taken into account. This energy loss is calculated using input from ATIMA and has been included into the code of the LSA framework. The setting generator was simulated and benchmarked by comparison to results of earlier measurements. The modeling of slits and their magnetic-rigidity-changing properties as well as modeling of the propagation of charge states and isotopes through matter was included.

FRS

The FRS is a high resolution magnetic forward spectrometer with a maximum magnetic rigidity of 18 Tm and a momentum and angular acceptance of ±1 and ±7.5 mrad, respectively [1]. It is used to separate and identify reaction products from nuclear reactions such as fission or fragmentation of heavy-ion beams with matter. It consists of 9 focal planes with 1 dipole (green) and 5 quadrupole (yellow) magnets in between each plane to allow separation and beam focussing (Fig. 1). Furthermore the FRS is equipped with multiple multi-wire-proportional chambers (MWPC), time-projection-chambers (TPC), scintillators (SCI), ionization chambers (IC), targets, and degraders to allow mass and charge of the ions to be identified in flight on an event-by-event basis.

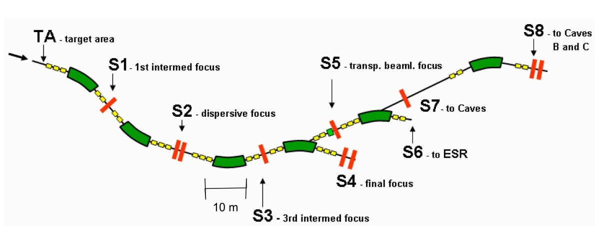


Figure 1: Magnetic setup of FRS and focal planes. [2]

In 1990 the FRS has been completed at the GSI Helmholtzinstitut für Schwerionenforschung and has been in operation ever since. With the construction of the new FAIR facility (Facility for Antiproton and Ion Research [3]), the existing GSI facility is currently being upgraded to accommodate the current experimental beam time of FAIR-Phase-0 and future beam times at both the GSI and FAIR facilities [4]. Following the completion of FAIR, the facility will feature also a new superconducting fragment separator (Super-FRS [5]) with enhanced acceptance.

CONTROL SYSTEM

Following the upgrade, the GSI/FAIR facility will provide heavy-ion beams at higher intensities than before. With the construction of the new FAIR center, a control system upgrade is also required in order to provide streamlined operability for the combined GSI/FAIR facility for increased operation efficiency. These control system upgrades encompass both hard- and software.

Hardware

Like in GSI's main control room, the hardware for the FRS controls has been replaced. Figure 2 shows the old control panels and the refurbished control room of the FRS in comparison. Old consoles and computers - operating on a VMS platform - have been removed. New computers operating on Linux and Windows 10 and monitors were installed. A set of 2x3 monitors and 2 computers is dedicated to the operation and controlling of the FRS by using a combination of the LSA [6] framework and applications additional to self developed monitoring applications for pneumatic drives and stepper motors, called DRIVESTAT, dipoles, quadrupoles and sextupoles, with FMGSTAT, and FMGSKAL, which is being used to launch the magnetic pre-cycling sequence outside of the LSA framework. DRIVESTAT and FMGSTAT communicate both with LSA and the devices directly to read out and monitor LSA setting values, set device values, current device values and status reports. Next to it monitors and a dedicated computer is placed for calculation and online simulation purposes utilizing LISE++ [7] and MIRKO [8]. On top of this set up a big monitor is used for online detector readout via ROOT [9].
The new setup facilitates access to the FRS controls via LSA, other monitoring applications, simulations tools and online detector readout. The equipment thus allows an efficient interplay between simulation, experimental data on beam properties and controls to be implemented.

* Work supported by BMBF (05P15RDFN1 and 05P19RDFN1)
† j.-p.hucka@gsi.de

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA068

A CONTROL SYSTEM USING EtherCAT TECHNOLOGY FOR THE NEXT-GENERATION ACCELERATOR

M. Ishii#, M. T. Takeuchi, JASRI/SPring-8, Hyogo-ken, Japan
C. Kondo, JASRI, Hyogo, Japan
T. Fukui, RIKEN SPring-8 Center, Hyogo, Japan

Abstract

The construction of a new 3GeV Light Source is in progress. Furthermore, we have an upgrade project of SPring-8 that we call SPring-8-II. We adopted EtherCAT technology as a network fieldbus for the next generation control systems. Currently, a low-emittance electron gun system and a digital control system with a magnet power supply have been built and bench-tested at for the 3 GeV Light Source. These systems are controlled from the MADOCA control framework via EtherCAT. Additionally, we are proceeding with the design of a new high-power RF (HP-RF) control system based on the HP-RF control system by SACLA and will introduce the system into the 3GeV Light Source. These new systems will be validated in a prototype accelerator for the 3GeV Light Source at the SPring-8 site, and will be installed in the 3GeV Light Source.

INTRODUCTION

Multiple fieldbus protocols have been used for data communications with remotely installed devices for over 20 years at SPring-8. Currently running remote I/O systems using a master/slave topology are as follows: a serial remote I/O (RIO) that was introduced at the start of SPring-8 operation, opt-VME [1] developed at SPring-8 in 2003, optical FA-link [2], and DeviceNET. These protocols are used with a magnet power supply control, a vacuum control, and an undulator control. However, products that work with RIO, opt-VME, and optical FA-Link have already been discontinued.

The construction of a new 3GeV Light Source is in progress. High-speed bus access, transfer of large amounts of data, and synchronization with a timing signal are required in the next-generation control system. As a network fieldbus, we adopted Ethernet for control automation technology (EtherCAT) [3]. Because its cyclic data transfer time is less than 1 ms, EtherCAT is suitable for a fast control and feedback system. As the control system using EtherCAT, a low-level RF (LLRF) system, a new standard in-vacuum undulator system at the SPring-8 storage ring, and a patter power supply with a kicker magnet at SACLA have been operating [4].

In this paper, we describe an improved way of handling an EtherCAT slave that requires a control sequence. We also describe the design and installation plan of the new control systems using EtherCAT technology for the 3GeV Light Source.

ishii@spring8.or.jp

CONFIGURATION AND COMMUNICATION

The EtherCAT master requires an EtherCAT network information (ENI) file for network initialization and configuration. An ENI file is generally generated from individual EtherCAT slave information (ESI) files by using a software configuration tool. An ESI file contains vendor and product information, initialization information, process data, a distributed clock synchronization configuration, etc. The main purpose of the EtherCAT master is cyclic exchange of process data with the configured slaves. There are two types of process data:

- Input process data object (Input PDO, data received from the EtherCAT slaves)
- Output PDO (Output PDO, data transmitted to the EtherCAT slaves).

Figure 1 shows a system configuration using EtherCAT. We adopted an EtherCAT master AdXMC1573 with an XMC form factor. EtherCAT communication is processed on the master protocol stack, and the operating system accesses PDO and service data object (SDO) via a shared memory mapped by the device driver of this module. We installed the EtherCAT master module into a VME, a MicroTCA.4 and a PC.

Overwrite of a Request

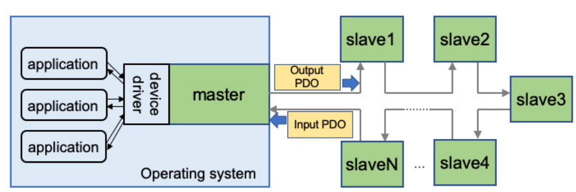


Figure 1: System configuration using EtherCAT.

We adopted a Melec F3200/EC as a motor controller. The F3200 has been used to control cavity tuners in the digital LLRF system [5] and to control undulator gap in a new standard in-vacuum undulator (IVU-II) [6]. In addition, the F3200 will be used in the 3GeV Light Source and SPring-8-II. In the MADOCA control system, at least two processes, an equipment management process (EM) and a data logging process (MDAQ), run on one host. When requests are issued from multiple processes to an EtherCAT slave, exclusive control may be necessary. Figure 2 shows the PDO map image of the F3200. The current pulse count and status of each axis can always be read from the input PDO without issuing a request. However, requests such as

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA069

babyIOC - CONTROL SYSTEM IN A BOX SMALL FACTOR SOLUTION*

O. Ivashkevych†, M. Cowan, L. Flaks, D. Poshka, T. Smith
BNL, Upton, USA

Abstract

In the world of increasing complexity and integration, experiments often stretch over multiple beamlines or several facilities. Users may come with their own sample environments and detectors. It is always a challenge to integrate user end-station equipment into the hosting facility controls. Recognizing this trend, NSLS2 has developed babyIOC Control System in a Box, portable small-form-factor IOC solution.

The selected hardware is from innovative hardware designer UDOO. This SBC has microSD card storage, 64-bit Intel architecture, 4-core 2.56 GHz, 8 GB of RAM, x3 1 Gbit interfaces (including dual port daughter card). The cost of board and extensions is ~$400 US. We've configured it so the system boots and runs from microSD card. Building another system comes to copying the image to another microSD card. We believe this board with the easy downloadable image can be used at any facility and/or experimental stations. Given a growing interest to areaDetector software from the Tango community, babyIOC could serve as evaluation starting point.

MOTIVATION

Back in March 2018 we faced the need to support mobile experiments at NSLS2. It was clear that the controls system solution would need to travel with the experiment, thus, it should be small factor and powerful enough for the job. It would need to blend seamlessly with the NSLS2 control system, be low maintenance, and easily upgradable to support evolving needs of the experiment. In addition, we were looking for an inexpensive, small factor solution for quick prototyping and testing in the real production environment.

HARDWARE SELECTION

From analyzing areaDetector [1] performance on older hardware, we determined that 2GB memory and 8 core 1GHz CPU should be enough to run areaDetector IOC, the most resource demanding application. We also needed at least two ethernet interfaces.

We have looked at single board computers offered by Beagleboard, Raspberry Pi, Nvidia, PINE64, UDOO, Boundary Devices, Hardkernel, PC Engines, and others. We stopped on the UDOO Ultra [2] board originally funded on Kickstarter in 2016. This board offered Intel architecture 4-core 2.56 GHz, 8 GB of RAM. NSLS2 computing hardware is standardized on Intel architecture and all system's installation is provisioned using Foreman, and configuration managed (orchestrated) with Puppet. Because of the matching architecture, installing our standard operating system on UDOO Ultra was simple. UDOO Ultra also has three 1 Gbit Ethernet interfaces: one on the board plus two provided by a daughter card. It also comes with a microSD slot, HDMI, miniDP++, USB 3.0, UART and other interfaces. It should be noted that the board boots and runs from microSD card. It also has a possibility to add a hard drive via an onboard SATA connector. We did not use this option opting instead for mircoSD because it is small, low-power, low-heat, solid state, and easy to swap. Another advantage of using microSD card for storage is that one can create a master image with all necessary software deployed and configured, which could then be copied on a new microSD card. Additional systems can be created by simply acquiring the hardware and copying the system image onto a microSD card. Figure 1 shows the arrived hardware and Figure 2 the assembled board.

Figure 1: Unassembled hardware: UDOO Ultra board, two 1Gb Ethernet daughter card, metal case, cooling fan.

Figure 2: Assembled babyIOC.

* This research used resources of the National Synchrotron Light Source II, a U.S. Department of Energy (DOE) Office of Science User Facility operated for the DOE Office of Science by Brookhaven National Laboratory under Contract No. DE-SC0012704.
†oksana@bnl.gov

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA071

TIMING SYSTEM INTEGRATION WITH MTCA AT ESS

J. J. Jamroz, J. Cereijo García, T. Korhonen, J. H. Lee, ESS, Lund, Sweden

Abstract

European Spallation Source (ESS) organization has selected cutting-edge technologies to satisfy performance and scalability expectations:

- Micro Telecommunications Computing Architecture (MTCA).
- Micro Research Finland (MRF) based timing system with delay compensation.
- Experimental Physics and Industrial Control System (EPICS).

To achieve optimal data acquisition quality, the control system is built on top of the timing system which gives the same absolute time reference to all EPICS process variables (PVs). The MTCA system gives configurable cableless access to manage connections among different electronic mezzanine cards, therefore reducing installation workload.

INTRODUCTION

ESS is a collaboration of 17 European nations and its objective is to be the world's most powerful spallation neutron source [1].

The neutrons are produced by a 5 MW proton beam (125 MW peak) hitting a solid, rotating tungsten target in 600 m distance from the ion source. The target weights 11 tonnes and rotates at 23 1/3 rpm containing 36 sectors.

The acceleration of the proton beam is ensured by a Radio-Frequency linear accelerator (RF linac) operating on 352.21 MHz (RF_L) up to 216 MeV and 704.42 MHz (RF_H) up to the nominal energy of 2 GeV. Main characteristics are 14 Hz repetition rate, 62.5 mA peak beam current, and 2.86 ms beam pulse length.

The linear accelerator consists of Normal Conducting Linac (NCL) and Superconducting Linac (SCL) with the following parameters:

- NCL: length = ~50 m, nominal energy = ~90 MeV.
- SCL: length = ~550 m, nominal energy = 2 GeV.

Superconductor cavities are cooled down to ~2 K using liquid helium. The design considers 180 klystrons with 1 MW peak power per each (40 kW average). The machine targeted operational availability is 95%.

TIMING NETWORK

Figure 1 presents the timing network. The installation is based on single mode OS2 duplex fibers (double sided arrows) due to the bandwidth–length product, monolithic network structure and compatibility to the IT infrastructure and easier maintenance. The RF Front End, TM, TD11, TD12 chassis are located within one cabinet making the tree trunk and connected to uninterruptible power source. The RF Front End electronics feeds TM with $RF_L/4$ (88 MHz) and Pulse per Second (PPS) synchronized to Global Positioning System (GPS). The TD21 up to TD2X chassis are distributed along with the linac inside the klystron gallery (service area) and supplying timing signals to other subsystems via TDMS. TDMS is an MTCA chassis with EVR which belongs to a specific subsystem. The EVR injects timing signals into the MTCA backplane to other AMC cards within the subsystem.

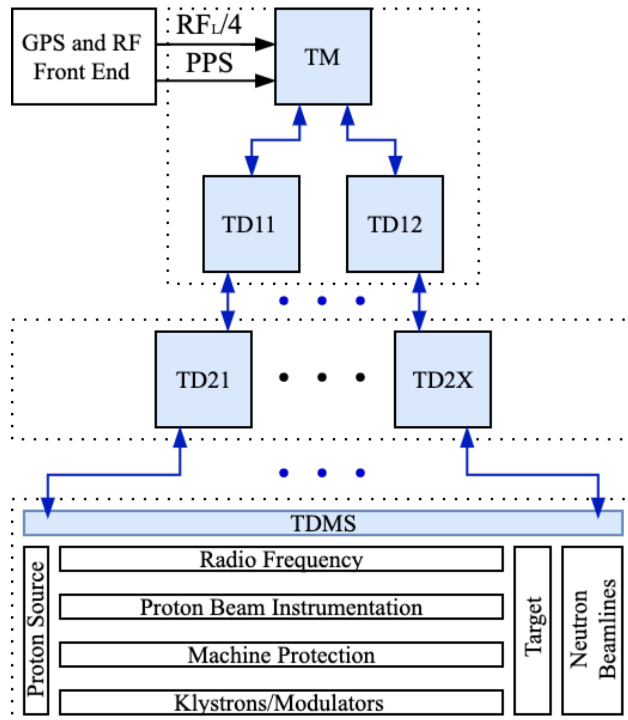


Figure 1: ESS timing network - tree topology. Timing Master (TM); Timing Distribution (TD); TDX_1X_2: X_1 - tree layer, X_2 - ID, TD2X contains X = 41 currently; Timing Distribution to Machine Subsystem (TDMS).

The MRF timing protocol assigns automatically each timing device to a topology ID. This feature has been used to develop a timing network builder [2] which plots actual network connections with states and power levels of transceivers. The builder is used to troubleshoot installation, to verify documentation and to perform maintenance checkups. In addition, the protocol supports a delay compensation of an optical signal. It makes the system resilient to aging and temperature impact on optical fiber (propagation delay) as the signal delay drifts are continuously compensated.

MTCA

MTCA is an open standard embedded computing specification created by PCI Industrial Computer Manufacturers Group [3]. Figure 2 presents the specification capabilities. MCH acts as a switch of different interfaces connected to AMCs within the system. Therefore, each AMC can utilize

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA075

EPICS ALSO FOR SMALL AND MEDIUM SIZED EXPERIMENTS*

H. Junkes†, Fritz-Haber-Institut der Max-Planck-Gesellschaft (FHI), Berlin, Germany

Abstract

The Experimental Physics and Industrial Control System (EPICS) consists of a series of software tools and applications for the development, implementation and operation of distributed control systems. It is used worldwide in many, mostly larger facilities such as particle accelerators, free electron lasers and telescopes. EPICS scales from very large to very small systems due to the efficient memory usage and support of many hardware platforms. This article describes some of these small systems as well as typical experiment controls in chemistry and physics at research institutions. These are placed in the context of large plant controls.

BASIS: SOMEWHAT LARGER EXPERIMENT

A mid-infrared FEL has been commissioned in 2013 at the Fritz-Haber-Institut (FHI) in Berlin. It is be used for spectroscopic investigations of molecules, clusters, nanoparticles and surfaces. The oscillator FEL is operated with 15 – 50 MeV electrons from a normal-conducting Sband linac equipped with a gridded thermionic gun and a chicane for controlled bunch compression. Construction of the facility building with the accelerator vault began in April 2010. First lasing was observed on Februar 15th, 2012 [1]. The EPICS software framework [2] was choosen to build the control system for this facility. The facility management system is integrated using BACnet/IP. Graphical operator and user interfaces are based on the Control System Studio package. The EPICS archiverAppliance, an electronic logbook, a web based monitoring tool, and a gateway complete the installation.

The Max-Planck-Gesellschaft (MPG) has now funded a significant upgrade to the FHI FEL. A second, short-Rayleigh-range undulator FEL beamline is being added that will permit lasing from <5 µm to >160 µm. Additionally, a 500 MHz kicker cavity will permit simultaneous two-colour operation of the FEL from both FEL beamlines over an optical range of 5 to 50 µm by deflecting alternate 1 GHz pulses into each of the two undulators [3].

ADDON: STANDARD/SMALL EXPERIMENTS

Based on the EPICS infrastructure, which is available at the institute through the FEL project, this environment has been used for experiment control in many standard experiments at the institute since then. Some special solutions could also be implemented cheaply and efficiently.

* Work supported by Max-Planck-Gesellschaft, Germany
† junkes@fhi-berlin.mpg.de

Small but Fine Devices

Due to the small footprint of EPICS and the support of the ARM processor family it is possible to realize an Input/Output Controller (IOC) on the Raspberri Pi Single Board Computer (SBC).

Measurement@20 kV The IOC runs on a Raspberry Pi Zero W. It has all the functionality of the original Pi Zero, but comes with added connectivity, consisting of [4]:

- 802.11 b/g/n wireless LAN
- Bluetooth 4.1
- Bluetooth Low Energy (BLE)

A mounting device was built to allow the Pi Zero be attached on top of the Fluke 287 (True RMS Multimeter). For the communication with the Fluke, the infrared connector of the multimeter is used. The conversion into a serial signal (RS232) is done on the holding device. The battery of the multimeter is used to power the Pi Zero. This allows the instrument to be operated (incl. EPICS IOC) at 20 kV. This is used, for example, to detect voltage breakthroughs in cold molecule traps.

Figure 1: Holding plate with Pi Zero.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA078

A VIRTUALIZED BEAMLINE CONTROL AND DAQ ENVIRONMENT AT PAL

S. W. Kim*, H. J. Choi, H. S. Kim, and W. W. Lee†
Pohang Accelerator Laboratory, Pohang, South Korea

Abstract

At least three different computers are used in the beamlines of PAL, first for EPICS IOC, second for device control and data acquisition(DAQ), and third for analyzing data for users. In the meantime, stable beamline control was possible by maintaining the policy of separating applications listed above from the hardware layer. As data volumes grow and the resulting data throughput increases, demands for replacement of highly efficient computers has increased. Advances in virtualization technology and robust computer performance have enabled a policy shift from hardware-level isolation to software-level isolation without replacing all the computers. DAQ and analysis software using the Bluesky Data Collection Framework have been implemented on this virtualized OS. In this presentation, we introduce the DAQ system implemented by this virtualization method.

VIRTUALIZATION

As the paradigm shifts from single-core to multi-core programming, interest in virtualization is increasing. The beamline control software were categorized into three major parts, IOC / DAQ / Analysis, and has been run on different computers. However, as multi-core processors become common, the traditional approach is inefficient in terms of not using idle computing resources. By dividing the system resources through virtualization, it can allocate dynamically to the environment with high resource utilization and save time for installation and programming environment construction. Unlike server virtualization, a workstation requires terminals such as monitors, keyboard, and mouse to control each virtual machine. Allocating physical hardware to a virtual machine is called *passthrough*. The types of hypervisors will be discussed in the following session.

Virtualization Solutions

Virtualization solutions can be classified into three major types depending on the hypervisors: KVM, Xen, and ESXi (See Table 1). Kernel based Virtual Machine (KVM) converts the Linux kernel into a hypervisor, and takes advantage of Linux's process and memory management, network, I/O, and driver features. KVM has been integrated into the kernel since Linux version 2.6.20 and can be used immediately without rebuilding the kernel. Xen, like KVM, turns the Linux kernel into a hypervisor and can also take advantage of the Linux kernel, but requires a new build of the kernel with the Xen module loaded. KVM and Xen are both open-source and available for free on most Linux distributions and BSDs. Libvirt can manage virtual machines running on top of these hypervisors, and can be easily managed with Virt-manager, a GUI interface that utilizes it. ESXi is a hypervisor provided by VMware. The Free version has some limitations, including up to two physical CPUs, a maximum of eight vCPUs per virtual machine (VM), and other API limitations. The hypervisors listed above enable operating system virtualization and passthrough for free.

Table 1: Virtualization Solutions for GPU Passthrough

Solutions	Hypervisor	Pricing
Linux/kvm	KVM	Free
PROXMOX	KVM, LXC	Free
Linux/Xen	Xen	Free
XCP-ng	Xen	Free
Citrix XenServer	Xen	$763.00 / CPU
VMware	ESXi	Free (w/ limit)

Based on these hypervisors, there are commercial solutions that provide virtual machine, network and storage management tools, and commercial support. Some examples are VMware vSphere (ESXi), Citrix XenServer (Xen), XCP-ng (Xen), and PROXMOX (KVM). Citrix XenServer has removed the GPU passthrough feature from the community edition since version 7.3, but it is still available for free in its open source version, XCP-ng. XCP-ng and PROXMOX are free to use all features without any limitations, but security updates and support are available with commercial annual subscription.

Virtual machine orchestration is required to remotely manage and monitor hypervisors installed in various beamlines without having to log in directly. For most cases, it is sufficient to connect to the hypervisor with SSH and manage the VMs via Libvirt and Virt-manager. A virtualized workstation using CentOS 7 / KVM as a hypervisor will be introduced in the following section. For reference, Orchestration solutions include oVirt (KVM), XOA (Xen), and vCenter Server (ESXi).

Virtual Machines on a Workstation

The workstation's resources consisted of 32 logical CPUs, 32 GB of RAM, two graphics card, a PCIe to USB hub, and hard disks have been allocated to the virtual machines. The procedures of setting up CentOS 7 as a hypervisor and GPU passthrough are summarized separately [1]. CentOS 6 (IOC VM), CentOS 7 (DAQ VM), and Windows 7 (Analysis VM) operating systems were installed, respectively, and eight vCPUs and 8 GB of RAM were allocated. And the remaining resources were assigned to the hypervisor. Since the IOC

* physwkim@postech.ac.kr
† lww@postech.ac.kr

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA080

A COMMUNICATION PROTOCOL FOR MOTION CONTROL APPLICATIONS AT THE JCNS NEUTRON INSTRUMENTS

H. Kleines†, F. Suxdorf, Forschungszentrum Jülich, Germany

Abstract

Main focus of slow control in neutron scattering is motion control for the movement of around 25 mechanical axes in a typical neutron instrument. The implementation of motion control functions in the JCNS neutron instruments at the FRM II research reactor in Garching, Germany, is based on Siemens S7 PLCs. A communication protocol called PMcomm which is optimized for motion control applications in neutron instruments has been developed at JCNS. PMcomm (PROFI motion communication) is based on PROFINET or PROFIBUS as the underlying transport protocol in order to facilitate the easy integration into the PLC world. It relies on the producer/consumer communication mechanism of PROFINET and PROFIBUS for the efficient direct access to often-used data like positions or status information. Coordinated movement of groups of axes is facilitated by a generic controller/axes model that abstracts from the specifics of the underlying motion control hardware. Simplicity was a major design goal of the protocol in order to allow an efficient and easy implementation on PLCs.

INRODUCTION

JCNS, the neutron science division of Forschungszentrum Jülich, developed and operates 15 neutron instruments at its outstations ILL in Grenoble, the Spallation Neutron Source in Oak Ridge and at the FRM-II in Garching near Munich. Further neutron instruments are being developed for the future European Spallation Source in Lund. The control and data acquisition systems of these neutron instruments are responsible for a variety of tasks including detector readout, vacuum systems, cryogenic systems, sample environment devices and motion of many mechanical axes.

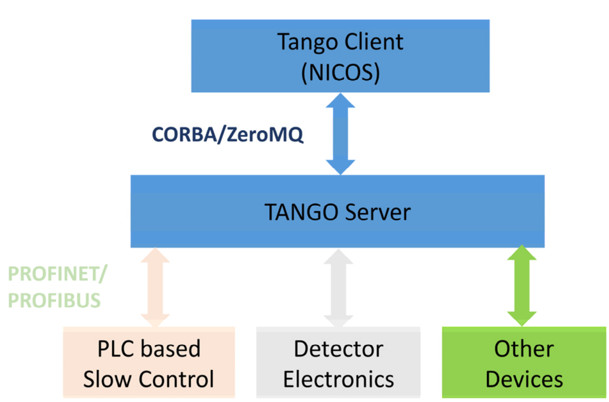


Figure 1: Jülich-Munich Standard System Architecture.

In order to reduce the overall development effort und the number of spare parts, these control systems are highly standardized and follow a common framework, the so-called Jülich-Münich standard [1], that evolved gradually over a period of more than 20 years. Originally being based on the TACO framework, it is now based on the successor TANGO [2], as shown in Fig. 1. Main client application is the standardized measurement program NICOS [3] providing a GUI interface as well as a scripting interface for scan definition and execution. All slow control tasks - especially motion - are implemented with industrial PLC technologies including fieldbus communication and decentral peripheral systems in the frontend. Main motivations for this approach are:

- low prices induced by mass market,
- inherent robustness,
- long term availability and support from manufacturer,
- powerful development tools.

Because of its strong market position, Siemens has been selected as the PLC vendor for all JCNS instruments.

In the past the communication of PLCs with peripheral devices and with supervisory computers was implemented with PROFIBUS DP, since this field bus standard was naturally supported by Siemens PLCs. In the recent years Ethernet-based industrial communication systems like PROFINET IO [4], Ethernet/IP, Ethercat, Powerlink or Modbus TCP became increasingly important for the communication with industrial devices and today PROFINET IO (see section III) is the most commonly used industrial Ethernet system and inherently supported by all PLCs from Siemens. As a consequence, all new developments for JCNS instruments are based on PROFINET IO and PROFIBUS is being replaced by PROFINET in older instruments.

For the communication between motion related PLCs and controlling computers running the corresponding TANGO servers, a dedicated application protocol called PMComm (PROFI Motion Communication) has been developed. Originally designed for PROFIBUS DP it supports also PROFINET IO, since both fieldbus systems share an identical communication model, requiring just a minor software adaptation on the PLC side as well as on the host side.

PROFINET IO

Originally defined by PI (PROFIBUS and PROFINET International), PROFINET IO is now internationally standardized in IEC 61158 and IEC 61785. It is based on 100 Mbit/s Ethernet supporting wireless, copper and optical media in full conformance to the corresponding IEEE standards. As a consequence, a PROFINET communication interface can be used simultaneously for PROFINET real time communication as well as for standard internet

† h.kleines@fz-juelich.de

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA081

ANALYSIS AND DIAGNOSTIC TOOLKIT FOR OPERATION EVENT IN THE NSRRC

T. Y. Lee, W. Y. Lin, T. W. Hsu, Y. K. Lin, H. H. Chen, B. Y. Chen, S. J. Huang, B. Y. Huang, H. C. Chen, J. A. Li, C. H. Kuo, NSRRC, Hsinchu, Taiwan

Abstract

Taiwan Photon Source (TPS) and Taiwan light source (TLS) have been operated in the same time. TPS is a 3 GeV electron energy, 518 m circumference, low-emittance synchrotron storage ring which will offer one of the synchrotron x-ray sources, provide cutting-edge experimental facilities and novel multidisciplinary scientific research. TLS is a 1.5 Gev electron energy. The control system is difference between two facilities. Amount of instruments and devices these must be monitored and controlled by operator. The difference diagnostic tools will be difficult to operate and analysis between two system. These utility toolkits are effective to reduce operator loading. However, these tools are developed with same concept, combined with two difference machine is effective and reduce maintenance efforts. These applications of software will be reported in this conference.

INTRODUCTION

TLS is a small, state-of-the-art and compact synchrotron radiation facility featuring with adapted energy for users. This machine are still operated and supported with high reliability and stability beam quality. The operation performance is shown in the Fig. 1. The schedule user time is more than five thousand hours. The user availability is 98.7%. The beam time between fail (MTBF) is 365.8 hours. For this operation request, alarm in advance and analysis after event, it must be quickly found out and solved to avoid same event again in the next time [1]. The TPS is opened for users from 2016. The operation time includes of user time, sub-system commission and beamline commission. It isn't only for users. The schedule user time is 4479 hours in the 2018. Thea beam availability is 99.2%. The MTBF is 93 hours. There are many events are processed in the short time. But it still 48 times trips. In the 2019, the statistics are until August. The operation performance is shown in the Fig. 2. Amount of signals analysis follow trip that are heavy duty in the TPS. For specially, large scale signals of TPS are more than TLS. The utility are developed to scan signal automatically and find sub-system problem after event that are necessary in the big data analysis. Following subsystem from beam trip event and statistics are to classify signal that is effective to reduce searching time and CPU loading [2].

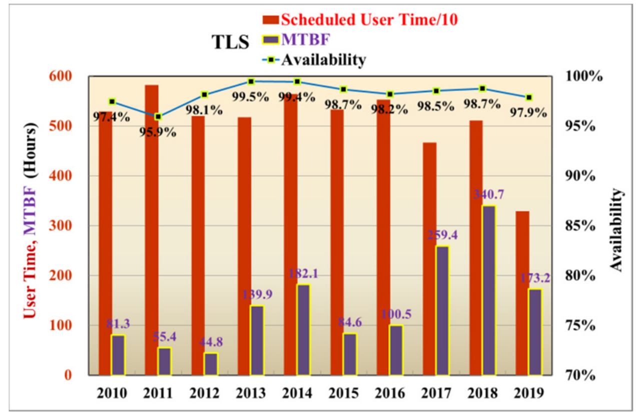


Figure 1: Annual operation statistic of TLS.

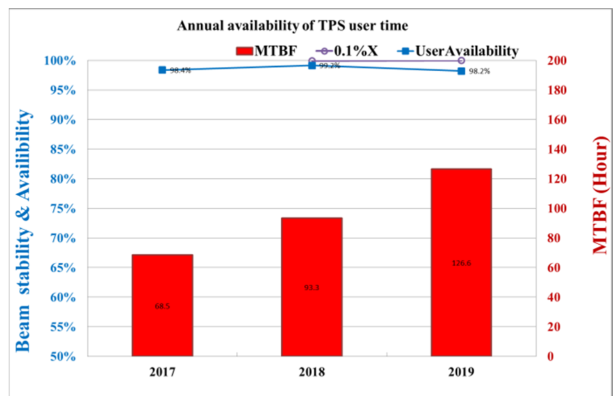


Figure 2: Annual operation statistic of TPS.

TLS BEAM OPERATION EVENT AND ANALYSIS UTILITY

In 2006, the super conductivity RF cavity system is installed in the storage ring. The main trip rate is contributed in the beginning operation of this system. In 2011, TLS is operated in 360mA topup from 200mA. Meanwhile it is always aimed to improve the performance of facility as indicated by availability, mean time between failures (MTBF) and beam stability index. Availability is defined as the ratio of delivered user time to the scheduled user time; MTBF as the ratio of scheduled user time to number of faults. After 2012, digital bunch by-bunch feedback systems were working. There are many instability are reduced. The storage ring reliability is improved to stable status. The annual beam trip statistics are shown in the Fig. 3. In the 2015, there is some new power amplifier configuration problem of bunch-by-bunch feedback system in the beginning operation. After 2016, everything is solved. The MTBF is up very much. However, there are

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA083

`ophyd` DEVICES: IMPOSING HIERARCHY ON THE FLAT EPICS V3 NAMESPACE [*]

K. Lauer[†], SLAC National Accelerator Laboratory, Menlo Park,USA

Abstract

EPICS V3 provides simple data types accessible over the network through Channel Access identified by a flat process variable (PV) name. This flexibility is often regarded as a strength of EPICS, as the user can easily pick and choose the information they require. However, such data is almost always inter-related in some manner, pushing the burden of reconstructing that relationship to the end-user/client.

ophyd represents hardware in Python as hierarchical classes, grouping together related signals from the underlying control system. ophyd devices make imposing this hierarchy simple, readable, and descriptive. This structure allows ophyd to provide a consistent interface across a wide-range of devices, which can then be used by higher-level software for any number of tasks: from command-line inspection, to scanning/data collection (bluesky), or even automatic GUI generation (typhon, adviewer). ophyd contains a number of pre-built devices for common hardware (and IOCs) as well as the tools to build custom devices.

BACKGROUND

EPICS and PVs

Standard EPICS IOCs host a process database of records. Records, which generally hold a primary value in the field `.VAL` along with related metadata in other appropriately-named fields (e.g., `.DESC` for description, `.EGU` for engineering units, etc.), are made available to clients and other servers over the Channel Access (CA) protocol. In a properly configured network of IOCs, records are almost always uniquely named such that only one IOC on one machine serves information from its database. At that point, a specific field of a record `RECORD.FIELD` over CA is often referred generically to as getting or putting to a Process Variable (PV).

Values along with a fixed set of metadata can be retrieved over CA in a single request. In CA, it is not possible to group together multiple requests and have the server return an atomic result.

A Short Note about PVAccess

Much additional work has been put into the most recent major version of EPICS (V7) in recent years to bring structured data to the protocol level, which is not currently possible in Channel Access, with a new protocol called PVAccess. This addition allows for keeping structured data accessible and synchronized at the IOC server level.

Such synchronization is outside of the scope of ophyd, which currently relies on CA to retrieve or change PVs on IOCs.

While ophyd does not currently have PVAccess support, it is a feature that is currently in the planning stages. The composability, configurability, and consistent API of devices, as described in later sections, will still apply when ophyd is PVAccess-capable – even allowing for CA, PVAccess, and soft/simulation signals to be mixed in as-needed.

There are thousands of deployed EPICS V3 servers that may never see an upgrade to V7 for a variety of reasons, meaning that the relevance of CA and related higher-level applications will likely persist for decades.

SIGNALS

`ophyd.Signal`

An ophyd `Signal` represents the smallest set of data a user might be interested in – a single temperature value, a PID setpoint or readback, and so on. The data held by a `Signal` may be structured and may have additional metadata associated with it, including timestamps and control limits.

Signals can be used in isolation, instantiated as needed. The strength of ophyd comes in when signals are used in conjunction with devices, which is detailed in the next sections.

`ophyd.EpicsSignal`

A subclass of `ophyd.Signal`, an `EpicsSignal` bridges the gap between CA and the ophyd signal interface.

As setpoints and readback values are often separate PVs in EPICS, an `EpicsSignal` allows for specifying a PV to write to (setpoint) and a PV from which to read (readback).

A simple example might be that of the motor record[1], where the user-setpoint `.VAL` and the user-readback `.RBV` are fields of the same record:

```
motor = ophyd.EpicsSignal(
    write_pv='MOTOR.VAL',
    read_pv='MOTOR.RBV',
    name='value')
status = motor.set(3.0)
```

Signals may also be enforced to be read-only at the ophyd layer, on top of any access rights enforced at the IOC level. These signals are differentiated easily by the `RO` suffix on the class – i.e., `EpicsSignalRO`.

For example, the following is effectively `caput PV:NAME.VAL 3.0`:

[*] Work supported by U.S. D.O.E. Contract DE-AC02-76SF00515.
[†] klauer@slac.stanford.edu

[1] There is first-class `Device` support in ophyd for motor record. See `ophyd.EpicsMotor`.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA086

A FAST WIRE SCANNER SYSTEM FOR THE EUROPEAN XFEL AND ITS IMPACT ON SAFETY SYSTEMS

T. Lensch*, T. Wamsat, Deutsches Elektronen Synchrotron, Hamburg, Germany

Abstract

The European-XFEL is an X-ray Free Electron Laser facility located in Hamburg (Germany). The 17.5 GeV superconducting accelerator will provide photons simultaneously to several user stations. Currently 12 Wire Scanner stations are used to image transverse beam profiles in the high energy sections. These scanners provide a slow scan mode for single bunch operation. When operating with long bunch trains (>100 bunches) fast scans are used to measure beam sizes in an almost nondestructive manner. To operate fast scans multiple impacts on the beam loss system (BLM) and the charge transmission interlock (TIS) have to be taken into account. This paper focusses on the interaction between these systems and first experiences performing measurements.

INTRODUCTION

The E-XFEL is a superconducting accelerator with an energy of up to 17.5 GeV. Within one RF pulse of 600 µs up to 2.700 bunches can be accelerated. With a repetition rate of 10 Hz this corresponds to up to 27.000 X-ray pulses per second that can be distributed to the different undulator lines to allow for simultaneous operation of experiments [1].

Wire Scanners at E-XFEL

Wire Scanners are widely used for beam profile measurements. A fork equipped with thin wires passes through the electron beam. The wire interaction with the beam produces scattered electrons and showered particles downstream the wire scanner unit which are detected by photomultipliers. The signal plotted over the wire position represents the beam profile (see Fig. 1).

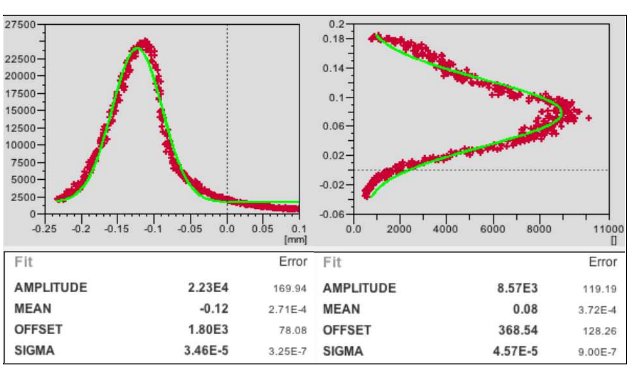


Figure 1: Profiles measured by wire scanners with 20 µm wire at 250 pC per bunch. Left the horizontal plane (x-axis: wire position, y-axis: photmultiplier signal) and right the vertical plane (axes swapped) is shown. A Gauss fit is displayed in green.

At the E-XFEL there are 14 wire scanner units installed. Each wire scanner unit consists of two motorized forks (horizontal and vertical plane) driven by a linear servo motor. This 90 ° configuration of motors helps to avoid vibration influences. The motion unit is integrated by a custom front end electronic into the MTCA.4 [2] environment. A set of three 90 ° tungsten wires (50, 30 and 20 µm) and two crossed 60 ° wires (10 µm) is mounted on each titanium fork (see Fig. 2). The wire position is measured with an optical linear ruler. [3]

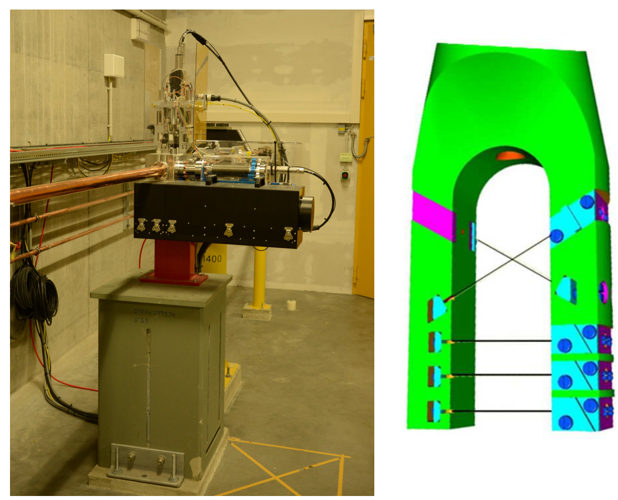


Figure 2: Left: wire scanner unit with horizontal and vertical plane installed in the E-XFEL with a screen station in the foreground. Right: 3D drawing of the fork with the different wires. Wire thicknesses from bottom to top: 50, 30 and 20 µm) and two crossed 60 ° wires (10 µm).

Wire scanner units are installed in groups of three upstream of the collimation section and upstream the undulator systems. Two locations in the post linac measurement section are equipped with an additional wire scanner unit each. Scattered electrons and showers are detected by several dedicated photo multipliers installed downstream each set of wire scanner units. Figure 3 shows the distribution of wire scanner units in the E-XFEL.

SCAN MODES

The wire scanner system developed for the E-XFEL supports *Slow* and *Fast Scans* for different measurement purposes.

Slow Scan

Slow scans are performed with single bunches (one bunch every 100 ms) at the E-XFEL. Thereby the wire is driven slow (i.e. 0.2 mm/s) for a distance of a few millimeters

* timmy.lensch@desy.de

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-WEPHA089

DESIGN AND IMPLEMENTATION OF SUPERCONDUCTING BOOSTER CONTROL SYSTEM

A.L. Li, Z. Peng, J. Zheng, China Institute of Atomic Energy, Beijing, China

Abstract

In order to improve beam energy, a superconducting booster is built behind the tandem accelerator. The Control system is designed based on EPICS according to its functional needs. It gives a detailed description of hardware and software. The control system realizes data acquisition, network monitoring, Process variable (PV) management, database services, historical data analysis, alarm and other functions of remote device. The running result shows that the control system has fast response time and works stably and reliably, which meets the control requirement.

INTRODUCTION

Superconducting booster is a post-accelerator device of HI-13 tandem accelerator in CIAE. It is combined with tandem accelerator by bunching and chopping method to obtain higher beam energy. Such a combination of two devices enlarges the species of ions that exceed the coulomb barrier energy [1], which has scientific research value. The control system should not only keep the phase and amplitude of RF electric field in superconducting cavity stable, but also realize the functions of remote sequential switch, condition monitoring and interlocking protection for vacuum, cryogenic, radio frequency and motor subsystems. In addition, it should have good reliability and expansibility. Experimental Physics and Industrial Control System (EPICS) is a control system software tool for accelerators and other large-scale scientific experimental devices.It adopts server/client structure and has strong plasticity and expansibility. Since the 1990s, EPICS has been widely used in accelerator laboratories all over the world. Considering the requirements of upgrade project and the advantages of EPICS, the control system of booster is based on EPICS. All equipment control is integrated into EPICS system in order to save manpower and material resources, efficiently realize booster control and operation condition monitoring also.

SYSTEM STRUCTURE AND PRINCIPLE

Superconducting booster is located behind tandem accelerator for improving beam energy. In the physical design, a beam pulsing system, including traveling wave chopper and double drift buncher, is added at the front end of the tandem accelerator to improve the beam utilization and match the longitudinal acceptance of the booster. At the same time, a third stripper is installed before the beam of tandem accelerator injected into the superconducting booster to improve the ionic charge state and make full use of the voltage gain of the superconducting cavity of the booster. In order to ensure that the beam has a sufficiently small envelope in the superconducting cavity, a new set of three-unit quadrupole lenses is installed at the front end of the booster. A set of phase stabilization system is installed to stabilize the phase of the pulse beam entering the amplifier. The structure of the superconducting booster is shown in Fig. 1. It is distributed at both ends of the tandem accelerator, which consists of XY guider, traveling wave chopper, double drift buncher, phase stabilization system, three-unit quadrupole lens, and auxiliary equipment such as radio frequency system, cryogenic system and vacuum system. According to the difference of beam energy, the front end of tandem accelerator is called low energy end, and the back end is called high energy end.

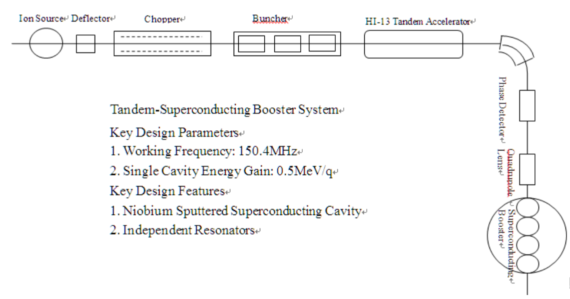


Figure 1: Systematic schematic diagram.

The basic principle of superconducting booster: the DC beam from ion source is cut into uniform pulsed beams by the transverse electric field generated by traveling wave chopper and modulated to 1ns pulse width by the longitudinal time-varying electric field generated by double drift buncher, then accelerated by tandem accelerator. When the phase of the pulse beam is synchronized with the phase of the high frequency electric field in the cavity, the particles are accelerated synchronously. When the pulsed beam enters the phase detector, it excites high frequency signal, which is used to adjust the phase of the pulsed system in the low energy band through the phase stabilization system, so that the phase of the pulsed beam which enters the booster is stable. The booster consists of four QWR (Quarter-Wave Resonator) copper-niobium sputtering superconducting cavities. The design index of the energy-increasing section is 2Mev/q.

HARDWARE DESIGN OF CONTROL SYSTEM

The equipment needed to be controlled in the pulsing system includes a set of frequency source to provide frequency and phase reference for superconducting booster and beam pulsation system; a set of chopper

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA090

TESTING TOOLS FOR THE IBEX CONTROL SYSTEM

T. Löhnert, F. A. Akeroyd, K. Baker, D. Keymer, A. J. Long, C. Moreton-Smith, D. Oram,
ISIS Neutron and Muon Source, Didcot, UK
J. R. Holt, T. A. Willemsen, K. J. Woods, Tessella, Abingdon, UK

Abstract

At the ISIS Neutron and Muon Source [1], we are currently in the process of replacing the legacy SECI control system with the next generation IBEX control system [2] based on EPICS [3]. Since IBEX replaces a fully functional control program, users and developers need to have confidence that the new system works as well as the old one, ensuring that the migration does not cause any undue disruption to beamline operations. Automated testing is an indispensable tool to continuously ensure our software is up to the highest standard of quality. This paper gives an overview of the various types of testing tools we utilize at ISIS, with a particular focus on our IOC test framework [4]. This framework has been developed in-house for testing drivers with simulated devices in order to circumvent testing limitations due to beamline/device availability.

INTRODUCTION

For the last 5 years, we in the Experiment Control team have been developing the next-generation instrument control system called IBEX for use on beamlines in the ISIS facility. The migration to IBEX is an ongoing process, with around half of our beamlines currently running on the new system. Being such a large, complex and long-running project make IBEX prone to failure if proper care is not taken. Consider the following facts:

- IBEX is a complex ecosystem consisting of many different services. It is not always obvious what impact changes to one component will have on another
- Over the course of its lifetime, over 30 developers have contributed to the IBEX project, many only for a limited duration and sometimes early in their software engineering careers
- The IBEX system is too complex even for permanent staff to be familiar with every part of it at any one time, let alone have in-depth knowledge of it
- Devices are often fixed on the beamline they are used on. Since beam time is highly valuable, it is often difficult to get sufficient time to test a new device driver before it is needed in production. This can similarly apply to re-testing if changes/updates are later made

Automated testing helps meet all of the above challenges. A robust set of tests ensures any unforeseen changes in behaviour are caught before they are deployed to production beamlines. Once written, robust tests benchmark any newly developed code to forever protect it from regression. Tests can also act as a form of documentation for unknown parts of the system (in addition to other forms of documentation), as they demonstrate the expected behaviour of the component being tested, which is especially helpful if the original author has since moved on. Finally, having a well-designed test framework with strong simulation capabilities goes a long way to ensure the surprises when testing with a real beamline are kept to a minimum.

Robust testing practices raise the confidence of the developers that new code works as intended, and that existing code continues to work when subjected to changes. It also helps raise the confidence of users, who are generally risk-averse that the new control system they are being given performs its function satisfactorily.

In the following sections, we will explore a selection of relevant IBEX components, and the tools in place to ensure their continued functionality.

IBEX COMPONENTS

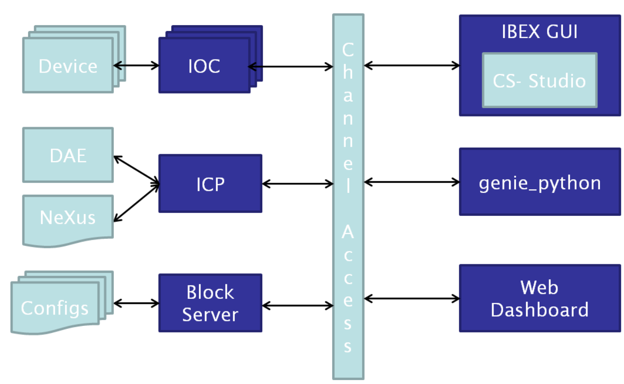


Figure 1: Overview of (selected) IBEX components.

IBEX covers a wide range of responsibilities related to experiment control, from interacting with individual bits of beamline equipment such as temperature or pressure controllers, choppers, jaws etc. to managing data collection and writing the final experiment data file. To this end, it comprises a number of technologies and services.

Figure 1 shows a schematic of the core system. On the left side, we have the server comprised of various backend components:

- A number of devices which are driven through EPICS Input-Output Controllers (*IOCs*), which expose relevant values to be read/written to over the network using the EPICS Channel Access protocol.
- The Instrument Control Program (ICP), which deals with the neutron data (collection, processing and writing to file) obtained from the Data Acquisition Electronics (DAE)
- The Block Server, which is a python process that manages the beamline configuration – i.e. which devices to control, which sample environment values to log etc.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA091

GENERALISING THE HIGH-LEVEL GEOMETRY SYSTEM FOR REFLECTOMETRY INSTRUMENTS AT ISIS

T. Löhnert, A. J. Long, ISIS Neutron and Muon Source, Didcot, UK
J. R. Holt, Tessella, Abingdon, UK

Abstract

At the ISIS Neutron and Muon Source [1], we are currently in the process of rolling out the next generation IBEX control system [2] across all beamlines. One class of beamline we have yet to migrate to the new system is reflectometers. These beamlines require equipment to track the path of the neutron beam to high levels of precision over various experimental configurations, which results in a unique set of requirements for the motion control system. We have implemented a higher level geometry layer responsible for linking the positions of beamline components together so that experimental parameters such as the incident beam angle θ (Theta) are preserved across different beamline configurations. This functionality exists in the legacy system, but needed to be re-implemented for IBEX. The new reflectometry system [3] is written as a Python server running as part of the server on the local instrument machine. This paper provides an overview of the architecture of the new system, specifically how it supports the design goal of making the system easy to extend and reconfigure while preserving the functionality of the existing solution, as well as an outlook on future plans for a more sophisticated motion control system enabled by axis synchronization in real-time.

INTRODUCTION

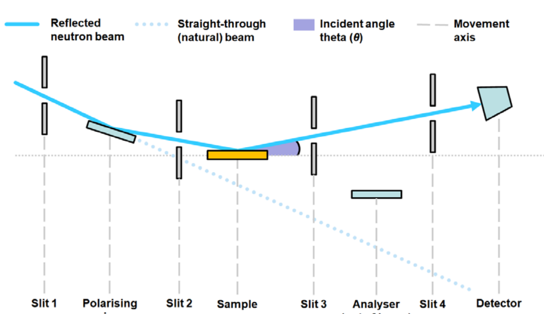


Figure 1: Schematic of a typical reflectometry beamline.

Reflectometers are complex beasts that require the user to keep tight control of a multitude of experimental parameters in order to achieve the intended results. As control system developers, we strive to make this process as easy and intuitive as possible.

In the case of reflectometers, we do this by hiding the complexity of low-level motor control behind a higher level parameter layer. We achieve this by linking the positions of devices on the beamline in physical space so that when one of them (e.g. a polarising mirror) alters the beam path, other devices further along the beam automatically move to track the new path. Currently, the high-level motion control for our 5 reflectometry beamlines is provided by the legacy SECI control system through dedicated LabVIEW [4] VIs. This is currently being replaced by the new IBEX control system based on the open source EPICS toolkit [5]. In order to support reflectometers in the new system, we had to replicate the functionality available in SECI from scratch.

The existing solution under SECI is rather fragmented, having been extended and changed over the course of many years to deal with requirements as and when they arise. Each beamline has its own variation of control interfaces, scripts and workflows with limited documentation. As such, we took this as an opportunity to create a generalised design under IBEX, so that we only have one system that can then be configured for different beamlines in terms of the composition and geometry of the beamline, i.e. which moving parts exist, where on the beamline do they sit, and what is their range of motion. This has a multitude of benefits: We only need to maintain one system, any improvements to the system are available to all beamlines simultaneously, and it helps establish standards and consistency for all ISIS reflectometers.

In the following sections, we will explore the design of the reflectometry server under IBEX, and how it achieves these goals while preserving the workflows the scientists have developed and grown accustomed to over the course of ISIS' many years of operation.

SERVER ARCHITECTURE

Overview

IBEX is implemented in a client-server architecture. Individual devices are controlled by EPICS Input Output Controllers (*IOCs*). These IOCs can be run as part of the server and provide device-specific Process Variables (*PVs*), values which can be read or written to over the network using the EPICS Channel Access protocol.

The Reflectometry Server is implemented in python, using the PCASpy library [6] to expose values and methods in the code via PV addresses. To the outside, the reflectometry server looks and acts like any other EPICS IOC.

The server's main function is transforming the positions of physical motors into higher level parameters that take the geometry of the beamline and the current path of the neutron beam into account. The reflectometry server itself consists of three layers to accomplish this (see Fig. 1). Each contains a list of items, strictly in order from closest to the beam source to furthest from it since any change may affect items further along the beam. The layers are as follows (see Fig. 2 for schematic):

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA092

SNS CREDITED PULSE ENERGY LIMIT SYSTEM CONCEPTUAL DESIGN*

C. Deibele†, K. L. Mahoney††, D. C. Williams†††,
Oak Ridge National Laboratory, Oak Ridge, USA

Abstract

The Controls Group at the Spallation Neutron Source (SNS) is designing a programmable signal processor based credited safety control that calculates pulsed beam energy based on beam kinetic energy and charge. The SNS Pulsed Energy Limit System (SPELS) must reliably shut off the beam if the average power exceeds 2.145 MW averaged over 60 seconds. This paper will cover the architecture and design choices needed to develop the system under the auspices of a programmable radiation-safety credit control. The authors will also introduce the concept of a graded failure approach that allows the credited system to continue operation in the presence of some faults.

PROBLEM DESCRIPTION

The SNS is reliably operating at its initial design beam power of 1.4 MW, which is well below the facility safety envelope of 2 MW. After the Proton Power Upgrade (PPU) at the SNS, the machine will be capable of producing a beam power of up to 2.8 MW although the safety envelope will remain at 2MW. The disparity results in a requirement for a credited system that can be implemented through the Personnel Protection System (PPS) to shut down the beam if it is too high. This system is known as the SNS Pulsed Energy Limit System (SPELS) or the Beam Power Limit System (BPLS).

A table of 2.8 MW beam parameters is shown in Table 1.

Table 1: PPU Beam Parameters

Nominal Current Pulse Width	$0.75\ \pm 0.1\ \mu sec$
Kinetic Energy	1.3 GeV
Nominal Rep Rate	60 Hz
Peak Current	100 A
Average Current	2.15 mA
Bandwidth	$\leq 22\ MHz$

For background, the power delivered to the target is calculated as follows:

$$P = R_{rate} E_b \int_{t_o}^{t_o+w} I(t)dt \tag{1}$$

where R_{rate} is the repetition rate of pulses on the target, E_b is the beam energy, t_o is the time domain start of the beam, w is the current pulse width, and I(t) is the time domain current distribution.

The present plan is to operate the FTS at a 60 Hz repetition rate until the Second Target Station (STS) comes online, then operate the FTS at 45 Hz and the STS at a 15 Hz repetition rate. Additionally, when the STS comes online, it is envisioned to deploy a second BPLS system to limit the STS beam power.

Traditional protection and safety systems that interface with the PPS system are controlled via a Safety Programmable Logic Controller (Safety PLC). The nature of the dynamic range of currents that are expected to be measured and the relatively narrow beam pulse width makes for a challenging implementation with a Safety PLC. Additionally, the known electrical noise environment is challenging since the near by beam extraction kickers induce a large signal on the ground when the beam is extracted out of the accumulator ring.

IMPLEMENTATION

Repetition Rate

An examination of Eq. (1) reveals that three separate measurements are needed to measure the beam power. The first measurement of the repetition rate is a function of the timing system at the SNS. This is can vary between 1 and 60 pulses per second. The accelerator can only be run in a setup that is less than or equal to 60 pulses per second, and so an algorithm that works for 60 pulses per second is needed.

Beam Energy

The next necessary measurement is the kinetic beam energy. In normal operation, the beam energy is determined by a measurement of the time-of-flight (TOF) using a beam position monitor system in the Linear Accelerator (linac). This is then used to set the main dipole magnets in both the accumulator ring and in the Ring-To-Beam-Target (RTBT) beamline. Since the magnetic field necessary to bend the trajectory (θ) of the beam is a function of the beam energy (*T*), and the magnetic field is a function of the magnet current (*BL(I)*), an independent measurement of the magnet current to calculate energy is possible.

$$T = \sqrt{E_0^2 + \left(\frac{cBL(I)}{\theta}\right)^2} - E_0 \tag{2}$$

The magnet current will be measured with a Safety PLC using three Direct Current Current Transformers (DCCTs). Each DCCT has a 4-20 mA output for a fail-safe connection. This is required so that the Safety PLC can sense if the DCCT amplifier is working correctly or if it has lost power.

* This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Basic Energy Sciences, under contract number DE-AC05-00OR22725.
† deibele@ornl.gov
††mahoneykl@ornl.gov
††† williamsdc@ornl.gov

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA093

CODE GENERATION BASED ON IFML FOR THE USER INTERFACES OF THE SQUARE KILOMETRE ARRAY (SKA)

M. Brambilla, S. Pavanetto, M. Gasparini, Politecnico di Milano, Milano, Italy
A. Marassi†, R. Cirami, INAF-Astronomical Observatory of Trieste, Trieste, Italy

Abstract

The Square Kilometre Array (SKA) project is responsible for developing the SKA Observatory, the world's largest radiotelescope ever built. In this context, a number of Graphical User Interfaces (GUI) have to be designed and built to be used for monitoring and control, testing, simulation, integration, commissioning and maintenance. The Tango framework and its UI tools, selected for SKA in 2015, support the types of basic control interfaces currently used at both radio telescopes and within high energy physics experiments. This paper reports on the development of a Qt/Taurus code generator prototype based on the IFML (Interaction Flow Modeling Language) standard and respective modeling tools, that are extended for supporting the platform-specific code generation. The purpose of this work is to enable the use of low-code development in SKA GUI design, thus enabling increased efficiency, reliability and coherency of the produced UI. We present a simple GUI use case as complete example of software development cycle starting from requirements and including IFML modelling, Qt/Taurus automatic coding, interface evaluation and validation.

INTRODUCTION

The Square Kilometre Array (SKA) project is responsible for developing the SKA Observatory, the world's largest radiotelescope ever built: eventually two arrays of radio antennas - SKA1-Mid and SKA1-Low - will be installed in the South Africa's Karoo region and Western Australia's Murchison Shire, each covering a different range of radio frequencies. In particular, SKA1-Mid array will comprise 133 15m diameter dish antennas observing in the 350 MHz-14 GHz range, each locally managed by a Local Monitoring and Control (LMC) system plus the 64 MeerKAT dishes, arranged in a dense core with quasi-random distribution, and spiral arms going out to create the long baselines that go up to 200km [1] and remotely orchestrated by the SKA Telescope Manager (TM) system.

Four sub-elements can be identified in the SKA-Mid1 dish element: the Dish Structure (DS), the Single Pixel Feed (SPF), the Receiver (Rx) and the Local Monitoring and Control as described in [1].

Dish LMC will provide a Graphical User Interface (GUI) to be used for monitoring and Dish control in standalone mode for testing, TM simulation, integration, commissioning and maintenance.

The performed technological prototyping of Qt and Taurus based upon Python and PyQt has shown they fulfill the basic SKA.DISH UI requirements and could be used to implement desktop UIs like SKA.DISH UIs.

Therefore, we focused on the development of a Qt/Taurus code generator prototype based on the IFML (Interaction Flow Modeling Language) standard, with the aim of automating the user interface implementation production.

The purpose of this work is to enable the use of low-code development in SKA GUI design, thus enabling increased efficiency, reliability and coherency of the produced UI. We present a simple GUI use case as complete example of software development cycle starting from requirements and including IFML modelling, Qt/Taurus automatic coding, interface evaluation and validation.

The paper is organized as follows: we first introduce the features required in the SKA.DISH LMC user interface, then we discuss the background concepts and technologies that are foundational for our approach, spanning usability, accessibility, user-centered design and Tango control. We describe our user-centered design activities and then we describe the model-driven development process for graphical user interfaces using IFML.

DISH USER INTERFACES

In the present paper we are considering SKA.DISH engineering user interfaces to be used by engineers for test, diagnostic, maintenance of DSH sub-elements, already identified and described in [1].

In particular, we have chosen to design and model LMC Engineering Interface. DSH sub-elements engineering interfaces will be accessible either directly from LMC (to be connected with keyboard/mouse and a screen) as desktop application.

LMC will provide GUIs to be used for testing and DISH control in stand-alone mode for testing, commissioning and maintenance, offering basic functionalities of DSH control & monitoring, set-up, control and testing, health monitoring, alarm management, lifecycle support, direct access monitoring in case of TM failure.

BACKGROUND CONCEPTS

Usability and Accessibility

The ISO 9241 standard Ergonomics of Human-System Interaction (ISO, 2008) defines *usability* as “the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use”, specifying:

- *Effectiveness*: the accuracy and completeness with which users achieve specified goals

† marassi@oats.inaf.it

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA095

MANAGING ARCHIVER RULES FOR INDIVIDUAL EPICS PVS IN FRIB'S DIAGNOSTICS SYSTEM*

B. S. Martins†, S. Cogan, S. M. Lidia, D. O. Omitto,
Facility for Rare Isotope Beams, East Lansing, USA

Abstract

The Beam Instrumentation and Measurements group at the Facility for Rare Isotope Beams is responsible for maintaining several EPICS IOC instances for beam diagnostics, of different IOC types, which end up generating tens of thousands of PVs. Given the heterogeneity of Diagnostics devices, the need to archive data for scientific and debugging purposes, and space limitations for archived data storage, there is a need for having per-PV (as opposed to per-Record) archiving rules in order to maximize utility and minimize storage footprint. This work will present our solution to the problem: "IOC Manager", a custom tool that leverages continuous integration, a relational database, and a custom EPICS module to allow users to specify regular-expression based rules for the archiver in a web interface.

INTRODUCTION

FRIB's Beam Instrumentation and Measurements department is responsible for a myriad of devices used in FRIB's beamline for data collection and monitoring of operational parameters, such as Allison Scanners, Profile Monitors, Cameras, Beam Position Monitors, Beam Current Monitors, among many others. All of these devices' parameters are made available to operators, scientists and engineers through the EPICS [1] framework as Process Variables (PVs). Currently, there are more than 200,000 PVs in FRIB's diagnostics system alone. Managing this many PVs across dozens of different IOCs presents numerous challenges, especially with regards to data archival: FRIB diagnostics PVs archival requirements are difficult to implement given some current limitations of the EPICS framework and of the EPICS Archiver Appliance [2]. These limitations and the solutions to them will be presented in this paper.

Archiver Configuration Enforcement

Given the limited features of the Archiver Appliance API, it is essential that there is a mechanism in place to enforce that the desired archiver configuration is in fact, in place.

Non-Regular PV Aliases

Different users of FRIB's diagnostics system expect to have different views over the same devices: controls engineers are typically interested in operational parameters of the devices themselves, while operators care about the readings and actuation capabilities that devices provide.

For example, μTCA crates used in FRIB's diagnostics system, such as the one shown in Fig. 1, can host several types of fast acquisition cards, each having a number of acquisition channels. Controls engineers might be interested in the state of the card itself, whereas operators might be interested in the readings of a particular channel attached to a particular physical device. EPICS provides an aliasing mechanism that can be used in this case to provide one PV name to engineers and a second, aliased name to operators. However, aliasing PVs in batches can be cumbersome if the aliases are not uniform.

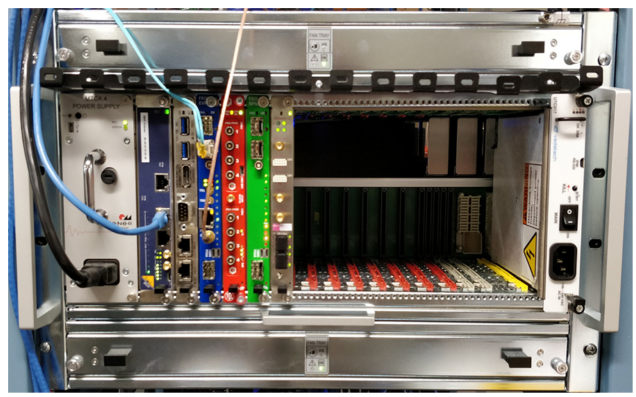


Figure 1: A μTCA crate with MCH, CPU, event receiver, Pico8, BPM and BCM cards.

Non-Regular PV Archival Policies

The second challenge faced by FRIB's controls engineers is the need for having data archival rules on a per-PV basis, rather than on a more typical per-Record basis. For instance, all ADC channels on a CAENels Pico8 picoammeter card have the same kind of EPICS Record that provides the channel readout, but the archival policy for a particular channel depends on what it is connected to: channels connected to Faraday Cups may have a different policy than channels connected to Halo Monitor Rings, for example. In other words, the archival policy must be based on a PV *alias*, not on the underlying *record*.

Centrally Managing Archiver Rules

Lastly, given the variety of device types and the sheer amount of diagnostics PVs, it is important to make it easy for controls engineers to add, modify and remove archiver rules in a central place (as opposed to in each individual IOC), as well as to allow the engineer to assess the coverage of each rule.

* This material is based upon work supported by the U.S. Department of Energy Office of Science under Cooperative Agreement DE-SC0000661, the State of Michigan and Michigan State University.
† martins@frib.msu.edu

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA096

TIMING SIGNAL DISTRIBUTION FOR SYNCHROTRON RADIATION EXPERIMENTS USING RF OVER WHITE RABBIT

T. Masuda†
Japan Synchrotron Radiation Research Institute (JASRI), Sayo, Hyogo, Japan

Abstract

In synchrotron radiation experiments, measurements such as nuclear resonant scattering, time-of-flight, and time-resolved measurements require an RF clock and fundamental revolution frequency (zero-address) signals synchronized with a storage ring accelerator as a light source. At SPring-8, these timing signals are delivered directly over dedicated fibers and cables from an accelerator timing station to each experimental station. Considering the upcoming internet of things (IoT) era, it is preferable to distribute these signals over a network using state-of-the-art digital technology. Consequently, I am building a proof of concept system (PoCS) that aims to achieve distribution of the accelerator RF clock (508.58 MHz at SPring-8) and the zero-address signals synchronized with the storage ring using RF over White Rabbit (RFoWR). The PoCS consists of a master node, which receives the RF clock and the zero-address signals from the accelerator, and a slave node, which reproduces these timing signals near experimental stations. Each node employs a Simple PCI Express FMC Carrier (SPEC) board and a new FMC DAC 600M 12b 1cha DDS (FMC DDS). The slave node generates the synchronous clock with the specified division rate and phase shift. This paper describes the achieved functions and performance of the PoCS.

MOTIVATION

Some measurements in synchrotron radiation experiments require high-precision timing signals synchronized with a storage ring (SR) accelerator, which is a light source for the experiments. These measurements include nuclear resonant scattering, time-of-flight, and pump and probe time-resolved measurements using laser pulses.

At SPring-8, a 508.58 MHz RF clock (F_{RF}) of the 8 GeV SR and a 208.8 kHz fundamental revolution frequency signal (F_{rev0}) called a “zero-address signal” are delivered using dedicated long-distance cables from the accelerator timing station to each experimental station. This requires precise timing signals. Here, 208.8 kHz is derived from F_{RF} divided by 2,436, which is a harmonic number of the SR.

To use these signals at the experimental station, long RF cables and electronics, such as divider modules and delay modules, must be deployed. These installations are costly and require experts to adjust the timing. In addition, these timing signals are available for very limited beamlines because use of the dedicated fibers/cables restricts expandability of the signals. To maximize experimental results, the precise timing signals must be easy and convenient to utilize. Considering the upcoming IoT era, it is preferable that these signals will be distributed as digital information over a network by using state-of-the-art digital technology.

RF over White Rabbit (RFoWR) technology [1-2] exactly fits this purpose. Considering there are some practical application plans using RFoWR, such as plans at CERN SPS and ESRF EBS [3], I will conduct timing signal distribution for synchrotron radiation experiments at SPring-8.

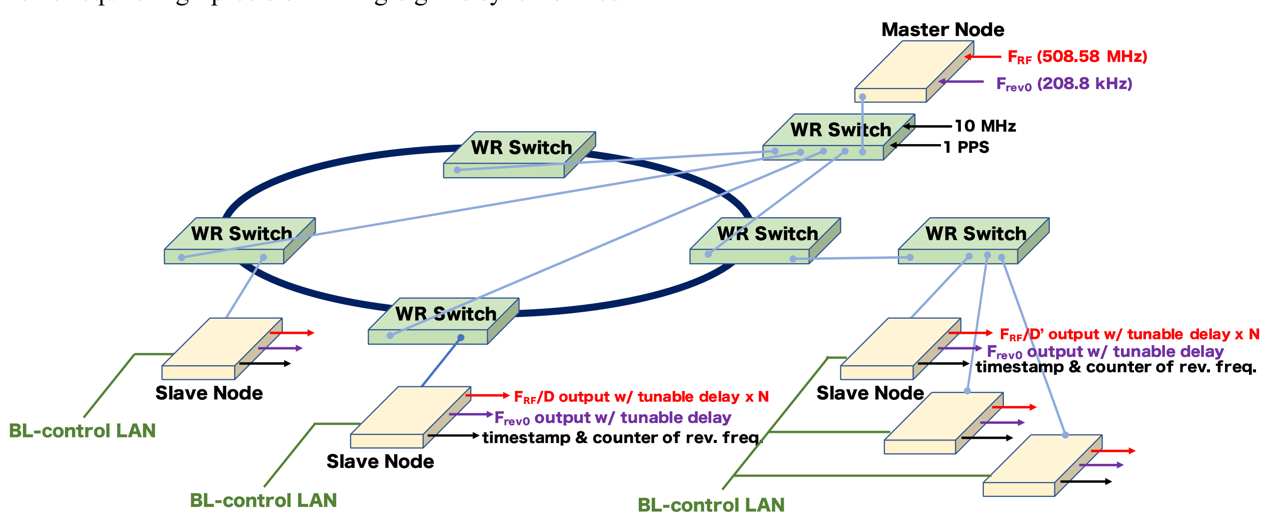


Figure 1: Conceptual design of new timing distribution system using RF over White Rabbit technology.

† masuda@spring8.or.jp

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA097

DEVELOPMENT OF A TANGO INTERFACE FOR THE SIEMENS-BASED CONTROL SYSTEM OF THE ELETTRA INFRASTRUCTURE PLANTS

P. Michelini, I. Ferigutti, F. Giacuzzo, M. Lonza, G. Scalamera, G. Strangolino, M. Trevi
Elettra-Sincrotrone Trieste S.C.p.A., Trieste, Italy

Abstract

The control system of the Elettra Sincrotrone Trieste infrastructure plants (cooling water, air conditioning, electricity, etc.) consists of several Siemens PLCs connected by an Ethernet network and a number of management stations running the Siemens Desigo software for high-level operation and monitoring, graphical display of the process variables, automatic alarm distribution and a wide range of different data analysis features. No external interface has been realized so far to connect Desigo to the Elettra and FERMI accelerator control systems based on Tango, making it difficult for the control room operators to monitor the conventional plant operation and parameters (temperature, humidity, water pressure, etc.), which are essential for the accelerator performance and reliability. This paper describes the development of a dedicated Desigo application to make selected process variables externally visible to a specific Tango device server, which then enables the use of all the tools provided by this software framework to implement graphical interfaces, alarms, archiving, etc. New proposals and developments to expand and improve the system are also discussed.

INTRODUCTION

Four years ago an Elettra internal project was approved to carry out a general survey on the infrastructure plants acquisition systems, installed sensors, actuators and control software. The first step was to create a plant data base and to optimize the process control software written in several years running on non-homogeneous hardware. The idea was to provide the laboratory Energy Manager with detailed information about the energy fluxes in order to evaluate whether and where to intervene for their possible optimization. From further analysis it became evident that this was not enough. In fact, plant data are necessary also for operators and machine physicists to properly manage the accelerators. A way to exchange information between infrastructure control systems and machine control systems was required to improve the accelerators reliability, implement preventive maintenance and allow for potential energy savings. The final decision was to develop an unified management of the plant monitoring and regulation systems, managing all infrastructure Programmable Logic Controllers (PLC).

Scenarios

Built in different years, two infrastructure plant control systems are operating at Elettra, one for the Elettra synchrotron light source and the other for the FERMI Free Elecron Laser. The first one was based on Siemens S5 PLCs with Citect [1] Supervisory Control And Data Acquisition (SCADA) system for the operator console software. The second one relies on new Siemens S7 PLCs with Desigo Insight [2] supervision for higher-level operation and monitoring, graphics-based display of the process, automatic alarm generation and data analysis.

As first step, we decided to integrate the old Elettra plant control system supervision into Desigo. For this purpose, we had to upgrade the hardware with new Siemens PLCs, porting the control software on them and creating new graphic pages on Desigo. This process is still running.

Furthermore, we had to create a way to exchange data between the Desigo world and the accelerator control systems based on Tango, in order to make process variables visible externally and exploit the tools provided by this software framework.

A Practical Example

The production, distribution and use of energy flows in the infrastructure plants have to be continuously monitored to ensure the proper functioning conditions of the two accelerators.

Each power distribution panel is equipped with an electricity meter. The main hot and cold water flows are controlled by flow meters.

Hot and cold water and electric energy are supplied by two trigeneration plants. This kind of plants burn methane in a combustion engine, which produces electricity. Its high temperature fumes are used to generate hot water for users and for the absorption refrigerators (see Fig. 1). An exhaustive control of the distribution system allows to anticipate possible machine downtime and to intervene directly on the identified breakdown.

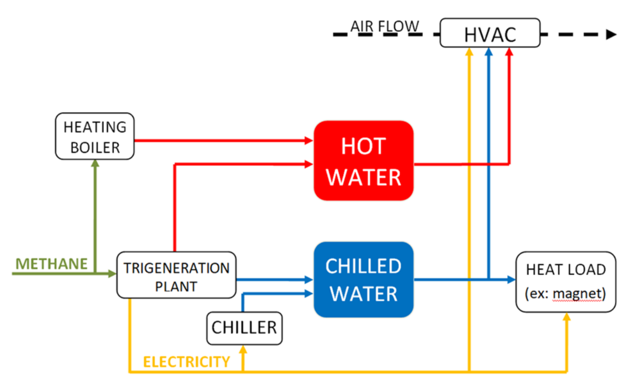


Figure 1: Energy block diagram.

DESIGO ARCHITECTURE

The conventional plants are controlled by Siemens Desigo Insight 6.0, a Building Automation and Control System (BACS) provided with functions such as alarm management, time scheduling and trend logging.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA099

XLEAP-II MOTION CONTROL

M. A. Montironi, H. Bassan, M.A. Carrasco, E.M. Kraft, A. Marinelli
SLAC National Accelerator Laboratory, Menlo Park, USA

Abstract

The XLEAP project was conceived with the main scope of extending the generation of ultrashort pulses at LCLS to the sub-femtosecond (sub-fs) regime. As the project produced the expected results, an upgrade called XLEAP-II is being designed to provide the same functionality to LCLS-II. The XLEAP project utilized one variable gap wiggler to produce sub-fs X-ray pulses. The upgrade will involve four additional wigglers in the form of repurposed LCLS fixed gap undulators mounted on translation stages. This paper describes the design of the hardware and software architecture utilized in the motion control system of the wigglers. First it discusses how the variable gap wiggler was upgraded to be controlled by an Aerotech Ensemble motion controller through an EPICS Soft IOC (Input-Output Controller). Then the motion control strategy for the additional four wigglers, also based around Aerotech controllers driving servomotors, is presented. Lessons learned from operating the wiggler and undulators during LCLS operation are discussed and utilized as a base upon which the upgraded motion control system is designed and built. Novel challenges are also identified and mitigations are discussed.

INTRODUCTION

The X-Ray Laser-Enhanced Attosecond Pulse (XLEAP) Generation project was conceived with the main scope of extending the generation of ultrashort pulses at LCLS to the sub-femtosecond (sub-fs) regime for energies up to 1.25 keV. The technique is based on the concept of shaping the LCLS electron beam with a high-power infrared (IR) laser whose energy is modulated by a variable gap wiggler. The experimental setup developed for the XLEAP project in order to achieve such short pulses is shown in Figure 1 [1, 2].

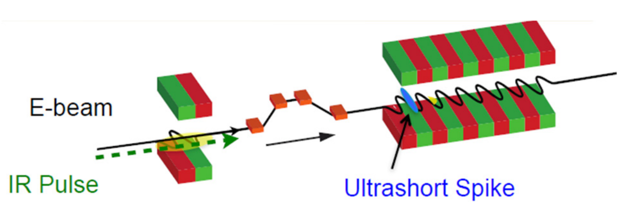


Figure 1: XLEAP experimental setup. The combined effect on the electron beam of the high energy IR pulse, of the magnetic chicane, and of the wiggler produce an ultrashort current spike [2].

As the project produced the expected results [3], an upgrade called XLEAP-II is being designed to provide the same functionality to LCLS-II. Differently than XLEAP though, in this case energy modulation of the electron beam will be induced in four wigglers built by modifying decommissioned LCLS fixed gap undulators. The existing variable gap wiggler utilized for XLEAP will be reused as a space-charge booster after a magnetic chicane. Figure 2, taken from the LCLS Strategic Development Plan [2] is a schematic representation of the XLEAP-II experimental setup.

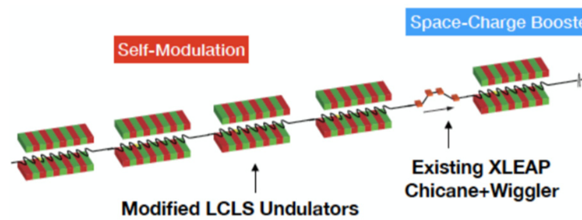


Figure 2: Schematic representation of XLEAP-II experimental setup. The four repurposed LCLS undulator, located upstream of the XLEAP chicane and wiggler provide energy modulation to the electron beam [2].

Motion control is a fundamental component of the experimental setup described in this section. The four repurposed undulators are in fact mounted on translation stages used to control their insertion level into the beamline. Gap control is required for the wiggler in order to modulate beam at different frequencies. The rest of the paper describes the design of the hardware and software architecture utilized in the motion control system of the wiggler and of the repurposed undulators.

MOTION CONTROL SYSTEM

Variable Gap Wiggler

The variable gap wiggler, shown in Figure 3 has a period of 33 cm and a gap adjustable between a minimum of 8 mm and a maximum of 200 mm. Gap actuation is achieved through two Slo-Syn (MH112 series 15-Amps) stepper motors, one located at the upstream end and one at the downstream end. A passive motion transmission system couples the top and bottom jaws. In order to maximize the reliability of the system through redundancy, two independent position feedbacks are implemented for each axis. The primary feedback system is provided by AMOSIN BiSS-C radiation hardened absolute linear encoders measuring the upstream and downstream gap. The secondary feedback system consists of incremental rotary encoders made by Gurley and positioned directly on the motor axis. The motion controller chassis, based on two Aerotech CP20 drives, was developed to control the wiggler during the XLEAP experiment and will be reused. The controller allows to drive the two motors in a coordinated manner thus minimizing the risk of tapering the device. Moreover, it provides dedicated inputs for the primary and secondary

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA101

VR AS A SERVICE: USE OF VIRTUAL REALITY IN A NUCLEAR ACCELERATOR FACILITY

L. Pranovi, M. Montis, INFN-LNL, Legnaro, Italy

Abstract

A nuclear plant, for energy or for nuclear physics, is a complex facility where high level security is mandatory, both for machines and people. But sometimes the status of danger is not correctly felt, inducing workers to misinterpret situations and, as consequence, not act in the best way. At the same time problems related to area accessibility can occur during normal machine operations, limiting actions related to local maintenance and environment supervision.

It would be suitable to have the opportunity to perform these tasks independently from environment limitations and machine operations. In order to overcome these limits, we applied Virtual Technology to the nuclear physics context. As consequence, this new tool has given us the chance to reinterpret concepts like training or maintenance planning. In this paper the main proof of concept implemented are described and additional information related to different VR technology usages are exposed.

VIRTUAL REALITY TECHNOLOGY IN THE RESEARCH

A nuclear accelerator laboratory can be described as a complex system composed by several functional sub-systems (vacuum system, cryogenic plant, control system, RF system, etc.) which have to be coordinated properly in order to provide the service to the end user. Every single system has to work correctly and requires a big effort in continuous maintenance in order to guarantee a stable beam for experiments. In addition, every single operator has to be fully trained to work and control every single part of the accelerator. Because of the nature of the experiments and the kind of facility, different problems and limitations related to area accessibility occurred during and after operations. These limitations cause delays and uncertainty in planning process at every level (management, logistics and operations). It would be suitable to get the access to all the restricted areas regardless limitations and machine operations. The aim is to find a method and a tool to overcome these constraints through a real representation of the area, allowing regular operations in the facility. Virtual Reality (VR) Technology is an interactive computer-generated experience taking place within a simulated environment, that incorporates mainly auditory and visual, but also other types of sensory feedback. It is rapidly diffusing among several different professional environments, such as medical, architecture, military and industry, with different level of interactions, based on the experience required.

For the developer, the VR technology is a powerful tool which allows to reproduce environments and objects with a high level of detail.

According to the actual standards and applications available on the market, we tried to design and implement a VR experience dedicated to the nuclear facilities such as linear accelerators.

CASES OF INTERESTS

This kind of technology can be adopted to help different users (managers, developers, operators) in different critical tasks. Formally, based on the principal characteristics of a VR experience (photorealism, immersion, interactivity, real time), different types of experiences can be designed and implemented, depending on the goal the application wants to achieve.

Through the experience matured in our daily work, it has been possible to focus and define three particular macro areas of interest which has been used for the first prototype of the VR application:

- Training: one of the biggest challenges in physical laboratory is having all operators properly trained to work with the accelerator machine and ancillaries. As technology evolves and new solutions are continuously implemented into the facility, every single operator has to be cyclically trained. This can be done in a secure simulated environment, independent of the physical machine. In this scenario, the trainer can also simulate emergencies and evaluate behaviour feedback, estimating the response time in a hazardous environment.
- Machine operations and maintenance: using the same virtual environment provided for the training, it is possible to evaluate and prepare maintenance planning (ordinary and extraordinary) and machine upgrades. As the entire environment is rebuilt starting from CAD models, the final 3D virtual model guarantees sub-millimetre resolution where VR users can operate in the virtual simulation to evaluate, for example, device positioning for machine upgrades.
- Data collection: the virtual model is defined starting from 3D CAD models. As different groups manage the different sub-systems constituting the facility, the tool can be very useful to test and verify data integrity and coherence. This aspect results in reducing design errors and optimizing communication and data exchange among the groups.

CONTROL PROCESS AND ITS INTEGRATON

An important aspect kept in mind during the preliminary study and design for the test-bench application was how to integrate a VR solution into the normal common design and organization process: users have their own set of tools and methodology to execute their own tasks, and these two

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA102

A SOFTWARE SUITE FOR THE RADIATION TOLERANT GIGA-BIT TRANSCEIVER - SLOW CONTROL ADAPTER

P. Moschovakos*, P.P. Nikiel, S. Schlenker, CERN, Geneva, Switzerland
H. Boterenbrood, Nikhef, Amsterdam, Netherlands
A. Koulouris, National Technical University of Athens, Athens, Greece

Abstract

The future upgrades of the LHC (Large Hadron Collider) will increase its luminosity. To fulfil the needs of the detector electronic upgrades and in particular to cope with the extreme radiation environment, the GBT-SCA (Giga-Bit Transceiver - Slow Control Adapter) Application-specific integrated circuit (ASIC) was developed for the control and monitoring of on-detector electronics. To benefit maximally from the ASIC, a flexible and hardware interface agnostic software suite was developed.

A hardware abstraction layer - the *SCA software package* - exploits the abilities of the chip, maximizes its potential performance for back-end implementations, provides control over ASIC configuration, and enables concurrent operations wherever possible. An OPC UA server was developed on top of the *SCA software library* to integrate seamlessly with distributed control systems used for detector control and Trigger/DAQ (Data AcQuisition) configuration, both of which communicate with the GBT-SCA via network-attached optical link receivers based on FPGAs.

This paper describes the architecture, design and implementation aspects of the *SCA software suite* components and their application in the ATLAS experiment.

SCA SOFTWARE SUITE CONTEXT

Introduction

The GBT-SCA, or SCA for short, is a radiation-tolerant ASIC and part of a chip-set of the GBT project, in which a bi-directional 4.8 Gbps optical link architecture has been developed using a SEU robust protocol [1], providing simultaneous transfer of readout data, timing and trigger signals as well as slow control and monitoring data, by multiplexing multiple logical electrical data links of 80, 160 or 320 Mbps, called E-links [2] onto a single optical link using the rad-tolerant GBTX ASIC on the front-end side. The SCA's purpose is to interface to control and monitoring signals of front-end electronics on the detectors, using two redundant 80Mbps E-links to connect to a GBTX.

The SCA employs the HDLC (High-level Data Link Control) protocol on its E-links in a synchronous request-reply communication pattern. On top of the HDLC data link layer, a custom protocol has been implemented to address the different hardware devices (or channels) present on the SCA chip.

The SCA has 16 independent I^{2}C (Inter-Integrated Circuit) serial bus masters, an SPI serial bus master, a JTAG serial bus, 32 GPIO ports (General Purpose I/O), an ADC with 31 analogue inputs, an embedded temperature sensor and 4 independent DAC. The SCA request and reply message layout is shown in Fig. 1

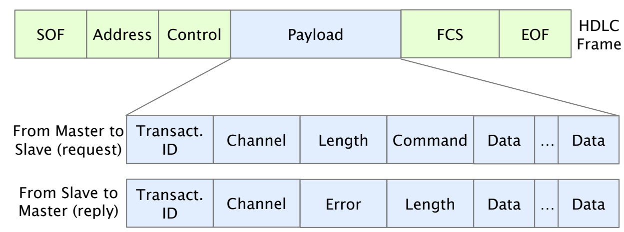


Figure 1: The SCA request and reply message layout.

The channels operate independently from each other in order to allow concurrent transactions[1] and perform concurrent transfers from their end-devices [3].

Functionality and Requirements

The software package is required to provide a high level of abstraction and means for interfacing to all communication channels of an SCA profiting from the hardware parallelism in-between independent channels. In order to ensure reliability, the software needs to do the necessary bookkeeping for the synchronous communication and transaction tracking.

Moreover, the software is required to achieve high performance and low latency, including features like grouping of requests to perform lengthy operations, such as field-programmable gate array (FPGA) programming, requiring transfer of large amounts of data over JTAG. Since thousands of SCAs will be used in the detector systems, scalability is an important design aspect. At the same time, monitoring and control tasks require a high availability of close to 100%, implying the need of a high level of robustness.

Finally, the software needs to adapt to different communication scenarios of the SCA. A simulated chip needs to be supported as well as prototype board communication interfaces for development and testing purposes. For the final production system, the *SCA software* is interfaced with the optical link receiver system - FELIX [4, 5] via a dedicated communication link called netIO [6].

Integration Overview

Figure 2 shows an overview of the SCA integration chain, illustrating the interplay of the *SCA software suite* components.

* paris.moschovakos@cern.ch

[1] Data are gettiing serialized in the physical line, while HDLC sequence number is used to keep the traffic in order when the data are ready to be transmitted

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA103

BACKWARD COMPATIBLE UPDATE OF THE TIMING SYSTEM OF WEST

A. Barbuti, G. Caulier, Y. Moudden*, T. Poirier, B. Santraine, B. Vincent and the WEST Team†
CEA, IRFM, Saint Paul Lez Durance, France

Abstract

Between 2013 and 2016, the tokamak Tore Supra in operation at Cadarache (CEA-France) since 1988 underwent a major upgrade following which it was renamed WEST (Tungsten [W] Environment in Steady state Tokamak). The synchronization system however was not upgraded since 1999. At the time, a robust design was achieved based on AMD's TAXI chip : clock and events are distributed from a central emitter over a star shaped network of simplex optical links to electronic crates around the tokamak. Unfortunately, spare boards were not produced in sufficient quantities and the TAXI is no longer available. Designing replacement boards provides an opportunity to investigate new clock and data recovery (CDR) solutions and extended functionalities However, backward compatibility is a major constraint given the lack of resources for a full upgrade of the synchronization network and electronics. This contribution reports on the implementation of a custom CDR in full firmware, using the IOSerDes of Xilinx FPGAs. Preliminary results on Xilinx development boards are provided.

INTRODUCTION

Making the most of information gathered or distributed over a large network of sensors and actuators for measurement and control requires some shared sense of simultaneity. In some cases, it is provided by sufficiently frequent external events simultaneously observable over the entire array and on which all measurements can be readjusted in time. Most often, a facility's timing system is in charge of generating and propagating complementary *internal* events to all its subsystems in order to achieve the specified rigidity of the shared time frame in terms of accuracy, precision and offset of the local time on remote nodes with respect to master time. As an example, the remarkable state of the art White Rabbit network [1] provides a generic scalable solution to synchronize hundreds of nodes with better than nanosecond accuracy over up to10 km optical fibers. Many other different solutions have been engineered to closely match the synchronization needs of specific physics experiments with very different sizes, granularity and constraints ([2–7]). And the trend is set by the upcoming high luminosity runs at CERN which demand for even greater accuracy on the order of a few tens of picoseconds [8].

A common feature of many such synchronization solutions is the use of fast serialisers and deserialisers either as individual chips or as specific functional blocks inside modern FPGAs. Deserialisation is possible only if a clock signal is propagated along with the data telling where to sample and read individual data bits. At higher transmission rates, a very tight phase relationship between data and clock is required and is obtained thanks to proper digital encoding of the data stream acting as a modulation of the carrier clock frequency. This serves well the purpose of synchronization: the carrier clock embedded in the high speed serial stream of data words is distributed and recovered in the physical layer on each remote node hence providing means to increment local counters at a common rate and to send specific synchronous command words for instance to reset these counters, with either known or measurable latency. Last upgraded in 1999 [9], the timing system of WEST (WTS) was designed based on these ideas and AMD's TAXI receiver and emitter chips [10] used for serial communications up to 17.5 Mbyte/s. Unfortunately, the TAXI chip was discontinued shortly after and spare boards had not been produced in sufficient quantities with respect to the long life span of the Tokamak. While a functionally compatible circuit is available from Cypress [11], the fact that serial communication standards have now reached multigigabit rates reasonably questions the future availability of such low rate SerDes. The obvious need to design replacement boards in order to ensure continued operation of the WTS during the next ten years, is also an opportunity to investigate new clock and data recovery (CDR) solutions and extend current board functionalities (e.g., error detection, measurements, etc.) using up-to-date FPGAs and embedded resources.

Backward compatibility is however a major design constraint given the span of the network, the lack of resources for a thorough upgrade and the fact that WEST is in operation. Hence state of the art timing solutions based on high speed multigigabit serializers [1] such as provided in latest generation FPGAs are not applicable. The following sections describe the current state of the timing network of WEST and plans for gradual conservative upgrades. Next, the implementation of a custom replacement CDR in full firmware is discussed. Finally, preliminary results obtained with prototype firmware and software on Xilinx FPGA development boards are presented.

WEST TIMING SYSTEM

Current Status

A sketch of the timing infrastructure of WEST (WTS) is given in Fig. 1: events and clock are distributed from a central TAXI emitter board over a star shape network of simplex optical links to remote receivers in electronic crates all around the tokamak for control, safety and data acquisition. The TAXI receiver includes a CDR circuit to recover the

* yassir.moudden@cea.fr
† http://irfm.cea.fr/WESTteam/

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA104

IT INFRASTRUCTURE FOR SAFETY SYSTEMS DEVELOPMENT AT ESS

S. Armanet*, R. Mudingay†, European Spallation Source, Lund, Sweden

Abstract

The Control System Infrastructure team has deployed a dedicated isolated environment to support Safety Systems development at ESS.

We have tried to take advantage of our standardised infrastructure components for controls like virtualization, centralized storage, system orchestration and software deployment strategy. Because we already have all these components in place for our Control System IT infrastructure we have decided to treat engineering workstations as disposable components in an isolated and dedicated virtualized environment. We have designed the environment to control who and when users can access the development environment, from which device, to which workstations and what they can run in this environment.

INTRODUCTION

Safety systems design and development are critical parts of ESS development and operation phases. They are controlled and reviewed by the Swedish nuclear safety authorities and a priority in our cyber security plans. The development process has to follow a well defined security plan and risk assessment and the development environment has to follow these rules. Basically these rules are to answer these questions:

- who can access the development environment?
- when can these users can access it?
- from which devices?
- to which engineering workstations?
- to do what?

In the ICS infrastructure team we have decided to implement engineering workstations has disposable workstations so that we simplify the requirements on managing Windows workstations. These workstations run on a network that is completely stand alone and isolated from any other network.

The only way to access them is using the Remote Desktop Protocol from controlled clients. This environment is fully virtualized which permit us to achieve these goals in a flexible and automated way.

We provide all required tools to program safety systems in a controlled way and we are evaluating a way to interact with external hardware from this environment by using dedicated transfer workstation with ad hoc controls and restrictions.

SYSTEM DESCRIPTION

Infrastructure

Because our orchestration process allows us to easily target where we want to deploy virtual machines, we have

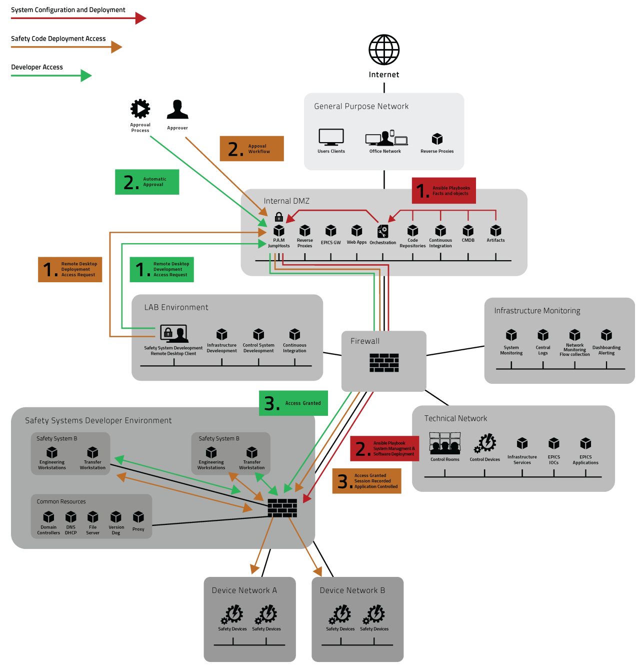


Figure 1: System description.

decided to split our infrastructure (see Fig. 1) in different virtualization cluster and network layers [1].

This way we can easily adapt each cluster to its specific use case and manage network connectivity specifically for each cluster. We use a simple open source Virtualization solution for our clusters and our orchestration tool talk to it's API to deploy virtual machine. This is to avoid complex virtualization solution and keep the management of different cluster easy within a small team.

The virtualization solution supports multi-user, distributed management, 2 factor authentication, High availability and live VM and storage migration. VM location are defined in our CMDB based on network membership, functional group membership or host based definition. Our clusters are composed of at least 3 nodes for redundancy, maintainability and high availability.

We have a shared storage back-end.At the moment our storage is based on replicated ZFS NAS. We use ZFS snapshots send/receive as backup tool for safety systems.

Our plan is to deploy our clusters across 3 data centres/server rooms to avoid split brain scenario. Our storage cluster will also be redundant across these 3 server rooms.

We will share a common storage cluster but with dedicated pools for each environment.

Orchestration

We use Ansible as automation tool. Our roles and playbooks are hosted on an internal gitlab repository and use our CMDB (CSentry) as dynamic inventory.

* stephane.armanet@esss.se
† remy.mudingay@esss.se

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA105

BEAM SYNCHRONOUS DATA ACQUISITION USING THE VIRTUAL EVENT RECEIVER

G. Mun, W. W. Lee, G. Kim, J. Hu, C. B. Kim, H. S. Kang
Pohang Accelerator Laboratory, Pohang, Korea

Abstract

The 4th generation light source, PAL-XFEL, is an X-ray free electron laser in Pohang, Korea. One of key features of the event timing system in the PAL-XFEL, the beam synchronous acquisition is used in many beam diagnostics and analysis and the species of that increase gradually. In order to reduce the cost for event receivers which are required for operating the beam synchronous acquisition and to resolve the difficulty of the limited platform dependant on event receivers, we developed the virtual event receiver system receiving timestamps and BSA information from an event generator not using real event receivers. In this paper, we introduce the software architecture of the virtual event receiving system and present test results of it.

BEAM SYNCHRONOUS ACQUISITION

Event timing system is one of the key infra systems to control accelerators, which basically controls triggers for each instrument and system in an accelerator system. The event timing system generally consists of event generators and event receivers. The event generator, EVG, generates event definitions which are scheduled in a time sequence according to the operation policy in an accelerator system. The event receiver, EVR, generates trigger signals for clients such as modulators, instrument for diagnostics, and low level RF controllers based on the information transferred from the EVG. This event information includes each timestamp corresponding to each event. The event timing system of the PAL-XFEL consists of the hardware implemented by the MicroResearch Finland Co. and the software developed for the LCLS [1-3].

The event timing system of the PAL-XFEL supports the beam synchronous acquisition, BSA, which is useful for analysing the correlations among several client devices by collecting data in terms of same beam pulse as realized in the LCLS [4]. Figure 1 shows a schematic drawing to describe the concept of BSA by using the EVG and EVRs. Each client input output controller, IOC, has an EVR which receives event pattern information from the EVG IOC.

Figure 2 shows the orbit correction system used in PAL-XFEL. This orbit correction system uses the real-time beam position data taken in BSA by the beam position monitors. The BSA is also used to measure beam profiles by using the wire scanners in the PAL-XFEL, which is especially essential to correct the jitter of the raw data from the wire scanners.

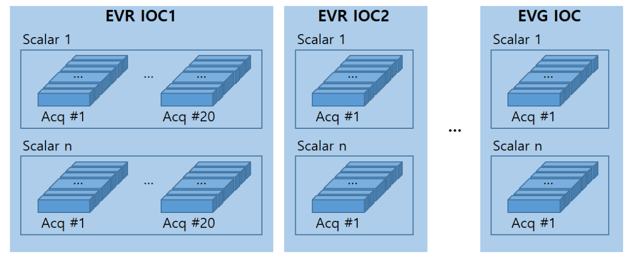


Figure 1: Data acquisition among IOCs in an event timing system using EVRs and an EVG.

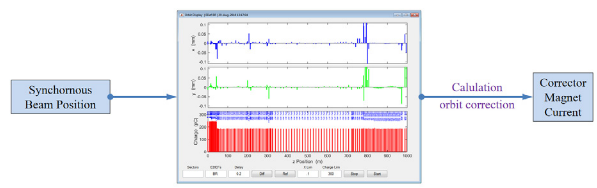


Figure 2: Orbit correction system using BSA to take synchronous beam positons with the beam position monitors in the PAL-XFEL.

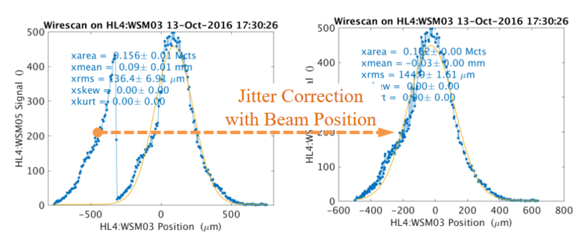


Figure 3: Typical example of beam profile measurement with wire scanner using BSA.

VIRTUAL EVENT RECEIVER

The EVR has three major roles in the event system of the PAL-XFEL. The first one is to generate trigger signals. The generation of trigger signal are processed by the FPGAs embedded in the EVR. The other functions are to transfer the information related with the timestamp and BSA from the EVG as mentioned before.

The actual works to generate timestamps and taking data in BSA can be processed by the CPU of an IOC. The EVRs are simply interconnect the information related with the events between the EVG and the client IOCs or the processors. Therefore, if we realize the function of delivering information related with BSA and timestamp by using a separated simple micro-computer and if the client system can make the trigger signal based on the received event data

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA108

MODERNIZATION PLANS FOR FERMILAB'S ACCELERATOR CONTROL SYSTEM

Dennis J. Nicklaus, Fermilab, Batavia, USA

Abstract

The control system, ACNET, for Fermilab's accelerator complex has enabled the lab's scientific mission for decades. ACNET has evolved over the years to incorporate new technologies. However, as Fermilab prepares to enter a new era with its PIP-II superconducting linear accelerator, ACNET is at a crossroads. There are several components that are either obsolete or outdated, or certainly will be over the long lifetime of PIP-II. We have begun a plan to modernize our accelerator control system. This paper discusses some of the obsolete hardware and software that needs to be replaced and lays out options and technologies that we might adopt as part of this modernization effort.

BRIEF HISTORY

Fermilab's control system, ACNET [1], was developed in the early 1980s for the Tevatron. It was developed as a three-tied system, having 1) front-end systems responsible for the data acquisition from field busses, 2) a central services layer featuring a centralized relational database containing information about all the devices connected through the control system, and 3) the user interface and application layer. A custom, connectionless UDP datagram protocol supports the communication between the computer nodes in the control system. This architecture has proven very resilient over the decades, reliably supporting the physics program of the lab as the set of accelerators and beam lines has grown more complicated over the decades, growing through the Main Injector and Recycler era, to our current set of neutrino, muon, and other fixed target experiments.

MODERNIZATION NEEDS

Hardware and software need updating at all three tiers of the control system. There are a variety of updates that we are beginning to plan, and these can range from a simple upgrade to a more modern computer, to a complete redesign of a particular sub-system.

Front-end and Field Bus

Many of the front-end data acquisition systems are VME-based. Most of these run under the VxWorks operating system, but a substantial number (125) of general purpose I/O units in our linac and elsewhere, known here as IRMs, run PSOS as their operating system. There is very little experience in PSOS at Fermilab (or anywhere), so moving off that platform is a priority. But we would also like to begin moving away from VxWorks to Linux. VxWorks has served our needs well, but with the evolution of processor speed and Linux real-time extensions, we no longer think that VxWorks is providing much benefit for its cost.

Inside these VME crates, we have a substantial number of processor cards based on the 68040 CPU, which came out in 1990. Obviously, upgrading these obsolete processors is a priority. Continued support of these older processor cards means we have to make compromises in rolling out features for our front-end software framework. Beyond the very ancient 68040 processors, we have several systems that rely on end-of-life cards such as the MVMEV2400 series cards. Since we need to upgrade these old processor cards, we are also evaluating other bus architectures to replace the VME bus with something such as MicroTCA.

One priority in the modernization effort is to replace our CAMAC hardware. Fermilab has a lot of CAMAC data acquisition and control hardware. This technology (dating to the 1970s) has not been commercially available for approximately 20 years, and we can no longer buy chips and parts for our custom-built cards. Some of the CAMAC cards could be replaced by general purpose I/O available in PLCs. We have prototype implementations of replacements for some other CAMAC functionality but need to deploy multiple instances across the accelerator complex.

Central Services

One of the key features of ACNET is its central database for information about all the devices and nodes in the control system. The central database is implemented with the Sybase relational database product. The Fermilab accelerator controls department is currently working on replacing Sybase with the open source Postgress.

On the hardware side of our central services we also have aging computing infrastructure. The cluster of computing nodes that run the core software and the operations consoles and user applications are due for replacement.

Applications and User Interface

Our core ACNET user interface is based on X-Windows. While highly functional, this results in a somewhat dated look and feel for our interface, and we need to be cognizant of the possibility that X-Windows may not be supported by the Linux community indefinitely. Since we have over 500 custom applications, porting them to any new technology is a significant effort

Power Supplies

High power (20kW to 500kW) power supplies are another example of our infrastructure that is aging and needs replacement. We have 181 of these power supplies that are over 35 years old. Here again, we can no longer buy parts for these critical accelerator components. These were built on 1960s and 1970s engineering standards and one reason for upgrading them is to get more modern designs with up-

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA112

DATABASE SCHEME FOR ON-DEMAND BEAM ROUTE SWITCHING OPERATIONS AT SACLA/SPring-8

K. Okada†, N. Hosoda, T. Ohshima, T. Sugimoto, M. Yamaga,
Japan Synchrotron Radiation Research Institute, Hyogo, Japan
T. Fujiwara, T. Fukui, H. Maesaka, T. Maruyama, T. Okada,
RIKEN SPring-8 Center, Hyogo, Japan
O. Morimoto, Y. Tajiri, SPring-8 Service Co. Ltd., Hyogo, Japan

Abstract

At SACLA, the X-ray free electron laser (XFEL) facility, the electron linac operates in time-sharing (equal duty) mode between beamlines. The upgrade plan of the facility includes varying the duty factor on an on-demand basis and bringing the beam into the SPring-8 storage ring. Low-emittance beams are ideal for the next generation storage ring. In every 60 Hz repetition cycle, we must deal a bunch of electrons to each beamline properly. The challenge here is we must keep the beam quality for the XFEL demands while responding occasional injection requests from the storage ring. This paper describes the database system that supports both SACLA/SPring-8 operations. The system is a combination of RDB and NoSQL databases. In the on-demand beam switching operation, the RDB maintains the parameters to define sequences, which include a set of 1-s route patterns and a bucket sequence for the injection. Data analysis is a post-process to build an event for a certain route, because not all equipment receives the route command in real time. This paper presents the preparation status toward the standard operation for beamline users.

INTRODUCTION

The upgrade plan of SPring-8 is to build a fourth-generation ring-based synchrotron radiation light source [1]. The role of SACLA will be to provide high quality electron beams as an injector. In the top-up mode, SACLA is expected to respond to requests from the storage ring while running as a source of XFEL beam lines. This route switch must be operated in a selected bunch of the accelerator's 60 Hz cycle. After the injection into the storage ring, characteristics of electron bunches must be returned back to meet the demand of XFEL emission. In addition to the kicker magnet for routing, the RF units must be tuned to set energies and bunch shapes accordingly.

Though the upgrade project is still in the planning stage, realizing SACLA as an injector to the current SPring-8 storage ring has benefits which include reductions in the electric power consumption and the maintenance cost of the current dedicated injector apparatus.

Restructuring of the accelerator control system to enable a unified operation of SACLA and SPring-8 [2] is mostly established. In this paper, we describe the scheme of database at the site with a focus on the beam route switching operation.

†k.okada@spring8.or.jp

DATABASE AT THE SITE

Overall Scheme

The database at SACLA/SPring-8 supplies a platform for operating the accelerators. Instead of having various independent systems, all subsystems are to follow a common rule in the data storage and this allows the simplification of the correlation analyses. A schematic view of the data stream from a viewpoint of databases at the site is presented in Figure 1. Parameter DB manages signal information that are read by each DAQ component before data logging, and DAQ components push the data to Online DB. Operations of slow feedback control often use Online DB as the data source. The data in Online DB are eventually moved to Archive DB with a size reduction process. Parameter DB also stores various operation setup parameters and calibration factors in an organized format.

The unified data format allows for the provision of a global alarm service. The alarm service monitors signals periodically and compared with a simple set point. The results are stored in Parameter DB.

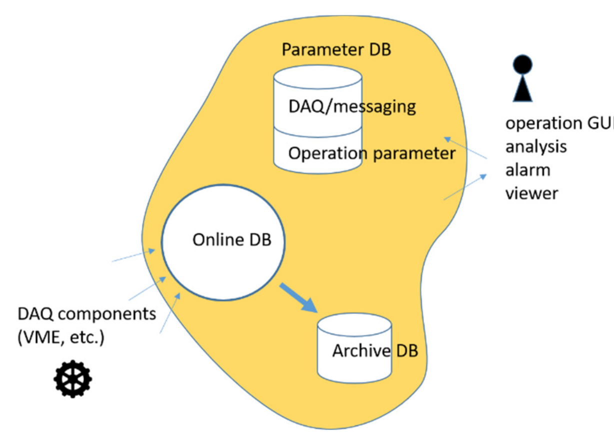


Figure 1: Data stream from the viewpoint of a database.

Parameter Database

Parameter DB is the entry point of every application that need to access data. It is so important that a limited number of librarians control updates of contents. Users who add new equipment will submit a request to register it onto the database.

DAQ/messaging: Parameter DB has a list of signals for the DAQ and messaging. The signals are for equipment and are sorted by group/subgroup categories. Each host

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA113

EPICS MAINTENANCE TOOLS AND PRACTICES AT FRIB'S DIAGNOSTICS DEPARTMENT*

D. O. Omitto†, S. Cogan, B. S. Martins
Facility for Rare Isotope Beams, East Lansing, USA

Abstract

The Beam Instrumentation and Measurements department is responsible for dozens of different diagnostics devices deployed at multiple locations at the Facility for Rare Isotope Beam. To manage such a high number of devices, different tools were created to address preventive and corrective maintenance tasks and check the overall health of the equipment. This work will present how the EPICS tools and frameworks, such as archiver, channel finder, and pyDevSup, were integrated with our environment to help achieve high availability for the beam diagnostic devices

INTRODUCTION

The FRIB's diagnostics group under the Beam Instrumentation and Measurements department is responsible for a variety of instruments and devices used to measure beam parameters that are essential for the machine's overall tuning and operation. While other groups at FRIB typically have a large number of similar devices, diagnostics devices are heterogeneous systems that have specific needs maintenance-wise. The team manages 15 device types with about 350 total device instances. In the controls layer, there are 36 Experimental Physics and Industrial Control System (EPICS) [1] Input/Output Controllers (IOC) types with about 250 instances. From a maintenance perspective, the diagnostics devices can be classified into intercepting and non-intercepting concerning the ion beam, each with their own set of preventive and corrective actions/requirements and tools.

INTERCEPTING DEVICES

Intercepting devices are usually large systems that have to be moved into the path of the beam and then either partially or fully block the beam. Sometimes they also need to be moved into a shared space in the beam pipe and require some extra insertion coordination via interlocks, which is handled by Allen Bradley PLCs to avoid potential collisions. Examples for such devices are Allison Scanners, Profile Monitors and Faraday Cups. They are usually critical systems responsible for determining the machine mode and helping mitigate beam power loss and trajectory deviation. Because they are physically moving devices, they require some extra attention regarding mechanical wear and tear.

* This material is based upon work supported by the U.S. Department of Energy Office of Science under Cooperative Agreement DE-SC0000661, the State of Michigan and Michigan State University.
† omitto@frib.msu.edu

NON-INTERCEPTING DEVICES

Non-intercepting devices are non-invasive systems that are installed in a fixed position alongside the beam path with electronics that help monitor beam energy, beam position stability, and beam power losses. They are usually not subject to mechanical wear and tear but require active monitoring and calibration of the underlying electronics responsible for their data acquisition. Examples of such devices are Beam Position Monitors, Beam Current Monitors, Neutron Detectors and Halo Monitor Rings. Figure 1 shows examples of both intercepting and non-intercepting devices that share space in the beam pipe.

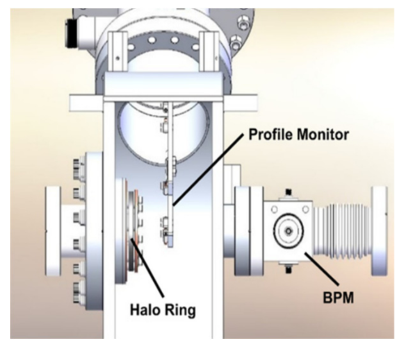


Figure 1: Intercepting and non-intercepting devices.

HARDWARE AND SOFTWARE STANDARDIZATION

Standards are essential to achieve an integrated, maintainable, and affordable control system [2]. To help minimize installation efforts and beam downtime during the initial commissioning of the machine, the diagnostics group has adopted a few devices and technologies as standard, most of which are widely used by other laboratories. For instance, we adopted the Delta Tau Geobrick LV as the standard stepper motor controller, which is also the standard controller at the Diamond Light Source and the National Synchrotron Light Source II. Also, the MicroTCA system, developed by Desy, was chosen as the standard for fast-acquisition electronics.

The EPICS framework and some of its tools, such as iocStats [3] for IOC health monitoring, the Archiver Appliance [4] for PV data archival and Channel Finder [5] as a PV directory server, are used as the central distributed control system. Jenkins [6] is used for automatic building of Debian packages [7], Stash [8] is used for software version control, and Puppet [9] is responsible for automatic deployment of Debian packages and EPICS IOCs.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA114

INTEGRATION OF NEW SIEMENS S7-1500 PLC FAMILY IN UNICOS-CPC: ENGINEERING CHALLENGES AND PERFORMANCE EVALUATION

M.Vazquez Muñiz, J. Ortolá Vidal, CERN, Geneva, Switzerland

Abstract

UNICOS-CPC (UNified Industrial COntrol System - Continuous Process Control) framework is a CERN standard solution for the design and implementation of continuous industrial process control applications [1]. This paper reports on the design and test results for the integration of a new PLC platform, the new S7-1500 Siemens PLC (Programmable Logic Controller) series. Special focus is given to the challenges faced during the integration, due to the new software architecture of the PLC, as well as to the early stage of the development and communication interfaces provided by the supplier. The paper shows the openness of the PLC development tool (TIA Portal) and presents a comprehensive evaluation of the PLC-SCADA communication mechanisms, and their integration in UNICOS-CPC.

INTRODUCTION

The first phase of the re-engineering process to support the new SIMATIC S7-1500 series in the UNICOS-CPC framework was initiated as soon as Siemens released a first stable version of its S7-1500 programming environment, the Siemens' Totally Integrated Automation Portal (TIA Portal) version 13. The solution produced in this first stage was based on the previous architecture of the framework implementation to support the SIMATIC S7-300 and S7-400 series and their corresponding programming environment: Siemens' SIMATIC Step 7 [2]. This implementation was fully stable and provided a first architectural approach and a proof of concept in terms of functionality and performance. All the necessary features needed to cope with TIA Portal were re-engineered without making use of any of the new functionality extensions and resources provided by the new programming environment together with the new PLC series (S7-1500). This implementation was successful and very useful to reveal the point where the framework must be redesigned.

After the completion of the first stage, a new UNICOS-CPC version was planned in order to incorporate the new functionality offered by the new technology (e.g. additional communication drivers, optimised data blocks, name variable addressing, increased code and data memory capacity). TIA Portal version 15 was chosen as the target version of the S7-1500 series programming environment to develop this framework version. The new implementation was focused on the re-engineering of major components of the framework, such as the device type definitions and specially the communication mechanism between the PLC and the SCADA layer based on the selection of three available communication drivers: S7 Driver, S7+ Driver and OPC UA. The limitations and design choices are presented in detail in the following sections.

DEVICE IMPLEMENTATION

The UNICOS-CPC framework relies on a library of standard object types that can model the operational behavior of the most commonly used physical devices in an industrial plant, such as sensors (temperature, flow, pressure, etc.) and actuators (heaters, control valves, motors, switches, etc.). Those devices are grouped in different families depending on their functionality and hierarchy in the plant definition: I/O Objects, Interface Objects, Field Objects and Control Objects.

Each object type is modeled and implemented as a single object in the PLC, in the case of Siemens, this functionality is implemented in a function block (FB). This FB will be instantiated, following an object oriented model, as many times as objects of the same type are needed in the control system. Then, for each instance of each object, a data structure is created to contain the information regarding its particular instance configuration. As an example, the structure for a Digital Input (DI) is shown in Fig. 1.

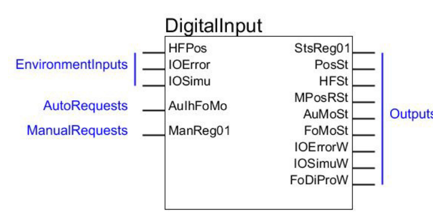


Figure 1: UNICOS CPC Digital input function block diagram.

In the version of UNICOS-CPC framework that supports the S7-300 and S7-400 series, the most common object types (IO Objects and Interface Objects) are instantiated as User Data Type (UDT) structures inside a single multi-instance data block (DB), which contains a data structure for every object instance. This encapsulation is the technical solution chosen to cope with the limitation of most of S7-300 CPUs regarding the maximum number of DBs that can be declared in a single application (maximum 4000 DBs in a S7-319-3 PN/DP CPU). The limits for a S7-1516-3 PN/DP CPU are shown in Fig. 2.

As a consequence of using the multi-instance DB, the number of DBs declared in any application is drastically reduced, however accessing from the user program to the Input-Output interface of each object instance, which is required to implement the control logic in the application

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA115

FROM MXCuBE3 TO BSXCuBE3 A WEB APPLICATION FOR BioSAXS EXPERIMENT CONTROL

M. Oskarsson, A. Beteva, D. D. S. De Sanctis, S. Fisher, G. Leonard, P. Pernot, M. D. Tully
ESRF, Grenoble, France
J. B. Florial, A. A. McCarthy, EMBL Grenoble Outstation, France

Abstract

A new version of the beamline control application BSXCuBE (BioSAXS Customized Beamline Environment) designed to control BioSAXS experiments at the new ESRF Extremely Brilliant Source (EBS) is under development. The new application is implemented as a Web application and it is based on MXCuBE3 (Macromolecular Crystallography Customized Beamline Environment version 3) from which inherits the same technology stack and application structure. This approach allows for faster development and easier maintenance. The advances in architecture and the design of new features in BSXCuBE3 are intended to enhance the automation on BioSAXS beamlines and facilitate the integration of new sample setups, such as microfluidics. As for MXCuBE3, the access to the application from any web browser natively allows the execution of remote experiments. Moreover, the ergonomics of the interface further simplifies beamline operation even for non-experienced users. This work presents the current status of BSXCuBE3 and demonstrates how the development of MXCuBE3 has contributed to the construction of a BioSAXS application.

BACKGROUND

MXCuBE3 (Macromolecular Crystallography Customized Beamline Environment version 3) is the latest generation of a beamline control application allowing beamline users to carry out experiments in Macromolecular Crystallography (MX). Originally designed and developed at the ESRF [1] the MXCuBE project has evolved to become a collaborative development which now involves eleven institutes [2]

MXCuBE3 is currently in production at the ESRF MX-beamlines ID29 [3] and ID23-2 [4], BioMAX at MAX-IV [5] and XRD2 at Elettra [6]. The application has also been installed for testing on three other ESRF MX beamlines ID30A-1, ID30A-3 and ID30B, with the aim of full deployment after the ESRF-EBS upgrade [7]. MXCuBE3 is built on well-established libraries for web development, including React [8], Redux [9], SocketIO [10] and React-Bootstrap, in order to create an intuitive, user-friendly application [11].

First released in 2005 MXCuBE [12] is now the most frequently used software for MX experiment control and data acquisition in Europe. The user experience of the application has always been important and has gained more focus in recent years, making the application easier to use such that the user can focus on the experiment at hand rather than the complexities of the beamline hardware.

The success of MXCuBE3 has inspired and influenced the development of a new general framework for beamline control applications, capable of serving both web and Qt front ends. This framework consists of reusable UI components, many already existing in MXCuBE3, and a general purpose backend. The framework further facilitates good development practices by providing patterns and abstractions for both back-end and front-end development. This design effectively allows different applications to share the same application logic regardless of the frontend technology used.

One of the first applications to be implemented in this new framework is an experiment control application for BioSAXS experiments, BSXCuBE3, scheduled for deployment in August 2020. BioSAXS is often used as a complementary technique to MX and many potential BSXCuBE3 users are already familiar with the MXCuBE3 user interface (UI). Building the BSXCuBE3 UI with the same technology and design as MXCuBE3 thus not only shortens development time and eases maintenance, it also makes the two interfaces mutually consistent from a user point of view.

GENERAL APPLICATION FRAMEWORK

Many beamline control applications at ESRF have been successfully developed and deployed using the ESRF FWK2, a Qt3 based general purpose beamline application framework [13]. However, the EBS upgrade program, the introduction of the BLISS beamline control system [14] and the obsolescence of Qt3 required the introduction of a new application framework. MXCuBE3 was developed as a web application to facilitate remote access experiments at MX beamlines, while most other ESRF beamline control applications are designed with Qt. The structure of the new application framework makes it possible to use the same back-end for both Qt and Web front-end technologies. While MXCuBE3 was developed on top of the MXCuBE2 back-end which is today exclusively used for MX experiments, BSXCuBE3 shares the same back-end solution with a larger set of beamline applications, further decreasing the maintenance and development time.

Back-end

The back-end of the application framework (Fig. 1) is written in Python 3 and can be divided into four main parts: Web server, Control system integration, Common application logic, and Application specific components.

Web Server As is the case for MXCuBE3, the BSXCuBE3 web server is based on a Web Server Gateway

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA119

ASYNCHRONOUS DRIVER EVALUATION AND DEVELOPMENT FOR DIGITAL SYSTEMS AT THE ARGONNE TANDEM LINEAR ACCELERATING SYSTEM*

C. E. Peters†, J. Reyna, D. Stanton, Argonne National Laboratory, Lemont, USA

Abstract

The ATLAS (Argonne Tandem Linear Accelerating System) accelerator at Argonne National Laboratory, near Chicago, IL., has recently been upgraded via the addition of a pulsed mode Electron Beam Ion Source (EBIS). Pulsed operation requires finer levels of control of various digital systems like fast switching high-voltage power supplies and remotely controlled function generators. Additionally, pico-level and femto-level ammeters need per-device zero correction and calibration to accurately read beam intensities. As the facility moves away from fast register-based analog signals, new and slower digital protocols adversely affect the perceived execution time of the control system. This work presents options, research, and results of implementing an asynchronous layer between high level user interfaces and the low level communication drivers in order to increase the perceived responsiveness of the system. Solutions are evaluated ranging from in-house codes, which implement system-wide mutual exclusion and prioritization, to drivers available from the EPICS control system. Key performance criteria include ease of implementation, cross platform availability, and overall robustness.

INTRODUCTION

The ATLAS accelerator is located at the United States Department of Energy's Argonne National Laboratory in the suburbs of Chicago, Illinois. It is a National User Facility capable of delivering ions from hydrogen to uranium [1] for low energy nuclear physics research in order to perform analysis of the fundamental properties of the nucleus. The majority of the current control system has been based on a CAMAC Serial Highway [2] (SH) architecture since the 1980s. Access to this hardware bus from software relies on PCI based personality cards which in turn connect to the serial highway. While this system is clearly outdated from a technology progress perspective, it continues to provide distributed serial networking with low latency and high reliability. This improves the perceived responsiveness of the control system and allows simple single-threaded access via the use of the operating system's register-based PCI subsystem interface.

Moving away from CAMAC and fast register access has commonly been accomplished by interfacing to (non-CAMAC) serial devices in the form of USB/RS-232/RS-485 specifications. However these devices use slower baud rates and typically control more complicated devices. This results in longer latency delays for each command. It should be noted this application is for a 'slow' control system and all values are only updated at about ~1-2Hz.

* This work was supported by the U.S. DOE, Office of Nuclear Physics, under Contract DE-AC02-06CH11357. The research used resources of ANL's ATLAS Facility, a DOE Office of Science User Facility.
† ChrisPeters@anl.gov

ATLAS Control System Software Description

The ATLAS Control System (ACS) group only consists of 2 – 5 full-time members, depending if the definition includes students and temporary assignments. Therefore, a third-party vendor Vista Control Systems, Inc. [3] is used to provide software libraries to supplement the creation of database structures, operator interfaces, logging tools, etc. The EPICS control system is acknowledged to be the largest and most comprehensive offering in the space, however the amount of overhead to implement and maintain a large and diverse open-source package can be prohibitive for small groups. Even borrowing individual modules like the EPICS Asyn driver [4] can be resource prohibitive unless the group has already committed to the full EPICS ecosystem.

BENCHMARKS

In order to implement a modern solution to register base CAMAC which do not cause the main operator interface (OPI) to lock, we need to understand the current level of latency in the existing software/hardware loop.

CAMAC/PCI/OPI Latency

- Single core 400 MHz Alphaserver 1200 CPU running OpenVMS 8.2 with idle CPU usage and 1GB memory.
- Kinetic Systems 2115 PCI Serial Highway Driver running in byte-wise mode at 2.5MHz clock speed
- Single 16-bit CAMAC NAF Operations

Table 1: CAMAC Execution Latencies

Operation	# of Calls	Time	Latency
16-bit Read	1,000,000	47 sec	47 µSec
16-bit Write	1,000,000	47 sec	47 µSec
Fast Process*	100,000	38 sec	380 µSec
OPI Slider	5000	262 sec	5,240 µSec

* A Data acquisition process running at its fastest software loop

Non-CAMAC Serial Latency

It is noted here that raw CAMAC latency is quite low (see Table 1). This will be difficult to match. However as more and more software overhead is added, the latency for each loop of software <u>adds</u> to the hardware latency, and the actual required latency of any replacement system becomes more reasonable. The fastest process running on the SH is only about 0.5msec of latency, and the human interfaces

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA120

MANAGEMENT OF THE MicroTCA SYSTEMS AND ITS COMPONENTS WITH A DOOCS-BASED CONTROL SYSTEM

V. Petrosyan†, K. Rehlich, E. Sombrowski, DESY, Hamburg, Germany

Abstract

The extensive management functionality is one of the key advantages of the MicroTCA.4 standard [1]. Monitoring and control of more than 350 MicroTCA crates and thousands of AMC and RTM modules installed at XFEL, FLASH, SINBAD and ANGUS experiments has been integrated into the DOOCS-based [2] control system. A DOOCS middle layer server together with Java-based GUIs - JDDD and JDTool - developed at DESY, enable remote management and provide information about MicroTCA shelves and components. The integrated management includes inventory information, monitoring current consumption, temperatures, voltages and various types of the built-in sensors. The system event logs and collected histories of the sensors are used to investigate failures and issues.

MOTIVATION

The main goal was:

- Monitor basic health of the shelves
- Receive event reports and failure notifications from the boards and other intelligent FRUs
- Manage power, cooling & interconnect resources in the shelves
- Report anomalies
- Take corrective actions when needed
- Retrieve inventory information
- Read sensors and store their values for further investigations

IMPLEMENTATION

The integration of IPMI interface into DOOCS system gives the possibility to monitor and control hundreds of crates and modules during operation. A DOOCS server communicates to MicroTCA Carrier Hub (MCH) of the crate via IPMI over LAN interface and provides overall management functionality for the DOOCS control system (Fig. 1).

VIEWS OF THE MicroTCA CRATES IN DOOCS

JDDD – Java DOOCS Data Display GUI (DESY designed) - was used for visualisation and management of MicroTCA crates (Figs. 2 and 3).

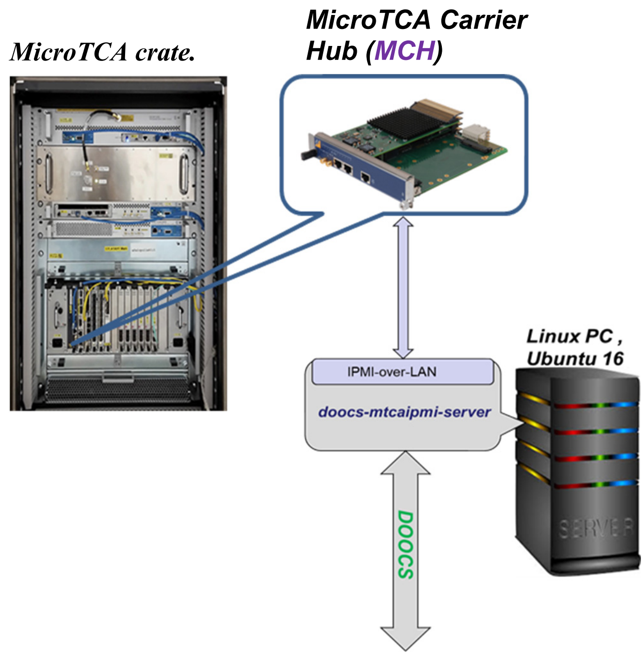


Figure 1: Hardware and software components.

Figure 2: MicroTCA 12-slot crate view JDDD.

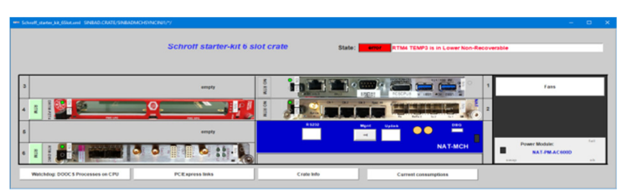


Figure 3: MicroTCA 6-slot crate view JDDD.

† vahan.petrosyan@desy.de

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA121

DEEP NEURAL NETWORK FOR ANOMALY DETECTION IN ACCELERATORS

M. Piekarski, W. Kitka, NSRC SOLARIS, Jagiellonian University, Krakow, Poland
J. Jaworek-Korjakowska, AGH University of Science and Technology, Krakow, Poland

Abstract

The main goal of accelerators like SOLARIS is to provide scientific community with high quality synchrotron light. In order to achieve this it is essential to monitor subsystems that are responsible for beam stability. In this paper, a deep neural network for anomaly detection in time series data is proposed. A pre-trained, 19-layer convolutional neural network called VGG-19 has been chosen. The main aim is to identify abnormal status of sensors in certain time step. Each time window is a square matrix so can be treated as an image. Any kind of anomalies in synchrotron's subsystems may lead to beam loss, affect experiments and in extreme cases can cause damage of the infrastructure, therefore when anomaly is detected operator should receive a warning about possible instability.

INTRODUCTION

The National Synchrotron Radiation Centre functions under the auspices of the Jagiellonian University (Fig. 1). The main goal of SOLARIS is to provide the scientific community with high-quality synchrotron light for research. At the moment SOLARIS has two beamlines PEEM/XAS and UARPES, and four more are under construction or are planned. The PEEM/XAS is a bending magnet based beamline dedicated to microscopy and spectroscopy in the soft X-rays energy range. Ultra angle-resolved photoemission spectroscopy beamline allows for measurements of fundamental quantities, i.e. the energy and the momentum, describing a photoelectron state in the space outside the solid sample.

Figure 1: NSRC SOLARIS [1].

Synchrotron is a complicated and complex device that requires the use of a distributed control system. To obtain a stable beam it is necessary for all subsystems to work correctly. Proper calibration is possible thanks to hundreds of diagnostic signals, which carry a lot of important information about the state of the machine. However, manual inspection performed even by an experienced operator is not able to extract full information from them. The purpose of this work is to present a system that could help operators in detecting potential threats and, in the future, to serve as a tool for predicting anomalies, beam loss or equipment failure.

The use of artificial intelligence techniques, including machine learning and neural networks, for signal analysis, prediction or anomaly detection has a long history. Also in the accelerator field BigData is being developed and finds applications in existing problems that engineers and scientists are struggling with. Neural networks used to correct beam orbit were proposed in a synchrotron in Australia [2]. This model consists of two neural networks trained on archived beam position data in the storage ring to determine the appropriate correction in such a way as to minimize losses described by the cost function. The topic of detection of anomalies in the control system has been raised at the last ICALEPCS Conference in 2017 by CERN [3]. The authors proposed three different mathematical approaches that have been designed and developed to detect anomalies. Those methods are dynamic, as behaviour of the system is changing in time, unsupervised detection systems for finding anomalies in live data.

ANOMALY DETECTION SYSTEM

In this section an anomaly detection model based on deep, convolutional neural network is presented. Main idea behind is to treat a bunch of diagnostic samples as an image and use transfer learning methods to build suited classifier on top of the pre-trained architecture.

Neural Networks

In recent years, neural networks have become very popular, resulting in their appearance in many applications and models. It is an attempt to map to some extent the activity of the human brain. Neural networks quickly proved to be effective in solving problems that typical programs or algorithms do not cope with or become too complicated to use. An important feature of the networks is that the they are effective even if the creator of the network himself does not quite know the algorithm that could solve the problem. Only knowledge of the problem and the appropriate selection of parameters and architecture are required. This greatly expands the possibilities of using such models for practical cases. The neural networks after proper training are able to

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA124

CERN ACCELERATORS BEAM OPTIMIZATION ALGORITHM

E. Piselli[1], A. Akroh[1], K. Blaum[2], M. Door[2], D. Leimbach[1,3], S. Rothe[1]
[1]CERN, Geneva, Switzerland
[2]Max Planck Institute for Nuclear Physics, Heidelberg, Germany
[3]Johannes Gutenberg University Mainz (IKP) Institute for Nuclear Physics, Mainz, Germany

Abstract

In experimental physics, computer algorithms are used to make decisions to perform measurements and different types of operations. To create a useful algorithm, the optimization parameters should be based on real time data. However, parameter optimization is a time consuming task, due to the large search space. In order to cut down the runtime of optimization we propose an algorithm inspired by the numerical method Nelder-Mead. This paper presents details of our method and selected experimental results from high-energy (CERN accelerators) to low-energy (Penning-trap systems) experiments as to demonstrate its efficiency. We also show simulations performed on standard test functions for optimization.

INTRODUCTION

Particle accelerators are, together with detectors, essential components for any experiment in nuclear and sub nuclear physics. Big accelerators are complex devices or machines that may serve many experiments at the same time.

Nowadays accelerator physics is a complex research field with applications far beyond subatomic physics. In order to deliver the best particle beams many adjustments are necessary. Controls plays a key role in all the technologies needed to run particle accelerators properly. All accelerator parameters should be controlled and adapted to the requested experiments. In this jungle of different setting configurations, algorithms for automated optimization can make a big difference on the time needed for setting up the machine. This saving of time is automatically translated into gain of operation time for the user.

Optimization is the discipline that deals with formulating useful models in applications, using efficient methods to identify the best possible solution. In mathematics, optimizing means finding the values which maximize or minimize a function.

Different approaches of finding optimal designs for a system are summarised in Table 1.

It is clear that the main advantages using optimization algorithms is due to the fast modelling and design process.

All processes in the accelerators are modelled and adjustments are made according to or with the help of these models. However, these models do not 100% represent the reality and therefore "intelligent" optimisation algorithms are vital to replace the often time consuming and handmade scans. The direct results are of course a gain in time and the enhanced performance of the machine, but we should not forget that the outcome of these scan can help us to refine the models.

Researchers have developed many different algorithms using the most disparate methods like linear and gradient search methods, machine learning tools, genetic algorithm and many others. The algorithm we have developed and successfully tested is derived from the numerical method Nelder-Mead.

Table 1: Optimization Approaches Design for a System

Experimental optimization	Simulated models optimization	Optimization algorithm
- Expensive - Tedious - Time consuming - Human involvement - Not always accurate	- Cheap - Slow design process - Medium human involvement - Error prone	- Fast modeling - Fast design process - Automated (minimum human involvement) - Low error - Complex optimization algorithm - Difficult of solving real world problems

BEAM OPTIMIZATION ALGORITHM

Direct-search methods do not use any information about derivatives and are therefore very robust with respect to small perturbations in the function's values. Nelder-Mead [1] is the most known method within this group and the popularity of its practical applications is based on its simplicity.

The Nelder-Mead technique was invented by J. Nelder and R. Mead in 1965 as an evolution of the method of Spendley et al. [2]. From an initial suitable solution, the algorithm tries, at each iteration, to build an improved solution until the optimum is reached.

The idea is to define a simplex (polytope in n-dimensional space with $n+1$ vertices), each of which are connected to all other vertices (e.g. a triangle in $\mathbb{R}^2$, a tetrahedron in $\mathbb{R}^3$, etc.).

The initial simplex **S** is formed by $n+1$ vertices $\overrightarrow{x_0}, \overrightarrow{x_1}, \dots, \overrightarrow{x_n}$ around an initial point $\overrightarrow{x_0} \in \mathbb{R}^n$. The other vertices are created in order to get a full starting configuration:

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA125

INTEGRATING IoT DEVICES INTO THE CERN CONTROL AND MONITORING PLATFORM

B. Copy*, M. Bräger, A. Papageorgiou Koufidis, E. Piselli, I. Prieto Barreiro
CERN, Geneva, Switzerland

Abstract

The CERN Control and Monitoring Platform (C2MON) offers interesting features required in the industrial controls domain to support Internet of Things (IoT) scenarios. This paper aims to highlight the main advantages of a cloud deployment solution, in order to support large-scale embedded data acquisition and edge computing. Several IoT use cases will be explained, illustrated by real examples carried out in collaboration with CERN Knowledge Transfer programme.

INTRODUCTION

C2MON [1] is a monitoring platform developed at CERN and since 2016 made available under an LGPL3 open source license. C2MON is at the heart of the CERN Technical Infrastructure Monitoring (TIM) that supervises the correct functioning of CERN's technical and safety infrastructure. TIM handles about three million messages per day, and serves as the central alarm management system for more than one hundred and fifty thousand alarms. C2MON relies on Java Messaging Services (JMS) [2], as well as caching and clustering technologies, to deliver transactional fail-safe data distribution. C2MON exhibits features specifically targeted at industrial control systems [3]. The Publisher-Subscriber pattern enabled thus, is key to a scalable and robust IoT infrastructure [4]. As exposed in the form of a development roadmap in 2017 [5] and thanks to recent development in cloud technologies, the C2MON deployment model was transitioned to adopt more agile runtime platforms, which had the immediate effect of simplifying the investigation and resolution of complex JMS issues encountered in production. The transition to a cloud deployment model based on Kubernetes [6] also makes C2MON more suitable for instant deployment on commercial hosting platforms such as provided by Amazon or Google.

C2MON ARCHITECTURE OVER KUBERNETES

Kubernetes as a deployment platform [6] provides a clean and efficient abstraction for scaling concerns and orchestration: individual units of process execution (called *pods*) are added or removed as required by runtime health metrics (such as CPU load, memory usage or the presence of critical errors), while process configuration and stateful requirements (such as data persistence or process cluster ordering) are transparently provided to individual pods according to templates. The recent introduction of tools such as Kustomize [7] makes it even simpler to tailor and combine a common set of Kubernetes templates for multiple scenarios, injecting configuration and writing scaling directives in a much more efficient and reproducible manner than traditional cluster management tools such as Ansible [8], due to the fact that the deployment environment is completely factored out and dissociated from hardware and operating system concerns.

* brice.copy@cern.ch

ADAPTING C2MON FOR BETTER INTEGRATION TESTING AND DEPLOYMENT

Simplifying Adoption for C2MON Users

C2MON aims to make it easier for new and existing users to (leverage its monitoring value) by focusing on the following points:

- Shipping a one click deployment.
- Allowing extension and (re)configuration of a running stack.
- Maximizing service resilience and fault tolerance.

One Click Deployment Kubernetes is the project of the Cloud Native Computing Foundation with the highest percentage of production usage, as of October, 2018 [9]. C2MON users can bootstrap a complete stack, such as the one specified below in Fig. 1, using a single command.

Configuring the C2MON Stack Managing the YAML [10] files for a Kubernetes stack can grow to quite a complex task, since the format provides no support for extension, or any form of versioning. YAML is also criticized for being unsafe and producing unexpected behaviors [11]. Addressing some of these issues, Kustomize [7] is an open source tool which provides:

- A declarative approach to configuration management.
- Component reuse capabilities.
- Integration into existing version control workflows.

By complying with Kustomize workflows, the C2MON Kubernetes deployment has been modularized into directories which correspond to layers. Additional configuration options, such as resource properties files can be added in those directories and they are automatically converted into Kubernetes ConfigMaps [12] and Secrets [13], which are then used in the pods. Thus, every component of C2MON software can be customized over the same base image and even reconfigured during runtime by applying a rolling update [14].

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA127

THE IRRAD PROTON IRRADIATION FACILITY CONTROL, DATA MANAGEMENT AND BEAM DIAGNOSTIC SYSTEMS: AN OUTLOOK OF THE MAJOR UPGRADES BEYOND THE CERN LONG SHUTDOWN 2*

F. Ravotti†, B. Gkotse[1], M. Glaser, I. Mateu, V. Meskova, G. Pezzullo
Experimental Physics Department, CERN, Geneva, Switzerland
P. Jouvelot, MINES ParisTech, PSL University, Paris, France
J.-M. Sallese, EDLAB, EPFL, Lausanne, Switzerland
[1]also at MINES ParisTech, PSL University, Paris, France

Abstract

The IRRAD proton irradiation facility at CERN was built during the Long Shutdown 1 (LS1) to address the irradiation experiment needs of the community working for the High-Luminosity (HL) upgrade of the LHC. The present IRRAD is an upgrade of a historical service at CERN that, since the 90's, exploits the high-intensity 24 GeV/c PS proton beam for radiation-hardness studies of detector, accelerator and semiconductor components and materials. During its first run (2015-2018), IRRAD provided a key service to the CERN community, with more than 2500 samples irradiated. IRRAD is operated via custom-made irradiation systems, beam diagnostics and data management tools. During the Long Shutdown 2 (LS2), IRRAD will undergo several upgrades in order to cope also with new requirements arising for projects beyond the HL-LHC. In this paper, we (1) describe the various hardware and software equipment developed for IRRAD, and (2) present the main challenges encountered during the first years of operation, which have driven most of the improvements planned for LS2 such as applying machine-learning techniques in the processing and real-time analysis of beam profile data.

INTRODUCTION

The proton IRRADiation facility at CERN (IRRAD), located in the East Area of the Proton Synchrotron (PS) accelerator, is an experimental infrastructure tailored for the qualification of material, electronics and detector components for High-Energy Physics (HEP) experiments. The IRRAD facility was originally built in 1999 [1] on the T7 beam line of the PS East Area and underwent a major upgrade in 2014 during the CERN Long Shutdown 1 (LS1). In the LS1, IRRAD was extended and moved to the T8 beam line, to cope with the increasing demand of irradiation experiments of the HEP community working for the High-Luminosity LHC [2].

The PS delivers to IRRAD a Gaussian proton beam of momentum 24 GeV/c, in ~400 ms spills (of about 5×10^{11} p) every 10 s on average, and with a typical beam spot of $12 \times 12\ \mathrm{mm}^2$ FWHM. In this beam, objects (also called "samples") up to $10 \times 10\ \mathrm{cm}^2$ in size can be exposed to a particle fluence of up to several 10^{15} p/cm^2. Smaller objects can be irradiated up to a fluence beyond 10^{17} p/cm^2. Irradiation experiments with a defocused proton beam or with larger objects are also possible. Furthermore, irradiation experiments at low temperature (−25 °C) or in cryogenic conditions (1.9 K) can be also performed at IRRAD.

The samples to be irradiated are placed on remotely controlled stages (irradiation tables) that can be moved along three axis (x,y,θ) as shown in Fig. 1. The irradiation tables, nine in total, are grouped in three consecutive zones along the proton beam trajectory and require the users to access the irradiation area, during weekly technical stops, for installing them. In addition, a conveyor (shuttle) system is available for the irradiation of smaller samples with maximum dimensions of 5×5 cm^2. This irradiation system can be moved from the outside area to the irradiation position, without the need to interrupt the proton beam operation. In order to precisely align the proton beam over T8, a dedicated beam instrument is used to provide a real-time display of the beam profile [3]. Moreover, dedicated software interfaces are used to control and monitor the IRRAD facility operations [4].

Figure 1: Irradiation tables (front) and shuttle system (back) in IRRAD.

Every year in IRRAD hundreds of samples are exposed to the proton beam (see Fig. 2). As a result, the registration, planning and follow-up of these experiments require the management of a considerable amount of data. In the early days, a dedicated software application was used in a

* This work has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement no. 654168 (AIDA-2020).

† Federico.Ravotti@cern.ch

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA129

SYNCHRONISING LabVIEW DEVELOPMENT AND DEPLOYMENT ENVIRONMENT

O. Ø. Andreassen†, C. Charrondière, H. Reymond, A. Rijllart, M. Miskowiec
CERN, Geneva, Switzerland

Abstract

LabVIEW™ with its graphical approach is suited for engineers used to design and implement systems based on schematics and designs. Being a graphical language, it can be challenging to keep track of drivers, runtime engines, deployments and configurations since most of the tools on the market aimed towards this are implemented for textual languages. Configuration management is possible in the development environment via version control systems such as Perforce™, however at CERN and in the open source software development community in general, the tendency is moving towards Git. In this paper we demonstrate how the combination of automated builds, packaging, versioning and consistent deployment can further ease and speed up development, while ensure robustness and coherency across systems. We also show how an in-house built tool called "RADE Installer" synchronises both development environments and drivers across workstations, empowering graphical development at CERN, by merging the open source toolchains with the workflow of LabVIEW. RADE installer represents a solution for LabVIEW to keep track of drivers, runtime engines, deployments and configurations.

BACKGROUND

The LabVIEW™ based Rapid Application Development Environment (RADE) was conceived as a result of an increasing need to quickly prototype and release controls, analysis and test-bench tools in the CERN accelerator domain (Fig. 1). The framework was designed to reduce the traditional development time without compromising the applications' stability and longevity. RADE's multi-tier, plugin-based architecture made it possible to develop simple control applications in hours and at the same time maintain them for years [1].

As the framework grew, so did its dependencies and complexity. Adding a change to the framework typically would take a day to release. With several changes being added every day, it could take weeks to months before they were distributed, depending on priority and available resources. This led us to invest in build automation and Continuous Integration (CI) [2].

With the introduction of CI, the release process was reduced from one day to about 1 hour (53 minutes) unattended and automated, freeing up the development time.

In addition, automating the tasks removed typical "operator errors" from repetitive work and made it possible to introduce new toolkits in the framework continuously. An added bonus from adopting CI was also the early feedback it gave. Since all unit test is executed on every release, the developer gets immediate feedback if a change broke a module and can start working on solving the issue at once [3].

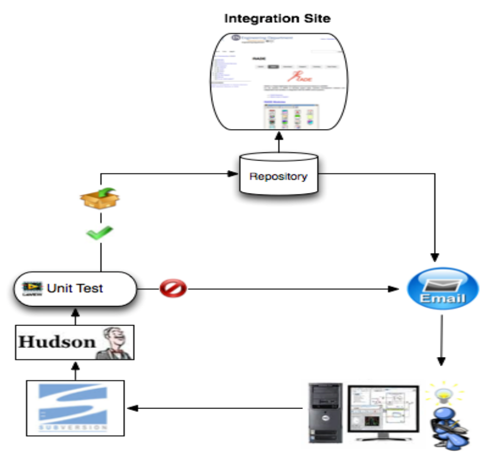


Figure 1: RADE Continuous Integration Process early version.

When the libraries have passed all the unit tests, the CI process builds and bundles the libraries in to several single target specific installers.

Reducing the release time made it possible to integrate the latest changes and any new features immediately. This change in the build philosophy worked well for libraries and applications the development team controlled, however it made it challenging for external users to be in synch with the constant change, especially if the release introduced changes to the dependent interfaces that where not backwards compatible. This led us to rethink the strategy of bundling all RADE libraries in to one big versioned installer and was the main drive behind investing efforts in to developing the RADE installer [3][4].

THE CHALLENGE

Version control and tools that automatically updates software is nothing new in the development world. There are hundreds if not thousands of different tools available on the market. The LabVIEW community even has two tools that is aimed against installing third party libraries, however CERN's unique infrastructure and platform requirements are limiting factors in what one can choose [5].

Selecting the Right Tools

There are several considerations one has to take when integrating network-based components in the CERN infra-

† odd.oyvind.andreassen@cern.ch

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA131

EVALUATION OF AN SFP BASED TEST LOOP FOR A FUTURE UPGRADE OF THE OPTICAL TRANSMISSION FOR CERN's BEAM INTERLOCK SYSTEM

R. Secondo, M.A. Galilée, J-C. Garnier, C. Martin, I. Romera Ramírez, A. Siemko, J. Uythoven
CERN, Geneva, Switzerland

Abstract

The Beam Interlock System (BIS) is the backbone of CERN's machine protection system. The BIS is responsible for relaying the so-called Beam Permit signal, initiating in case of need the controlled removal of the beam by the LHC Beam Dumping System. The Beam Permit is encoded as a frequency signal travelling over a more than 30 km long network of optical fibres all around the LHC ring. The progressive degradation of the optical fibres and the aging of electronics affect the decoding of the Beam Permit, thus potentially resulting in an undesired beam dump event and by this reducing the machine availability. Commercial off-the-shelf SFP transceivers were studied with the aim to improve the performance and continuous monitoring of the optical transmission of the Beam Permit Network. This paper describes the tests carried out in the LHC accelerator to evaluate the selected SFP transceivers and it reports the results of the test loop reaction time measurements during operation. The use of SFPs to optically transmit safety critical signals is being considered as an interesting option not only for the planned major upgrade of the BIS for the HL-LHC era but also for other interlock systems in use at CERN.

INTRODUCTION

The Beam Permit Loops are one of the most critical parts of the BIS [1]. They are used to carry User Permit requests from the Beam Interlock Controllers (BICs) to the LHC Beam Dumping System (LBDS). Its working principle is simple (see Fig. 1); the Beam Permit is encoded as a frequency signal and sent over optical fibres around the LHC ring. When the Beam Permit is TRUE, the frequency is generated and re-transmitted by every BIC crate, arriving back to the generator. When one of the users connected to the BIS network opens the loop, the Beam Permit becomes FALSE and the re-transmission is interrupted, stopping the generator. The missing frequency signal is detected by the beam dumping system, resulting in the activation of the beam dumping system. A failure to remove the beam when requested can potentially result in massive damage to the LHC machine [2].

The layout of the LHC BIS is reported in Fig. 2. The Beam Permit information is broadcasted over two separate channels A and B, respectively anticlockwise and clockwise for each beam of the LHC. A pair of BIC crates is located in each point of the LHC, plus one in the CERN control center (CCR). In addition, a board for the generation of a redundant asynchronous dump request (CIBDS) is located in point 6, together with the generator of the TRUE frequency

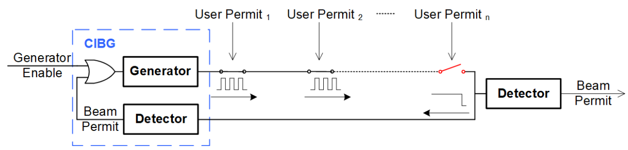


Figure 1: Basic principle of the Beam Permit Loop: a generator (CIBG) creates a TRUE frequency signal. User Permits are issued by users connected on each BIC, in order to open the Beam Permit loop and trigger a dump

(CIBG) and the Trigger Synchronization Unit (TSU), which ultimately triggers a synchronous dump by the LBDS. The frequencies associated to a TRUE value of the Beam Permit were chosen as 9.375 MHz for channel A and 8.375 MHz for channel B.

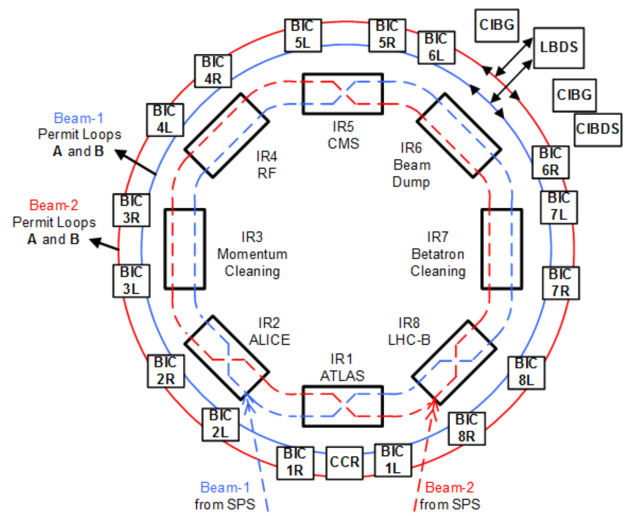


Figure 2: Layout of the operational BIS, showing the particle beams and the optical loops. BIC crates are located in the CCR and at the eight LHC interaction points.

Limitations of the Present System

The BIS has proven to be an extremely reliable system which has been working without major issues since the LHC started beam operation in 2009 [3,4]. Nevertheless, operational experience has shown that the progressive degradation of optical elements, e.g., due to the exposure to ionizing radiation, could compromise the availability of the LHC. As a result, one of the main limitations of the present BIS is given by the characteristics of the ELED based transceivers used for the optical communications. Among them:

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA132

THE DEVELOPMENT OF OBJECT DETECTION SYSTEM FOR INDUSTRIAL LINAC PROJECT AT SLRI

R. Rujanakraikarn†, P. Koonpong, S. Tesprasitte
Synchrotron Light Research Institute, Nakhon Ratchasima, Thailand

Abstract

The prototype of linear accelerator for industrial applications has been under development at Synchrotron Light Research Institute (SLRI). The primary purpose of this new project is for food irradiation application using x-ray. For efficient beam scanning purpose, a real-time object detection system has been developed by using a machine vision USB camera. The software has been developed by using OpenCV which is run on an embedded system platform. The result of the image analysis algorithm is used to control a beam scanning magnet system of the linac in real-time. The embedded system, both hardware selection and software design, running the object detection task will be described in this paper.

INTRODUCTION

Food irradiation is the process of exposing food and food packaging to ionizing radiation, such as from gamma rays, x-rays, or electron beams, without direct contact to the source of the energy (radiation) capable of freeing electrons from their atomic bonds (ionization) in the targeted food [1, 2]. Accelerator-based system is one of the platforms that can provide a good facility for food irradiation. There are three key elements of the accelerator-based system to be considered, an accelerator system to deliver the energetic beam, a scanning system to provide uniform beam coverage of the product, and a material handling system that moves the product through the beam in a precisely controlled manner [3].

Synchrotron Light Research Institute (SLRI) has been developing the prototype of linear accelerator for industrial applications. One of the main purposes of this new project is for food irradiation application, which is globally utilized during recent years. This proposed project is targeted to increase the availability of the low-cost machines for domestic uses since agricultural products are Thailand's primary economy.

In the prototype of this irradiation facility there are several main components for each key element. The accelerator system consists of an electron linear accelerating structure of the S-band standing wave type, a 3.1 MW magnetron driven by a solid-state modulator, and a hot-cathode electron gun. The scanning system comprises a beam scanning magnet and a scanning horn. The material handling system is composed of conveyor system, motor drive system, and electronic control system. The diagram of this accelerator-based irradiation facility prototype can be shown in Fig. 1.

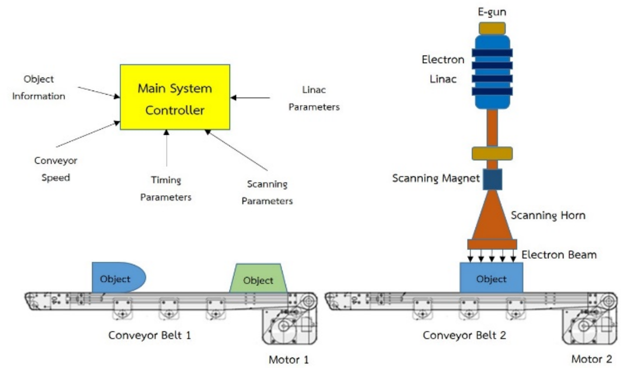


Figure 1: A prototype of accelerator-based irradiation facility.

The primary goal of irradiation facility is to deliver the specified amount of the required radiation to the products without unnecessary, wasteful, and excessive dose. Thus, monitoring and control of the process parameters and the information of objects to be scanned are important. Applying machine vision system to the irradiation facility is one way to detect object information on the conveyor belt. This system can support the material handling system in order to improve the efficiency of the facility.

This paper describes a real-time object detection system developed for this irradiation facility. The system design with selected hardware and image analysis algorithm software is described in the next section. Result and discussion, together with the relationship to the beam scanning magnet control system, are presented in the following section. Concluding remark is discussed in the final section.

SYSTEM DESIGN

This section describes the brief description of system design, both hardware and software, for object detection system developed in this project. In order to complete the object detection purpose, we apply machine vision to the material handling system. Figure 2 shows a diagram of the designed irradiation facility with the object detection system. It also shows scanning magnet controller and motion controllers necessary to be implemented in the system in order to complete all tasks to operate the facility.

Hardware

Typical machine vision system consists of lighting, lenses, vision processing unit, image sensor, and communication between sensor and processing unit. For this prototype, we consider choosing vision camera as an image sensor and lenses, with appropriate resolution and interface, for test and installation. Lighting is left for consideration once the system is installed. For processing unit, a

† roengrut@slri.or.th

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA133

SIRIUS DIAGNOSTICS IOC DEPLOYMENT STRATEGY

L. M. Russo*, LNLS, Campinas, SP, Brazil

Abstract

Sirius beam diagnostics group is responsible for specifying, designing and developing IOCs for most of the diagnostics in the Booster, Storage Ring and Transport Lines, such as: Screens, Slits, Scrapers, Beam Position Monitors, Tune Measurement, Beam Profile, Current Measurement, Injection Efficiency and Bunch-by-Bunch Feedback. In order to ease maintenance, improve robustness, repeatability and dependency isolation a set of guidelines and recipes were developed for standardizing the IOC deployment. It is based on two main components: containerization, which isolates the IOC in a well-known environment, and a remote boot strategy for our diagnostics servers, which ensures all hosts boot in the same base operating system image. In this paper, the remote boot strategy, along with its constituent parts, as well as the containerization guidelines will be discussed.

INTRODUCTION

Sirius [1], like many other particle accelerators and high-energy physics experiments, employs hundreds to thousands of device control system entities. It abstracts the so called *Front-End Controller* (*FEC*) layer, of modern control systems, from the *Client Application* layer and, in the case of a three-tier architecture, *Middle-Layer Services* layer [2, 3].

In the case of facilities based on EPICS [4], like Sirius, *Input-Output Controllers* (IOCs) act as the *FEC* and export its functionalities into *Process Variables* (PVs). As such, *IOCs*, after completing its development cycle (i.e. analysis, design, implementation, test), need to be deployed to the control system in a consistent way, minimizing downtime and coupling to other systems, whereas maximizing the flexibility of changes and rollback strategies in case something fails.

Modern installations typically have thousands of *IOC* instances and hundreds of thousands of *PVs*, in which control, monitoring, archiving and complex interactions with each other take place. Thus, the process of manually compiling, bundling the necessary configuration files and downloading the *IOC* application becomes a burden and an error prone task to the system maintainer. In this sense, the control system environment that has in its foundation principles like consistency and reliability becomes more fragile.

In order to achieve the desired capabilities of such a deployment system, many laboratories and institutes have developed strategies to solve and guide this process employing a myriad of techniques, such as Agile Development, Continuous Integration and Continuous Deployment, and tools, such as *rsync* file synchronizing tool, *Network File System* (NFS), Jenkins, Puppet, Ansible and Containers [5–9].

In the next sections, the standards, development cycle and tools chosen by the Sirius beam diagnostics group will be discussed.

DIAGNOSTICS SYSTEMS

Sirius Diagnostics systems can be summarized by Table 1 extracted from [10]:

Table 1: Summary of Beam Diagnostics Components. LINAC is the Linear Accelerator, LTB is the Linac to Booster Transfer Line, BO is the Booster Ring, BTS is the Booster to Storage Ring and SR is the Storage Ring

	Linac	**LTB**	**BO**	**BTS**	**SR**
DCCT	-	-	1	-	2
FCT	-	1	-	1	-
ICT	2	2	-	2	-
Beam Flag	5	6	3	6	-
Horizontal Slit	-	1	-	-	-
Vertical Slit	-	1	-	-	-
Button BPM	-	-	50	-	160
Stripline BPM	3	6	-	5	-
Front-End Photon BPM	-	-	-	-	80
Filling Pattern Monitor	-	-	-	-	1
Horizontal scraper	-	-	-	-	1 pair
Vertical scraper	-	-	-	-	1 pair
Tune shaker	-	-	1	-	2
Tune pick-up	-	-	1	-	1
Bunch-by-Bunch kicker	-	-	-	-	2
Bunch-by-Bunch BPM	-	-	-	-	1
X-ray port	-	-	-	-	2
Visible light port	-	-	-	-	1
Streak camera	-	-	-	-	1
Beam Loss Monitor	-	tbd	tbd	tbd	-
Gas Bremsstrahlung Monitor	-	-	-	-	tbd

Most of the diagnostics presented on Table 1 follow the same development and deployment strategy described in the next section. The exception are the following systems:

- BPM (Button, Stripline, Photon): the system [11] is based on MicroTCA.4 with an x86 AMC CPU board and a CentOS7 operating system. For historical reasons this system, at the time of its inception, did not have the deployment infrastructure available today. As such, management is performed manually with a set of scripts based on parallel *ssh* connections and remote *bash* commands. The plan is to convert it to the herein described deployment strategy.
- Bunch-by-Bunch (Kicker, BPM): the system is based on a custom solution by *Dimtel* [12] with the *IOC* running

* lucas.russo@lnls.br

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-WEPHA134

MONITORING SYSTEM FOR IT INFRASTRUCTURE AND EPICS CONTROL SYSTEM AT SuperKEKB

S. Sasaki†, T. T. Nakamura, KEK, Tsukuba, Japan
M. Hirose, Kanto Information Service, Tsuchiura, Japan

Abstract

The monitoring system has been deployed to efficiently monitor IT infrastructure and EPICS control system at SuperKEKB. The system monitors two types of data: metrics and logs. Metrics such as a network traffic and a CPU utilization are monitored with Zabbix. The data stored in Zabbix are visualized on Grafana, which allows us to easily create dashboards and analyze the data. Logs such as text data are monitored with Elastic Stack, which provides log collection, searching, analysis and visualization. We have applied it to monitor broadcast packets in the control network and EPICS control system. In addition, we have developed the EPICS Channel Access client software that sends PV values to Zabbix server and the status of each IOC is monitored with it. We have also developed the Grafana plugin and API gateway to visualize the data from pvAccess RPC servers. Various data such as CSS alarm status data is displayed on it.

INTRODUCTION

SuperKEKB [1] is an asymmetric-energy electron-positron double ring collider. Its target luminosity is 8×10^{35} $cm^{-2} s^{-1}$ which is 40 times higher than that of the preceding project, KEKB.

The SuperKEKB control system is based on EPICS [2]. The control system involves hundreds of computers which are networked together to allow communication between them. Therefore, monitoring the IT infrastructure is essential for a stable accelerator operation. We have deployed the monitoring system to monitor the IT infrastructure and EPICS control system. The system monitors two types of data: metrics and logs. This paper describes the system architectures and its applications that we have implemented.

METRIC MONITORING

Metrics are collection of a measurement at a certain point in time. They are typically collected at fixed-time intervals and referred to as a time series data. We monitor the metrics such as a network traffic and a CPU utilization with Zabbix [3].

Zabbix

Zabbix is an open-source monitoring software tool. It is integrated multiple features for monitoring as shown below:

- Data gathering.
- Alerting.
- Data visualization.
- Historical data storage.
- Network discovery.

We have monitored 22 computers for servers and operator consoles and 88 network switches. About 30000 items and 10000 triggers have been monitored. Data gathering is performed with SNMP or Zabbix agent.

Alerting settings

Zabbix supports triggers, which are logical expressions to evaluate gathered data and represents the current system state. Zabbix allows taking place some operations when trigger status changed. We have applied it to notify system problems by e-mail.

Trigger has severity parameter to defines how important a trigger is. Zabbix supports following trigger severities: Not classified, Information, Warning, Average, High and Disaster. We have configured notification behavior according to trigger severities. Alerts for High or Disaster severity are immediately notified. On the other hand, alerts for Warning or Average severity are notified when its trigger state remains PLOBLEM for 24 hours. There are no notifications for Information. This prevents major severity notifications from being buried in minor severity notifications.

Data Visualization on Grafana

Grafana [4] is an open-source software tool for data visualization and analysis. It allows to visualize data from various data storage backend. We can easily create and view dashboards via a web browser with Grafana.

Grafana is extendable with plugins to add a new panel or a datasource. We have applied Zabbix plugin [5] to Grafana to visualize the data stored in Zabbix data storage. This plugin provides metric processing functions to transform and shape the data. Figure 1 shows computer performance metrics on Grafana.

Figure 1: Grafana dashboard for computer performance metrics. The metrics are retrieved from Zabbix data storage.

† shinya.sasaki@kek.jp

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA136

THE SOFTWARE-BASED MACHINE PROTECTION SYSTEM USING EPICS IN J-PARC MR

K. C. Sato*, T. Kimura, S. Yamada, N. Kamikubota, N. Yamamoto
J-PARC, KEK & JAEA, Ibaraki-ken, Japan
S. Yoshida, Kanto Information Service (KIS), Accelerator Group, Ibaraki, Japan

Abstract

In J-PARC, a dedicated MPS (Machine Protection System) stops accelerator beam operation automatically when an interlock signal comes. The MPS accepts interlock signals of accelerator components by hard-wires. Recently, a software-based MPS, called "Soft-MPS", was introduced, which uses EPICS PVs without wiring. A PLC controller running Linux was configured to watch at EPICS PVs over network. After logical calculations, it outputs an interlock signal to arise a MPS event. From the first Soft-MPS setup in 2018 spring, nine Soft-MPS setups are currently used.

INTRODUCTION

J-PARC (Japan Proton Accelerator Research Complex) is a high-intensity proton accelerator facility in Japan. It has been operated collaboratively by Japan Atomic Energy Agency (JAEA) and High Energy Accelerator Research Organization (KEK). It consists of three accelerators: a linear accelerator (LINAC), a Rapid Cycling Synchrotron (RCS), and a Main Ring (MR). MR started beam operation in 2008 [1].

A MPS (Machine Protection System) is a mechanism to stop accelerator operations automatically. In general, a MPS accepts various faulty signals from accelerator components: interlock signals of power-supplies, over-threshold alarm from beam-loss monitors, and so on. In J-PARC MR, we have a dedicated MPS [2]. When a faulty signal is detected during beam-delivery operation, the MPS starts a procedure to extract beams to the abort dump immediately (order of 10 microseconds). The MPS is essential to avoid damages caused by high-intensity proton beams.

As default, all signals come into the MPS are hard-wired. Recently we have introduced software-based MPS (Soft-MPS), with which input signals are given by an EPICS-based control system (i.e. software). This report provides implementation and operation details of our Soft-MPS.

DEVELOPMENT OF SOFT-MPS

Why We Need Soft-MPS ?

There are 2 reasons of using Soft-MPS. (1) To install interlock signals quickly. A Soft-MPS can implement a new interlock signal easily, because of no cabling. This type of Soft-MPS should be switched to standard-MPS using hard-wire later. (2) To use non-hardware parameters. For example, machine operation modes, beam bunch information, etc. Such signals are difficult to be given by hard-wire cables.

* kenichi.sato@j-parc.jp

Overview

In J-PARC, we have been using an EPICS-based control system for the accelerator equipment [3] [4]. In J-PARC MR, a PLC controller (a CPU module running Linux with I/O modules of Yokogawa FA-M3 series [5]) are widely used as an EPICS input/output controller (IOC) [6] . A Soft-MPS setup is also an EPICS IOC, which consists of a CPU module and a Digital-Output module.

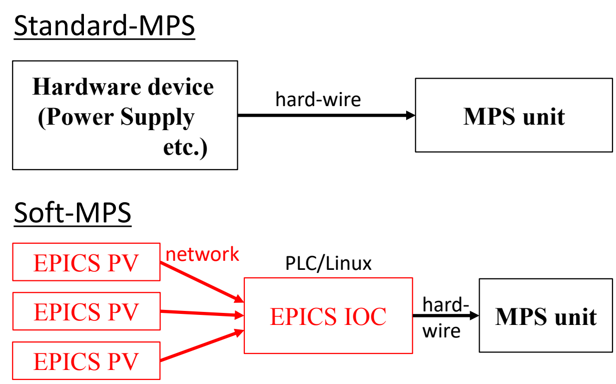


Figure 1: Standard-MPS vs. Soft-MPS.

Differences between standard-MPS and Soft-MPS are shown in Fig. 1. A standard-MPS has a hard-wire cable from a hardware device to a MPS unit. While a Soft-MPS uses EPICS process variables (PVs), which correspond to device interlock signals. Using remote PV parameters over network, an IOC would generate an output interlock signal to a MPS unit. Figure 2 shows photos: a PLC-type IOC for Soft-MPS and a MPS unit. Two are connected by a hard-wire cable. These are installed in the D1 building.

Typical process flows in an IOC are shown in Fig. 3. Four different setups of Soft-MPS are implemented in one IOC. Each of setups watches at remote PVs, located in distant buildings (D2, D3 and CCR). Logical outputs from setups are summarized, and finally a real output signal (a blue "DO") would be generated.

Reliability Mechanism

A Soft-MPS watches at PVs via a network. In general, signal-inputs over network are less reliable than using hard-wire. To improve reliability, Soft-MPS has two mechanisms: (a) detection of PV communication loss, and (b) latch and hold alert status of each input.

PV Communication Loss When a PV, which is located at a remote IOC, becomes unavailable to communicate,

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA137

INTEGRATION OF A MODEL SERVER INTO THE CONTROL SYSTEM OF THE SYNCHROTRON LIGHT SOURCE DELTA

D. Schirmer*, A. Althaus†
Center for Synchrotron Radiation (DELTA), TU Dortmund University, Germany

Abstract

During the past decades, a variety of particle optics programs have been applied for accelerator studies at the storage ring facility DELTA. Depending on the application, most programs were used offline without dynamic machine synchronisation. In order to centralize and standardize storage ring modeling capabilities, a dedicated online model server was developed and integrated into the EPICS-based control system. The core server is based on Python/EPICS service modules using OCELOT and COBEA as simulation tools. All data, actual machine readings/settings, conversion coefficients, results of simulation calculations as well as manual parameter settings, are handled via EPICS process variables. Thus, the data are transparently available in the entire control system for further processing or visualisation. To improve maintainability and adaptability, the remote presentation model controller concept was realized in the implementation. The paper explains the setup of the model server and discusses first use cases.

INTRODUCTION

DELTA is a 1.5–GeV electron storage ring operated as a synchrotron light source by the TU Dortmund University [1]. Since 2011, a short-pulse facility for coherent sub-picosecond pulses in the vacuum ultraviolet (VUV) and Terahertz regimes has been established [2, 3]. In recent years, various programs for storage ring modeling and tracking calculations were employed. Early accelerator design studies were carried out with the Pascal/Delphi program OPTICS [4]. Later, depending on the application focus, MAD [5] (optics and orbit correction studies [6]) and ELEGANT [7] (CHG/EEHG model calculations [8]) were used. These programs are usually applied in offline mode, i. e., without direct machine connectivity. For online modeling, the Accelerator Toolbox (AT) has proven to be useful [9, 10]. The AT is a collection of software tools to model storage rings and beam transport lines in a MATLAB workbench environment [11] and is mainly used as a stand alone program for prototypical applications. Direct read/write access to all EPICS [12] process variables (PVs), i. e., to all machine parameters, is enabled via the channel access (CA) protocol software interfaces 'labCA' [13] and 'mCA' [14]. In order to enable model calculations and storage ring simulations with live machine data, a novel Python/EPICS-based model server was integrated into the DELTA control system.

* detlev.schirmer@tu-dortmund.de
† andre.althaus@tu-dortmund.de

MODEL SERVER SOFTWARE SETUP

The model server implementation makes use of the Remote Presentation Model (RPM) [15] software design pattern [16]. It is commonly applied for developing client-server-based user interfaces (UI). Like the Model View Controller (MVC) pattern [17], it divides software code into three interconnected parts (model, view, controller). Thus, it provides a more effective way to create clearly structured, more maintainable and reusable applications. The basic concept is shown in Fig. 1. The server side controller (e. g., optics calculator or conversion scripts, see next section) contains all the controller logic for a specific view (e. g., Python/Qt-based or Tcl/Tk-based GUI panels). The model data of this controller is automatically synchronized with the view data via the EPICS channel access (CA) protocol.

The RPM paradigm was largely used during the implementation of the DELTA model server. The data exchange between individual program modules as well as their configuration control is handled by dedicated EPICS PVs. Thus, all input/output simulation data as well as the software module configurations are transparently available in the entire EPICS control system.

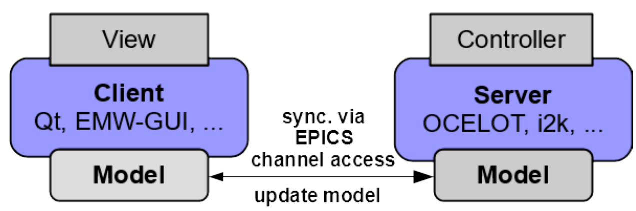


Figure 1: Remote Presentation Model (RPM) software design pattern used for the EPICS-based model server.

Since the EPICS-based DELTA control system is currently being converted from Tcl/Tk-based to Python-based applications [18], the simulation library OCELOT [19] was chosen as the modeling core software. OCELOT is a software framework for beam dynamics simulation studies mainly written in Python. The software package calculates, i. a., linear and nonlinear storage ring optics and requires a lattice description file. Therefore, a detailed lattice data file was created which models the DELTA storage ring as accurately as possible. It contains all currently (status 2019) installed storage ring components including drift spaces, dipoles, quadrupoles, sextupoles, slow and fast orbit steerer magnets, beam position monitors (BPMs), RF cavity, undulators as well as a dipole-like edge-model of the superconducting asymmetric wiggler magnet (SAW). This modeling input file was cross-checked by linear optics calculations with different simulation programs (Matlab-AT, MAD, OP-

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA138

ORBIT CORRECTION WITH MACHINE LEARNING TECHNIQUES AT THE SYNCHROTRON LIGHT SOURCE DELTA

D. Schirmer*
Center for Synchrotron Radiation (DELTA), TU Dortmund University, Germany

Abstract

In the last years, artificial intelligence (AI) has experienced a renaissance in many fields. AI-based concepts are nature-inspired and can also be used in the field of accelerator controls. At DELTA, various studies on this subject were conducted in the past. Among other possible applications, the use of neural networks for automated correction of the electron beam position (orbit control) is of interest. Machine learning (ML) simulations with a DELTA storage ring model were already successful. Recently, conventional Feed-Forward Neural Networks (FFNN) were trained on measured orbits to apply local and global beam position corrections to the 1.5–GeV storage ring DELTA. First experimental results are presented and compared with other orbit control methods.

INTRODUCTION

DELTA is a 1.5–GeV electron storage ring facility operated by the TU Dortmund University supplying radiation ranging from THz to the hard x-ray regime [1, 2]. The transverse position of the electron beam is measured at 54 capacitive multiplexed pick-up monitors (BPMs) installed along the 115 m long vacuum chamber of the storage ring [3]. For beam position correction, 30 horizontal (HC) and 26 vertical dipole correctors (VC) are available. A slow orbit feedback system (SOFB), ranging from 0.2 Hz to 1 Hz and based on a singular value decomposition (SVD) algorithm of measured orbit response matrices (ORM) is in operation since 2005 [4].

Because of increasing software issues and for control system maintenance reasons, a revised software version was required. Therefore, a cone-programming-based global correction approach has been developed and is in a testing phase now [5, 6]. An alternative concept applies machine learning techniques as an heuristic method, inspired by the pioneering work done at NSLS/BNL [7, 8].

ORBIT CORRECTION AT DELTA

All steerer magnets are additional coils on quadrupole yokes which can be ramped to a current of max. ± 10 A according to beam kicks of max. ±3 mrad at 1.5 GeV. They are controlled via 12-bit digital-to-analog converters (DACs) integrated on CAN to serial bus converter modules [9]. The DACs allow current changes with a granularity of 2.4 mA which corresponds also to the minimum read-back resolution [10].

The analog BPM signals are read out via a mix of I-Tech Libera [11] and Bergoz MX [12] electronics. The MX-BPMs provide the measured beam position as an analog voltage that is digitized by 12-bit analog-to-digital converters (ADCs) [13] and fed over a CAN-bus into the EPICS control system [14]. A 10 Hz low-pass filter reduces sampling noise while maintaining sufficient bandwidth for the SOFB. The measuring accuracy is approx. ±5 µm mainly limited by the resolution of the ADCs. At beam currents above 2 mA the 'slow acquisition' data from Libera BPMs is of roughly the same quality as the data from the MX-BPMs. For a more detailed description see [3, 15].

The 'zero orbit', i. e., the orbit with all correctors switched off, deviates from the design orbit of the storage ring due to alignment and field errors of magnets as well as unavoidable non-linear magnetic fringe fields. At DELTA, this orbit is normally not usable for standard synchrotron user operation, since it implies large beam amplitudes and angles and thus causes vacuum chamber heatings, reduces injection efficiency and does not optimally illuminate the synchrotron radiation beamlines. Through empirically adjusted orbit shifts and optionally by putting weights at dedicated BPM positions, which increase the impact of orbit deviation at important ring positions (e. g., beamline source points or injection septum), a new so-called reference orbit is defined. Nevertheless, due to temperature drifts, malfunctions and miss-calibrations of BPMs, failure of steerer magnets as well as variations in BPM weight and offset values, the currently measured orbit still deviates slightly from the desired reference orbit. An orbit correction (OC) algorithm keeps these deviations as small as possible. The quality of an OC algorithm can be expressed by the Euclidean norm $E_{x,z} = \sqrt{\sum_{i=1}^{N} (\Delta_{x,z})_i^2}$. It considers the orbit errors in both planes ($\Delta_{x,z}$) between the currently measured orbit and the reference orbit at all 54 BPM positions i=1..54. For a more detailed description see [4].

MACHINE LEARNING DESIGN STEPS

The development of an ML-based OC application passes six major steps (see Fig. 1). The first steps are data acquisition (DAQ) and cleaning. Afterwards, the neuron network topology is defined and optimized. Finally, after multiple training sessions with continuous performance tests, the OC application has to be tested in real machine operation. A number of special programming environments are available for implementing ML applications. First tests were carried out within the frameworks TensorFlow [16] and Keras [17]. For investigations conducted in this work, the computing environment MatLab (Matrix Laboratory) and corresponding

* detlev.schirmer@tu-dortmund.de

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA139

SCALING UP THE DEPLOYMENT AND OPERATION OF AN ELK TECHNOLOGY STACK

S. Boychenko, P. Martel, B. Schofield CERN, Geneva, Switzerland

Abstract

Since its integration into the CERN industrial controls environment, the SCADA Statistics project has become a valuable asset for controls engineers and hardware experts in their daily monitoring and maintenance tasks. The adoption of the tool outside of the Industrial Controls and Safety Systems group scope is currently being evaluated by ALICE, since they have similar requirements for alarms and value changes monitoring in their experiment. The increasing interest in scaling up the SCADA Statistics project with new customers has motivated the review of the infrastructure deployment, configuration management and service maintenance policies. In this paper we present the modifications we have integrated in order to improve its configuration flexibility, maintainability and reliability. With this improved solution we believe we can propose our solution to a wider scope of customers.

INTRODUCTION

The scale and increasing complexity of modern SCADA applications deployed in a wide range of domains in CERN's accelerator complex urged the development of a range of support frameworks for the back-end infrastructure's operation and maintenance. Until recently, data from different sources was stored and processed independently; this made data analysis a tedious task, requiring the manual matching of information from multiple domains. In order to combine data from multiples sources, make it more accessible and provide seamless access to the underlying measurements, the SCADA Statistics service was developed [1].

There are around 200 controls applications maintained by the Industrial Controls and Safety systems group and most of those archive data from thousands of devices into a persistent storage. The equipment experts working with historical data are generally interested in analysing the alarms and value changes (for example to determine if there are misconfigured devices). Despite the fact that the database allows extraction of the required data, due to its internal structure and huge amount of persisted measurements, the task is tedious and time-consuming. The SCADA Statistics project was developed to solve this problem, enabling fast and easy access to alarms and value changes data. In addition to the historic data analysis, the system collects and processes the data from controls applications logs, distributed across many servers, preserving the information which was previously discarded after some time (log files on the servers are rolling, i.e. overwritten when determined threshold is reached).

The SCADA Statistics service (see Figure 1) uses the Elastic stack, an open source search and analytics engine. The Elastic stack is composed of Filebeat (a lightweight way to follow and centralize logs and files), Logstash (a data processing pipeline), Elasticsearch (the search and analytics engine) and Kibana (a visualization tool for Elasticsearch data). The system can be split into three separate layers: data source, backend and frontent. The data source layer is composed by a large number of servers running the controls applications and a centralized storage where historical data is being persisted. These services are hosted on the dedicated hardware cluster, which is operated by the department's system administrators. The frontend layer functionality is assured by the IT department, and most of the maintenance tasks are performed by CERN's Elastisearch experts. The backend layer is running on the virtual machines (provided through Openstack technology), having a limited support, generally constrained to network and basic Operating System issues.

After three years in production, the SCADA Statistics service has generated over 1 TByte of data with more than $3.7x10^9$ indexed measurements. The input rate increases on a continuous basis as new applications and use cases are integrated into the service. In addition to the wide user base within the same working group [2], external clients (from ALICE [3] and potentially other domains at CERN) become interested in running the service on their own infrastructure.

Even though the existing solution fulfils the initial requirements, the "community" (i.e. "free") version of Elastic stack (which SCADA Statistics is based on) lacks some functionality like monitoring and failure recovery. These features are required if we are to extend the scope of the project for external users. Additionally, in its initial implementation, the system configuration and deployment was based on manually edited, long shell scripts which made the maintenance and integration of new features very difficult.

In this paper, we describe the consolidation project that made the SCADA Statistics service more flexible, maintainable and configurable.

SCADA STATISTICS INFRASTRUCTURE MONITORING

The initial experience with the maintenance of the SCADA Statistics stack proved that the system requires a constant monitoring, as unexpected failures were observed in most of the system components. The deployment of the conventional Metricbeat-based solution was not an option, since the "community" version of Elasticsearch, which was used to make the monitoring information persistent, did not allow the definition and configuration of alarms. Furthermore, this solution does not support an automated failure recovery mechanism, requiring the person in charge to perform

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA143

HIGH-LEVEL APPLICATION ARCHITECTURE DESIGN FOR THE APS UPRADE*

G. Shen†, N. Arnold, S. Benes, D. Jarosz, A. Johnson, D. Stasic, I. Usmani, S. Veseli,
Argonne National Laboratory, Lemont, USA
D. Liu, C. McChesney, Osprey DCS LLC, Ocean City, USA

Abstract

A modular software platform is under active design and development for high-level applications to meet the requirements of the Advanced Photon Source Upgrade (APS-U) project. The design is based on modern software architecture, which has been used in many other accelerator facilities and has been demonstrated to be effective and stable. At APS-U, we are extending the architecture in order to efficiently commission, operate, and maintain the APS-U. Its open architecture provides good flexibility and scalability. This paper presents the current status of high-level application architecture design, implementation, and progress at APS-U.

INTRODUCTION

The control system of the Advanced Photon Source Upgrade (APS-U) adopts an EPICS-based [1] controls system standard model, which is widely used in the accelerator control community. As illustrated in Fig. 1, its three-tier controls software infrastructure consists of:

- Distributed front-end layer. This layer is often referred to as process controls of technical systems, which includes an IOC for classic technical systems like the magnet power supply, vacuum, radio frequency/low-level rf (RF/LLRF), diagnostics, and IOC for high-speed data acquisition (DAQ) technical system.
- Service layer. The service layer collects data from various sources, for example, the front-end technical controls, database, and storage; organizes the data in a predefined data structure; publishes to its upper layer and/or accepts data from its upper layer; and ships data to database or front-end systems.
- Application layer. The application on this layer often interacts with the end user through either an operator graphic interface or other software-like physics application.

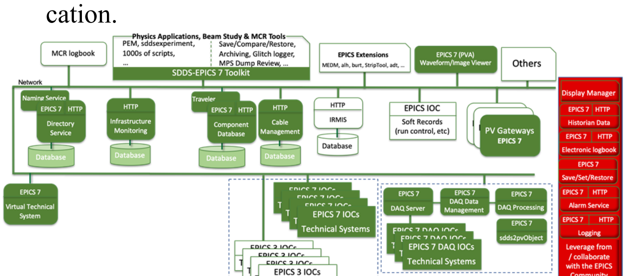


Figure 1: APS-U controls software architecture.

* 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.
† gshen@anl.gov

The high-level application (HLA) for APS-U presented in this paper is a software collection to satisfy the requirements to support APS-U engineering construction. It does not cover the physics application or the control room applications like display manager, alarm handling, data archiving, save-set-restore, etc.

HLA REQUIREMENT

To meet the APS-U project requirements, the engineer tool suite shall be modular, incrementally upgradeable, scalable, and extendable. Expansion of the tool suite to accommodate the build-up of the accelerator complex from early testing, through installation and commissioning and during the life of the facility, should not impact the performance. The tool suite shall be available to support all aspects of the project schedule from component tests during prototyping to beam characterization and optimization at commissioning and operation. To achieve this, the APS-U high-level engineering tools are required to be based on open standards and commercial off-the-shelf software applications, whenever possible.

The current Advanced Photon Source (APS) has a thorough set of tools deployed to support its operation, and most of those tools will be inherited by APS-U and/or improved upon where needed. In addition, APS-U has identified an additional set of high-level engineering tools driven by the unique requirements of the project, which are listed as below:

- Component database (CDB) [2, 3]
- Electronic traveler (eTraveler) [4, 5]
- Cable management system [3]
- EPICS directory service [6]
- Naming system
- Controls infrastructure monitoring

The latest status of those tools will be presented in this paper.

HLA STATUS

The high-level application is under active development at APS-U to satisfy the requirements of supporting project construction. Some tools have been deployed to support ongoing construction activities, and some tools are under evaluation. Detailed progress for each tool is presented in the following sections.

Component Database

The CDB manages component data and supports the full APS-U project lifecycle including both high-level and low-level design, procurement and assembly, installation and

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA148

CUMBIA-TELEGRAM-BOT: USE CUMBIA AND TELEGRAM TO READ, MONITOR AND RECEIVE ALERTS FROM THE CONTROL SYSTEMS

G. Strangolino, Elettra-Sincrotrone Trieste S.C.p.A., Basovizza, Trieste, Italy

Abstract

Telegram is a cloud-based mobile and desktop messaging app focused on security and speed. It is available for Android, iPhone/iPad, Windows, macOS, Linux and as a web application. The user signs in the cumbia-telegram-bot to chat with a Tango or EPICS control system from everywhere. One can read and monitor values, as well as receive alerts when something special happens. Simple source names or their combination into formulas can be sent to the bot. It replies and notifies results. It is simple, fast, intuitive. A phone number to register with telegram and a client are the necessary ingredients. On the server side, cumbia-telegram-bot provides the administrator with full control over the allocation of resources, the network load and the clients authorized to chat with the bot. Additionally, the access to the systems is read only. On the client side, the bot has been meticulously crafted to make interaction easy and fast: history, bookmarks and alias plugins pare texting down to the bone. Preferred and most frequent operations are accessible by simple taps on special command links. The bot relies on modules and plugins, that make the application extensible.

TECHNOLOGY

The server is written in C++and employs the Telegram BOT API[1], that is an HTTP-based interface for building *bots* for Telegram. Qt [1] is used to manage network connections, support JavaScript integration, store user data into a local database (*SQLite*) and draw graphs of spectrum variables (by means of the *Qwt* library). Figure 1 shows the whole infrastructure.

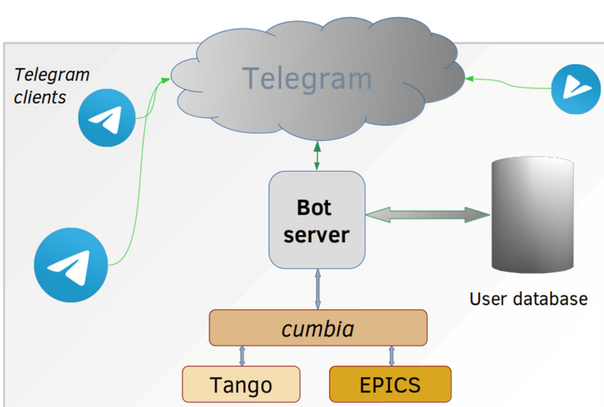


Figure 1: The whole *cumbia-telegram bot* architecture.

After joining the *bot*, clients must be authorized before messaging. Operations on the control systems are read only and may be subject to limitations (e.g. upper limit on active monitors) according to the account privileges that can be set up on a user basis. Further restrictions curb the access to the engines: the maximum read rate is established by the administrator and the bandwidth is divided across the connected users. Those monitoring more than one source in *polling* mode will face major restraint. On the other hand, event based updates are not limited at the moment, due to their theoretical low impact on the control system and network. The application is organised in modules. Some of them are built in, while additional ones can loaded as plugins, that register with the server, receive and process messages. Bookmark, search and command components are examples of plugins. The *modules* command returns a list of currently loaded modules and plugins, with their description.

FEATURES

Overcoming Messaging App Limitations

Telegram is an advanced cloud-based mobile and desktop messaging application. The design of the *bot* had to face the limitations of a text centred interface. Source names are not easy to remember and keyboard dictionaries cannot help writing often complex patterns. Frequent operations must be accessed easily, and handwriting must be replaced by *tap* operations as much as possible. *Tapping* is possible on commands either configured through a special bot named *The BotFather* or nested in the message (and identified by a leading / character, like */help*). Those manually configured with *The BotFather* are available through a button at the right of the message input text area.

The following components have been implemented to address these limits and make the user interaction effective, intuitive and funny:

- *history*, to keep a record of recent successful operations;
- shortcut to repeat the *last* action;
- *bookmarks*, where preferred commands can be saved and sent over again with a *tap*;
- *alias*. An alias can be associated to any sort of instruction, so that it can be quickly recalled.

Supported Engines

At the moment, the Tango [2] and EPICS [3] control systems are seamlessly integrated in the application. Since the *bot* is based on *cumbia*, more engines can be added with little effort.

[1] https://core.telegram.org/bots/api

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA150

SLED TUNING CONTROL SYSTEM FOR PAL-XFEL*

Y. J. Suh†, G. Mun, K.H. Kim, Y.J. Park, H. Heo, C.B. Kim, H.S. Kang
Pohang Accelerator Laboratory, Pohang, Korea

Abstract

A total of 42 SLED(SLAC Energy Doubler)s are installed at the Pohang Accelerator Laboratory-XFEL linac section . The PAL-XFEL SLED has a system that controls the resonance of the cavity by installing the motor in a way that adjusts the cavity volume. The controller uses Beckhoff [1] PLC and TwinCAT3, and the communication with EPICS IOC uses TCP-IP and it was made with Asyn motor. 42 slave unit and master unit are connected by optical cable and 1: 1 connection. So, if one unit is faulty and does not work, the remaining 41 units can operate normally. In this paper, we describe the introduction and results of the SLED tuning control system.

HARDWARE FEATURES

The Sled Tuning Control system is a device for controlling the motor with a motor connected to the center axis of the SLED cavity. Figure 1 shows the shape of the mechanism and motor installed in the tunnel.

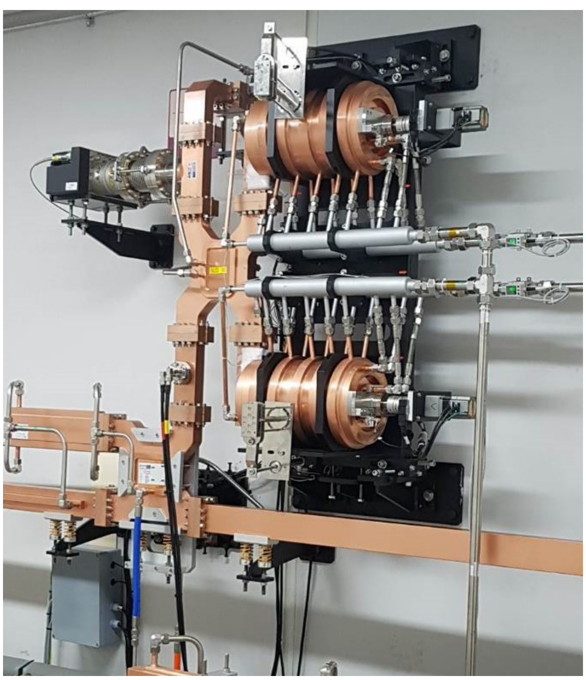


Figure 1: Sled cavity in Tunnel wall.

Figure 2 shows the shape of the junction box installed on the inner wall of Tunnel. The upper part is the amp of LVDT and the lower part is the cable of motor, limit sensor, motor brake.

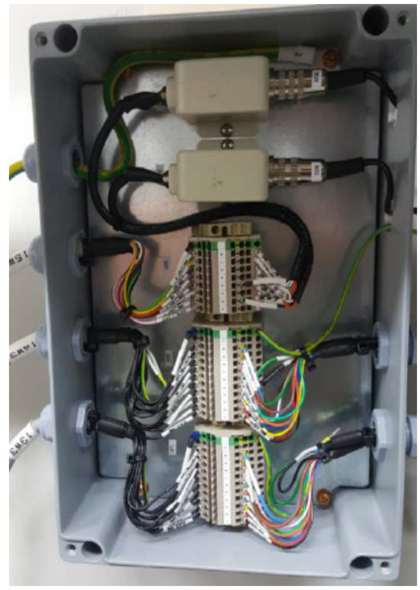


Figure 2: Junction Box in tunnel wall.

The controller uses Beckhoff PLC and it can be controlled by adding various devices to the I/O part even if the device is completed with good scalability and usability. Actually, a device called wire scanner will be connected and controlling. Figure 3 shows the slave unit installed in the gallery and Fig. 4 shows the master unit installed in Server Room.

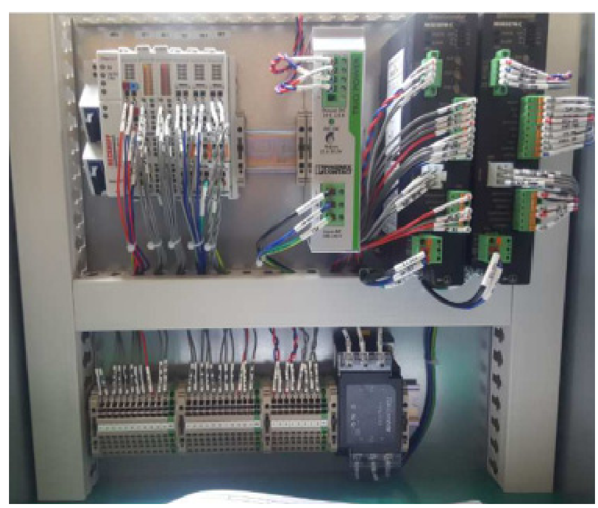


Figure 3: EtherCat Slave unit in Gallery Rack.

* Work supported by Ministry of Science and ICT, South Korea
† yjseo@postech.ac.kr

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA151

A VERY LIGHTWEIGHT PROCESS VARIABLE SERVER*

A. Sukhanov†, J. P. Jamilkowski, Brookhaven National Laboratory, Upton, USA

Abstract

The liteServer is very lightweight, low latency, cross-platform network protocol for signal monitoring and control. It provides very basic functionality of popular channel access protocols like CA or pvAccess of EPICS. It supports request-reply patterns: 'info', 'get' and 'set' requests and publish-subscribe pattern: 'monitor' request. The main scope of the liteServer is: 1) control and monitoring of instruments supplied with proprietary software, 2) seamless connectivity to existing control systems, 3) restricted access to process variables (PVs), 4) possibility to implement the server in FPGA without CPU core. The transport protocol is connection-less (UDP) and data serialization format is Universal Binary JSON (UBJSON). The UBJSON provides complete compatibility with the JSON specification, it is very efficient and fast.

INTRODUCTION

Modern instruments are often supplied with rich proprietary software tools, which make it difficult to integrate them to existing control systems. The initial motivation of developing the liteServer was to build a simple server, which controls a device using supplied proprietary software tools (DLLs or shared libraries) and expose the process variables for access from an existing control architecture. The basic features of the communication protocol are suitable for implementation in FPGA fabric which lacks a CPU core, such firmware could be made fault tolerant using TMR [1] technique.

UBJSON

The UBJSON serialization format have been chosen for communication because it provides complete compatibility with the omnipresent JSON – there is a 1:1 mapping between standard JSON specification and UBJSON [2]. In addition, it has following advantages:

- Ease of implementation. It is feasible to implement it in FPGA fabric.
- Ease of use.
- Speed and efficiency – UBJSON uses data representations that are (roughly) 30% smaller than their compacted JSON counterparts and are optimized for fast parsing. Streamed serialization is supported, meaning that the transfer of UBJSON over a network connection can start sending data before the final size of the data is known.

The Universal Binary JSON specification utilizes a single construct with two optional segments (length and data) for all types:

```
[type, 1-byte char]([integer length])([data])
```

Similarly to JSON, UBJSON defines two container types: **array** (analog of a Python list) and **object** (analog of a Python dictionary).

MESSAGING

The message, received by the server is an UBJSON object with obligatory keys **'cmd'**, **'username'** and **'program'.** The 'cmd' is 2-element UBJSON array, the first element is a liteServer command, the second is an array of arguments. The liteServer command may be one of the following: **'info', 'get', 'set', 'monitor'** and **'retransmit'**. The 'retransmit' command is special. It is used to recover data lost during the transaction. The array of arguments comprises up to four other arrays: device names, parameter names, parameter properties and values. All elements of an array can be requested by specifying an empty array. Example of the command to change frequencies of two devices:

```
{{'cmd': ['set', [['dev1', 'dev2'],
['frequency'], ['value'], [1.0, 2.0]]],
'username': 'JohnDoe', 'program':
'liteAccess.py'}
```

The response message is an object with keys representing device:PV and values representing the requested properties as shown below:

```
{'dev1:frequency': {'value': [1.0]},
'dev2:frequency': {'value': [2.0]}}
```

The **'username'** and **'program'** keys are used for optional access restriction.

THE SERVER

The server listens for request and sends responds through an UDP sockets. Messages, which are too big to fit into a single UDP transfer are chopped into smaller chunks. The position of the chunk is transferred as a first word of the socket message. The client acknowledges the server when the whole message has been received. If a chunk has been lost in the transfer, then the client issues a 'retransmit' request.

In the Python implementation, the liteServer module provides base classes Device and PV for devices and its Process Variables objects to be handled by the server. The

* Work supported by Brookhaven Science Associates, LLC under Contract No. DE-SC0012704 with the U.S. Department of Energy.
† sukhanov@bnl.gov

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA153

A STATE MACHINE SOLUTION TO CONTROL SUPERCONDUCTING CAVITIES

D. Touchard*, R. Ferdinand, M. Lechartier, F. Pillon, L. Valentin, GANIL, Caen, France
Y. Lussignol, IRFU, Saclay, France

Abstract

For the commissioning of the SPIRAL2 accelerating cavities at GANIL, a whole EPICS control-command system has been developed to start the radio-frequency (RF) system. The description of the RF constraints, the functions performed will be discussed to understand the operation of state machines that have been developed. The first results of the commissioning of the control-command of the cavities will be presented.

INTRODUCTION

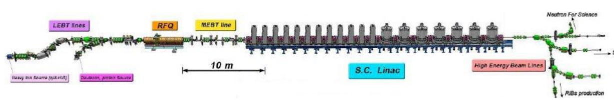


Figure 1: SPIRAL2 facility.

Phase 1 of the SPIRAL2 project includes two ion sources, a Radio frequency quadrupole (RFQ), a linear supra conducting accelerator (LINAC) and two experiment rooms Fig. 1. SPIRAL2's LINAC is capable of accelerating particles such as protons, deuterons, or ions from helium to nickel. The control command system is made up of 40 RF systems and is controlled by 70 tuning applications and 70 programmable logic controllers (PLC) [1]. Very early in the project phase of SPIRAL2, the choice was made to control RF systems using state machines with the State Notation Language (SNL), a domain specific language provided in the EPICS framework. The following article will present the global control command architecture for RF systems. It details the significant contribution of state machines, and the advantages of EPICS. The solutions put in place to control superconducting cavities and the radio frequency quadrupole will be detailed. In July 2019, the French Nuclear Safety Authority authorized to start the LINAC and beam commissioning. First results of the RFQ and LINAC RF commissioning are briefly described.

THE DIFFERENT TYPES OF CAVITY

- Radio frequency quadrupole : this continuous wave warm cavity is a 4 vane RFQ type, 5m long, that can accelerate a 5mA beam of proton or deuteron and a 1 mA beam of light ions (Q/A>1/3) up to 0.73 MeV/A.
- Rebunchers : 3 rebunchers providing up to 120 kV of effective voltage are located in the Medium Energy Beam Transport to keep the beam longitudinal phase dimension
- Supra conductor cavities: the SPIRAL2 LINAC is composed of two families of SC Quarter Wave resonators, providing an accelerating field of 6.5 MV/m and coupled to transfer 7 and 12 kW to the 5mA beams [2].

THE SPIRAL2 RF CONTROL COMMAND SYSTEM

The EPICS framework was chosen from the pre-project phase in order to standardize the software developments of the SPIRAL2 community. A considerable effort has been made to organize and modelize equipment interfaces in order to facilitate high-level applications development [3,4]. The ease of use of the Channel Access (CA) communication protocol which allows very simple access by name to facility beam process values, names Process Variable in EPICS (PV), has considerably reduced the development effort for control applications. These choices allowed an integration of the different projects in progress, such as SPIRAL2, NFS, S3 [5]. The RF control-command system is divided into 6 subsystems: cavities, low level RF (LLRF), frequency tuning systems (FTS), amplifiers, waveguides, and global amplification channels. The LLRF plays a central role in cavity regulation and can be accessed by high level control-command beam tuning applications Fig. 2 [6].

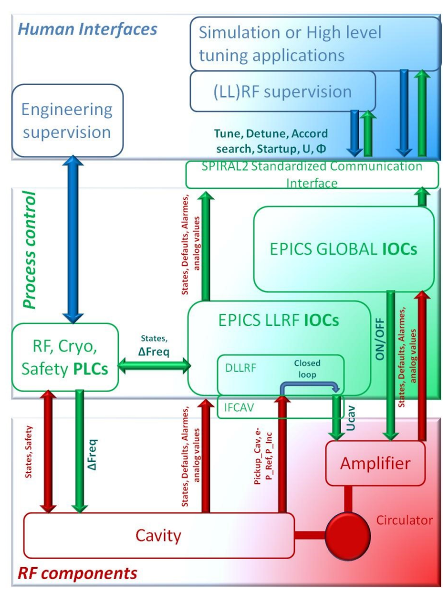


Figure 2: Spiral2 RF control command architecture.

FINITE STATE MACHINES (FSM)

State machine programming methods appeared in the 1990s. Finite state machine method is based on the prin-

* dominique.touchard@ganil.fr

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-WEPHA159

INTEGRATING CONVENTIONAL FACILITIES SYSTEMS VIA BACnet*

S. B. Webb†, ORNL, Oak Ridge, USA

Abstract

Conventional Facility (CF) controls, such as those used for Heating, Ventilation, and Air Conditioning (HVAC) are often developed and operated independent of the accelerator control system using commercial Building Automation Systems (BAS). At the Spallation Neutron Source (SNS), these systems are fully integrated into the machine control system based on the Experimental Physics and Industrial Control System (EPICS) toolkit. This approach facilitates optimal machine performance and enhances reliability. BACnet (Building Automation and Control Networks) is the predominant communication protocol used in the building automation industry, thus inspiring SNS to develop a BACnet/IP software driver [4] for EPICS to enable this integration. This paper describes how SNS uses BACnet and standard EPICS tools to perform custom (complex) chiller sequencing and data collection to study the cooling system performance.

THE SNS CHILLED WATER SYSTEM

The central cooling plant at SNS [1] consist of four 1,200-ton Trane centrifugal chillers, four chilled water pumps, four condenser water pumps, and two cooling towers. The plant provides chilled water to accelerator equipment and non-accelerator equipment (e.g., office complex).

History

The entire chilled water system was originally controlled independent of the accelerator control system by proprietary vendor software. The software was Windows dependent and installed on a local desktop computer in a remote location, separate from Central Control Room (CCR). This remote computer was the only interface to the chilled water system, its alarms and historical data. The sample rate and number of samples of the historical data were restricted by the hardware limitations of the local desktop computer.

PROBLEMS

- Single operator interface to the chilled water system
- Not readily accessible to operations staff
- No direct control or monitoring of the system from EPICS
- Reliance on offsite vendor to resolve problems
- Limited historical data
- Unseen/unknown alarms

* Work supported by Oak Ridge National Laboratory for UT-Battelle, LLC, under contract DE-AC05-00OR22725 for the U.S. Department of Energy.
†webbsb@ornl.gov

BUILDING AUTOMATION

Almost all commercial and industrial heating/cooling equipment is designed and built to work with building automation control systems, such as those provided by Johnson Controls, Honeywell, Trane, Carrier, etc.

The building automation industry standardizes a few communication protocols known as LonWorks, Modbus and BACnet [2] — BACnet being the most popular and the default protocol embedded in nearly all HVAC equipment.

EPICS SUPPORT FOR BUILDING AUTOMATION

Originally, EPICS [3] had no support for building automation. Any integration to HVAC equipment required external relays, instrumentation, and/or third-party gateway devices. This type of integration was slow and provided minimal datapoints from the system. This left some display screens incomplete and operators questioning certain events.

By developing an EPICS BACnet/IP driver [4], these HVAC systems can be fully integrated, directly monitored, and controlled from EPICS. This increases system flexibility, system performance, and improves system understanding and reliability.

INTEGRATION VIA BACNET

After developing the BACnet/IP driver [4], SNS fully integrated the chilled water system into EPICS, and have removed the vendor PC running proprietary software. SNS now manages their entire central cooling plant (chillers, pumps, and cooling towers) independently with an EPICS soft-IOC (Input Output Controller). The soft-IOC interfaces with the chillers directly via their native communication bus, BACnet and also communicates with an Allen-Bradley PLC that controls the pumps and the cooling towers. This made it possible to view and control the entire chilled water system from EPICS based OPIs (Operator Interface) in the CCR.

DRIVER DETAIL

Define a BACnet Device (st.cmd)

Each BACnet device requiring integration must be defined in an EPICS st.cmd file: '*drvBACnetDefineDevice("Chiller1", 21, "eth0", 47808)*'. The definition will include a user given EPICS name (Chiller1) that is unique in your st.cmd file for all devices defined, a BACnet device address/identifier (21), the network interface name of your ethernet card (eth0), and a port number (47808).

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA161

REVISITING THE BUNCH-SYNCHRONIZED DATA ACQUISITION SYSTEM FOR THE EUROPEAN XFEL ACCELERATOR

T. Wilksen†, A. Aghababyan, L. Froehlich, O. Hensler, R. Kammering, K. Rehlich, V. Rybnikov
Deutsches Elektronen-Synchrotron (DESY), Hamburg, Germany

Abstract

After about two years in operation the bunch-synchronized data acquisition as used with the accelerator control system at the European XFEL is being revisited. As we have now gained quite some experience with the current system design it was found to have some shortfalls specifically the offered methods and tools for data retrieval and management. In this paper issues of the current implementation are being discussed and taken as an input for an evaluation of new frameworks readily used by many internet and business companies in the context of modern data collection and management technologies. The main focus is currently put on streaming technologies which are being reviewed with respect to feasibility and adaptability for control system architectures at DESY's accelerator facilities.

INTRODUCTION

The European XFEL is a 3.4 km long X-ray Free-Electron Laser starting in Hamburg, Germany, and ending south of Schenefeld, a city in the neighbourhood of Hamburg. The linear, super-conducting accelerator comprises a photocathode laser-based RF gun followed by a 1.6 km accelerating linac section with 96 superconducting cavity modules installed. The electron energy operating points can be chosen between 11 GeV up to 16.5 GeV with present number of RF stations. The machine can provide up to 2700 electron bunches for each shot at a repetition rate ranging from 1 Hz to 10 Hz. The bunch repetition rate can vary between 100 kHz and 4.5 MHz. After a collimation section in the main tunnel the electron bunches are distributed according to a highly configurable bunch pattern selection into two electron beam lines with each one resp. two undulator sections to produce X-ray photon beams for six possible experiment stations as shown in Fig. 1. Currently the full machine is operated with up to 1000 bunches in the main linac and up to 400 bunches each in the two electron beam lines.

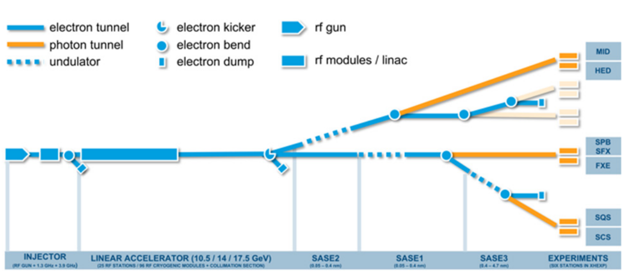


Figure 1: Layout of the European XFEL accelerator.

ACCELERATOR CONTROL SYSTEM

General Layout

The accelerator control system for the European XFEL had been designed using a standard hardware platform based on the MicroTCA.4 technology [1] and a software framework suitable to control a pulsed linear accelerator driving a free-electron laser. As laid out in [2] the choice made uses primarily DOOCS [3] as well as TINE [4] which is integrated by default into the DOOCS core libraries. One of the key requirements was to be able to synchronize all accelerator beam diagnostics data and RF- resp. LLRF information for diagnostic displays and high-level controls or physics applications. For this the accelerator control system layout integrates shot-synchronized and bunch-resolved data acquisition instances as part of the overall control system. This is illustrated in Fig. 2.

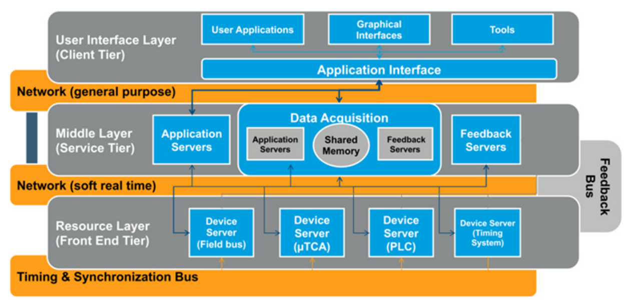


Figure 2: Accelerator control system layout with three layers and integrated data acquisition system on the middle layer.

The integral nature of the data acquisition part within the control system allows for synchronized and efficient online access of all shot-related data. This provides the possibility to plug in services on the middle layer using the shot-synchronized data for high-level applications, doing online processing and computations or even providing slow feedback capabilities. Those services can work seamlessly together with classic middle-layer applications using standard DOOCS calls to retrieve the data.

On the user interface layer the shot-synchronized data is available through dedicated interfaces alongside with data from any other DOOCS channel. Though there is a limitation on the amount of data and its selection with respect to the time range, it is possible to visualize these together with archived DOOCS data.

†Tim.Wilksen@desy.de

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA163

NXCALS - ARCHITECTURE AND CHALLENGES OF THE NEXT CERN ACCELERATOR LOGGING SERVICE

J. Wozniak†, C. Roderick, CERN, Geneva, Switzerland

Abstract

CERN's Accelerator Logging Service (CALS) is in production since 2003 and stores data from accelerator infrastructure and beam observation devices. Initially expecting 1TB / year, the Oracle based system has scaled to cope with 2.5TB / day coming from >2.3 million signals. It serves >1000 users making an average of 5 million extraction requests per day. Nevertheless, with a large data increase during LHC Run 2 the CALS system began to show its limits, particularly for supporting data analytics. In 2016 the Next CERN's Accelerator Logging Service (NXCALS) project was launched with the aim of replacing CALS from Run 3 onwards, with a scalable system using "Big Data" technologies. The NXCALS core is production-ready, based on open-source technologies such as Hadoop, HBase, Spark and Kafka. This paper will describe the NXCALS architecture and design choices, together with challenges faced while adopting these technologies. This includes: write / read performance when dealing with vast amounts of data from heterogenous data sources with strict latency requirements; how to extract, transform and load >1PB of data from CALS to NXCALS. NXCALS is not CERN-specific and can be relevant to other institutes facing similar challenges.

INTRODUCTION

The CERN Accelerator Logging Service (CALS) [1] was designed in 2001, has been in production since 2003 and stores data from all of CERN's accelerator infrastructure and beam observation devices. Initially expecting 1TB of data per year, the Oracle–based CALS system has scaled to cope with a throughput of 2.5TB per day coming from more than 2.3 million signals. It stores 1TB per day for the long-term and serves more than 1000 users from across CERN, who collectively submit 5 million extraction requests per day on average.

CALS is considered as being mission-critical and the go-to service when investigating problems with accelerator equipment or unexpected beam behaviour. Whenever a new system is commissioned, or a new mode of operation is established – there are inevitably subsequent requests to setup the corresponding data logging. This is magnified when new machines or facilities are commissioned.

Since the start of LHC, the scope of CALS and the demands placed upon it have evolved significantly and continue to do so. Figure 1 shows the restart of LHC at the end of 2009, there was an order of magnitude increase in data being logged – mainly coming from the new Quench Protection System. More recently, for the restart of LHC post LS1 (Long Shutdown 1) there was another almost order of magnitude increase in data being logged – mainly due to the need for more beam related data on a bunch-by-bunch and turn-by-turn basis.

Figure 1: CALS long-term daily storage evolution.

The trend of evolving logging needs has continued in recent years with the arrival of new facilities such as AWAKE and MEDICIS, as well as for the commissioning of new machines such as LINAC4.

The CALS system has scaled well in terms persisting acquired data and providing linear response times for data extractions. However, with basic accelerator operation reaching a high level of maturity, attention has turned to more complex analyses such as studying beam effects over longer periods of time. In other words, CALS has increasingly been subjected to extraction of much larger datasets over longer periods of time to support advanced data analytics. It is in this domain, during LHC Run 2, that the CALS system quickly started to show its limits. The CALS Oracle-based architecture is difficult to scale horizontally and does not perform particularly well for large data processing for signals with complex data structures. A key issue is that in order to perform moderately advanced analyses, the data first needs to be extracted and in certain scenarios this takes too long (e.g. for some use cases, it takes 12 hours to extract 24 hours of data). With increasing data volumes, more challenging analyses to be performed and a desire to quickly get answers to questions that can support operations – it was clear that actions needed to be taken.

In 2016, with all of the above knowledge and also aware that the Hi-Luminosity LHC (HL-LHC) machine is scheduled for commissioning in the not so distant future, the NXCALS project was launched with the aim of fully replacing CALS from LHC Run 3 (2021) onwards. The aim being to gain solid operational experience with NXCALS during several years and then have time to further adapt the system as needed during LS3 (Long Shutdown 3), while still ahead of HL-LHC commissioning.

NEED OF "BIG DATA" TECHNOLOGIES

In recent years, the so-called "Big Data" technology landscape has evolved significantly to support large scale

† jakub.wozniak@cern.ch, chris.roderick@cern.ch

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA164

CAFlux: A NEW EPICS CHANNEL ARCHIVER SYSTEM*

Kanglin Xu†, Los Alamos National Laboratory, Los Alamos, USA

Abstract

We introduce a new EPICS channel archiver system that has been developed at LANSCE of Los Alamos National Laboratory. Different from the legacy archiver system, this system is built on *InfluxDB* database. InfluxDB is an open source time series database system that provides a SQL-Like language for fast storage and retrieval of time series data. By replacing the old archiving engine and index file with InfluxDB, we have a more robust, compact and stable archiving server. On a client side, we introduce a new implementation combined with asynchronous programming and multithreaded programming. We also describe a web based archiver configuration system that is associated with our current IRMIS system. To visualize the data stored, we use Javascript Plotly graphing library, another open source toolkit for time series data, to build front-end pages. In addition, we also developed a viewer application with more functionality including basic data statistics and simple arithmetic for channel values.

CAFlux ARCHITECTURE AND SUBSYSTEMS

CAFlux EPICS channel archiver system is developed to replace the legacy archiver system as it is no longer an active project. CAFlux system consists of several components as shown in Fig. 1.

- *A Data Collection Engine* is developed with *Python* asynchronous programming and multithreaded programming based on PyEPICS [1] and EPICS channel access.
- *InfluxDB Server* is used as a data storage engine. InfluxDB [2] is a database system optimized for storage and retrieval of time series data. It supports a few hundred data nodes initially and is able to scale to a few thousand nodes. Its single-node edition is free but the clustering system is sold as commercial product.
- *HTTP Web Server* as a logic tier provides a web service for archiver configuration, data visualization and data stream.
- *MySQL Server* as another data tier holds all the user configuration information.

IMPLEMENTATION OF CAFlux DATA COLLECTION ENGINE

The Data Collection Engine is the core CAFlux subsystem that collects the data through EPICS channel access, caches it in the memory and saves it to the *InfluxDB* server. In this section, we describe the engine in details.

Multitier Architecture

The Data Collection Engine is designed as a multitier architecture, i.e. a lower level engine, a upper level manager and at the highest level, a monitor.

- Lower level engine for core jobs: reading configurations, collecting and caching data, and saving data.
- Lower level engine designed as a daemon and developed with the Python asyncio and threads module.
- Upper level manager for monitoring the low level engine: checking the PID file, restarting the lower level daemon process if it is dead or in zombie status, logging error messages and sending emails if any issue occurs.
- The upper level manager designed and implemented as simple as possible in order to make it more robust and stable enough to run 24 hours a day and 7 days a week with low probability for any issues.
- The highest monitor is a simple script scheduled by *LINUX Cron Daemon* to run every 10 minutes to check the status of the running processes. When a process is dead, it will restart it and send an email notice.

Threads and Asynchronous Task Loop

When we designed the data collection engine, an obvious approach is to use a timer thread for each record to collect data, to save data, and then to sleep for a preset time and wait for the next cycle. The drawback of this approach was that large a number of threads that do work for a short time but sleep most of the time. In addition, the number of threads is limited by resources when the record volume becomes large. Another approach is to start a thread to do work and then to let it die after the work is finished. This approach is complicated because we need a mechanism to start a thread at a preset rate and synchronize a large number of threads to make sure only one thread can access the shared resources. We also consider an *asynchronous* approach that does all the work in the main thread. But we find that one thread is not enough to meet the performance requirement for high volume records.

In this system, we developed an engine through a combination of *Python* asynchronous programming and multithreaded programming. In this way, we can circumvent complex implementation for synchronizing a large number of threads and still have enough working threads to handle a large amount of records with high writing frequency. Specifically, the main thread is

- to read inputs, initialize global data containers and start up
- to initialize CA library and create CA context

* Supported by the US Department of Energy, Los Alamos National Laboratory. Managed by Triad National Security, LLC for the DOE National Security Administration (Contract 89233218CNA000001).
† kxu@lanl.gov

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA165

UPGRADE OF THE EUROPEAN XFEL PHASE SHIFTERS

M. Yakopov†, S. Abeghyan, M. Bagha-Shanjani, S. Karabekyan, J. Pflueger,
F. Preisskorn, European XFEL, Schenefeld, Germany
G.Chen, CAEP, Sichuan, Peoples Republic of China

Abstract

To eliminate the impact of radiation shower on the incremental encoder readout and provide a better dynamic movement the upgrade of all phase shifters of the European XFEL have been successfully done without interruption of the operation schedule. The implementation steps, as well as the results of the hardware and software tests made in the laboratory, are presented. The sensitivity of the Renishaw RGH22O15D00A encoder to the radiation shower was measured in the SASE3 undulator system, and the results are presented.

INTRODUCTION

At the European XFEL undulator systems 88 gap variable phase shifters are in use [1]. They had been equipped with the "Oriental Motor" stepping motors and "Renishaw" linear incremental encoders. To bring the phase shifters into the operational state, the homing procedure was required for the linear incremental encoder. During the operation in the tunnel under the beam condition, at some undefined state, the counts of the encoder have been randomly lost. This effect has not been observed during the tests performed in lab. To bring then phase shifters back into the operational state, the rehoming procedure must have been done, which was breaking the operational routine. Additionally, in the coupled mode [1] at the small undulator gaps the phase shifters have been stuck during the movement because of the speed limitation of the stepping motor.

Based on the experience, gained during the first operation of the phase shifters in the tunnel, the decision has been made to build a test setup using the Beckhoff servomotor with the integrated absolute rotary encoder instead of the old stepping motor to create a solution, which will avoid future operational difficulties of the upgraded phase shifters.

HARDWARE UPGRADE

For the phase shifter upgrade the following main steps were performed:

- Exchange of the motor with the bearing
- Upgrade of the intersection control rack (ICR)[1]
- Integration of the servo axis in the undulator local control system [1]

The ICR has been modified in order to operate the phase shifter with the "AM8122-0F21-0000" servomotor instead of the old stepping motor. The following hardware components have been uninstalled from the ordinary ICR during the upgrade.

† mikhail.yakopov@xfel.eu

Although the servomotor is equipped with the integrated absolute rotary encoder, which is an indirect gap measurement system, the functionality of the existing feedback system based on the incremental linear encoder has been retained and used as a reference direct gap measurement system, to be able to make the initial calibration of the indirect gap measurement system. All related hardware and connections remained identical to the ordinary ICR.

The following hardware components have been installed during the ICR modification:

- EL 7211-0010 PLC servomotor terminal
- PULS CP10.481 DIN rail power supply
- Cable from the ICR to the servomotor

All stepping motor control related hardware was preliminary deinstalled from the ICR:

- EL 2521-0024 PLC pulse train terminal
- EL 2004 PLC digital output terminal
- RKD514LMC, stepping motor driver
- Terminal units
- Cable from the ICR to the stepping motor
- Corresponding wiring

SOFTWARE UPGRADE

The servo axis has been implemented in the system manager configuration instead of the stepping motor axis in order to operate the upgraded phase shifter. To keep the functionality of the existing linear encoder as a direct gap reading system one more special type of axis has been created named "Linear Encoder". This axis is not involved in the motion system and was only used later as a reference direct gap measurement feedback system for the correction curve calculation.

Before replacing the stepping motor with servomotor the scaling factor has been calculated, set in the system manager and the servomotor has been driven to the 40 mm, which is the half of the 80 mm previously set phase shifter gap. Due to the specific mechanic of the phase shifter, the value of the absolute rotary encoder in the servomotor corresponds to the double value of the linear encoder. Once they have been equated to each other, the motor has been replaced together with the flange and the coupling. The appeared mismatch between two feedback systems after motor replacement has been compensated by position bias value. The linear encoder value has been also corrected by position bias value to the 80 mm.

Calculation of the Scaling Factor

The scaling factor (SF) is a crucial parameter for the positioning accuracy. The value of the scaling factor has been calculated by using the formula specified below. The

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA166

DEVELOPMENT OF WEB-BASED PARAMETER MANAGEMENT SYSTEM FOR SHINE

H. H. Lv, Shanghai Institute of Applied Physics, Shanghai, China
Y. B. Leng, Y. B. Yan, Shanghai Advanced Research Institute, Shanghai, China
C. P. Chu, Institute of High Energy Physics, Beijing, China

Abstract

A web-based parameter management system for Shanghai High repetition rate XFEL aNd Extreme light facility (SHINE) is developed for accelerator physicists and researchers to communicate with each other and track the modified history. The system is based on standard J2EE Glassfish platform with MySQL database utilized as backend data storage. The user interface is designed with JavaServer Faces which incorporates MVC architecture. It is of great convenience for researchers in the facility designing process.

INTRODUCTION

SHINE is a hard X-ray free electron laser user facility which is recently proposed. When completed, it will provide sophisticated research methods for physics, chemistry, life sciences, materials science, energy science and other disciplines.

Generally speaking, the modern accelerator is complex and composed of many subsystems. Each subsystem is composed of several units with independent functions, while each unit is composed of thousands of devices. During the designing process, accelerator physicists devise the basic parameters according to the design goals of the facility. After that, researchers working in kinds of subsystems devise the specific devices' parameters for their respective subsystem. Moreover, the parameters of one subsystem depend on the other. For example, the design of the power supply depends on the requirements of the magnet. In reality, the official version of parameters will be identified through several revisions. Traditionally, the parameters are saved in static text files, which is not only hard to guarantee consistent values of plenty of parameters for several subsystems and a great number of devices, but also violence against the management and analysis [1]. As a consequence, it's very essential to develop an advanced system to store and manage the design parameters for SHINE. A user interface on which users could conveniently retrieve or modify the parameters is also desired. Then a web-based parameter management system has been developed to use relational database for data storing. Details of the application architecture, database schema and system functions are described in the following sections.

APPLICATION ARCHITECTURE

The system is based on standard J2EE (Java 2 Platform, Enterprise Edition) Glassfish platform with MySQL database used as backend data storage. J2EE platform contains various components, service architectures and technical levels which have common standards and specifications. It's compatible between different platforms following the J2EE architecture [2]. Therefore, the parameter system could be well integrated with other systems. This makes it feasible to integrate all the data involved in SHINE design, building, installation and commissioning process together.

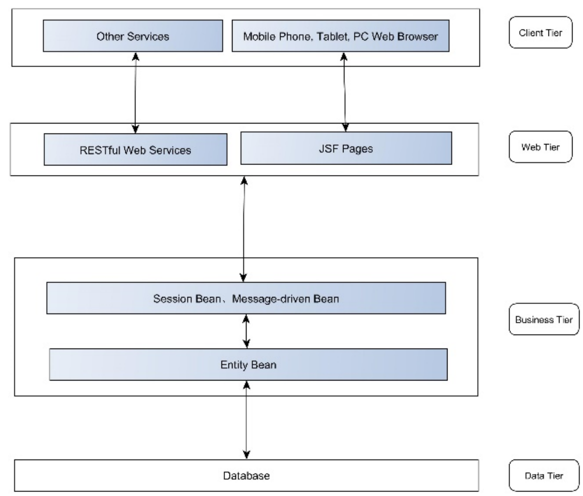


Figure 1: Application architecture.

As shown in Fig.1, the application architecture diagram is composed by four parts: database, business layer, web layer and client layer. The database is a general data storage container for parameter data. The business layer implements the business logic such as retrieving, searching, batch storing, updating and deleting records from database. They are handled by enterprise beans. Web layer includes web pages created by JavaServer Faces technology and services which could be invoked by other services. JavaServer Faces technology builds on servlets and JSP technology. It provides a user interface component framework for web applications. Client layer is a thin client that do not query databases, execute complex business rules, or connect to legacy applications. Such heavyweight operations are offloaded to enterprise beans, where they can leverage the security, speed, services, and reliability of server-side technologies.

Database Schema

The database schema saves the properties as Name/Value pairs and has built a universal database schema. Details are illustrated in Fig. 2.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA167

STATUS OF THE SHINE CONTROL SYSTEM

Y.B. Yan†, J.G. Ding, J.F. Chen, G.H. Chen, Y.B. Leng
H.F. Miao, Y.J. Liu, Q.R. Mi, H. Zhao, C.L. Yu, H.H. Lv, H.Y. Wang, P.X.Yu
Shanghai Advanced Research Institute, Chinese Academy of Sciences, Shanghai, P.R. China

Abstract

The high-gain free electron lasers have given scientists hopes for new scientific discoveries in many frontier research areas. The Shanghai HIgh repetition rate XFEL aNd Extreme light facility (SHINE) is under construction in China, which is a quasi-continuous wave hard X-ray free electron laser facility. The control system is responsible for the facility-wide device control, data acquisition, machine protection, high level database or application, as well as network and computing platform. It will be mainly based on EPICS to reach the balance between the high performance and costs of maintenance. The latest technology will be adopted for the high repetition rate data acquisition and feedback system. The details of the control system design will be reported in this paper.

OVERVIEW

Owing to the wide range of applications of X-rays in the research fields of physics, chemistry and biology, facilities with the ability to generate X-rays were developed continuously in the last century. The free electron laser (FEL) is a novel light source, producing high-brightness X-ray pulses. To achieve high-intensity and ultra-fast short wavelength radiation, several X-ray FEL facilities have been completed or under construction around the world [1].

The first hard X-ray FEL light source in China, the so-called Shanghai HIgh repetition rate XFEL aNd Extreme light facility (SHINE), is under construction. It will utilize a photocathode electron gun combined with the superconducting Linac to produce 8 GeV FEL quality electron beams with 1 MHz repetition rate.

The control system of SHINE is responsible for the facility-wide device control, data acquisition, machine protection, high level database or application, as well as network and computing platform. It will provide operators, engineers and physicists with a comprehensive and easy-to-use tool to control the machine components to produce high quality electron beam and free electron laser.

According to the experience of SSRF and SXFEL, the control system of SHINE will be mainly based on EPICS (Experimental Physics and Industrial Control System) to reach the balance between the high performance and costs of maintenance. EPICS is a set of open source software tools, libraries and applications developed collaboratively and used worldwide to create distributed soft real-time control systems for scientific instruments such as particle accelerators, telescopes and other large-scale scientific experiments.

ARCHITECTURE

As shown in Fig. 1, the control system can be divided into four layers to ensure the performance and scalability, which are operator interface layer, middle layer, device control layer and data acquisition layer.

The operator interface layer offers graphical user interface (GUI), command line interface (CLI) and high-level application programming interface (API) to the operators, engineers and physicists. It allows them to interact with the machine components.

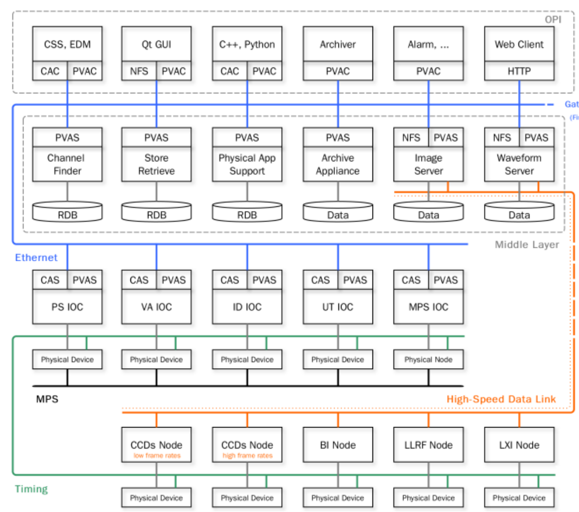


Figure 1: Architecture of the control system.

The middle layer consists of compute, storage and network devices. It provides the runtime environment for the whole accelerator control system. It also undertakes the centralized processing tasks for image and stream data acquisition system.

The device control layer is responsible for the facility-wide input and output device control, such as magnet power supply control, vacuum gauge control, stepper motor control and so on. The machine protection and timing system will be also implemented in this layer. They are the basis components of the control system.

The data acquisition layer is designed for the high speed image and stream data acquisition, processing and storage. It involves the beam and laser diagnostics, microwave related system. Some custom software will be used at this layer.

† yanyingbing@zjlab.org.cn

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA168

STATUS OF THE TPS VACUUM CONTROL SYSTEM

Y.C. Yang†, Y.Z. Lin, J.Y. Chuang, C. K. Chan and C. C. Chang
NSRRC, 101 Hsin-Ann Road, Hsinchu Science Park, Hsinchu, Taiwan, R.O.C.

Abstract

The Taiwan photon source (TPS) is a 3 GeV photon source. For the vacuum system NI CompactRIO controllers with embedded real-time processors and programmable FPGAs were selected to design the interlock system to maintain ultra-high vacuum conditions and protect vacuum devices. The vacuum pressure protection function and component protection logics worked well during the past years of operation. Besides, basic function and other applications such as TCP/IP Modbus communication and real time message APIs were developed. The architecture of the vacuum control system is presented in this paper.

INTRODUCTION

The Taiwan Photon Source (TPS), a low-emittance 3-GeV synchrotron ring, started in December 2014 and is now currently operated in top-up mode at 400mA for users. During the past years of operation, the vacuum system worked well [1], the dynamic pressure per beam current continuously decreased. Until the last machine shutdown in August 2019 and reached a value of $1.10x10^{-10}$ Pa/mA with a 6417Ah accumulated beam dose.

In the TPS vacuum control system, Compact-RIO (C-RIO) real time controllers from National Instrument serve for vacuum data acquisition, monitoring and safety interlock. All programs were developed in the LabVIEW language, which offers a graphical programming approach that visualizes the application. EPICS channels were built via embedded EPICS I/O client and server in the LabVIEW project to connect vacuum system with the TPS control system. In addition, distributed EtherCAT I/O modules were chosen to expand temporary monitor signals.

One real time message system, combining EPICS, LabVIEW, python and LINE APIs, was developed. Real time messages could be received on cell phones by inquiry and message push functions. The hardware architecture of the vacuum control system, data archive system, safety interlock logic and real time message API will be presented in this paper.

HARDWARE ARCHITECTURE

The design of the 518.4m circumference TPS storage ring is divided into 24 sections and each section is assigned to one control instrument area (CIA). In each vacuum section, NI CompactRIO (C-RIO) controllers are used to control the vacuum system, serving as data acquisition, numerical transfer and interlock protection function. Between the C-RIO and vacuum system, the interface of the I/O communication is used by an I/O port or terminal for the vacuum components, such as vacuum gauges, pumps and vacuum valves. A total of eight I/O modules, including three analogue inputs (AI), three digital inputs (DI) and two digital outputs (DO), are used and installed and one CIA serves 1/24 of the vacuum sections as shown in Fig 1. Analogue signals, like pressure readings of vacuum gauges and pumps are taken as the basic logic signals for safety interlock and cooling water flow rate as a vacuum component protection issue. Digital input and output signals provide the status, set-point, logic trigger and remote control of the vacuum system.

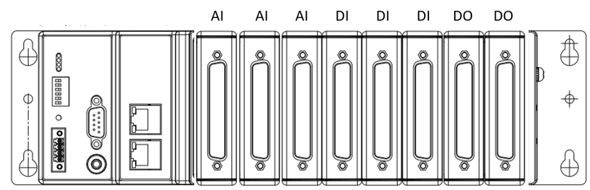


Figure 1: The C-RIO controller with modules.

All vacuum C-RIO controllers are operated under vacuum private network. In each C-RIO, an EPICS server was built to publish shield variables in the LabVIEW project to process variables (PV) in the TPS control network. The setting page of the EPICS server is shown in Fig. 2. Furthermore, one additional C-RIO was assigned as the EPICS client to communicate with the TPS EPICS server, which acquires machine information, such as operating beam current. The architecture of the vacuum control system is shown in Fig. 3.

During the past operations period, few events occurred. Temporary signals built and monitored were necessary. An EtherCAT I/O module was chosen to communicate by TCP/IP Modbus protocol. All EtherCAT modules are administrated by one TCP/IP Modbus server built into a C-RIO. In the TPS vacuum system, several RTD modules were used to monitor the temperature distribution near beam position monitors (BPM) which reach a higher temperature during machine operation.

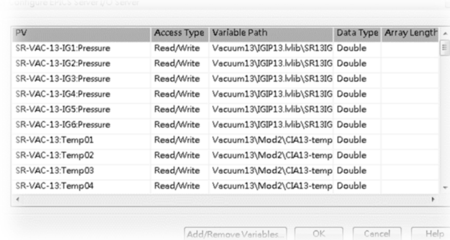


Figure 2. Embedded EPICS server in the LabVIEW project.

†jolas@nsrrc.org.tw

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA170

FIRST STEPS IN AUTOMATED SOFTWARE DEVELOPMENT APPROACH FOR LHC PHASE II UPGRADES CO_2 DETECTOR COOLING SYSTEMS

L. Zwalinski†, J. Daguin, L.T. Davoine, N. Frank, D. Giakoumi, M. Ostrega, P. Petagna, P. Tropea, B. Verlaat, CERN European Organization for Nuclear Research, Geneva, Switzerland

Abstract

In the context of the development of a new class of movable, medium size, evaporative CO_2 cooling systems, the EP-DT group at CERN has designed and prototyped the systems Light Use Cooling Appliance for Surface Zones (LUCASZ). The use of Schneider M580 PLC with Ethernet IP distributed Inputs/Outputs (I/Os) and automated software generation tools is the novel approach which is essential for multiple unit production. The paper describes how the selected software and controls hardware concepts, used for the control system of this new class of units, will be implemented as baseline solutions for the CO_2 cooling systems for Phase II upgrade of ATLAS and CMS silicon detectors. The main challenges for future control system development will come from the number of cooling plants, the modularity requirements for production and operation, and the implementation of the backup strategy requested by the detector operation constraints. The introduction of automated software generation for both PLC and SCADA is expected to bring a major improvement on the efficiency of the control system implementation. In this respect, a unification step between experiments is highly required, still without neglecting the specific needs of ATLAS and CMS.

INTRODUCTION

The LUCASZ [1] CO_2 cooling plant, designed for detector testing, is a movable system that uses the so-called I-2PACL concept [2]. I-2PACL stands for Integrated 2 Phase Accumulator Controlled Loop. The I-2PACL cycle is very similar to a standard 2PACL (2 Phase Accumulator Control Loop) cycle [3], but the accumulator saturation pressure control is done via a single PID heater controller rather than by a split range PID controller. In other words, the accumulator cooling is done by the cold liquid supplied directly from the pump rather than by a branch from the primary chiller. The advantages of such configuration are a larger operational temperature range, a simplified design, reduced cost and programming simplification. The disadvantages are reduced cooling capacity and lower system efficiency.

To cope with the various detector needs, the LUCASZ system has been designed in two versions: LUCASZ^{full} Fig. 1 and LUCASZ^{lite}. The full version is equipped with a detachable LocalBOX, featuring two cooling loops and the dedicated control instrumentation to allow for multiple automated functionalities which facilitate the system operation for non-expert users.

The characteristics of distributed architecture, long term availability, modularity and small series of units were the driving factors for the selection of industrial controls hardware. Although the PREMIUM series of Schneider Electric, have been widely employed in similar projects at CERN, their commercialization will be discontinued and we have decided to identify an alternative family of products to be tested and validated.

Figure 1: Two LUCASZfull units at CMS.

CONTROLS

Controls Architecture

Each LUCASZ^{full} unit is composed of two major elements: the plant and the LocalBOX. In order to provide the maximum installation flexibility, the two parts might be kept together or physically separated. As an example, for one of the CERN CMS application, the CO_2 instrumentation is distributed over of about 20 m distance. The LocalBOX is kept inside the detector testing clean room and the cooling plant is placed outside, for noise reduction purposes and ventilation.

The master control cabinet showed on Fig. 2, located on the plant skid, is hosting the M580 Schneider Programmable Logic Controller (PLC) equipped with its own I/Os cards connected on the same backplane. The plant interfaces to the deported LocalBOX control cabinet are done via Ethernet IP filed network.

† lukasz.zwalinski@cern.ch

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA174

ADUVC - AN EPICS AREADETECTOR DRIVER FOR USB VIDEO CLASS DEVICES

J. Wlodek[1], K. J. Gofron†, Brookhaven National Laboratory, Upton, USA
[1]also at Stony Brook University, Stony Brook, USA

Abstract

Most devices supported by the Experimental Physics and Industrial Control System [1] (EPICS) areaDetector [2] project fall under one of two categories: detectors and cameras. Many of the cameras in this group can be classified as industrial cameras, and allow for fine control of exposure time, gain, framerate, and many other image acquisition parameters. This flexibility can come at a cost; however, with most such industrial cameras' prices starting near one thousand dollars, with the price rising for cameras with more features and better hardware. While these prices are justified for situations that require a large amount of control over the camera, for monitoring tasks, and some basic data acquisition, the use of consumer devices may be sufficient while being less cost-prohibitive. The solution we developed was to write an areaDetector driver for USB Video Class (UVC) [3] devices. This driver allows for a variety of cameras and webcams to be used through EPICS and areaDetector, with most costing under $100.

INTRODUCTION

The EPICS areaDetector project enables universal camera and detector control for EPICS installations through the use of camera-specific driver implementations that all share an underlying base, called ADCore. Currently supported devices include many models of X-ray detectors which are used extensively for data collection, as well as a large array of industrial cameras such as Prosilica (Vimba) [4] devices, Flir (Point Grey) [5] devices, and more. These industrial cameras are used for some data collection, but are primarily used for monitoring purposes, allowing scientists to see a live feed of their experimental stage and experiment itself during its duration.

These industrial cameras often have expansive feature sets with various image acquisition and processing modes that allow for fine-tuned control of the feed shown to the scientists. This level of customizability is important in several use-cases (particularly for taking images with different exposure times); however, in some cases it may be unnecessary. In such situations, a cheaper more flexible alternative to these expensive industrial cameras would perhaps be more suitable.

To solve this problem, we looked for a set of cameras that were inexpensive, used a standardized communication interface, and had a readily available C/C++ library that could be integrated into an areaDetector driver. Taking into account these criteria, we settled on the UVC standard that is used for most consumer class webcams and USB video devices. This standard encompasses a vast number of supported cameras ranging from under $20 webcams to several hundred dollar industrial cameras that support many of the same features allowed by the more traditional ones already included with areaDetector. In this paper we will discuss how we added UVC device support to EPICS areaDetector, the construction of the ADUVC driver [6], and some supported cameras and deployment results and challenges.

THE UVC STANDARD

The UVC standard is utilized almost universally among USB cameras and video devices, most notably in consumer webcams. Most operating systems that allow for viewing camera images via a dedicated application (i.e. the Windows 10 "Camera" application) are internally using a custom UVC driver for camera control. The UVC standard focuses on defining camera settings and variables common to most cameras and abstracting them away with a high-level communication protocol that can be reused between most devices, much like how ADCore contains functionality common to all areaDetector cameras. This universality allows for UVC to be used to control and use a huge number of devices with varied output resolutions, framerates, and image formats. UVC enforces certain requirements that must be satisfied by the firmware of each device:

- Each UVC device must have a series of predefined formats within firmware that can be read according to the standard, supplying all required image information to the driver prior to streaming initialization, including the data type of the output, and the color mode of the resulting image (Table 1).
- Each UVC device must support standard commands sent over the USB interface for starting and stopping streaming.
- Each UVC device must send frames back to the driver in a standardized format.

Table 1: Some Supported UVC Frame Formats

Format	Data Type	Color Mode
MJPEG	UInt8	RGB
RGB	UInt8	RGB
Grayscale	UInt8/UInt16	Mono
YUYV	UInt8	RGB/Mono
Uncompressed	UInt8/UInt16	RGB/Mono

† kgofron@bnl.gov

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WESH1002

NEW JAVA FRAMEWORKS FOR BUILDING NEXT GENERATION EPICS APPLICATIONS

K. Shroff, National Synchrotron Light Source II, Upton, USA
C. Rosati, G. Weiss, European Spallation Source, Lund, Sweden
K. Kasemir, E. Smith, Oak Ridge National Laboratory, Oak Ridge, USA

Abstract

Phoebus is a Java/JavaFX framework for creating state-of-the-art, next-generation desktop applications for monitoring and controlling EPICS systems. The recent developments in Java and JavaFX have made it possible to reconsider the role of the Eclipse Rich Client Platform (RCP) in the development of client applications. Phoebus's aim is to provide a simple to use and yet "rich-enough" application framework to develop modular JavaFX desktop applications for the most recent Java platform. Phoebus is an extensible framework for multiple control system protocols. It provides features for developing robust and scalable multi-threaded client applications. Key features include event rate decoupling, caching and queuing, and a common set of immutable data types to represent controls data from various protocols. The paper describes the framework as used to implement applications and service for monitoring EPICS PVs. The benefits highlighted will provide the EPICS community a new development perspective.

MOTIVATION

Since 2006 Control System Studio[1, 2, 3] has been adopted as the graphical user interface for the control systems of many Accelerators at various universities and laboratories. The growth of an active collaboration has accompanied this increase in adoption. In 2018 a survey was conducted of both the end users of the tools and applications included in the CS-Studio products as well as the developers that built and maintained them. The results of the survey highlighted a consistent set of liked and disliked aspects of the CS-Studio project.

The users generally liked the available set of applications and the integrated environment of CS-Studio allowed for effective and intuitive workflows however the product was too much like an IDE with performance and reliability issues. The developers appreciated the extensible architecture which allowed them to easily add new applications, expand the functionality of the common product, and integrate with their existing infrastructure however the learning curve was very steep. In particular, understanding the build system and OSGi life cycle presented a significant barrier to new participants. Most of the identified issues were associated with CS-Studio's dependence of the Eclipse Rich Client Platform [4] and thus the Phoebus project [5, 6] was initiated to serve as a replacement of the Eclipse RPC framework.

INTRODUCTION TO PHOEBUS

Historical Considerations

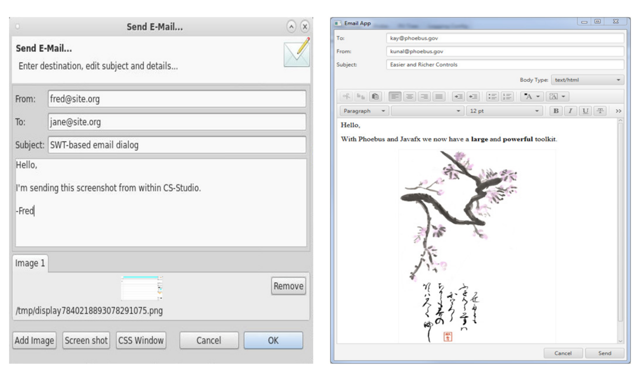
In 2004 the Eclipse rich client platform provided a great array of features for developing extensible client applications. It supported a modular architecture with support for extensibility via pluggable extension points and plugin life cycle management. It had its own build system and a workbench consisting of views, editors, and perspective. It also provided support for managing preferences, logging, native language support, and updates.

Current Situation

The benefits of using the Eclipse RCP framework are currently accompanied by some major drawbacks. The build system has increased in complexity which has made it difficult to understand and manage. For example, a complete compilation of CS-Studio from sources can take almost an hour

A lot of the features of the Eclipse RCP framework are now part of the java language itself and are better supported by a larger community. The modular architecture of Eclipse built on OSGi bundles/ Eclipse plugins could be replaced with the use of Java modules. The Java service provider interface (SPI) provided a viable alternative to the Eclipse extension points.

Additionally, the Eclipse framework ties us to the standard widget toolkit (SWT) [7], which was the best GUI library a decade ago, but is being overshadowed by JavaFX [8]. The JavaFX library, which was developed to replace swing as the standard GUI library for the java se, is a far more feature rich and better performing alternative to SWT. The library comes with a rich set of well designed easy to use widgets, supports properties bindings, CSS, etc.



SWT dialog Javafx Dialog

Figure 1: Comparison of dialogs created using SWT and JavaFX GUI libraries.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WESH1003

jddd MIGRATION TO OpenJDK11+: BENEFITS AND PITFALLS

E. Sombrowski, K. Rehlich, G. Schlesselmann, DESY, Hamburg, Germany

Abstract

The Java Doocs Data Display (jddd) is a Java-based tool for creating and running graphical user interfaces for accelerator control systems. It is the standard graphical user interface for operating the European XFEL accelerator. Since Java 8 Oracle introduced a number of major changes in the Java ecosystem's legal and technical contexts that significantly impact Java developers and users. The most impactful changes for our software were the removal of Java Web Start, Oracles new licensing model and shorter release cycles. To keep jddd up to date, the source code had to be refactored and new distribution concepts for the different operating systems had to be developed. In this paper the benefits and pitfalls of the jddd migration from Oracle Java8 to OpenJDK11+ will be described.

INTRODUCTION

jddd [1, 2] is a common tool for designing and running control system windows (also called panels) at DESY Hamburg and DESY Zeuthen. It has a graphical editor with a rich set of ready-made widgets for control panel design. Synoptical displays can easily be created without any programming knowledge. So far, more than 15000 panels have been designed in Hamburg and Zeuthen.

At DESY Hamburg statistical data about the jddd usage are collected. The average and maximum number of panels started per day, users per day and sessions started per day are displayed in Table 1. In recent years usage has been steadily increasing.

Table 1: jddd Usage Statistics 2018 at DESY Hamburg

	#started panels per day	#started diff. panels per day	#started jddd sessions per day
av	2982	558	169
max	7221	1089	434

JAVA UPGRADE

Since Java 8 Oracle made significant modifications in the Java ecosystem [3]. To keep jddd up to date, new solutions had to be found for the following changes:

Oracles New Licensing Model

Starting with Java 11 Oracle offers two distinct Java releases with different license models:

- Oracle JDK under commercial OTN License Agreement for Java SE [4]: This release offers a long term support, but is only free of charge for development and tests. For commercial use high costs incur.
- Oracle OpenJDK under the open source GNU General Public License v2 with Classpath Exception (GPLv2+CPE): This release is free of charge, but does not offer long term support.

As an alternative to Oracle a zoo of long term support OpenJDK distributions from different communities and vendors is available, like AdoptOpenJDK, Corretto by Amazon or Read Hat OpenJDK.

Oracle has been and will stay the reference implementation for Java. We decided to use Oracle OpenJDK, because the license model fits our needs and environment best.

Dealing with Shorter Release Cycles

Some months ago Oracle introduced a new release cycle for Java. It changed from a feature-based to a time-based release cycle. There are now two types of releases:

- Major releases for Oracle JDK and OpenJDK every 6 months, which are only supported until the next release.
- Commercial Oracle JDK Long Term Support (LTS) releases every 3 years: The most recent LTS release is Java 11, which came out in September 2018 and will be supported until 2026. The next will be Java 17 in September 2021.

The advantage of short release cycles is that new Java features will be usable much faster.

It also makes migration easier. The changes are smaller and more incremental, so each upgrade is easier and less of change.

We started jddd migration from Java 8 to OpenJDK11+ in September 2018 with OpenJDK 11, which turned out to be quite easy. No major code changes had to be done. Since OpenJDK11 jddd application are always compiled with the latest OpenJDK release without any problems.

Replacement of Java Web Start

During the past years Java Web Start and Java Network Launching Protocol (JNLP) were used for the internal distribution of jddd to the users. To access the application the users had to install a current Java that contained support for JNLP. To start jddd, they would click on a JNLP link and the Java Web Start program would download the JNLP file, interpret it, download the current version of the application, and run it in a security sandbox.

Java Web Start provided an easy and uncomplicated way to distribute a completely configured application to everyone's desktop independent of the operating system. As Java Web Start is no longer part of Java, a substitution for JNLP had to be found.

Java 8 is the last version providing a separate Java Runtime Environment (JRE). Since Java 9 there is only the Java Development Kit (JDK) available, which is basically the JRE plus a compiler, a debugger and several other tools. Instead of a separate JRE Oracle introduced the module system Jigsaw. The idea of Jigsaw is to modularize the Java

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WESH2001

CS-STUDIO ALARM SYSTEM BASED ON KAFKA*

K.-U. Kasemir, Oak Ridge National Laboratory, Oak Ridge, USA

Abstract

The CS-Studio alarm system was originally based on a relational database and the Apache ActiveMQ message service. The former was necessary to store configuration and state, while the latter communicated state updates and user actions. In a recent update, the combination of relational database and ActiveMQ has been replaced by Apache Kafka. We present how this simplified the implementation while at the same time improving performance.

OVERVIEW

The Control-System-Studio (CS-Studio) alarm system was developed to support control room operators [1]. It monitors a configurable list of process variables (PVs) from the Experimental Physics and Industrial Control System (EPICS), using either the Channel Access or the newer PV Access network protocol [2]. Whenever a PV enters an alarm state, the alarm system remembers the time and value of the alarm. It lists active alarms on a user interface and will optionally also verbally annunciate each new alarm. Operators can inspect the list of active alarms, access guidance on how to handle the alarm and open display panels that offer more detail for the affected subsystem. After operators acknowledge the alarm and the PV returns into a normal state, the alarm clears.

The alarm system is fully distributed. You may install one or more alarm configurations, and one or more operators can concurrently interact with them.

ORIGINAL IMPLEMENTATION: RDB & JMS

The alarm system needs some way to store its configuration, consisting of the PVs to monitor, their associated guidance and related display links. In addition, the system requires a message bus to communicate PV state changes, annunciation messages and operator actions like acknowledgements.

In the original implementation, a relational database (RDB) stores the configuration, and Apache ActiveMQ provides the message bus (JMS). Each alarm configuration is handled by one alarm server process, typically running as a Linux service. Operators can start one or more alarm clients to interact with the alarm system. All servers and clients can share the same RDB and JMS instance.

This appeared to be a good design choice at the time because an RDB is very well suited for storing information, while JMS is a high-performance message bus. The implementation of the alarm system toolkit was, however, complicated by the fact that interactions with storage and the message bus typically overlap.

When an alarm client starts up, it reads the configuration, which can take several seconds. While it is reading one such snapshot of the configuration, other alarm clients may concurrently change the configuration. Such changes are communicated via the message bus. A proper alarm client implementation thus needs to:

1) Subscribe to JMS and buffer received change indications in memory.
2) Read the last saved configuration from the RDB.
3) Apply the buffered changes to update the in-memory configuration to the current state.
4) From then on directly apply received JMS change indications to the in-memory state.

Similarly, the alarm server implementation needs to:

1) Send alarm state changes via JMS so that running alarm clients can receive them with minimal delay.
2) Write the most recent state of each alarm to the RDB, so that new alarm client instances can start with the most recent alarm state as persisted in the RDB and don't need to wait for a state change via JMS to be up-to-date.

While this was successfully implemented, details of the software appeared overly complicated. Of the two key support technologies, JMS and the RDB, the latter was a performance bottleneck. Sending all messages through JMS on the other hand proved to be very efficient and allowed the addition of logging and analysis tools, for example to determine which alarm triggers most frequently, without impacting the running alarm servers and clients [3].

NEW IMPLEMENTATION: APACHE KAFKA

When porting the alarm system toolkit from the original Eclipse-based CS-Studio development to a standalone platform [4], we used this opportunity to investigate new technologies for storage and message bus, i.e. the RDB and JMS functionality.

* This manuscript has been authored by UT-Battelle, LLC, under contract DE-AC05-00OR22725 with the US Department of Energy (DOE). The US government retains and the publisher, by accepting the article for publication, acknowledges that the US government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this manuscript, or allow others to do so, for US government purposes. DOE will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan (http://energy.gov/downloads/doe-public-access-plan).

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WESH2002

EPICS pva ACCESS CONTROL AT ESS

G. Weiss, ESS, Lund, Sweden

Abstract

At ESS (European Spallation Source), pva (PV Access) has been selected as the default EPICS protocol. However, initial releases of EPICS version 7 do not implement any access control of client requests in pva. In order to control write operations that may cause harm to the system, access control is highly desirable. This paper details how the ca (Channel Access) access security concept is reused and extended for pva access control. It also explains how ESS intends to deploy and manage access control in terms of infrastructure, tools and responsibilities. Limitations of the access control mechanism are also discussed.

SCOPE

The scope of pva access control is to provide means to restrict write operations on selected/critical PVs to a predefined group of users, sending requests from a predefined list of hosts. Access control of read requests is **not** in scope.

ca ACCESS SECURITY RECAP

Before going into the details of pva access control, it may be of value to recap the features of the ca access control.

When a ca client sends a read or write request on a PV, it may optionally send a client and host identity. These identities are arbitrary strings set by the client. In practice however, the client identity is set to the user name of the process running the client software, and the host identity is set the host name running the client software. This is the case with the EPICS ca command line clients *caget* and *caput*.

On the receiving side, the IOC server may use the client and host identity to grant or deny access to a PV. The rules governing the access control are defined in ACFs (Access Control File) read by the IOC on boot, or when an explicit ACF reload is requested. If the IOC does not load an ACF on boot, all requests are allowed from all users and hosts.

Rules defined in the ACF are based on matching identities to the client and host identities sent by the clients. As an example, consider the following ACF:

```
UAG(uag) {user1, user2}
HAG(hag) {host1, host2}
ASG(DEFAULT) {
    RULE(1, READ)
    RULE(1, WRITE) {
        UAG(uag)
        HAG(hag)
    }
}
```

Here the user access group (UAG) named **uag** lists client (user) identities **user1** and **user2**, and the host access group (HAG) named **hag** lists hosts **host1** and **host2**. The DEFAULT rule – which applies if no other rule is defined – states that all clients and hosts are granted read access, while write access is granted only for **user1** and **user2** on **host1** or **host2**.

By default, all database records in the IOC are associated with the DEFAULT ASG rule, but individual records may use the ASG field to point to some other rule in the ACF.

A detailed description of ca access control is found in reference [1].

pva ACCESS CONTROL

Starting from EPICS version 7.x (TBD), IOCs and PV Access gateways support access control along the same lines as ca access control. There are however some important differences to consider:

- Only write requests (e.g. through pvput) are subject to access control. All read requests are granted to all user identities on all hosts, irrespective of explicit READ rules in the ACF.
- UAG rules are more flexible as definitions may list user groups in addition to user names.
- HAG definitions may list either IP addresses or host names, or both.

To expand on the previous example ACF example, consider the following content of an ACF:

```
UAG(uag) {role/group1, role/group2, user1}
HAG(hag) {10.20.30.40, host2}
ASG(DEFAULT) {
   RULE(1, WRITE) {
      UAG(uag)
      HAG(hag)
   }
}
```

User groups are identified using the syntax **role/<group>**, where the prefix **role/** is fixed.

For the IOC to grant write access using the above ACF, the user identity provided in the client request must be member of either **group1** or **group2**, or the user identity may be **user1**.

The list of groups of a user identity is determined by the operating system of the IOC's host. In many cases such a list would be a union of local groups and groups managed by some authentication service like LDAP or Active Directory. The pva access control implementation does **not** directly query external services for group information, it only depends on data provided by the host operating system via a system call.

The above ACF also limits write requests to clients on a host with IP address **10.20.30.40** and a host named **host2**. In contrast to ca access control, HAG rules are based on the IP address associated with the incoming write request. Host name lookup is performed when the ACF is parsed. When

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WESH2003

TOWARD CONTINUOUS DELIVERY OF A NONTRIVIAL DISTRIBUTED SOFTWARE SYSTEM *

S. Wai†, SARAO, Cape Town, South Africa

Abstract

The MeerKAT Control and Monitoring(CAM) solution is a mature software system that has undergone multiple phases of construction and expansion. It is a distributed system with a run-time environment of 15 logical nodes featuring dozens of interdependent, short-lived processes that interact with a number of long-running services. This presents a challenge for the development team to balance operational goals with the continued discovery and development of useful enhancements for its users (astronomers, telescope operators). Continuous Delivery is a set of practices designed to always keep software in a releasable state. It employs the discipline of release engineering to optimise the process of taking changes from source control to production. In this paper, we review the current path to production (build, test and release) of CAM, identify shortcomings and introduce approaches to support further incremental development of the system. By implementing patterns such as deployment pipelines* and immutable release candidates we hope to simplify the release process and demonstrate increased throughput of changes, quality and stability in the future.

INTRODUCTION

The Control and Monitoring Subsystem (CAM) for the MeerKAT radio telescope consists of many different software components that work in tandem to allow operation of the telescope as a single, cohesive instrument[1]. However, the large amount of moving parts and differentiation presents a challenge in terms of software engineering complexity[1]. So far, this has been mitigated by leveraging virtualisation with automated deployments[3], as well as continuous integration and automated testing[4]. This has worked well, allowing the software team to adopt an incremental development model for extending the system. Enhancements are developed and released, usually 2-3 months apart[2].

The "Last Mile"

Despite having a high level of automation, the process of releasing newly developed functionality has not been problem-free. Misconfiguration, exceptional states and errors still manifest during some deployments. These prompt long, arduous fault-finding and troubleshooting efforts for the System Team[3] during releases.

Fortunately, releases are scheduled on "maintenance" days during which science operations are ceased so that engineering teams can work on the relevant subsystems and on-site maintenance – such as replacing of physical parts – can take place. This gives a window of opportunity for deploying CAM to site, with a full deployment (and verification activities) taking 2-3 hours in the best case scenario, but can take up the full day if issues are encountered.

This is a common problem in software development: the presence of bottlenecks, often in the final stages of software development lifecycle (the "last mile")[4]. Continuous Delivery[5] offers an approach to software delivery that addresses these problems by focusing on engineering for *feedback*, early and frequently. Some informal analysis was undertaken with this in mind to identify some of the root causes of the problems. They are outlined below:

Gaps in Automated Testing In spite of the sophisticated integration testing in place by the Automated Qualified Framework(AQF)[4], it failed to catch some categories of errors. AQF tests are executed in a static, single-node environment that is only partially deployed on each run: a python script, `kat-update.py` would pull the latest `master` branch of all CAM Python packages on the AQF node. In this scenario, the full deployment procedure is not exercised, thus some problems relating to configuration changes outside of code changes to Python packages cannot be discovered in AQF. Additionally, since it was a long-lived static environment it was vulnerable to *configuration drift*[6].

Build Provenance A consequence of running AQF on a scheduled timer instead of being triggered by changes, is that the provenance of a test result depends on the state of `master` at the time it was run and completed. Once an AQF test execution succeeds, a special branch (`stable`) is updated to mirror the state of `master`, for all relevant CAM packages on the AQF node. Given that an AQF run can take up to 3-4 hours, there is risk that the state of `master` at the time of completion is not the same as when the tests began[5].

* Work supported by South African Radio Astronomy Observatory, National Research Foundation
† swai@ska.ac.za
[1] The system design itself has proven to be robust as it is extensible: CAM has been chosen as the reference architecture for the Square Kilometre Array's Telescope Manager[2].
[2] At the time of writing (September 2019), version 23 of CAM system was deployed to production.
[3] A subset of the Software Team responsible for deployment of the CAM subsystem and overall health of the datacentre servers running CAM.
[4] It is apparent that *continuous integration* alone is insufficient, as most of the problems occur when trying to release already integrated code to the users.
[5] Fortunately, the team is small enough that this is addressed through clear communication and coordination of "code freezes" at the appropriate times.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WESH3002

CONTROL SYSTEM FOR FAST COMPONENTS OF ELECTRON BEAM WELDING MACHINES

A. V. Gerasev, P. B. Cheblakov, Budker Institute of Nuclear Physics, Novosibirsk, Russia

Abstract

Modern electron beam machines for different applications including welding, additive technologies and etc. consist of many different subsystems, which should be controlled and monitored. They could be divided by so-called fast and slow subsystems. Slow subsystems allow reaction time to be around a couple of seconds that can be implemented using PC. Fast subsystems require time to be around hundreds of microseconds combined with flexible logic.

We present an implementation of such fast system for mechanical moving platform and electron beam control. The core of this system is a single board computer Raspberry Pi. We employed a technique of fast waveform generation using Raspberry Pi on-chip DMA to manipulate stepper motors. Raspberry Pi was equipped by external CAN controller to operate an electron beam via CAN DACs. Special software was developed including libraries for low- and high-level technical process control written in C and Rust; and in-browser graphical user interface over HTTP and WebSockets. Finally, we assembled our hardware inside standard 19-inch rack mount chassis and integrated our system inside experimental electron beam machine infrastructure.

RASPBERRY PI AND ELECTRON BEAM WELDING MACHINE CONTROL

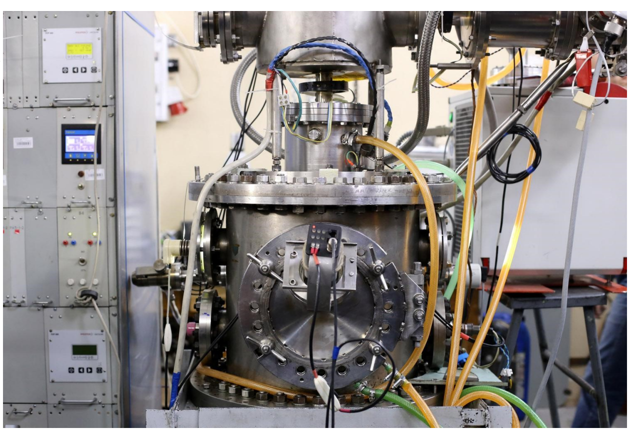


Figure 1: Small experimental electron beam machine in Budker Institute of Nuclear Physics.

Electron beam technologies become increasingly applicable in different areas. Their principle is based on beam of electrons hitting the object in vacuum and locally heating it. This approach is used for precise cutting and welding, and also for 3D-printing. In our institute research of such technologies is carrying out and a couple of different electron beam machines have already been constructed. One of them is experimental small electron beam machine shown in Fig. 1. One of successful experiments on this machine was 3D-printing with wolfram [1].

The machine contains so-called fast and slow components. Slow components like power supply and vacuum subsystem require the reaction time to be about one second and are successfully handled by general-purpose CX [2] control system. But for some experiments it was necessary to handle specific components like beam parameters control subsystem in more fast and precise manner. The components that require such control are called fast components.

Due to the experimental nature of the machine along with providing required performance and precision the control system of fast components should be flexible and easy to develop. We found that Raspberry Pi is the most suitable platform for such system implementation.

Raspberry Pi

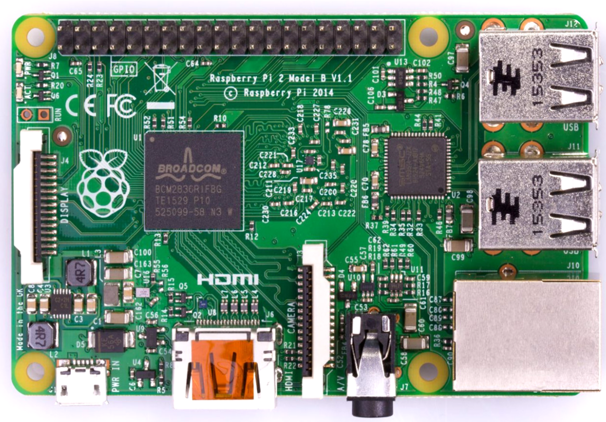


Figure 2: Raspberry Pi 3.

Raspberry Pi is a fully featured single board computer (shown in Fig. 2. There are several versions of this computer. We used versions 2 and 3. Version 2 is based on Broadcom BCM2836 system on chip (SoC) which contains 32-bit quad-core 900 MHz Cortex-A7. Version 3 is based on Broadcom BCM2837 SoC with 64-bit quad-core 1.2 GHz Cortex-A53. There is also Graphics Processing Unit (GPU) on the chip - VideoCore IV 250 MHz and 400 MHz accordingly. Along with CPU and GPU SoC also contains different peripherals devices (almost the same for both chips) including timers, interrupt, direct memory access (DMA), pulse width modulation (PWM) controllers. These models of Raspberry Pi has a lot of external interfaces like Ethernet, 4xUSB, General Purpose Input-Output (GPIO), 4xSPI and 2xUART.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WESH3003

WALTZ – A PLATFROM FOR TANGO CONTROLS WEB APPLICATIONS

I. Khokhriakov†, F. Wilde, Helmholtz-Zentrum Geesthacht, Geesthacht, Germany
O. Merkulova, IK company, Moscow, Russia

Abstract

The idea of creating Tango Controls web platform was born at Tango Users Meeting in 2013, later a feature request was defined aka v10 roadmap FR6 – to provide a generic web application for browsing and monitoring Tango Controls devices. The work started in 2017. Name "Waltz" was selected by voting at 32nd Tango Users Meeting, Prague, Czech Republic in 2018. Waltz is the result of joint efforts of Tango Community, Helmholtz-Zentrum Geesthacht and IK company.

This paper gives an overview of Waltz as a platform for Tango Controls web applications, the overall framework architecture and presents an end result of real-life applications used in HZG. The work shows that having Waltz platform web developer can intuitively and quickly create full web application for his/her needs. Different architectural layers provide maintainability. The platform has a number of abstractions and ready-to-use widgets that can be used by web developer to quickly produce web based solutions. Among Waltz features are user context saving, device control and monitoring, plot and drag-n-drop interface solutions. Communication with Tango Controls happens via Tango REST API using HTTP/2.0 and Server-Sent Events. Waltz can be also treated as a system for device monitoring and control from any part of the world.

INTRODUCTION

Waltz is a general purpose Tango Controls web application that provides the interface between the Tango Control system and the scientific users who define and calibrate their experiments. It can also be used for live monitoring of a big scientific installations like ESRF or DESY.

Initially, the idea of Waltz (ex. TangoWebapp) was born at the 29th Tango Users Meeting, Krakow, Poland in 2015 [1]. It was marked as Feature Request #6 within future Tango Controls evolution road-map [2-3]. Final project name "Waltz" was chosen by the community at the 32nd Tango Users Meeting, Prague, Czech Republic in 2018.

Waltz was designed to be used in two ways – as a platform and as an end-user application. In this article we are going to describe Waltz as a platform for Tango Controls web applications and will to list some of the features from the end-user application to show to possibilities of the platform.

WALTZ AS A DEVELOPMENT PLATFORM

Waltz considered as a platform that can be used to implement integrated and coherent web based GUIs for Tango Controls [4].

Waltz's flexibility is achieved by a layered architecture: internal event bus (e.g., OpenAjax [5]) and widgets (e.g., UI components) which can be used as building blocks for rich functionality.

Waltz offers developers a number of compact APIs: tango device model API; reusable functional components (e.g., mixins); UI builder API. These will be covered in more details later in this section.

Integration with SVG files [6] allows to implement extremely user-friendly widgets.

Subscriptions allow Waltz to use Server-Sent Events [7-8] to receive notifications about native Tango events with minimal overhead.

Since v0.7 [9] Waltz fully supports JS6 [10] features. So developers are able to use cutting edge features of modern JavaScript to implement required widgets.

Layered Architecture

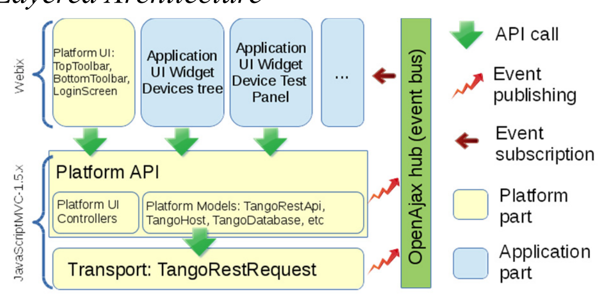


Figure 1: Waltz architecture layers.

The platform has 3 layers (from bottom to top): transport; models; UI (Fig. 1). Lower layers don't know anything about layer(s) on top and they send events via OpenAjax hub. Higher layers talk to the lower ones via API calls.

The Waltz platform is divided into *API* and *UI* parts. *Platform UI* is implemented following the concept of *smart components* – rich components build on top of used in the platform JS framework components (Webix components) [11]. Each UI component is completely standalone and can be used as a building block for more complex widgets or even dedicated applications. Using UI/non-UI components developer creates new UI components for specific use cases. *Platform APIs* is conceived for building UI, errors handling, interacting with Tango device model, etc (Fig. 2).

Another perspective of the layered architecture of Waltz is shown in Fig. 2. Different UI/non-UI components communicate with each other via OpenAjax event bus which is a part of JavaScriptMVC framework [12]. Non-UI components or API components are responsible for common functionality of the platform, e.g., storing user context data.

† igor.khokhriakov@hzg.de

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WESH4002

A PyDM USER INTERFACE FOR AN LCLS SIMULATOR

M. Gibbs*, W. Colocho, J. Shtalenkova, A. Osman, H. Slepicka
SLAC National Accelerator Laboratory, Menlo Park, USA

Abstract

PyDM (Python Display Manager) is a framework for building control system user interfaces. A user interface for the LCLS (Linac Coherent Light Source) simulator has been built in PyDM. The simulator interface gives a realistic experience of operating many parts of the LCLS accelerator, and can be used for training new accelerator operators on routine tasks. This interface also provides a good demonstration of the experience of using PyDM in a real-world environment.

SIMULACRUM: AN LCLS SIMULATOR

Recently at the SLAC National Accelerator Laboratory, an LCLS accelerator simulation system called Simulacrum has been developed. This simulator runs an accelerator model, and uses the model to populate EPICS PVs that mimic the real accelerator. The goal of Simulacrum is to let users take unmodified accelerator software applications used to operate LCLS, and run them against a simulator that could be running on a development server, or even on a user's laptop. The reverse is also true: users can develop new applications or displays against Simulacrum, using the same code that they would to interact with the real accelerator's control system.

While Simulacrum was originally developed as an aid in developing accelerator software, but it has also been identified as a training tool for operators. It can help new operators gain some intuition of how the electron beam behaves, and let them explore in an environment free of the time pressure and up-time demands that come with a facility delivering beam to user experiments. Rather than just reading training documents and procedures, the tools used to operate the machine can be used in the simulation.

PyDM

PyDM [1] is a framework for building control system graphical user interfaces (GUIs). PyDM is built with Qt [2] and the PyQt [3] Python-Qt bindings. It is packaged with a set of widgets appropriate for interacting with scientific and industrial equipment. PyDM is also highly extensible by users, who can add new widgets, support for new data sources, and add custom client-side logic to control system displays. At SLAC, some portions of the GUI used to run the LCLS accelerator are being replaced with new PyDM-based displays. Simulacrum is being use to test the new displays while most of the control system is offline during facility upgrades. The PyDM simulator interface presented here shares many of the same files that will be used for the real LCLS accelerator.

* mgibbs@slac.stanford.edu

AUTO-GENERATION OF DISPLAYS FROM DEVICE LISTS

PyDM displays can use Python code to build themselves from lists of devices. In this interface, each display queries an EPICS PVAccess table PV for a list of relevant device types, and generates user interface elements for every item in the list. By using this method, the same interface files can be used against any accelerator lattice definition used by the simulator.

PyDM's Template Repeater Widget is used frequently in these displays. This widget takes a user interface file representing the controls for a single device, along with a list of devices supplied as a list of Python dictionaries, and renders the user interface file once for each item in the list.

DISPLAYS

The top-level interface is organized by area and subsystem, laid out in a grid. For example, a user might select the column corresponding to "LI25" (linac sector 25), and the row corresponding to "Magnets". This will take the user to a listing of all magnets in the simulated sector 25. PyDM's browser-like navigation tools let the user page forward and backward through displays they have visited, or go directly back to the top-level "Home" display.

An alarm tree in the simulator collects low-level alarms into summary alarms. These summary alarms are propagated through the displays, up to the top level, allowing the user to quickly identify and traverse the displays to find the relevant screen to address the alarm.

Magnets

The magnet displays are organized in tabs for each type of magnet. In each tab is a straightforward list of controls, one row for each magnet. The controls provide the basic functions one would expect: magnet strength can be changed, via a line edit widget or slider. Other functions, saving the magnet field setpoint, or loading a saved setpoint, are available from a drop-down menu.

Collimators

The collimator displays contain controls to move collimator jaws directly, or set 'center' and 'gap' values that move two jaws together. The collimator displays utilize PyDM's "Widget Rules" feature to animate a schematic of the collimators: the on-screen position of two widgets representing the collimators are bound to the PVs that report the collimator jaw positions.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WESH4003

CONTINUOUS INTEGRATION FOR PLC-BASED CONTROL SYSTEMS

B. Schofield*, J. Borrego[1], E. Blanco Vinuela, CERN, Geneva, Switzerland
[1]also at IPFN - IST, Universidad de Lisboa, Portugal

Abstract

Continuous integration is widespread in software development, but a number of factors have thus far limited its use in Programmable Logic Controller (PLC) application development. A key requirement of continuous integration is that build and test stages must be automated. Automation of the build stage can be difficult for PLC developers, as building is typically performed with proprietary engineering tools. This has been solved by developing command line utilities which use the APIs of these tools. Another issue is that the program must be deployed to a real target (PLC) in order to test, something that is typically easier to do in other types of software development, where virtual environments may easily be used. This is solved by expanding the command line utilities to allow fully automated deployment of the PLC program. Finally, testing the PLC program presents its own challenges, as it is typically undesirable to alter the program in order to implement the tests natively in the PLC. This is avoided by using an industry standard protocol (OPC-UA) to access PLC variables for testing purposes, allowing tests to be performed on an unaltered program.

INTRODUCTION

Continuous Integration (CI) attempts to ensure the consistency of a project, by regularly and automatically integrating work from multiple developers into a single shared main version. Essentially its goal is to detect breaking changes as early as possible, by automatically running a set of tests when code changes are made. This process is commonly split into three automated stages, namely project build, deploy and test. Tests should be carefully designed, making sure the correctness of a software application is kept as new features are added.

Programmable Logic Controllers (PLCs) are the most common means of implementing industrial control systems. Such control system applications are often large and complex, and their specifications and requirements may change during their development or operation. For these reasons, it would be desirable to employ CI tools when developing these applications. However, automating the building, deployment and testing of PLC applications is a non-trivial endeavour. The main reasons for this include:

- Program compilation and deployment is typically done in proprietary engineering tools, which often do not support easy automation of these tasks;
- Having to resort to a physical PLC as a test target, as accurate and feature-complete simulators are not always available;
- The need to significantly modify the target test project, as normally it is required to add auxiliary interfaces to allow testing.

* Corresponding Author. E-mail: brad.schofield@cern.ch

Previous attempts to implement a PLC testing environment have resorted to using the Supervisory Control And Data Acquisition (SCADA) stack to exchange data with PLCs. However, this approach is not without its problems. Firstly, many internal variables often cannot be accessed via the SCADA interface, making it difficult to interact with the PLC program. Furthermore, relying on SCADA for testing the PLC program introduces a dependency on the communication protocol between them.

Rather than using a real PLC one could turn to simulation software, which can even provide additional desirable features. For instance, SIMATIC PLCSim Advanced is a simulator for Siemens S7-1500 PLCs which supports fine-grained cycle by cycle execution and breakpoints. However, each simulator and its set of engineering tools is specific to a model or set of PLC models and thus require considerable effort to integrate into a testing suite directly.

Other methods seek to translate the native PLC code to widespread languages such as Java or C [1] which run on x86 architectures and have support for step-by-step debugging.

We propose a novel workflow that allows us to write tests for generic PLC-based systems which removes the dependence on a SCADA stack and requires minimal changes to the program under test. The tests description should be readable yet powerful and able to describe high-level behaviour concisely. To achieve this we implement a Python framework and a proof of concept test suite. The communications interface employs OPC Unified Architecture (OPC-UA) [2], which provides read and write access to any symbol-mapped variable in the PLC memory during runtime. OPC-UA is an open communication protocol for industrial automation which supports multi-platform communication stacks. The automation of the build and deployment phases of the CI pipeline for the PLC are carried out by developing tools which leverage the Application Programming Interfaces (APIs) of the engineering tools provided by PLC manufacturers.

As long as a test PLC can communicate with an OPC-UA server, we can interact with it via a client running on virtually any platform, written in whichever programming language fits the software stack employed by the testing framework. What is more, we conceive that by providing a single test description that uses such an OPC-UA client, we can run tests on any PLC – provided smaller structural differences in the interface exposed by OPC-UA are taken in consideration.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-THAPP01

AUTOMATIC GENERATION OF PLC PROJECTS USING STANDARDIZED COMPONENTS AND DATA MODELS

S. T. Huynh†, H. Ali, B. Baranasic, N. Coppola, T. Freyermuth, P. Gessler, N. Jardón Bueno, M. Stupar, J. Tolkiehn, J. Zach, European X-Ray Free-Electron Laser, Schenefeld, Germany

Abstract

In an environment of rapidly expanding and changing control systems, a solution geared towards the automation of application dependent Programmable Logic Controller (PLC) projects becomes an increasing need at the European X-Ray Free Electron Laser (EuXFEL). Through the standardization of components in the PLC Framework, it becomes feasible to develop tools in order to automate the generation of over 100 Beckhoff PLC Projects. The focus will be on the PLC Management System (PLCMS) tool developed to achieve this. Provided with an electrical diagram markup (EPLAN XML export), the PLCMS queries the database model populated from the PLC Framework. It captures integration parameters and compatible EtherCAT fieldbus hardware. Additionally, inter-device communication and interlocking processes are integrated into the PLC from a defined user template by the PLCMS. The solution provides a flexible and scalable means for automatic and expedited deployment for the PLC control systems. The PLCMS can be further enhanced by interfacing into the Supervisory Control and Data Acquisition (SCADA) system for complete asset management of both PLC software and connected hard-ware across the facility.

INTRODUCTION

The European X-Ray Free-Electron Laser (EuXFEL) is a research facility that aspires to provide high brilliance X-ray beam for user experiments at six end stations with diverse experimental capabilities [1]. The facility is currently home to over a few thousand devices within the multiple photon beamlines, which are to be controlled remotely through the use of Programmable Logic Controllers (PLC).

As such, the PLC team was formed to bring about the design, implementation, deployment and commissioning of over a hundred PLCs with the various hardware components installed and/or proposed. Additionally, the PLCs are to be interfaced with the proprietary Supervisory Control and Data Acquisition (SCADA) system; Karabo [2], to provide a user interface for control.

In order to meet the demands relating to the rapid deployment of multiple PLC projects, it became apparent that an automated solution was required to compile and build the various PLC Projects. As such, the PLC Management System (PLCMS) tool was developed. Currently, the PLCMS performs several functions, all aimed to assist and expedite the existing manual processes related to PLC deployment. These can be largely divided into two categories: the PLC Framework and the PLC Projects, both of which are explored in more detail throughout the paper.

Firstly, the rationale behind the PLCMS will be introduced, followed by the functionality of the PLCMS, and how some of the shortcomings within the existing PLC generation process are addressed and resolved. Lastly, some of the ongoing developments to further enhance the PLCMS and its capabilities are explored.

† sylvia.huynh@xfel.eu

THE PROBLEM

Since the inception of EuXFEL, there has been a rapid campaign to bring the facility online. In order to provide a facility where scientists can come to conduct research where the only heavy lifting comes down to a mouse click, many PLCs were required to be commissioned and deployed within a short period of time.

The PLC projects are designed around the infrastructure, whereby hardware components which belong to a similar sub-system, are grouped together and allocated a PLC. Each project is subsequently composed of standardised "soft" components called softdevices which are essentially a software representation of the hardware component within the PLC. As the PLC Projects often contain over a hundred instances of softdevices, with various initialisation and configuration parameters, this process would become tedious, time-consuming and error prone.

Utilising a library of the aforementioned softdevices would thereby enable device instantiation rather than continuous creation, and also provide a means for consistency. To be useful however, the library needs to maintain a standardised structure or template across all the softdevices. As the library evolves however, keeping track of changes within each softdevice across multiple versions, deployed across an array of PLCs becomes increasingly challenging. Adjustments to interlock parameters, initialisation values, configuration parameters, or hardware reconfiguration of softdevices; all changes that are applied on the PLC project layer, add to the complexity of maintaining these PLC projects by hand. It would be a fallacy to continue without another means for the design, development and generation of PLC projects that would achieve the results within a more effective and efficient manner.

THE PLCMS

The PLCMS has been developed using Python 3 in addition to Object-Relational Mapping (ORM) to interface with an SQL database backend. The PLCMS performs multiple functions, aimed to expedite the PLC framework and project generation processes in order to cope with the number of ongoing changes and vast quantity of PLCs used throughout the facility. Additionally, it provides the ability

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-THAPP02

THE CONTROL SYSTEM OF THE ELLIPTICAL CAVITY AND CRYOMODULE TEST STAND DEMONSTRATOR FOR ESS

A. Gaget†, T. Joannem*, IRFU, CEA, Université Paris-Saclay, Gif-sur-Yvette, France

Abstract

CEA IRFU Saclay is taking part of ESS (European Spallation Source) construction through several packages and, especially in the last three years on the Elliptical Cavity and Cryomodule Test stand Demonstrator (ECCTD). The project consists of RF test, conditioning, cryogenic cooldown and regulations of eight cryomodules with theirs four cavities each. For now, two medium beta cavities cryomodules have been successfully tested. This paper describes the context and the realization of the control system for cryogenic and RF processes, added to cavities tuning motorization relying on COTS solutions: Siemens PLC, EtherCAT Beckhoff modules, IOxOS fast acquisition cards and MRF timing cards.

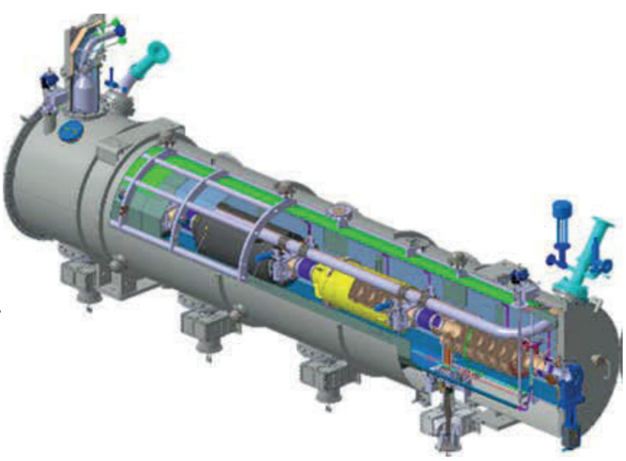


Figure 1: Medium-beta cryomodule general layout.

INTRODUCTION

The European Spallation Source ESS is a large European research infrastructure under construction in Lund, Sweden. It will be composed among others of 30 elliptical cryomodules. These will be integrated in the next few years with a delivery rate of one cryomodule per month. An international collaboration has been established to develop and construct the 30 elliptical cryomodules. CEA Saclay and IPN Orsay are collaborating to design, build and test a first Medium beta Elliptical Cavities Cryomodule Demonstrator (M-ECCTD) [1] (see Fig. 1). A second demonstrator with high beta cavities (H-ECCTD) is being developed by CEA before starting the production of the cryomodules. CEA will provide many other components, such as the power couplers and their RF processing.

The control for the high level control activities has been developed in EPICS, based on the ESS EPICS Environment (EEE) [2] developed by the ESS control team at Lund (ICS). For the critical needs (cryogenic, vacuum, interlock), we have used Siemens PLC with an homemade SCADA dedicated to those devices and communicating via TCP/IP: Muscade®.

CONTEXT

Before the cryomodule package the CEA has developed a platform for our previous contribution, the conditioning of the power couplers for the cavities. To fulfill this need, it has been decided to upgrade an existing 704MHz RF platform of our own. EPICS has replaced all the LabVIEW software that was controlling the RF source. This RF Source is now used intermittently to condition power couplers that will be used in the cryomodule, and to condition cavities in the cryomodule demonstrator as shown in Fig. 2.

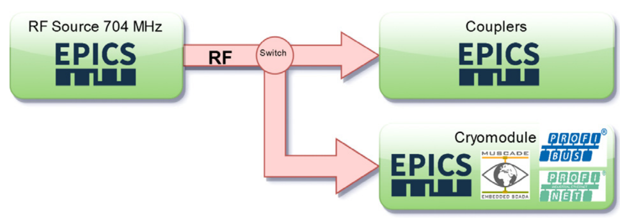


Figure 2: RF platform at CEA.

Likewise an existing cryogenic platform called Supratech is used to provide cryogenic fluids like Helium and Nitrogen to local experiments. In order to simulate the valve box for cryomodules cooling down an intermediate gateway control system was created.

Supratech control system is based on Siemens PLC and Profinet fieldbus to communicate. It was decided to follow the same logic for the cryomodule and the cryogenic gateway control systems.

Thanks to this homogeneity and more precisely Profinet, cryomodule control system can easily share data with other systems and send commands directly to the local cryoplant to refill it with cryogenic fluids.

HARDWARE SOLUTION

Fast Acquisition

For RF signal, photomultiplier and pickup electron measurement, we needed acquisition cards able to acquire data with a sampling rate up to 1Ms/s and with a frequency period up to 14Hz. Today ESS and CEA have now both evolved to MTCA standard but at the project beginning we chose the proven VME solution. We are using IOxOS [3] hardware solution with the VME64X CPU card IFC1210 coupled with the ADC3111 FMC (FPGA Mezzanine Card) (see Fig. 3). ESS ICS have developed the drivers.

† alexis.gaget@cea.fr
* tom.joannem@cea.fr

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-THAPP03

CONSTRUCTION OF BEAM MONITOR CONTROL SYSTEM FOR BEAM TRANSPORT FROM SACLA TO SPring-8

A. Kiyomichi†, N. Hosoda, M. Ishii, M. Yamaga, JASRI, Hyogo, Japan
T. Fukui, K. Maesaka, RIKEN SPring-8 Center, Hyogo, Japan

Abstract

As a part of the SPring-8 upgrade project, the SACLA (SPring-8 Angstrom Compact free electron LAser) linac will be used as the injector for the SPring-8 storage ring. We will upgrade the beam monitor system for beam transport, which consists of screen monitors (SCM), beam position monitors (BPM), and current transformers (CT). For the SCM, we adopted the GigE Vision standard for the CCD camera and EtherCAT as a field bus for the stepper motor control of the focusing system. We have developed camera control software using open source libraries to integrate various vendors' GigE Vision cameras with the SPring-8 control framework. A grabbed image is stored into the file server and properties, such as camera settings for image and event number, are stored into the database. The BPM is a key device for precise and stable injection. We adopted the commercially available MTCA.4 fast ADC/DAC module with modified firmware developed for the readout of the BPM and the CT. Acquisition software for MTCA.4 modules to synchronize with a beam trigger is being developed. The acquired data are stored into the database with a time stamp and an event number. The preparation of the beam monitor control system for the beam transport to injection from SACLA to SPring-8 is herein presented.

INTRODUCTION

The upgrade project, SPring-8-II [1], requires a low emittance beam for the injection because of its small dynamic aperture. The SACLA linear accelerator to be used has been designed as a full-energy injector for the SPring-8 storage ring, sharing with the XFEL operation. Then, an ultralow emittance electron beam with a short bunch length of 10 fs is delivered from SACLA. The frequency of beam injection into the SPring-8 storage ring is 10 Hz for an initial injection and once a few minutes for top-up operations, while the repetition rate of the SACLA linac is 60 Hz. The beam transport from SACLA to SPring-8 storage ring is about 600m. The beam monitor system has approximately 30 screen monitors (SCM), 30 beam position monitors (BPM), and 10 current transformers (CT). The beam monitor and its control system will be upgraded for the beam transport from SACLA injection.

In SPring-8, image diagnostic systems such as SCMs, widely use charge-coupled device (CCD) cameras with Camera Link interfaces (I/F) [2]. The image processing system was developed and operated based on a PC server or MicroTCA that supports the Camera Link I/F [3-5]. The Camera Link I/F has high bandwidth data transfer capability, trigger synchronization, and camera control via a serial bus. As grabber boards were released by multiple vendors and Linux APIs were available, the Camera Link standard was adopted when switching from analog cameras to digital cameras for SCMs approximately 10 years ago. However, the Camera Link I/F has a maximum data transfer length of 10m, which is too short for use in an accelerator facility; therefore, a transmission/reception device is required. The disadvantage of the Camera Link I/F is that it requires complex cabling with a chain of transmitters and receivers using optical conversion. This results in difficult and expensive maintenance. In the next phase of SPring-8, to identify an alternative, the GigE Vision [6] standard camera was evaluated. This would allow data transfer lengths of up to 100m, and significantly simplify cabling.

For the readout electronics of the BPM, two candidates are selected; our original design based on MTCA.4 [7] and Libera Spark [8]. They are developed and evaluated in parallel. The MTCA.4 based electronics is the same as the new low-level RF system [9] and it uses the commercially available MTCA.4 fast ADC/DAC module. This will also be used as the readout electronics of the CT.

NEW CONTROL SYSTEM FOR BEAM MONITOR

GigE Vision Camera

Post evaluation, the GigE Vison camera has been adopted for the SCM. The camera control software has been developed using open source libraries to integrate in the SPring-8 control framework. A PC-based image processing system is built to control the SCM with one unit by installing PCI Express cards such as Power-over -Ethernet (PoE) type network interface and trigger counter. Details of the development are described in the next section. The JAI Go-2400M (Fig. 1), which is a GigE version of the same series as the Camera Link camera used in SACLA, is used.

Figure 1: JAI Go-2400M.

† kiyomichi@spring8.or.jp

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-THAPP04

EPICS TOOLS FOR SMALL EXPERIMENT BASED ON PLC

K. Saintin†, P. Lotrus††, Q. Bertrand, G.A. Durand, F. Gohier, T.J. Joannem, N. Solenne
IRFU, CEA, Université Paris-Saclay, Gif-sur-Yvette, France

Abstract

Irfu [1] control software team is involved from feasibility studies to equipment deployment into many different experiments in their size and running time. For many years, Irfu is using PLC solution for controlling part of the experiment, and two different SCADA:

- MUSCADE®, an in-house SCADA dedicated to small experiments.
- EPICS [2] for bigger facilities.

With MUSCADE®, Irfu has developed a set of tools that gives an easy and a fast way for PLC developers to configure the SCADA. As EPICS projects are growing in our department, we are working now on adapting those tools to EPICS:

- PLCPARSER generates an EPICS database for PLC communication (S7PLC, Modbus).
- CAFEJava (Channel Access For Epics Java) API runs a simulated EPICS IOC to test EPICS synoptic, and provides EPICS process variables access for any Java application.
- DXF2OPI converts AutoCAD DXF files into OPI files for CSS [3] software.
- MOONARCH (Memory Optimizer ON ARCHiver Appliance) reduces EPICS Archiver Appliance [4] data files storage.

INTRODUCTION

Irfu is working on different kind of experiments from particle accelerator installed in big facilities to gas station installed in the middle of the fields. Irfu participates regularly to the construction of accelerator around the world and its control system team has developed EPICS skills for more than 20 years. EPICS and associated software are mainly designed for big facilities, which is not convenient for some of our experiment; especially small ones based on PLC control (see Fig. 1). Therefore, we have developed a set of software that will help our PLC developers to work with EPICS as SCADA, and on adapting EPICS tools to tiny structure.

To help PLC developers, we are working on generating EPICS communication from TIA Portal data blocks description, on a multi-platform IOC (Input Output Controller) for testing the communication, and on automating GUI (Graphical User Interface) generation from AutoCAD drawing. To adapt EPICS for tiny experiments that are using only one computer, we have developed a tool that will decimate or remove archiving data. The idea is to provide tools that gives enough autonomy to non-EPICS specialist for configuring and deploying an EPICS SCADA on one PC.

† katy.saintin@cea.fr
†† paul.lotrus@cea.fr

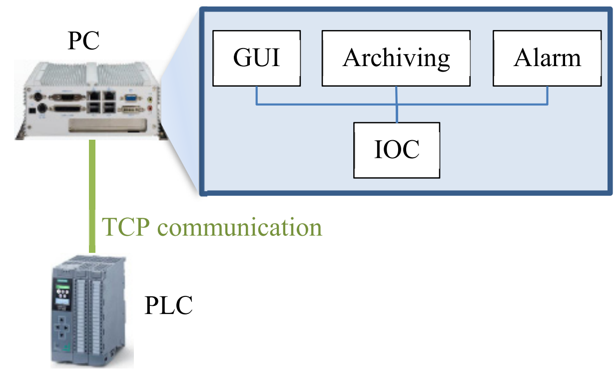


Figure 1: Small experiment architecture based on PLC.

PLCPARSER

At Irfu, a typical hardware configuration is a Siemens PLC and a PC with Linux and EPICS as SCADA. A recurrent development task is the communication between the two devices. There are several ways to automate this development; we have chosen to start from the PLC development code. The main reason is that PLC developers are implementing quickly the physicist requests and therefore PLC code is the core of the machine behavior, so it is natural to generate the communication from this reference. The principle is to read the PLC's memory spaces involve in the communication with EPICS, and then generate the IOC code. To do that, we are using an export feature of TIA Portal, the PLC developer can export each memory block in a text file format (Fig. 2). For Modbus/TCP or S7PLC (PSI) communication, we are exporting data-block in awl file format, which gives few information like the memory position, the name, the type and a description.

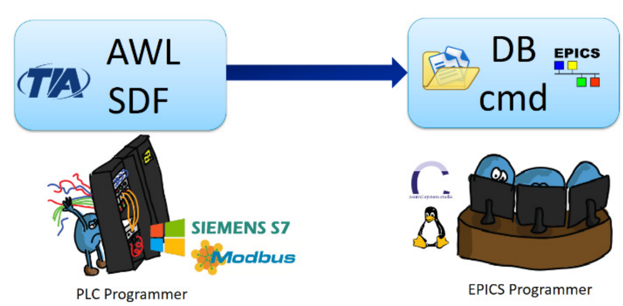


Figure 2: PLCPARSER tool principle.

The next step is to use PLCPARSER application, which parses all the awl files contained in a folder. Then, the user configures the communication parameters like IP address and adds information like alarm, physical units, etc... Once PLCPARSER is generating IOC files, the user can copy those files into the dedicated EPICS top folder (See Fig. 3).

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-THAPP05

OVERVIEW OF ACQUISITION AND CONTROL ELECTRONICS AND CONCEPTS FOR EXPERIMENTS AND BEAM TRANSPORT AT THE EUROPEAN XFEL

P. Gessler†, H. Ali, F. Babies, K.-E. Ballak, H. Bamaga, B. Baranasic, O. Bieler, N. Coppola, K. Dornack, J. Eilers, D. Emes, B. Fernandes, M. Fobian, T. Freyermuth, S. T. Huynh, N. Jardón Bueno, M. Meyer, O. Oshtuk, P. Parlicki, J. Reifschläger, S. Sayar, H. Sotoudi Namin, M. Stupar, J. Tolkiehn, H. Vega Pérez, S. Wagner, J. Zach
European X-Ray Free-Electron Laser Facility, Schenefeld, Germany

Abstract

FPGA based fast electronics to acquire and pre-process signals of detectors and diagnostics and PLC based hardware and software for motion, vacuum and other control and monitoring applications are key elements of the European X-Ray Free Electron Laser. In order to bring the newly developed scientific user facility up and running, the underlying electrical and electronic components require a diverse array of tools and processes to be developed in order to meet the continually adapting requirements and make use of technological advances. Many challenges were faced, including high availability and up-time, adaptability to a dynamic environment, rapid lead-time for integration of complex components, numerous instrumentation installations and commissioning, high time resolution and subsequently, high demands on data and sampling rates, synchronization and real-time processing. In this contribution we will provide an overview of the selected technologies, developed concepts and solutions along with generically designed frameworks and tools, which aim to provide a high degree of standardization on the control systems and even automatic generation from requirements to final install.

INTRODUCTION

The European X-Ray Free-Electron Laser Facility (EuXFEL) is located in northern Germany with a length of about 3.4 km. It provides coherent x-ray pulses between 260 eV and 24 keV and a duration of less than 100 fs at three beamlines. At present, six instruments are located at the end of the beamlines, which provide a wide span of environments and instrumentation to allow scientific research in many fields [1].

Due to the underlying super conducting accelerator technology, the time structure of the photon pulses delivery is organized in short bursts (also called trains) of up to 600 µs every 100 ms. In these bursts up to 2700 pulses could be generated and directed to up to three of the six instruments at a time. The minimal spacing between pulses is 220 ns and therefore equivalent to a repetition rate of 4.5 MHz. These parameters are of relevance for the design of the electronics and acquisition system since high-resolution measurement during the bursts is often required, and in the gaps between the bursts, time for data processing and transfer is available.

A significant number of equipments are required and have to be accurately remotely or automatically operated. These could be categorized as follows: (1) slow control and monitoring elements like vacuum systems, positioning, temperature control, etc., and (2) fast diagnostics and detectors. While the first category includes a very large number of individual channels of sensors, and actuators, which require only moderate time resolution of seconds to sub-ms range, the second category demands a high sampling rate (in the order of multiple GHz) and high data bandwidth (in the order of 10 GBytes/s on a single detector). On such speeds and band widths, the pre-processing of acquired signals is an important aspect in order to implement data reduction and to reduce the latency in the system to process and react on the measurement signals in real-time.

The mentioned aspects are important boundary conditions and significantly impacted the selection and development of the underlying electronics for control and acquisition. It should be mentioned here, that there is a split of responsibility for the accelerator part up to the undulators on one side and the photon beam transport and experiment stations on the other side. This work only focuses on the later part. Information about the accelerator and undulator related control electronics could be found, for example in [2].

VISION AND APPROACH

In the planning phase about 130 individual components were counted, where each one could be relatively simple in terms of control, like an imager with one motor and related position switches to insert a screen, digital outputs to control equipment powering and to enable or disable a LED for testing purposes. But one component could also be an x-ray split and delay line with about 100 individual motors and related switches and encoders. Taking a closer look to the quantity of the elements to be controlled, monitored, digitized and processed, one part of the challenge could be easily seen, which is the integration of thousands of individual elements like motors, pumps, valves, gauges, sensors and analogue and digital signals.

In order to master the task not only of how to integrate all these elements into the control system of European XFEL, but also how to be able to support and maintain it

† patrick.gessler@xfel.eu

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-THAPP06

DOUBLE CRYSTAL MONOCHROMATOR CONTROL SYSTEM FOR ENERGY MATERIALS IN-SITU LABORATORY BERLIN (EMIL)

A. Balzer, P. Sreelatha Devi, A. Ziegler
Helmholtz-Zentrum Berlin, Germany

Abstract

The control system for the *Double Crystal Monochromator* (DCM) at the *Energy Materials In-Situ Laboratory* (*EMIL*) is discussed in this paper. The DCM feeding the U17 hard X-ray beamlines was designed and optimized for stability and resolution. A multi modal set-up provides synchrotron radiation with a broad energy range of 80*eV* - 10*keV* and variable polarization. Two canted undulators, three monochromators, five end stations, more than twenty optical elements and a sample to source distances of more than 60*m* are challenges by its own.

The mechanical concept of the DCM puts high demands on the control software. For on-the-fly synchronization of crystal pitch, crystal translation and the cryogenic cooling system rotation (referred as *cryo* in this paper), a closed-loop feedback is needed to fulfill the control system requirements. Motion programs are used for compensation of the non-linearities of the pitch rotation. Target positions are approached on a well defined path, improving reproducibility and positioning time. A non-linear closed-loop control provides fine positioning.

A setup of the motion controller, based on the tpmac module, provides the abstraction interface to the complex DCM motion control software [1]. This paper discusses the DCM hardware, software model and experimental verification.

INTRODUCTION

EMIL is a joint venture of *Helmholtz-Zentrum Berlin* (HZB) and the *Max Planck Society* (MPG) and is aimed at combining a chest-tool of spectroscopic techniques and deposition tools at BESSY II synchrotron [2]. The EMIL laboratory uses as light source one of the most complex beamlines at *BESSY II*. Two undulators, one providing soft X-rays and one hard X-ray, are shining light into a beamline which contains three monochromators, two *Plane Grating Monochromators* (PGM) and one *Double Crystal Monochrmator* (DCM), ten mirror chambers and four exit slits. The two beams are dispersed and focused into two isolated pathways towards five interaction points of which three of them achieve the overlap of soft and hard x-rays. This is only possible due to selectable monochromators and mirrors. This design which demands a very high mechanical and thermal stability of its optical elements as well as motion reproducibility was built by Bestec.

The DCM monochromator, the subject of the present work, is a development of the aforementioned company and is equipped with three pairs of silicon crystals which are cooled to liquid nitrogen temperature in order to reduce the crystal deformation at high heat loads produced by the impinging beam.

A single crystal rotation stepper motor drive with harmonic drive gear is used to move the cradle of the selectable double crystals. The cryo system connected to the crystals has to be moved in sync with the crystal rotation. The worm gear of the cryo system rotation is driven by a second stepper motor.

The fixed exit beam position is achieved by the adjustment of the crystal height, driven by an in vacuum stepper drive. A piezo stage mounted on top of that drive is used to fine-tune pitch, roll and crystal translation (CT) (see Fig. 1).

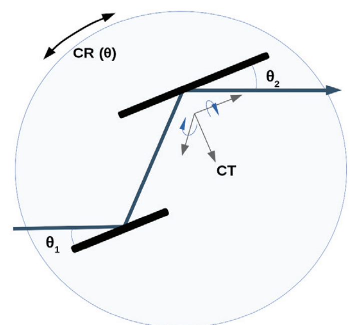


Figure 1: Double crystal positioning using stepper motors for crystal rotation (CR) and crystal translation (CT).

Challenges

Motion control for synchrotron applications is more discerning than ever before. Experimental constructions and prototypes of devices make engineers face various different challenges. None the less, they still have to be cost effective.

Some devices have to follow others on-the-fly, other moving on predefined paths. They have to proceed fast and definite at the same time, to ensure precise positioning of the motors on the endpoint and providing a high stability and robustness regarding vibrations.

Not only the end position precision of hardware is important, but also some beamline applications are very sensitive to disturbances and jerk, resulting from discontinuous acceleration profiles during motions as well. A stable and reliable communication is crucial for the experimental setups to take synchronized data.

In order for software engineers and scientists to get a better grasp of the hardware and software, diagnostic tools had to be implemented and improved. These tools help with the optimization of the devices movement or alignment of the beam, as well as commissioning. To further improve running beamline applications, shorter development cycles

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-THBPP01

BUILDING THE CONTROL SYSTEM TO OPERATE THE CRYOGENIC NEAR INFRARED SPECTROPOLARIMETER INSTRUMENT FOR THE DANIEL K. INOUYE SOLAR TELESCOPE

R. J. Williams*, A. J. Borrowman, A. Greer, A. Yoshimura,
Observatory Sciences Ltd, St Ives, UK
A. Fehlmann, B. D. Goodrich, J. R. Hubbard, DKIST/NSO, Boulder, USA
I. F. Scholl, University of Hawaii, Pukalani, USA

Abstract

The Cryogenic Near Infrared Spectropolarimeter (Cryo-NIRSP) will be one of the first-light instruments on the Daniel K. Inouye Solar Telescope (DKIST) currently under construction in Hawaii. Cyro-NIRSP is a near- and thermal-IR imager and spectrograph operating in a cryogenic environment. It will be used to study the faint solar coronal magnetic field across a large field of view. Such a complex and precise instrument demands equal requirements from the control system, which must manage the setup, timings, synchronization, real time motion, and overall monitoring of the many sub-components (e.g. cameras, polarimeter, mirrors). The control system is built within the pre-defined DKIST software framework, which provides consistency across all instruments. This paper will discuss how such a control system has been achieved for the Cryo-NIRSP instrument, detailing some of the challenges that were overcome relating to the synchronization of specific components and the complex inter-dependencies between configurables. It will also describe the data processing and visualization software development for the end-to-end functioning of the instrument.

INTRODUCTION

The DKIST (currently under construction) will be the world's largest ground-based solar telescope, with a 4-meter aperture. It will study the sun in visible to near-IR wavelengths and has adaptive optics to provide high spatial resolution of the sun's features. The telescope will have 5 first-light instruments, one of which will be the Cyro-NIRSP currently being fabricated by the Institute of Astronomy at the University of Hawaii. The main purpose of the Cryo-NIRSP is to study the solar coronal magnetic field over a large field of view (FoV) at near- to thermal- IR wavelengths and measure the full polarization state (Stokes I,Q,U,V) of spectral lines originating from the sun. The thermal IR capabilities mean that it will be able to study the solar disk in the CO lines and it will also be able to study both the near-limb and more distant corona, providing data on features such as prominences, flares, and other events in the low corona.

In parallel to the construction of the Cryo-NIRSP instrument itself has been the development of the control and data-processing software, both essential for the running of the instrument. The Cryo-NIRSP software system is responsible for: monitoring and controlling all mechanisms, interacting with the instrument cameras, interacting with the modulator system, providing an interactive control for the instrument operation, and providing instrument-specific data-processing plugins. This paper describes how the control software has been developed, including a brief outline of the instrument's many components and complexities, and details the implemented software solutions to manage these. The final section will address the data processing side, which is vital for correctly calibrating the instrument and ultimately meeting the end science goals of the instrument.

The Instrument

Cryo-NIRSP is both a near- to thermal-IR imager and a spectropolarimeter, with a spectral resolution of 30000 to 100000. Its critical optics are cooled to cryogenic temperatures giving it the capability of achieving photon-noise-limited observations. Warm pre-optics relay the image from the DKIST coude optics onto mirrors that scan a field of view for 2D spectropolarimetric imaging. Some of the key cooled optics include mirrors, wiregrid polarisers, a grating, filter wheels, and a slit wheel (accommodating both single- and mutli-slit spectroscopy). See Fig. 1 for a schematic of the instrument [1]. The Cryo-NIRSP can facilitate many different observing modes, and all the light can be sent exclusively to the context imager or exclusively to the spectropolarimeter or it can be used for simultaneous imaging and spectroscopy where the light is split off to both.

The camera system consists of the camera hardware itself and the supporting Camera System Software (CSS) [2]. The Cryo-NIRSP uses two IR cameras, one that records the spectra provided by the spectropolarimeter (SP) and another that images the field of view of the spectropolarimeter (i.e. the context imager - CI). Each camera consists of a 2048 x 2048 IR detector array with an accompanying array controller that transfers data collected from the detector to the machine hosting the CSS. The camera hardware sends raw frames to the CSS. Sets of Fully Processed Arrays (FPAs) are then sent to the Data Handling System (DHS). The CSS also contains a Time Reference and Distribution System (TRADS) for computing trigger times based on a reference time and rate. The cameras use nondestructive read ramps (NDRs) and support 3 different readout modes: slow up the ramp, fast up the ramp, and line by line. The up-the-ramp mode samples the signal at regular intervals throughout the exposure and

* rjw@observatorysciences.co.uk

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-THBPP02

DONKIORCHESTRA: A SOFTWARE TRIGGER-DRIVEN FRAMEWORK FOR DATA COLLECTION AND EXPERIMENT MANAGEMENT BASED ON ZeroMQ DISTRIBUTED MESSAGING

R. Borghes, F. Billè, V. Chenda, G. Kourousias, M. Prica
Elettra-Sincrotrone Trieste, Basovizza, Italy

Abstract

Synchrotron end-stations consist of a complex network of devices. The setup is not static and is often upgraded. The data acquisition systems are constantly challenged by such changes and upgrades, so scalability and flexibility are crucial skills. DonkiOrchestra is a ZeroMQ-based framework for data acquisition and experiment control based on an advanced software trigger-driven paradigm. In the DonkiOrchestra approach a software device, referred to as Director, provides the logical organization of the experiment as a sequential workflow relying on triggers. Each software trigger activates a set of Actor devices that can be hierarchically organized according to different priority levels. Data acquired by the Actors is tagged with the trigger number and stored in HDF5 archives. The intrinsic asynchronicity of ZeroMQ maximizes the opportunity of performing parallel operations and sensor readouts. This paper describes the software architecture behind DonkiOrchestra, which is fully configurable and scalable, so it can be reused on multiple endstations and facilities. Furthermore, experimental applications at Elettra beamlines and future developments are presented and discussed.

INTRODUCTION

Elettra Sincrotrone Trieste is a multidisciplinary international research center located on the outskirts of Trieste that hosts two advanced particle accelerators: Elettra and FERMI. It provides state-of-the-art techniques to lead experiments in physics, chemistry, biology, life sciences, environmental science, medicine and cultural heritage. Elettra is a third-generation synchrotron light source that delivers more than 5000 hours/year of synchrotron light from IR to soft x-rays to 28 beam lines. A substantial facility upgrade, named Elettra 2.0, is currently planned [1]. FERMI is a seeded free electron laser facility that provides fully coherent ultrashort 10-100 femtosecond pulses with a peak brightness ten billion times higher than that made available by third-generation light sources. Currently 6 versatile experimental stations carry out outstanding research with photon radiation coming from FERMI.

The Software For Experiments Team manages a set of core services for the beamlines of the two facilities, spanning from the instruments integration to the data acquisition and experiment management. The range of applications is wide and heterogeneous, it is crucial for the efficiency of the team to adopt common and reusable frameworks. For this reason in the recent years we designed and developed a set of software tools highly configurable and scalable that can be adapted and used on several beamlines.

The next sections provide the reader with a technical overview of DonkiOrchestra, a modular framework focused on data acquisition and experiment management. DonkiOrchestra represents a novel approach, a software trigger-driven architecture that solves efficiently the intrinsic problem of non-synchronicity of any heterogeneous and distributed experimental system. Thanks to the modular and configurable structure of DonkiOrchestra it was possible to use it on different Elettra beamlines. Two sections of the paper are dedicated to the presentation of the experimental applications of DonkiOrchestra at TwinMic and XRF beamlines. Lastly future developments are presented and discussed.

CONCEPTUAL DESIGN

Synchrotron and Free Electron Laser beamlines consist of a complex distributed network of devices like sensors, detectors, motors, but also computational resources. DonkiOrchestra draws on years of experience and has been designed in order to solve a set of critical challenges:

- *Scalability and Customizability*: each scientific setup is unique and designed for a specific technique but needs frequent adaptations for the integration of new instrumentats or for carrying out specific user experiments.
- *Parallelism and Synchronization*: sensors, motors and cameras in a distributed control system act independently but often do not share a common time base. Most of the experiments require the simultaneous acquisition of different data sources and a high level of synchronicity is essential for the data analysis phase.
- *Communication Efficiency*: data communication overheads directly affect the duration of the experiment and the quality of the data synchronization. The choice of an efficient, reliable and modern communication system is critical for the performance of the entire framework.

One of the most important aspects in a beamline experiment is how the sensors and actuators are best synchronized. Often it is fundamental to know not only what happened, but also when it happened. The use of a wired trigger signal is the quickest, easiest and most accurate way to synchronize events in different places. Nevertheless, in the heterogeneous environment of a photon source facility, this solution is not always adoptable. It could be due to the presence of non-triggerable instrumentation or the need of having an

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-THBPP03

DEEP LEARNING METHODS ON NEUTRON SCATTERING DATA

P. Mutti, F. Cecillon, Y. Le Goc, G. Song, Institut Laue-Langevin, Grenoble, France

Abstract

Recently, by using deep learning methods, a computer is able to surpass or come close to matching human performance on image analysis and recognition. This advanced methods could also help extracting features from neutron scattering experimental data. Those data contain rich scientific information about structure and dynamics of materials under investigation. Deep learning could help researchers better understand the link between experimental data and materials properties. Moreover,it could also help to optimize neutron scattering experiment by predicting the best possible instrument configuration. Among all possible experimental methods, we begin our study on the small-angle neutron scattering (SANS) data and by predicting the structure geometry of the sample material at an early stage. This step is a keystone to predict the experimental parameters to properly setup the instrument as well as the best measurement strategy. In this paper, we propose to use transfer learning to retrain a convolutional neural networks (CNNs) based pre-trained model to adapt the scattering images classification, which could predict the structure of the materials at an early stage in the SANS experiment. This deep neural network is trained and validated on simulated database, and tested on real scattering images.

INTRODUCTION

The fundamental properties of the neutron make it a powerful tool to investigate atomic-scale structure and dynamics of materials. Compared to other scattering techniques such as x-ray or light, neutron has advantages such as negligible radiation damage, exceptional penetration power and ability to selectively highlight specific parts of materials via isotope labelling [1]. However, because the flux of neutrons is lower than the flux of the photons at modern light sources, long measurement time (even tens of hours) is required for a high-quality data collection. In this case, early decision becomes a critical component in the experimental work flow. On the other hand, the instrument settings of neutron scattering experiments are manually decided by scientists according to their experiences and knowledges. Due to large differences in the level of experience of the different users, high quality data could become difficult to obtain and quite time consuming. Hence, we propose to use recent advanced machine learning methods to help interpreting experimental data in order to optimize effective use of limited beam time and ence increasing science productivity.

Deep learning has become one of the most active research field in the machine learning community since it was presented in 2006 [2]. Recently, it impacted deeply data science, and its popularity has grown exponentially, especially in computer vision related tasks [3]. Convolutional neural networks (CNNs) is a well-known deep learning architecture inspired by the natural visual perception mechanism of the living creatures. In recent years, CNNs have been widely adopted for a variety of applications in image classification [4, 5], video analysis [6] and natural language processing [7]. CNNs are widely used in image related tasks since they present several advantages. Firstly, CNNs decompose images into multiple patches that are partially overlapped. In this case, each cortex neuron only corresponds to a single patch, which enables the network to classify images with little data pre-processing [8]. Secondly, CNNs are weight sharing, which enable CNNs share learned features from a single sample via back-propagation with other samples. It efficiently decreases the necessary required input sample size and quantity [9].

In this paper we have applied transfer learning method on a CNNs architecture based pre-trained model, to analyze and classify neutron scattering image from SANS experiment. This model named Inception-v3 is pre-trained on ImageNet database. With fine-tuning, this model is first retrained and validated by simulated scattering images, then verified on real SANS experimental data collected at the Institut Laue-Langevin (ILL).

METHODS

Data Simulation

To generate two-dimensional Small-Angle Neutron Scattering (SANS) images we use the data reduction and simulation code GRASP [10].

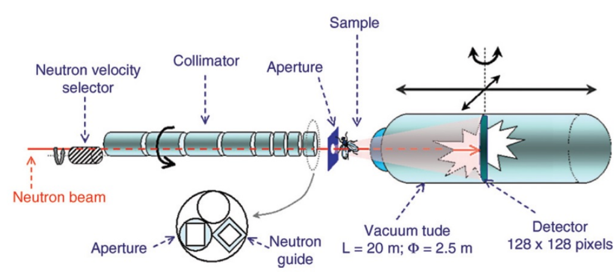


Figure 1: Schematic representation the steady-state instrument D22 at the Institut Laue-Langevin [11].

GRASP is a MatlabTM script application designed for the graphical inspection, analysis and reduction of position sensitive detector(PSD) data produced by the 3 SANS instruments of the ILL. GRASP is using a mixed analitical and Monte-Carlo approach for the neutron event generetion and can include the real intrument resolution function, the measured backgrounds from a variety of different samples and the influence of the sample environment. By using GRASP, we could generate simulated SANS 2D detector images with variable sets of parameters. We could select different materials' structures and also freely change the in-

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-THBPP04

HARD X-RAY PAIR DISTRIBUTION FUNCTION (PDF) BEAMLINE AND END-STATION CONTROL SYSTEM*

O. Ivashkevych†, M. Abeykoon, J. Adams, G. Bischof, E. Dooryhee, J. Li, R. Petkus, J. Trunk, Z. Yin, BNL, Upton, USA

Abstract

The Pair Distribution Function (PDF) beamline is the latest addition to the Diffraction and In Situ Scattering program at the National Synchrotron Light Source-II (NSLS-II) at Brookhaven national Laboratory. The DW100 damping wiggler on 28-ID is the source for two branch lines: X-ray Powder Diffraction beamline, XPD (28-ID-2) constructed in 2015 and Pair Distribution Function, PDF (28-ID-1) beamline constructed in 2018. The state-of-the-art gantry system (Bridge) of the PDF beamline end-station consists of two detector translation stages and one sample translation stage. The detectors and sample stage move at 300 mm/s using Linear Brushless DC motors that are controlled by a Geo Brick LV Delta Tau motor-controller. All three translation stages are equipped with absolute encoders and proximity sensors to avoid collisions. The control system slows down the stages when proximity switches are activated. A complex controls and safety systems with multiple custom features are required to provide the full functionality of the Bridge and to protect equipment and users. An optics conditioning module (OCM) located upstream of the Bridge contains clean-up slits, a fast shutter that is synchronized with detector frame rate, an alignment LASER, and an X-ray Energy Calibration System (ECS). The controls system of the OCM supports automatic operation of the ECS followed by unexpected beam dumps to recalibrate the X-ray wavelength.

INTRODUCTION

PDF is a powerful technique to study local structural fluctuations in complex materials, which are very often responsible for tuning their interesting properties. 28-ID-1 is the dedicated PDF beamline at the NSLS-II. The beamline is optimized for total scattering measurements over a large Q-range with very low background. Beamline end-station experimental setup is designed to provide complementary Wide angle X-ray Scattering (WAXS) and Small Angle X-ray scattering (SAXS) data along with PDF data to enable multi-modal approach. The beamline end-station Bridge provides fast exchangeability between PDF, SAXS and WAXS setups. A variety of sample environments including a (5-500) K liquid He cryostat coupled to a (0-5) T superconducting magnet, a (80-500) K liquid nitrogen cryostream, a hot air blower and a gas flow-cell heater are available on a large optical table located underneath the Bridge. Currently, the beamline can operate at two discrete energies, 75keV and 117 keV using 2 special cut monochromator crystals.

A schematic layout of the beamline optical configuration is shown in Figure 1. The first optical component of the PDF beamline, Side Bounce Monochromator (SBM) receives the part of white beam that is transmitted through the first crystal of the 28-ID-1 (XPD) beamline Double Laue monochromator. The SBM is used to select the X-ray photon energy of the PDF beamline and to focus the monochromatic X-ray beam in the horizontal plane. The second optical component of the PDF beamline, the Vertically Focusing Mirror (VFM) is used to focus the X-ray beam in the vertical plane. The Beam Diagnostic Module (BDM), located downstream of the VFM, contains beam-defining slits and a beam monitor. The PDF monochromatic beam travels between the First Optical Enclosure (FOE) and experimental hutch B inside a shielded beam transport.

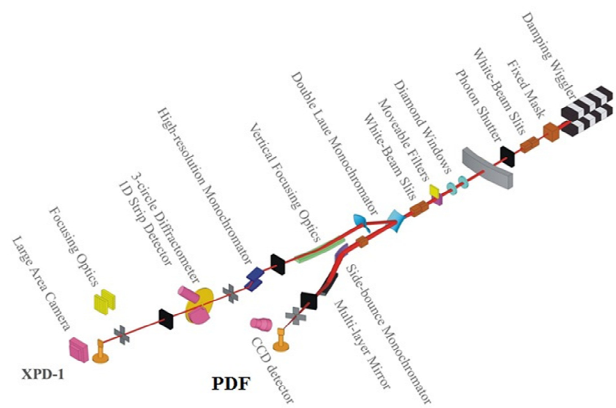


Figure 1: A schematic layout of the 28-ID optical configuration.

The Optics Conditioning Module (OCM) located upstream of the Bridge in hutch B consists of a fast photon shutter, a set of attenuators, clean-up slits, a beam alignment system using a LASER pointer, a telescopic beam guide, and a movable 2-circle Diffractometer for energy calibration. A 3-D CAD layout of the PDF end-station in shown in figure 2.

The two detector stages and the sample stage move independently from each other 3m along the path of the X-ray beam. Detector and sample stages are certified for carrying up to 200kg, and 100kg loads respectively. They are configured to move at 300 mm/s. The unique design of the bridge combines versatility of high speeds with experiment automation, and meets the NSLS-II Personal Protection System (PPS) and Equipment Protections System (EPS) requirements.

The NSLS-II uses Experiment Physics and Industrial Control Systems (EPICS) [1] as the controls framework.

* This research used resources of the National Synchrotron Light Source II, a U.S. Department of Energy (DOE) Office of Science User Facility operated for the DOE Office of Science by Brookhaven National Laboratory under Contract No. DE-SC0012704."
†oksana@bnl.gov

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-THBPP05

IMPLEMENTING ODIN AS A CONTROL AND DATA ACQUISITION FRAMEWORK FOR EIGER DETECTORS

G. Yendell, M. Taylor, U. Pedersen, Diamond Light Source, Oxfordshire, UK
T. Nicholls, A. Neaves, STFC/RAL, Chilton, Didcot, Oxon, UK
A. Greer, OSL, Cambridgeshire, UK

Abstract

The increasing data throughput of modern detectors is a growing challenge for back-end data acquisition systems. OdinData provides a scalable framework for data acquisition used by multiple beamlines at Diamond Light Source (DLS). While it can be implemented standalone, OdinControl is used to provide a convenient interface to OdinData. Eiger detectors at DLS were initially integrated into the Odin framework specifically for the data acquisition capability, but the addition of detector control provides a more coherent and easily deployable system. OdinControl provides a generic HTTP API as a single point of control for various devices and applications. Adapters can abstract the low-level control of a detector into a consistent API, making it easier for high-level applications to support different types of detector. This paper sets out the design and development of Odin as a control system agnostic interface to integrate Eiger detectors into EPICS beamline control systems at DLS, as well as the current status of operation.

INTRODUCTION

In an effort to stay on the forefront of scientific research in their respective fields, many beamlines at DLS have chosen to integrate Eiger detectors. Eiger detectors are very extensible, exemplified by the different use cases of the various deployments. Some beamlines require the maximum frame rate possible, while others run at much lower rates of less than 10 Hz and simply benefit from the quality of the data produced. With these varied use cases, a dynamic control and data acquisition framework is required to cope with large data output in some cases, while keeping more basic deployments as simple as possible. The system should only be as complex as is necessary to achieve the given requirements of the beamline.

Effective integration of Eiger detectors into control systems at DLS is enabling beamlines to provide world-leading research opportunities in various science cases and will also provide the scope for experiments that were not feasible before [1].

Eiger Detectors at Diamond Light Source

Eiger detectors have a photon count rate of $5 \times 10^8\ \mathrm{s^{-1}mm^{-2}}$, a pixel size of $75\,\mu\mathrm{m}$ and a read out time of $3\,\mu\mathrm{s}$, which, amongst many other cutting-edge features, has led to quick uptake on beamlines at DLS. The Eiger comes in a variety of sizes suited to different applications, of which DLS currently makes use of the Eiger2 XE 16M, Eiger1 X and Eiger2 X 4M and Eiger1 X 500K models [2]. Eiger detectors have become commonplace on beamlines at DLS in the last 18 months and more are scheduled to be commissioned in the near future. This presents many challenges, not least the file system load of running multiple instances at the full data rate simultaneously and in some cases with minimal down time between acquisitions. It also presents a challenge in continually commissioning control and data acquisition frameworks for detectors as they arrive at various beamlines, as well as keeping them up to date.

Having to support various detector models might complicate the software required to drive them, due to different demands. Presenting the data rate of different models of Eiger is not trivial, as it is obfuscated by variable bit depths, sustained/burst thresholds and compression ratio depending on signal strength. However, broadly speaking, the maximum data rate for all models is approximately the same and in any case primarily limited by the 40 Gb network link. The choice of model is effectively a decision balancing time-resolution against the collecting area. This makes the software design decisions somewhat simpler; because different models have roughly equivalent performance requirements, the scale of the system to be deployed is mainly defined by the use case of the beamline.

X-ray Imaging and Coherence beamline I13 employs an Eiger1 X 500K to make efficient use of the high coherent flux available at the sample. The major use case is for ultrafast diffraction imaging, primarily ptychography [3]. This is a method of collecting a series of diffraction patterns from a sample, in a grid with overlapping illumination regions, and processing the datasets to create an image. Ptychography allows higher spatial resolution images than using conventional lens imaging. The combination of high flux and the inherent noise resilience of the technique enables scans with exposures of the order of $100\,\mu\mathrm{s}$, which makes scans at 10 kHz feasible. This pushes the boundaries of trajectory scanning and data acquisition technologies.

Macromolecular Crystallography (MX) beamlines, such as Microfocus MX beamline I04, make the greatest use of Eiger detectors at DLS. The large collection area of the 16M and 4M models is ideal for collecting crystal diffraction data, while the small pixel size enables better sampling of diffraction spot profiles and improved signal-to-noise from smaller diffraction spots. The high frame rates are a particular asset for MX beamlines that invest in automation and high throughput of samples, due to reduced acquisitions times. For these beamlines, Eiger is the natural upgrade from the Pilatus detectors that have been depended on for many years.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-THCPL04

SCIBORG: ANALYZING AND MONITORING LMJ FACILITY HEALTH AND PERFORMANCE INDICATORS

J-P. Airiau, S. Vermersch, V. Denis, P. Fourtillan, C. Lacombe
CEA - CESTA, Le Barp, France

ABSTRACT

The Laser MegaJoule (LMJ) is a 176-beam laser facility, located at the CEA CESTA laboratory near Bordeaux (France). It is designed to deliver about 1.4 MJ of energy to targets, for high energy density physics experiments, including fusion experiments. It operates, since June 2018, 5 of the 22 bundles expected in the final configuration. Monitoring system health and performance of such a facility is essential to maintain high operational availability. SCIBORG is the first step of a larger software that will collect in one tool all the facility parameters. Nowadays SCIBORG imports experiment setup and results, alignment and PAM control command parameters. It is designed to perform data analysis (temporal/crossed) and implements monitoring features (dashboard). This paper gives a first user feedback and the milestones for the full spectrum system.

INTRODUCTION

Monitoring and analyzing critical data (or indicators) of the LMJ is essential to operate such a facility. It contributes to manage availability and planning actions for the new experiments and follow the global performances for the achieved experiments.

GOALS

The number of equipment to control (from laser bays and target chamber) is very large and logically multiplies the number of indicators to monitor. Each equipment and associated control system generates control data (configuration, log, results, etc.); indicators are extracted from these data directly or indirectly. These data are of different types such as structured (log), semi-structured (scalar results, tabulated data) and un-structured (image, pdf report, etc.). To be able to correlate these data among themselves, it is necessary to standardize their format when they are extracted from the source producer. The data extraction sampling rate is determinant to estimate the database storage capacity and the performance to access to the formatted indicators. In order to speed up the request to the database, some additional information is associated to the data in the form of metadata. These metadata include the specific facility structure (bundle, quad, beam, section, diagnosis), they contribute to simplify the data classification.

Once the extraction and transformation done (data + metadata), the indicators are stored in a database (see Fig. 1). The expected application above this database will perform the following features:

- Computation of complex data from elementary data (z=f(a,b,c,...) , ex.: total energy on target), physical value (transfer function), security factor (vacuum windows damages),
- Monitoring temporal evolution of indicator (elementary data) and performances (complex data or physical value), performing security analysis (security factor),
- Crossed analysis y = f(x) (ex. amplification gain vs energy bank efficiency),
- Smart GUI including rules on data (ex.: warning on threshold overcoming) and multiple types of chart for analysis,
- Dashboard creation based on analysis synthesis
- Predictive analysis: predictive maintenance (planning of optic replacement), calibration (fine tuning of coefficient used on embedded software).
-

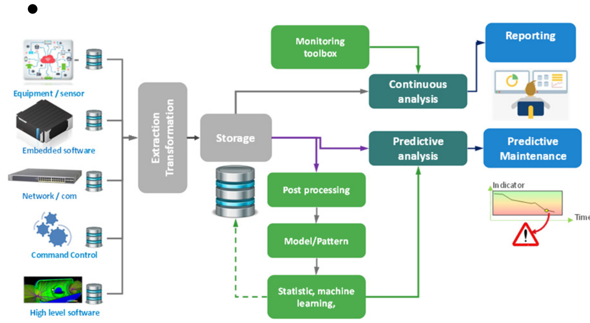


Figure 1: Functional architecture.

Hypothesis

In order to limit the data number and complexity, the type of indicator is limited to structured (log) and semi-structured (scalar and tabulated values) data. The extraction sampling rate is not less than 10 s. This hypothesis is based on the fact that most of the equipment to monitor has a response time bigger than this criterion. The embedded software and operating tools algorithms can generate data at very high frequency but we suppose that these indicators are available as tabulated data at the end of computation. This supposition limits the frequency of the extracting requests.

Data Perimeter

The global architecture of the LMJ facility is structured in layer from equipment to high level software [1].

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-THCPL05

SIGNAL ANALYSIS FOR AUTOMATED DIAGNOSTIC APPLIED TO LHC CRYOGENICS AT CERN

K. O. E. Martensson, B. Bradu, G. Ferlin, CERN, Geneva, Switzerland

Abstract

The operation of the LHC at CERN is highly dependent on its associated infrastructure to operate properly, such as its cryogenic system where many conditions must be fulfilled for superconducting magnets and RF cavities. In 2018, the LHC cryogenic system caused 172 hours of accelerator downtime (out of 5760 running hours). Since the cryogenics recovery acts as a time amplifier, it is important to identify not optimized processes and malfunctioning systems at an early stage to anticipate losses of availability. The LHC cryogenic control systems embeds about 60,000 I/O whereof more than 20,000 analog signals which have to be monitored by operators. It is therefore crucial to select only the relevant and necessary information to be presented. This paper presents a signal analysis system created to automatically generate adequate daily reports on potential problems in the LHC cryogenic system which are not covered by conventional alarms, and examples of real issues that have been found and treated during the 2018 physics run. The analysis system, which is written in Python, is generic and can be applied to many different systems.

INTRODUCTION

The LHC (Large Hadron Collider) at CERN (European Organization for Nuclear Research) consists of eight 3.3 km long cryogenically independent sectors with shafts to the surface in between them. These shafts between two sectors and associated infrastructure are denoted as *points*. Cryogenic plants are installed in five out of nine points as illustrated in Fig. 1 and together they cool down a mass of 36,000 tonnes to a temperature of 1.9 K, making it the largest cryogenic system in the world. This paper presents an analysis system constructed with the purpose of detecting non-urgent problems in the cryogenic system that are not caught by conventional alarms and are difficult for operators to detect manually. Archived data are analyzed daily by eleven different analysis algorithms, after which reports for operators are generated and published online.

The cryogenic plants are using similar setups which makes automatic and coherent checks of the systems advisable. The analysis is therefore distributed among eight VMs (Virtual Machines), each analyzing one sector. Approximately 40 analysis jobs, together analyzing about 1000 signals, is defined for each sector. The system is not replacing the daily operator surveillance, but is helping the cryogenic operations team fulfilling its main objective of maximizing the availability and performance of the LHC's cryogenic system by:

- Anticipating potential failures or loss of availability.
- Detecting unoptimized processes and malfunctioning sensors.
- Prolonging the life span of equipment.
- Reducing the need for repetitive manual operator checking and thereby freeing operator to other tasks.

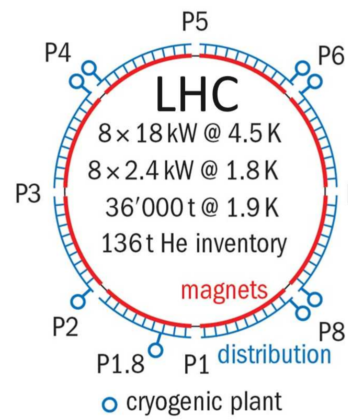


Figure 1: Distribution of cryogenic plants in the LHC [1].

INFRASTRUCTURE

Each sector is set up to be analyzed by a dedicated VM to ensure parallel and independent execution of new and updating analyses. These VMs are Linux machines that are set up on CERN's OpenStack. They are connected to a local machine using the software PuTTY©. New analyses are started automatically through cron scheduling every night and are performed on data spanning from the latest already available results up to the most recent midnight. It is also possible to redo a previous analysis with new parameters, and in that case only the explicitly queued time frame will be analyzed. This is useful when applying new or edited job specifications to dates which have already been analyzed. Triggered warnings are ranked by how severely the thresholds have been violated and are presented on a web site.

Python was chosen as programming language due to its versatility. Its many available modules and wide functionality gives great possibilities for easy expansion of the software to other types of systems and analyses. It is of great significance that the constructed analysis system can be adjusted with small effort to other systems at CERN, such as the computing cluster. At the time of construction, the necessary signals from the cryogenic system are not accessible from the computing cluster in a manner that it would make computations faster or more reliable. This is set to change in the future when the the currently undergoing change of primary logging service will be completed. Python is very commonly used at CERN and is therefore very probable to maintain

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-THCPL06

INTRODUCING BIG DATA ANALYSIS IN A PROTON THERAPY FACILITY TO REDUCE TECHNICAL DOWNTIME

P. Fernandez Carmona[†], Z. Chowdhuri, S. G. Ebner, F. Gagnon-Moisan, M. Grossmann, J. Snuverink, D. C. Weber, Paul Scherrer Institut, 5232 Villigen PSI, Switzerland

Abstract

At the Centre for Proton Therapy of the Paul Scherrer Institute (PSI) about 450 cancer patients are treated yearly using accelerated protons in three treatment areas. The facility is active since 1984 and for each patient we keep detailed log files containing machine measurements during each fraction of the treatment, which we analyse daily to guarantee dose and position values within the prescribed tolerances. Furthermore, each control and safety system generates textual log files as well as periodic measurements such as pressure, temperature, beam intensity, magnetic fields or reaction time of components. This adds up currently to approximately 5 GB per day. Downtime of the facility is both inconvenient for patients and staff, as well as financially relevant. This article describes how we have extended our data analysis strategies using machine archived parameters and online measurements to understand interdependencies, to perform preventive maintenance of ageing components and to optimize processes. We have chosen Python to interface, structure and analyse the different data sources in a standardized manner. The online channels have been accessed via an EPICS archiver.

INTRODUCTION

The Paul Scherrer Institute (PSI) in Switzerland started treating tumours using accelerated protons in 1984. Since then more than 9000 cancer patients have been treated at its fixed beamline for eye irradiation and three gantries.

The facility continuously produces large amounts of data originating from the particle accelerator, beamlines and control and safety systems needed for the dose delivery. Part of this data, especially everything directly related to patient irradiation, gets stored for online and future analysis. Patient treatment log files get analysed daily throughout the treatment duration to guarantee the delivery standards. Some sensor data is used for forensics after particular machine malfunctions, and other sources are used as surrogates to estimate when a part needs to be replaced due to ageing or wear.

Downtime due to unexpected failure of components is both inconvenient for patients and personnel, and expensive in financial terms due to unused clinical resources and paused revenue. With the goal in mind to reduce downtime and to improve preventive maintenance we started a pilot to introduce structured big data analysis techniques in our facility using the extensive available data. In the present paper we will describe the data available, our steps to classify it and process it as well as the first promising results.

DATA SOURCES

PROSCAN, the facility dedicated to treating patients at PSI, is a complex set of interconnected but largely independent and heterogeneous subsystems. There are several control, safety and monitoring systems generating status data at different rates and formats. The first part of this work was to list and attempt to classify all the available data for its future analysis.

Machine Data

A superconducting cyclotron provides protons to the treatment areas by means of 5 beam lines. These contain 17 steering magnets, 45 quadrupoles, 8 deflecting dipoles, 47 beam monitors, 14 beam blockers and several other auxiliary elements each with its set point and actual status values. Most of these values are permanently available, some on request as they are beam-disrupting.

The usage and sharing of the beam across the different areas also gets monitored and archived. It is always possible to know at any given time which area had mastership, that is, control and access to the beam, and at which energy and intensity. This can be useful for forensics, but also for statistics, see Figure 1, and to identify potential inefficiencies.

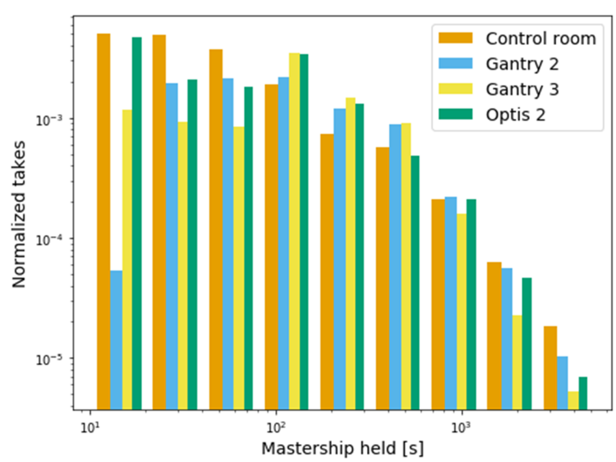


Figure 1: Histogram of mastership duration per area in 2019.

Additionally each treatment area stores dosimetry-relevant parameters not limited to but including humidity, pressure and temperature at different locations. The control systems also store technical values from electronics and sensors, namely supply voltage, power consumption or internal temperature.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-THCPL07

EXPERIENCE USING NuPIC TO DETECT ANOMALIES IN CONTROLS DATA*

T. D'Ottavio†, P. S. Dyer, J. Piacentino, Jr., M. R. Tomko
Brookhaven National Laboratory, Upton, USA

Abstract

NuPIC (Numenta Platform for Intelligent Computing) is an open-source computing platform that attempts to mimic neurological pathways in the human brain. We have used the Python implementation to explore the utility of using this system to detect anomalies in both stored and real-time data coming from the controls system for the RHIC Collider at Brookhaven National Laboratory. This paper explores various aspects of that work including the types of data most suited to anomaly detection, the likelihood of developing false positive and negative anomaly results, and experiences with training the system. We also report on the use of this software for monitoring various parts of the controls system in real-time.

INTRODUCTION

An *anomaly* can be considered a point in time when the behavior of a system is significantly different from previous, normal behavior. For the purpose of this paper, we consider only anomalies in numeric time-series data. Most controls data fall into this category.

Why work on anomaly detection? For most subsystems within controls, we can usually identify data that experts would consider normal - at least for some period of time. Normal behavior might be defined as data within a certain range, or with a particular pattern, or changing within a prescribed rate, or some combination of all of these. Experts can usually pick out deviations from normal. But the volume of data accumulated in modern control systems does not allow for such review. We need computers and algorithms to detect these anomalies.

NUPIC DESCRIPTION AND SETUP

NuPIC stands for Numenta Platform for Intelligent Computing. It is a software system designed to mimic the neural algorithms used within the neocortex of the human brain [1]. Numenta's goal is to reverse-engineer the neocortex and apply that knowledge to the creation of machine intelligence [2]. The NuPIC software, which originated in 2013, is open source on GitHub [3]. Originally written in C++, the most popular implementation is in Python. There is also a third-party port to Java [4]. The Python implementation provides three APIs. At the highest level (easiest to implement, least customizable) is the Online Prediction Framework API, or OPF. At the lowest level (easiest to customize, most difficult to implement) is the Algorithm API. A middle ground is provided by the Network API. The work in this paper was done with the Python version of the NuPIC OPF 1.05. [5].

The NuPIC software uses a software algorithm called Hierarchical Temporal Memory (HTM), which uses stored data sequences to make predictions about future data. These predictions are then compared to the actual data delivered to calculate prediction errors which, along with prediction errors from surrounding data, are transformed into anomaly likelihood values. It is these likelihood values that ultimately determine if the software found an anomaly [6]. (We actually enhanced these values using an extension that will be described later.)

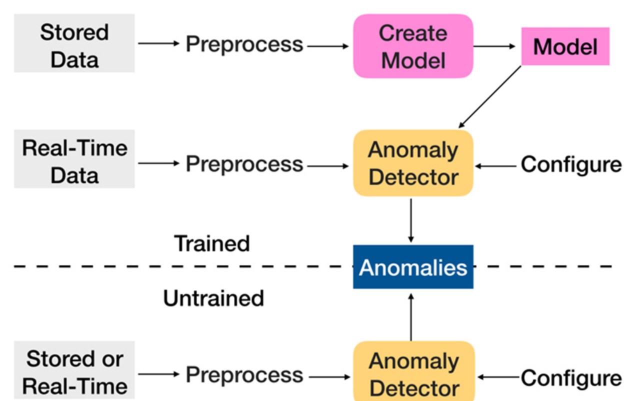


Figure 1: System diagram.

As seen in Fig. 1, the software system can be run in both a trained or untrained mode. When training, you create a NuPIC model using "normal" data. This model is then loaded into the NuPIC software prior to looking for anomalies. Once the model is loaded, you can instruct the software to continue learning from new data, or to turn learning off. In the untrained scenario, the NuPIC software is learning on new data as it arrives, though you can control which data is used for learning. Stored data, when it was used, was retrieved from our logging/archiving systems. Real-time data came from devices connected to the control system reporting at a rate of 1 Hz or slower. Preprocessing, when required, involved averaging, sampling or filtering of the data.

The heart of the system is the Anomaly Detector, which holds the NuPIC software. This software has about 30 parameters that can be tweaked to adjust the sensitivity and performance for various data sets [7]. However, Numenta delivers a set of parameters tuned specifically for anomaly detection as part of their Online Prediction Framework. With one exception described later, we used this set for all of the results that follow.

* Work supported by Brookhaven Science Associates, LLC under Contract No. DE-SC0012704 with the U.S. Department of Energy.
† dottavio@bnl.gov

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-THCPR01

NOVEL FPGA-BASED INSTRUMENTATION FOR PERSONNEL SAFETY SYSTEMS IN PARTICLE ACCELERATOR FACILITY

S. Pioli[1]*, O. Frasciello[1], M. M. Beretta[1], P. Ciambrone[1], D. G. C. Di Giulio[1], B. Buonomo[1], L. G. Foggetta [1], M. Belli[1], P. Valente[2], A. Variola[2]
[1]INFN-LNF - Frascati National Laboratory, Rome, Italy
[2]INFN-Roma1 - Section of Rome, Italy

Abstract

Personnel Safety System for particle accelerator facility involves different devices to monitor gates, shielding doors, dosimetry stations, search and emergency buttons. In order to achieve the proper reliability, fail-safe and fail-proof capabilities, these systems are developed compliant with safety standards (like the IEC-61508 on "Functional Safety", ANSI N43.1 "Radiation Safety for the design and operation of Particle Accelerator" and NCRP report 88) involving stable technologies like electro-mechanical relays and, recently, PLC.

As part of the Singularity project at Frascati National Laboratories of INFN, this work will report benchmark of a new FPGA-based system from the design to the validation phase of the prototype currently operating as personnel safety system at the Beam Test Facility (BTF) of Dafne facility. This novel instrument is capable of: devices monitoring in real-time at 1 kHz, dual modular redundancy, fail-safe and fail-proof, multi-node distributed solution on optical link, radiation damage resistance and compliant with IEC-61508, ANSI N43.1 and NCRP report 88.

The aim of this FPGA-based system is to illustrate the feasibility of FPGA technology in the field of personnel safety for particle accelerator in order to take advantage of a fully digital system integrated with facility control system, evaluate the related reliability and availability and realize a standard, scalable and flexible hardware solution also for other fields with similar requirements like machine protection systems.

INTRODUCTION

Particle accelerators require Personnel Safety Systems (PSSs) in order to reduce as much as possible the risk of an accidental exposition of workers to ionizating radiation. These kind of systems must provide access control to any area involved with the accelerator facility (monitoring gates, shielding doors, dosimetry stations, search and emergency buttons) and produce an enabling signal to allow operation to radio-frequency systems.

In order to design properly a PSS, regulation and best practice guide lines and industrial standards are available, like IEC-61508 on "Functional Safety" [1], NCRP report 88 on "Radiation Alarms and Access Control Systems" [2] and ANSI report 43 on "Radiation Safety for the Design and Operation of Particle Accelerator" [3].

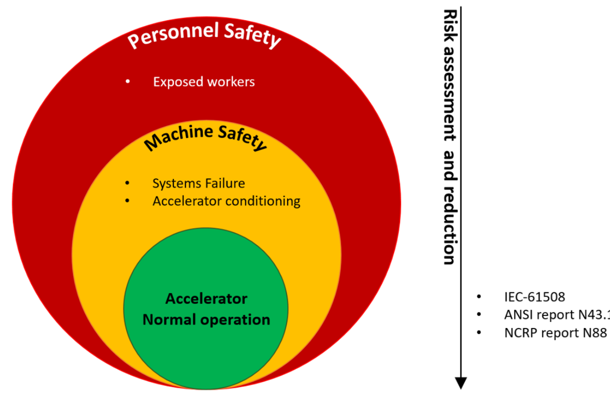


Figure 1: Risk assessment priority for particle accelerator facility.

At the National Laboratories of Frascati of the INFN, we developed a method to design and commissioning FPGA-based safety systems that could be involved for personnel and machine protection (MPS), compliant with the three standard listed in the previous paragraph. Such systems are designed, from both hardware and software point-of-view, to match with risk assessment and response time requirements, Fig. 1, of the hosting particle accelerator facility.

Up to this moment several commercial solutions are available to suite with PSS and MPS requirements. PLCs can easily achieve required performances in terms of response time nevertheless introducing the compliancy with functional safety standards only few product can be involved for PSS purposes, due to a latency in response time of ~100 ms, and anyway with compliancy only from the hardware point-of-view. For these reasons we chose to develop hardware and software of a new kind of high reliability FPGA-based device according with IEC-61508 standard, NCRP report 88 and ANSI report 43.1 to match all requirements of PSS and MPS. According with our experience with IEC-61508 compliant safety systems [4], in this paper will be presented prototypes developed to operate as PSS (because it has higher constraints in terms of reliability compared to MPS) in order to investigate the feasibility of our method to realize safety system suitable for new and old accelerator facility of the INFN with modern and mature technologies like FPGA and dismiss old and expensive relay crates.

The minimal features of such device will include:

- Response time able to perform real-time intervention (for accelerators with repetition rate of at least 1 kHz);

* stefano.pioli@lnf.infn.it

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-THCPR02

TARGET CONTROL AND PROTECTION SYSTEMS LESSONS FROM SNS OPERATIONS*

D. L. Humphreys†, Oak Ridge National Laboratory Spallation Neutron Source, Oak Ridge, USA

Abstract

The Spallation Neutron Source (SNS) at Oak Ridge National Laboratory has been in operations since 2006 and proposes a project to build a Second Target Station (STS) to effectively double potential scientific output. The SNS target controls operate in a harsh environment which includes high radiation, exposure to gaseous radionuclides, and activated liquid mercury and mercury vapor. These conditions necessitate protective interlocks and credited controls for protection functions to ensure proper response to off-normal conditions. In order to inform the design of target controls for the STS, we have examined lessons learned during SNS operations regarding the design and implementation of the control and protection systems for the first target station (FTS). This paper will examine various aspects of the performance of the target control and protection systems including reliability, maintainability and sustainability given the challenging environment created by 1.4 MW operations. Specific topics include distributed control of various target subsystems, response to loss of power, selection of nuclear grade instrumentation, and applying these lessons to the design for the STS project.

INTRODUCTION

The Spallation Neutron Source (SNS) [1] is an accelerator based neutron source at the Oak Ridge National Laboratory (ORNL) in Oak Ridge, Tennessee. The SNS is a U.S. Department of Energy user facility that hosted over 1300 users in 2018. It provides the world's most intense, pulsed neutron source for scientific research. The SNS machine consists of a 1.4 MW accelerator that delivers pulsed protons to a liquid mercury target which in turn spalls neutrons. The neutrons are moderated and guided through 16 instrument beamlines to a variety of sample environments for research in a broad range of scientific disciplines.

The existing complex was designed with provisions for a Second Target Station (STS) which would provide space for up to 22 additional instruments. The STS would share beam pulses from the existing SNS accelerator. The STS will provide intense cold neutrons with a longer wavelength which will significantly enhance neutron brightness as compared to the first target station (FTS). ORNL is currently working on the STS conceptual design. [2]

* This manuscript has been authored by UT-Battelle, LLC, under contract DE-AC05-00OR22725 with the US Department of Energy (DOE). The US government retains and the publisher, by accepting the article for publication, acknowledges that the US government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this manuscript, or allow others to do so, for US government purposes. DOE will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan (http://energy.gov/downloads/doe-public-access-plan)
† humphreysdl@ornl.gov

The STS target design calls for a segmented rotating assembly (Fig. 1) consisting of tantalum-clad tungsten blocks. The target rotation will be controlled so that it is synchronized to the proton beam using the existing accelerator timing system, which will be extended to the new target station.

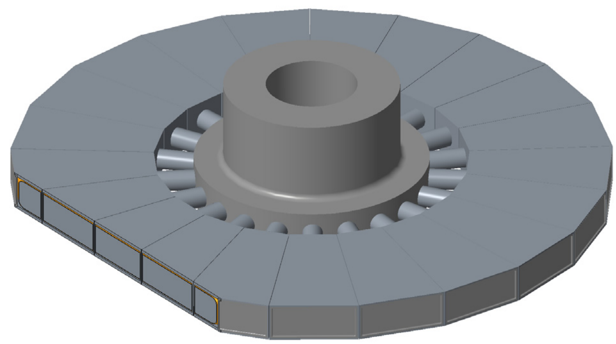


Figure 1: STS Target assembly disk conceptual design, with cutaway to show tantalum cladding on tungsten blocks.

In order to inform the design for STS target controls, we have examined lessons learned from operating and maintaining the FTS for over a decade. Relevant topics include distribution of processors for target subsystems, response to loss of PLC power, selection and maintenance of instrumentation, and credited controls and protective functions Lessons Learned from FTS

Integrated Control System Structure for Target Subsystems

The existing SNS Integrated Control System (ICS) uses the Experimental Physics and Industrial Control System (EPICS) toolkit as an integrating framework for a large set of diverse devices used to monitor and control the accelerator complex, target, and instrument suite. The scope for STS target controls includes control system hardware, software, interlocks and user interfaces integrated with the SNS ICS. The target controls design for STS will follow the FTS model of using commercial industrial PLCs with appropriate instrumentation to provide STS target control. The process instrumentation and control for the target systems will be designed to connect to the existing machine control system, in a similar manner as FTS, to provide both safety-related and non-safety-related control, equipment protection, and monitoring for the target systems. The target instrumentation will interface to the machine network so that the target systems can be controlled from either the Central Control Room or the STS Target Control Room. Target startup and control system maintenance will be performed from the STS Target Control Room.

The existing FTS target controls architecture uses IOC/PLC pairs shared across multiple target subsystems.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-THCPR03

A SAFETY RATED FPGA FRAMEWORK FOR FAST SAFETY SYSTEMS

F. Tao†, B.M. Bennett, D.G. Brown, J. Jones, M. Stettler
SLAC National Accelerator Laboratory, Menlo Park, USA

Abstract

In the accelerator community, majority of credited safety systems are implemented with safety PLCs to reduce the enormous burden on functional safety compliance. However, for safety functions require fast response or complex signal processing, FPGA is often the only feasible solution, and the system design has to comply with the IEC 61508 standard. In this paper, we will introduce a generic safety-rated FPGA design template. FMEDA analysis, hardware reliability modelling, firmware development, verification and validation will be described in details to demonstrate the IEC 61508 compliant development process. In this dual redundant design, each chain consists a FPGA chip from different manufacturers to minimize the potential common cause failure. Cross checks between FPGAs and end-to-end testing are performed to increase the diagnostic coverage and improve the reliability. Based on this safety FPGA template, an Average Current Monitor (ACM) system is developed at SLAC with the addition of a safety PLC for diagnostics and a HMI for user interface. The overall system is deployed as part of Beam Containment System (BCS) to limit the beam current with the target Safety Integrity Level (SIL) 2.

DEVELOPMENT BACKGROUND

In SLAC, Radiation Safety Systems (RSS), which includes Personnel Protection System (PPS) and Beam Containment System (BCS), traditionally use simple analog/digital electronics and proven concepts to avoid complex failure modes. In the modernization of RSS, PPS uses safety PLCs to replace relay logic both at the system level, zone level and down to the chassis level. The adoption of safety PLC not only improves the safety and reliability performance but also makes the configuration control easier and more reliable. However, when it comes to BCS, PLC alone is not a feasible solution. As some BCS sensor signals require a fast and/or complex logic processing, such as beam loss monitors or average current monitors. Using safety PLC can't meet the response time requirements. For this reason, we have to find a new engineering solution for electronics design of safety critical applications.

Just cite the LCLS-II (Linac Coherent Light Source) BCS as an example. Since the maximal designed beam power is 1MW, the corresponding response time for the BCS is 200 micro-second, which is beyond the capability of any safety PLC on the market. Therefore, for all LCLS-II BCS control systems and subsystems, we have to split the system into fast portion and slow portion. A Siemens S7-1515F safety PLC will interlock to those slow sensors and customized electronics deal with the fast portion of the system.

Complex logic processing required in the BCS instruments easily rules out the potential of using traditional electronics. There are only two options left for safety-critical electronics system development. The first one is safety rated micro-controller (MCU). This type of IC has been widely used in automotive applications or has been used by automation vendors to build safety controllers such as PLC, motor drive etc. At the time of writing this paper, the fastest general purpose safety micro-controller from Texas Instrument (TI), a market leader on safety MCU, is TMS570 with 330MHz frequency. Considering that MCU requires multiple instruction execution periods to finish a task, which make it still impossible to meet the requirements for dealing with 20MHz sampling, which is a minimal requirement when deals with the beam related analogue signal processing.

The other option is FPGA (Field Programmable Gate Array). This is a type of IC widely used in the accelerator instrumentation and control. It can process inputs in parallel, which is advantage over the MCUs. As the quick technology development, it is much easier to find FPGA solutions for applications with even higher sampling frequency requirements. For this reason, we decided to adopt FPGA in the development for fast interlock systems.

SYSTEM'S FUNCTIONAL SAFETY

The functional safety standard IEC 61508 was published in 2000 and the second edition has been available since 2010. This functional safety standard is applicable to customized electronics development such as BCS. In addition, there is a guidance document from the US Department of Energy (DOE) on use IEC 61511, a second tier functional safety standard for process industry applications, to BCS design.

IEC 61508 is a performance based standard that uses Safety Integrity Level (SIL) as the measure for system's reliability performance on a particular safety function. To apply this standard, safety functions within the system have to be defined first; then for each safety function, assign a SIL based on the risk assessment.

In a SLAC's typical BCS, there are two functions have the most criticality. One is the beam loss detection, and the other is the beam energy limit. The first is vital to protect the safety devices such as safety collimators, stoppers etc., while the second function is used to establish the basis for ray trace study, a method used by radiation protection physicists on risk assessment. During the risk assessment, we determine that the SIL rating is 2. For this reason, we will target the FPGA design for SIL2 applications.

† fengtao@slac.stanford.edu

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-THCPR04

THE EUROPEAN XFEL BEAM LOSS MONITOR SYSTEM

T. Wamsat*, T. Lensch
Deutsches Elektronen-Synchrotron DESY, Hamburg, Germany

Abstract

The European XFEL MTCA based Beam Loss Monitor System (BLM) is composed of about 470 monitors, which are part of the Machine Protection System (MPS). The BLMs detect losses of the electron beam, in order to protect accelerator components from damage and excessive activation, in particular the undulators, since they are made of permanent magnets. Also each cold accelerating module is equipped with a BLM to measure the sudden onset of field emission (dark current) in cavities. In addition some BLMs are used as detectors for wire- scanners. Experience from the already running BLM system in FLASH2 which is developed for XFEL and tested here, led to a fast implementation of the system in the XFEL. Further firmware and server developments related to alarm generation and handling are ongoing.

The BLM systems structure, the current status and the different possibilities to trigger alarms which stop the electron beam will be presented.

INTRODUCTION

The Beam Loss Monitor (BLM) system at the European XFEL is the main system to detect losses of the electron beam, thus to protect the machine hardware from radiation damage in particular the permanent magnets of the undulators. As part of the Machine Protection System (MPS) [1] the BLM system delivers a signal which stops the electron beam as fast as possible in case the losses get too high.

In addition there are Beam Halo Monitors (BHM) [2] in front of the beam dumps using the same digital backend as the BLM electronics.

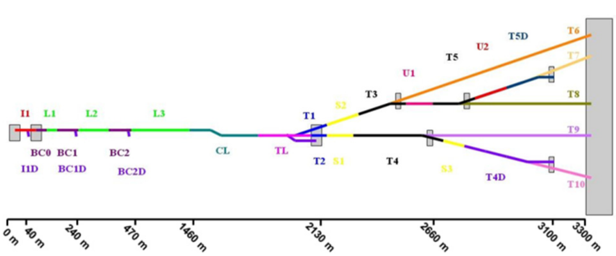


Figure 1: Section overview of the European XFEL accelerator [3].

About 470 BLMs are installed along the XFEL Linac which schematically is shown in Fig. 1. The BLMs are positioned at locations near the beamline, where losses can be expected or where sensitive components are installed, thus most of the BLMs are installed in the undulator area (see table 1). Since even a big number of BLMs cannot provide a complete survey of losses, there is also a toroid based Beam Current Monitor system [4] installed, which provides transmission interlock system to stop the beam if too much charge gets lost along the machine. Also each superconducting accelerating module is equipped with one BLM at the end to detect the field emission produced by cavities (Correlation work still ongoing, no paper available yet).

* thomas.wamsat@desy.de

Table 1: BLM Distribution

Section	#
I1, BC0	24
BC1	18
BC2	23
L1, L2, L3 (field emission)	98
CL, TL	71
S1, T4	80
S3	48
T4D	10
S2, T3	80
U1, T5, U2, T5D	20

Some selected BLMs are also used as additional detectors for wire scans [5].

SYSTEM OVERVIEW

The hardware consists of the BLM devices, a dedicated Rear Transition Module (RTM) in combination with the DESY Advanced Mezzanine Card DAMC2. Furthermore a MPS card is required for alarm output collection.

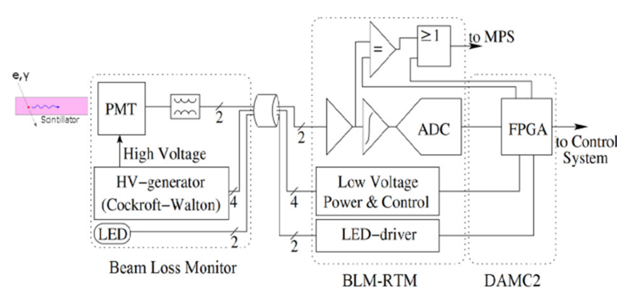


Figure 2: BLM system scheme.

The BLM includes either an EJ-200 plastic scintillator or a SQ1 quartz glass rod. The latter are used mainly in the undulator intersection. In contrast to scintillators, that are also sensitive to hard x-rays, the Quartz rods work with Cherenkov effect, that is sensitive to particle loss only.

The high voltage for the photomultiplier (PMT) is generated within the BLM, so no high voltage cables are needed (see Fig. 2), just a CAT 7 cable with RJ-45 connectors is used. A LED can be switched on within the BLM to provide test pulses to check if the PMT is still working.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-THCPR06

THE ITk COMMON MONITORING AND INTERLOCK SYSTEM

S. Kersten†, P. Kind, M. Wensing, Bergische Universitaet, Wuppertal, Germany
C. W. Chen, J.- P. Martin, N. Starinski, GPP, Montreal, Canada
E. Stanecka, IFJ-PAN, Krakow, Poland
I. Mandiz, JSI, Ljubljana, Slovenia
P. Phillips, STFC/RAL, Chilton, Great Britain
D. Florez, C. Sandoval, Universidad Antonio Narino, Bogota, Colombia
S. Connell, University of Johannesburg, Johannesburg, South Africa

Abstract

For the upgrade of the Large Hadron Collider (LHC) to the High-Luminosity Large Hadron Collider (HL-LHC) the ATLAS detector will install a new Inner Tracker (ITk), which consists completely of silicon detectors. The most inner part is composed of pixel detectors, the outer part of strip detector elements. All together ca. 28000 detector modules will be installed in the ITk volume. Although different technologies were chosen for the inner and outer part, both detectors share a lot of commonalities concerning their requirements. These are operation in the harsh radiation environment, the restricted space for services, and the high power density, which requires a very efficient cooling system. While the sub-detectors have chosen different strategies to reduce their powering services, they share the same CO_2 cooling system. The main risks for operation are heat ups and condensation, therefore a common detector control system is under development. It provides a detailed monitoring of the temperature, the radiation and the humidity in the tracker volume. Additionally, an interlock system, a hardware based safety system, is designed to protect the sensitive detector elements against upcoming risks. Another constraint is that - once the detector is installed - its components are not accessible for several years or even for the lifetime of the detector. Thus the control system must be fault tolerant and provide very good remote diagnostics. The components of the ITk common monitoring and interlock system are presented.

INTRODUCTION

In the third long shutdown of the LHC, when it will be upgraded to the high-luminosity LHC, the ATLAS experiment will install a new tracker. It consists of the pixel detector at small radius close to the beam pipe and a large area strip detector surrounding it [1], [2]. Although the segmentation of the sensor cells will be different for the pixel and strip detector, the ITk will be an all silicon detector.

Therefore, the design of the tracker is driven by major common requirements. Main constraints for the construction of the detectors and their control systems are the high power density, the harsh radiation environment and the low material budget requirement inside the tracker volume.

Pixel and strip detector modules are composed of the sensor and the related front end electronics. In total there are about 10000 pixel detector modules with 6 billion channels covering an area of ca. 13 m^2. The strip detector consists of roughly 18000 modules with 60 million electronic channels and covers an active silicon area of 165 m^2. These large number of readout channels cause the high power density inside the tracker volume. The modules are mounted on different carrier structures. These are staves in the barrel region, while rings and disks carry the modules of the endcaps. Thin cooling pipes are integrated into all carrier structures.

To handle the low material budget requirement the sub-detectors developed different strategies to reduce the volume of the power cables. The strip detector makes use of local DC-DC converters, while the pixel detector powers a group of up to fourteen detector modules in a serial chain. Further both sub-detectors foresee local monitoring chips, which reduce the number of sense lines. Due to the different powering schemes, the requirement concerning the Detector Control System (DCS) are also detector specific. This leads to different DCS Application Specific Integrated Circuits (ASIC). In case of the strip detector all DCS information is sent through data path, while in case of the pixel detector part of the data is sent through the normal data path, but some monitoring data is sent out through an independent path. Challenge for all ASICs is the radiation hardness.

Besides the sub-detector specific ASICs it was decided to search for common DCS tools where possible to reduce the development efforts and to simplify the operation and the long term maintenance. This is the subject of this article.

COOLING SYSTEM

Due to the highly power dissipation of the detector modules a high efficient cooling system is required. The cooling system will be based on evaporating CO_2 [1], using the 2-Phase Accumulator Controlled Loop cooling concept [3]. There will be a common cooling system for strip and pixel detector, which will be made up of several interchangeable cooling plants. Such a system does not require any active components inside the detector, just temperature sensing of the cooling pipes is needed. CO_2 was chosen for its significant mass savings inside the detector volume. Furthermore, CO_2 is radiation hard, cheap in use and environmentally friendly. The evaporation temperature is driven by the needs of the pixel detector. To ensure a thermal stability over the full operational lifetime, the coolant

† Susanne.kersten@cern.ch

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-THCPR07

ELECTRONICS FOR LCLS-II BEAM CONTAINMENT SYSTEM LOSS MONITORS*

R. Kadyrov†, C. Clarke, A. S. Fisher, M. Petree, C. K. Yee,
SLAC National Accelerator Laboratory, Menlo Park, USA

Abstract

LCLS-II is a new FEL facility which is under construction at SLAC National Accelerator Laboratory. Its super-conducting electron linac is able to produce up to 1.2 MW of beam power. In the event of electron beam loss, radiation can exceed allowed levels outside the thin shielding originally built for the lower power LCLS linac. Beam Containment System (BCS) loss monitors are employed to detect the beam loss and shut off the beam within 200 µs, limiting the radiation dose in occupied areas and minimizing damage to equipment associated with personal safety. Single-crystal (sCVD) diamond particle detectors are used as Point Beam Loss Monitors (PBLM) to detect losses locally. Long Beam Loss Monitors (LBLMs) measure losses throughout the beam path, from electron gun to beam dump, using optical fibers up to 200 m long. A PMT at the downstream end of each fiber detects light produced by Cherenkov radiation along the length of the fiber. A unified electronics design integrates the charge from the PMT or diamond detector, compares the loss with a predefined threshold and generates a fault if the limit is breached. Continuous self-checking is implemented for both types of sensors.

INTRODUCTION

Loss detection sensors previously used for LCLS are based on gas ionization chambers that have undesirable qualities at loss rates possible for LCLS-II [1]. Also a gas system deployed over the full accelerator complex would be unreliable and expensive. Therefore, new types of sensors were selected. These new sensors, and a more stringent shutoff time requirement, demanded a new system architecture and electronic design.

When a loss shower crosses the radiation-hard fused-silica optical fiber of an LBLM, Cherenkov light is emitted. A portion of the light is captured and transmitted through the fiber to the PMT installed at the downstream end of each fiber to measure the light [2].

In a PBLM, electron-hole pairs are generated within the diamond by ionising radiation. The pairs are collected with an HV bias applied to the faces of the sCVD sensor [2]. Negative biasing on the diamond sets an output signal polarity matching that of the PMT, and selecting proper charge integration parameters gives similar voltage levels. Therefore, a unified electronic front end was designed for these two sensors. This resulted in significant savings in engineering and design verification effort.

* SLAC is supported by the U.S. Department of Energy under contract DE-AC02-76SF00515
† rkadyrov@slac.stanford.edu

SYSTEM REQUIREMENTS

Requirements for functional safety are found in SLAC Radiation Safety Systems Technical Basis Document [3] and supplemented by functional and technical developed by LCLS-II project:

- The design shall be as fail-safe as possible.
- Built in automated test features are required that verify the interlock path of the electronics.
- A continuous non-invasive self-checking of the device should be implemented.
- System elements shall be under configuration control including labelling and locking sensors, electronics and connections where possible.
- Combined offsets, noise voltage and self-checking amplitude should be less than 10% of the lowest threshold expected. Input offset, bias and noise current shall be below 100 pA.
- Because loss detectors are very small current sources, the input impedance of the electronics is high; guard drives or similar elements are required in the design.
- The input integration time constant is 500 ms.
- Input protection from excess signals is required along with buffering of all external interfaces against any reverse signals.
- Electronics shall respond to a threshold breach within 10 µs. As discussed in [4], the beam shutoff time required for the entire system is 200 µs.
- The accelerator gallery is not temperature controlled and the racks are not cooled. The design shall withstand temperatures within the rack of 0 to 50°C and high humidity.

FUNCTIONAL DESIGN

The charge from the diamond sensor or PMT is integrated on the passive circuit shown in Figure 1. It is comprised of integration capacitor C_1 and discharging resistor R_1. R_2 is used for high frequency component termination. C_1 uses a film capacitor with an insulation resistance >10 GΩ and a 1-kV DC voltage rating. The capacitance is selected based on the expected charge and varies with the sensor type and location. For the PBLM, the capacitance of the long signal cable is taken into account in the calculation of the total integrating capacitance. R_1 is selected to keep the time constant of the capacitor discharge close to 500 ms. For the PBLM, R_1 is split into three series resistors of 4.99 MΩ each. Additionally, this area of PC board is made free of solder mask and potted with dielectric compound for greater stability and lower leakage.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-THCPR08

SPIRAL2 MACHINE PROTECTION SYSTEM STATUS REPORT

C. H. Patard*, Q. Tura†, C. Berthe, F. Bucaille, G. Duteil, P. Gillette, E. Lecorche,
G. Normand, R. J. F. Roze, GANIL, Caen, France

Abstract

The phase 1 of the SPIRAL2 (Second Generation of Ions Radioactive Production System in Line) facility, the extension project of the GANIL (Big Heavy Ions National Accelerator) laboratory in Caen, France, is to be commissioned. The accelerator, is composed of a normal conducting RFQ (Radio Frequency Quadrupole) and a superconducting Linac (Linear Accelerator). It is designed to accelerate high power deuteron and heavy ion beams up to 200kW. A Machine Protection System (MPS) has been implemented to protect the accelerator from thermal damages for this very large range of beam intensities. This paper presents the solutions chosen for this system, composed of three subsystems: one dedicated to thermal protection which requires a PLC and a fast electronic system, a second one dedicated to enlarged safety protection, and a third safety subsystem dedicated to fast vacuum valve protection. Both of those subsystems work associated with a global EPICS-based (Experimental Physics and Industrial Control System) control and HMI system, which gives the operation team global supervision of the accelerator and allows controlling sensor trigger thresholds, interlock system, beam initialization and power increase through the beam time structure. The MPS has been developed and is currently tested to be ready for the incoming SPIRAL2 commissioning.

INTRODUCTION

The SPIRAL2 facility has been constructed and is now in its commissioning phase.

The accelerator is based on an injector composed of a deuteron source and a heavy ions source, with the first step of pre-acceleration through an RFQ (see Fig. 1). The beam is then accelerated through a superconducting Linac, and then carried to one of the Linac Experiment Area (AEL), Neutron For Science (NFS) or S^3 (Super Spectrometer Separator). The Linac has been designed to accelerate particles ranging from deuterons to heavy ions at high power, up to 200kW for deuterons. Beam time structure may vary from continuous beam (88MHz) to chopped beams (1Hz to 1kHz) and single bunch mode beams.

A Machine Protection System (MPS), associated to an EPICS based control and HMI system, has been developed to protect the accelerator from thermal damages and control this very large range of beam power and various types of beam time structure.

MAIN FUNCTIONS

The main functions of the Spiral2 MPS are [1]:

- Protect the beam pipes and moving devices (slits, faraday cups, beam profile monitors, targets…) from thermal damages.
- Control the operating range of the facility.
- In case of beam losses, identify accelerator device activation.
- Ensure a reinforced safety protection of the beam dumps and target, which all have their own protection system addressing "beam off" requests to the MPS.
- Secure a high class protection of the fast vacuum valves.
- Providing an overview and an interface of the system's availability from the control rooms. The MPS transfers state-feedback of accelerator equipment and alarms to the EPICS interface, and receives instructions in return: handling insertion devices, acknowledgments, threshold management.

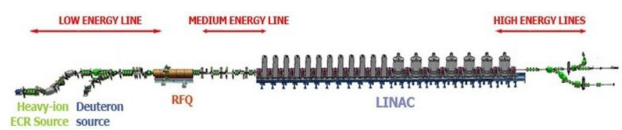


Figure 1: View of the SPIRAL2 accelerator.

Beam Cuts and Response Time

The MPS ensures those functions by receiving beam stop requests from diagnostics systems, or from ancillary systems (cryogenics, vacuum, radiofrequency…). Those beam cuts are of two types, fast and slow.

- The fast thermal protection system protects the accelerator chambers in case of 200kW instantaneous beam losses: it is designed for a response time of a few tens of µs. This fast cut is operated by activating chopper high voltage device in the low energy beam line. A second fast beam cut made is the safety classified fast cut. Its response time is determined by the time of insertion of the fast vacuum valves, and by the thermal resistance of the target that receives the beam. The beam must be cut fast enough to preserve the accelerator and the target in case of dysfunction of the thermal MPS control. This time is estimated to a few ms. This safety class fast beam cut is made by shutting down the RFQ because it is estimated simpler and safer than the activation of chopper high tension device.
- The slow cut is operated through the insertion of a beam stop or a faraday cup, some of them being safety classified to be used as a safety slow beam cut. Whenever a fast cut is operated, a slow beam cut is requested simultaneously, the fast beam cut devices maintaining their action until the permanent slow beam cut is achieved.

* charles-henri.patard@ganil.fr
† quentin.tura@ganil.fr

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-FRAPP01

THE LASER MEGAJOULE FACILITY: COMMAND CONTROL SYSTEM STATUS REPORT

H. Durandeau, R. Clot, P. Gontard, S. Tranquille-Marques, Y. Tranquille-Marques
Commissariat à l'énergie atomique et aux énergies alternatives (CEA), Le Barp, France

Abstract

The Laser MegaJoule (LMJ) is a 176-beam laser facility, located at the CEA CESTA Laboratory near Bordeaux (France). It is designed to deliver about 1.4 MJ of energy on a target, for high energy density physics experiments, including fusion experiments. The first bundle of 8-beams bundle was commissioned in October 2014. Today five bundles are in operation.

In this paper, we focus on two specific evolutions of the command control: the Target Chamber Diagnostic Module (TCDM) which allows the measurement of vacuum windows damages (an automatic sequence activates the TCDM that can be operated at night without any operator) and new Target Diagnostics integration. We also present a cybersecurity network analysis system based on Sentryo Probes and how we manage maintenance laptops in the facility.

LMJ FACILITY

The LMJ facility covers a total area of 40,000 m2 (300 m long x 150 m wide). It is divided into four laser bays, each one accommodating 5 to 7 bundles of 8 beams and a target bay holding the target chamber and diagnostics. The four laser bays are 128 m long, and situated in pairs on each side of the target chamber. The target bay is a cylinder of 60 m in diameter and 38 m in height. The target chamber is an aluminum sphere, 10 m in diameter, fitted with several hundred ports dedicated to laser beams injection and diagnostics introduction. A Supervisory and integrated computer control systems ensure the LMJ control system.

LMJ COMMAND CONTROL SYSTEM

LMJ Command Control System Functions

The main functions of the control system are shots execution and machine operations: power conditioning controls, laser settings, laser diagnostics, laser alignment, vacuum control, target alignment, target diagnostics [1]. The control system has also a lot of other major functions: personnel safety, shot data processing, maintenance management.

General Architecture

The LMJ control system has to manage over 500 000 control points, 150 000 alarms, and several gigabytes of data per shot, with a 2 years on line storage.

Software Architecture

All command control software developed for the supervisory layers uses a common framework based on the industrial PANORAMA E2 SCADA from Codra (Figure 1).

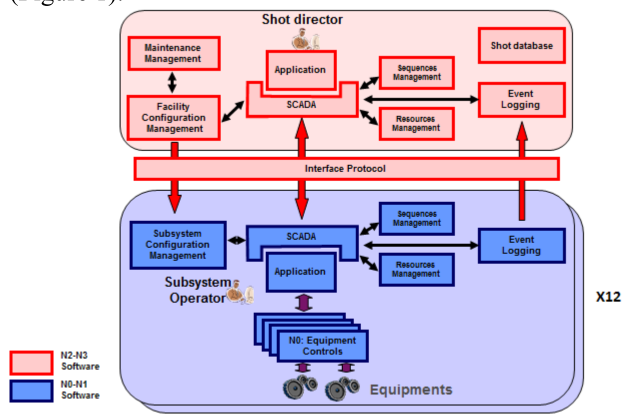


Figure 1: LMJ Software architecture.

The framework implements the data model described above as .net components inside the PANORAMA E2 SCADA and adds some common services to the standard features of PANORAMA E2:

- Resources management,
- Alarms management;
- Lifecycle states management;
- Sequencing [2];
- Configuration management;
- Event logging.

MAJORS EVOLUTION

Many version of command-control of each subsystem are provided using the integration policy of the maintenance phase [3]. This policy was based on a three steps process:

- Factory tests;
- Integration between subsystems on the dedicated platform (PFI);
- Functional integration on the LMJ facility.

We present two major evolutions depending on new equipment we have to manage.

Target Chamber Diagnostic Module (TCDM)

This new laser Diagnostic Module allows the measurement of vacuum windows damages (Figure 2:).

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-FRAPP02

PRELIMINARY ENGINEERING DESIGN OF THE CENTRAL INSTRUMENTATION AND CONTROL SYSTEMS FOR THE IFMIF-DONES PLANT

M. Cappelli†, ENEA, Frascati Research Center, Rome, Italy
A. Bagnasco, Ansaldo Nucleare, Genova, Italy
A. Ibarra, CIEMAT, Madrid, Spain

Abstract

IFMIF-DONES is the International Fusion Materials Irradiation Facility-DEMO Oriented NEutron Source, an accelerator-based neutron source where a high-energy deuterons beam is focused on a fast flowing liquid lithium jet to produce high-energy neutrons via stripping reactions with intensity and irradiation volume sufficient to generate material irradiation test data for design, licensing, construction and safe operation of the DEMO fusion reactor. This work presents the design of Central Instrumentation and Control Systems for the IFMIF-DONES plant and describes its most recent development. After a general overview of the current status of the design, the differences with respect to the corresponding system developed during the previous phases of the project will be highlighted. The paper describes the overall architecture (in terms of definitions, functions and requirements) and provides details about the identification of subsystems and equipment. A particular attention will be given to the I&C Networks connecting infrastructures.

INTRODUCTION

The irradiation environment in the future Fusion Power Plants is characterized by the presence of 14 MeV fusion neutrons in the first-wall region [1]. Understanding the degradation properties of materials and components throughout the operating life of the reactor is a main problem in allowing the design and plant licensing by the safety authorities. Presently available n sources are not adequate to reproduce fusion-like environment.

The availability of a fusion-relevant neutron source is therefore of the utmost importance for understanding the degradation of the fusion material under neutron bombardment during the life of the fusion reactor. In particular, this is a necessary requirement to a safe design for the DEMOnstration Fusion Reactor (DEMO), whose in-vessel materials will be exposed to neutron fluxes up to 5×10^{18} m^{-2} s-1 with a peak energy of 14.1 MeV. This could produce a displacement damage in excess of 10 dpa per year of operation with an He production rate of 10^{-13} appm/dpa [2].

Since more than 40 years, an international consensus has been reached on the development of an accelerator-based neutron sources exploiting D-Li stripping reactions (Li(d,nx)) as the optimal choice to provide the appropriate neutron flux and spectrum for reproducing the irradiation conditions of fusion reactors. In particular, since 1994 EU and Japan joined their effort in the so-called IFMIF-EVEDA (IFMIF-Engineering Validation and Engineering Design Activities) project, conducted in the framework of the bilateral EU-Japan Broader Approach Agreement [3, 4]. Recently, the EU supported the development of a Li(d,nx) neutron source called IFMIF-DONES (International Fusion Materials Irradiation Facility-DEMO Oriented NEutron Source) through the EUROfusion Work Package Early Neutron Source (WPENS) in collaboration with F4E Agency.

The objective of the IFMIF-DONES design [5] is to provide a high-energy neutron source with sufficient intensity and irradiation volume to generate material irradiation test data for the design, licensing, construction and safe operation of DEMO as defined in the EU Roadmap [6, 7, 8].

The general scheme of the current IFMIF-DONES Plant configuration is shown in Fig. 1, where five major groups of systems can be identified: Accelerator Systems; Lithium Systems; Test Systems; Plant Systems, and Central Instrumentation and Control Systems.

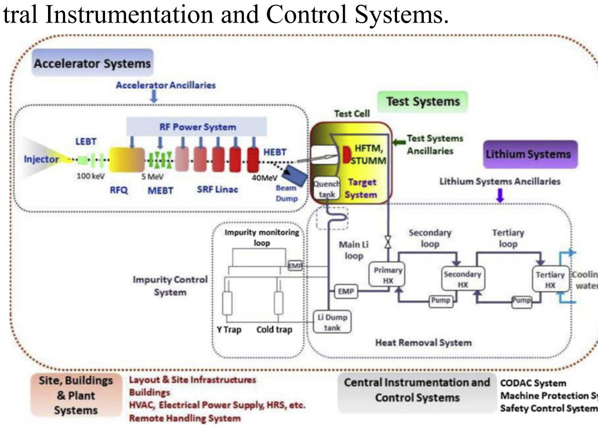


Figure 1: IFMIF-DONES Plant general scheme [9, 10].

The Accelerator Systems (AS) comprise all systems and components for the generation, acceleration and shape of the deuteron (D+) beam. Similarly to the IFMIF accelerator, the DONES accelerator consists of a sequence of acceleration and beam transport stages producing a 5 MW deuteron beam (125 mA, 40 MeV) with a rectangular cross section of [100, 200] mm × 50 mm, which impinges on the free surface liquid lithium target (25 mm thick, 260 mm wide), cross-flowing at 15 m/s on the opposite side. The stripping reactions generate neutrons that interact with the materials samples enclosed in

† mauro.cappelli@enea.it

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-FRAPP03

STATUS OF CSNS ACCELERATOR CONTROL SYSTEM

Y. L. Zhang†, C. P. Chu, W. C. Gao, F. Q. Guo, Y. C. He, D. P. Jin, M. T. Kang, G. Li, X. Wu, L. Wang, P. Zhu, Institute of High Energy Physics, Chinese Academy of Sciences, Beijing, China

Abstract

The China Spallation Neutron Source (CSNS) consists of an 80 MeV H^- LINAC, a 1.6 GeV Rapid Cycling Synchrotron, two beam transport lines and one tungsten target and three beam lines. The designed proton beam power is 100 kW in Phase-I. EPICS(Experimental Physics and Industrial Control System) is chosen as the software platform for the accelerator control system. The accelerator control system mainly consists of 21 sub-systems. VME64x based system with real-time embedded controllers is chosen for the timing system and fast protection system. PLCs and embedded industrial computers are used for the device level slow controls. CSS (Control System Studio) and RDB based techniques are adopted for high level applications. The overall control system was completed in 2018 and transitioned to routine operations in September of the same year. The design and implementation of the overall accele rator control system are introduced in this paper.

INTRODUCTION

CSNS accelerator consists an 80Mev H^- LINAC and a 1.6GeV Rapid Cycling Synchrotron (RCS) and two beam transport lines as shown in Fig. 1. The LINAC, which is mainly composed of a H^- ion source, a Radio Frequency Quadrupole (RFQ) and four Drift-Tube LINACs (DTL). LINAC and RCS both can be operated at 25Hz. At phase-I, the beam power is 100kW. The whole accelerator reserves the ability to upgrade the beam power to 500kW in the future [1]. In August 2017, the first proton beam was delivered to the target and the first neutron beam was generated after the first shot of proton beam. One year late, in August 2018, the whole facility was passed the national acceptance and open to the users.

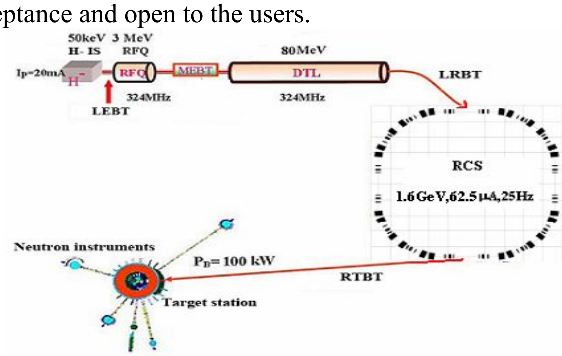


Figure 1: Schematic diagram of CSNS.

CSNS accelerators are designed to accelerate very high intensity proton beam and the uncontrolled beam loss may permanently damage or give a high radiation dose to the accelerator components along the beam line. Therefore, high reliability for control system is the basic requirement. The accelerator control system must be carefully designed so that we can avoid the unnecessary beam loss. Besides, the availability, scalability, maintainability and the budget was also carefully considered in the design and implementation stage.

CONTROL SYSTEM OVERVIEW

The accelerator control system can be divided into four parts: global systems, high level applications, low level and remote device controls and conventional facility integration. The global systems include timing system, machine protection system and control network. High level applications include services, console applications and databases. Low level and remote controls include vacuum control, front end and DTL control, power supply and RF remote control, stripper foil control and etc. Conventional facility integration includes UPS, isolated transformers, cables, racks and control rooms. The whole accelerator control system mainly consists of 21 sub-systems.

The main tasks of the accelerator control system are as the follow:

- To provide global control and communication platform.
- To provide global monitoring, data archiving and querying services.
- To provide global timing signals covering the whole facility, including accelerator, target and beam lines.
- To provide global machine protections.
- To provide data exchange interface to the conventional facility system, personnel protection system (PPS) and the experimental control system.
- Device layer or remote controls.

The design and implementation of accelerator control system is based on the EPICS framework [2]. EPICS toolkits provide standard tools with operator interface builder, data archiving, alarm handling and etc. CSNS control system benefits a lot from the existing EPICS tools and modules, such as Control System Studio(CSS) [3], Archiver Appliance [4], StreamDevice [5] and so on.

In order to minimize the hardware category of the control system, brand and type of the adopted hardware were normalized. For example, all slow control sub-systems must use the YOKOGAWA PLC and Moxa industrial computer. Most of the devices and equipment we chose which can be directly connected to EPICS by using the existing drivers in EPICS community. We only need to develop EPICS drivers for a few devices by ourselves. By this way, we saved a lot of time in the design and implementation stage.

The CSNS accelerator control system complies with the standard large distributed system architecture. The overall control system can be divided into three layers, the presentation layer, the middle layer and the device layer as shown in Fig. 2. IOCs, services and applications distributed in

† zhangyl@ihep.ac.cn

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-FRAPP05

REVIEW OF COMMISSIONING AND FIRST USER OPERATION IN RESPECT TO HIGH LEVEL CONTROLS AT THE EUROPEAN XFEL

R. Kammering[†], B. Beutner, W. Decking, L. Froehlich, O. Hensler, T. Limberg, S.M. Meykopff, M. Scholz, J. Wilgen, Deutsches Elektronen-Synchrotron (DESY), Hamburg, Germany

Abstract

In September 2017 the European XFEL entered user operation after several years of construction and commissioning. To provide a fast and flexible commissioning of the various sections of the machine, the high-level control software was essential already from the beginning. While progressing in commissioning and increasing operation parameter space, the enormous complexity of the European XFEL put hard requirements on the control and operation concepts. Having now the full baseline parameters reached, this paper will review the concepts and architecture of the control system in respect to effectiveness, reliability and ease of operation.

Basic software concepts and design ideas but also general operation concepts, interoperability between various systems can now be reviewed in respect to the overall facility performance.

SOME HISTORY – BIRTH OF THE HIGH-LEVEL CONTROLS GROUP

Already in summer 2014 a group of people from various machine related sub-groups came together to form the so-called High-Level Controls (HLC) group. This concept arose from the lessons learned at the Free Electron Laser in Hamburg (FLASH) in Germany and the commissioning of the Linac Coherent Light Source (LCLS) in the US. Within both of these projects it showed up that a simple bottom-up approach for implementing higher level software often falls short. The complexity of such modern machines today is that large that a step-by-step commissioning of individual sub-components is often either not possible or at least not economically efficient. These days the software plays a key role when it comes to integrating sub-systems and establishing full interoperability between the various components. The proper functionality of sub-systems can mostly only be established if the whole software envelope is in place and active.

To overcome these now known problems it has been decided to form an expert group concentrating on these topics exclusively. Even though at this point in time only the injector complex, consisting of a photo cathode gun and two accelerating modules existed, the group already addressed topics still years ahead. This allowed for grasping possible very complex and therefore time-consuming tasks and addressing these already at this early stage.

One such example is the proof of concept for the, that days only envisioned and later on implemented, central data acquisition system (DAQ) (see [1] and [2]). Using some of the already existing server nodes a simulated environment has been set up to mimic the estimated data rates such a DAQ system would need to cope with. Even if the aim of this setup has been to show that such an architecture can cope with the data rates, soon it showed up that such a system can serve for many testing and development purposes. Such a virtual accelerator not only allows to test and debug software components but also can serve as a testbed for graphical user interfaces. E.g. have here new concepts and ideas for the visualization of the complex beam distribution and bunch train dimension been developed. The system has later on been called the Virtual European XFEL and is still being developed further [3].

Even thus within the first years the commissioning of the injector complex had highest priority, the scope had always been to establish software components, interfaces and the overall architecture with the full facility in mind.

Fundamental decisions like supported languages, operating systems and control system interoperability have been a major topic at this stage. A lesson learned – here from the European XFEL commissioning – is to fix these decisions prior to starting the work, but also allow for late changes. Thus has the decision to support the python language been taken at a late stage, but the strong requests and lively discussions finally resulted in this outcome, which nowadays no one would question.

GETTING THINGS DONE

With the proven capability to cope with the expected data rates the already at FLASH running centralized DAQ architecture has been accepted. This paved the road for some of the foreseen central services we envisioned within the HLC group. Table 1 shows some of the servers and their purpose.

Table 1: Some of the DAQ Attached DOOCS (see [4] or [5]) Middle Layer Servers – these Servers offer most of the Core Functionality for e.g. the Display Layer

Server	Purpose
Charge ML	Transmission information
Charge Calc.	Long term charge integrator
Orbit ML	Various orbit information
Beam Energy Measurement	Beam based energy calc.
Energy Profile	Higher level energy calc.
Longitudinal FB	Slow energy, chirp, … FB
BLM ML	Higher level BLM calc.

With this architecture and concept at the hand the implementation and configuration of these servers could to great extend be ‘copied’ from FLASH to the European XFEL ecosystem. This is especially true since much software at FLASH has already been designed to be later on ported to the European XFEL.

† raimund.kammering@desy.de

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-FRAPP06

STATUS OF THE CONTROL SYSTEM FOR THE ENERGY RECOVERY LINAC bERLinPro AT HZB*

Thomas Birke†, Pablo Echevarria, Dan Eichel, Roland Fleischhauer, Ji-Gwang Hwang, Guido Klemz, Roland Müller, Christoph Schröder, Ervis Suljoti, Andriy Ushakov
Helmholtz-Zentrum Berlin für Materialien und Energie GmbH, Berlin, Germany
Karim Laihem, Rheinisch-Westfälische Technische Hochschule, Aachen, Germany

Abstract

bERLinPro is an energy recovery linac (ERL) demonstrator project built at HZB. It features CW SRF technology for the low emittance, high brightness gun, the booster module and the recovery linac. Construction and civil engineering are mostly completed. Synchronized with device integration, the EPICS based control system is being set-up for testing, commissioning and finally operation. In the warm part of the accelerator, technology that is already operational at BESSY and MLS (e.g. CAN-bus and PLC/OPC UA) is used. New implementations like the machine protection system (MPS) and novel major subsystems (e.g. Low Level RF (LLRF), photo cathode laser) need to be integrated. The first RF transmitter has been tested and commissioned. For commissioning and operation of the facility the standard set of EPICS tools form the back-bone. A set of generic Python applications already developed at BESSY/MLS will be adapted to the specifics of bERLinPro. Scope and current project status are described in this paper.

INTRODUCTION

The goal of bERLinPro is the production of high current, high brightness, low emittance CW beams and to demonstrate energy recovery at unprecedented parameters [1]. The three stage acceleration consists of an SRF photo electron gun, an SRF booster linac with an extraction energy of 6.5 MeV and an SRF main linac module equipped with three 7-cell HOM damped cavities. All magnets and the vacuum system of the low energy injector and dump line are installed. Commissioning of the diagnostic line and the low energy part of the machine, i.e. gun / booster / linac replacement straight / dump line, the *banana* (see eponymous shape in Fig. 1) is planed for 2020,

bERLinPro is designed to show energy recovery for high current (100 mA) beams. The damping of higher order modes in the SRF linac is demanding and led to a new design of the HOM damping waveguids [2]. Availability of a proper linac module is critical.

The MESA project (Johannes Gutenberg Universität, Mainz, Germany) is planned for 1 % of the bERLinPro current, with a possible upgrade to 10 mA. For HOM-damping, the same technology as used at ELBE (HZDR) and XFEL (DESY) is in operation. Intermediate installation of the MESA linac module into bERLinPro allows to proceed towards recirculation with beam at 32 MeV and some mA at bERLinPro [3].

* Work funded by BMBF and Land Berlin
† thomas.birke@helmholtz-berlin.de

OPERATIONAL MODES

Unlike cERL and cBETA, bERLinPro features numerous different use cases. These comprise the photo-electron-source only, straight diagnostic beamline, *banana* path and recirculation with and without energy recovery (see Fig. 1). Available modes also differ in beam power, bunch charge, acceleration voltage and bunch train pattern as well as methods to increase beam current. All of these and the individual operation states of accelerating units, booster and linac modules will have immediate impact and consequences on a challenging set of soft- and hardware machine protection systems and set-ups.

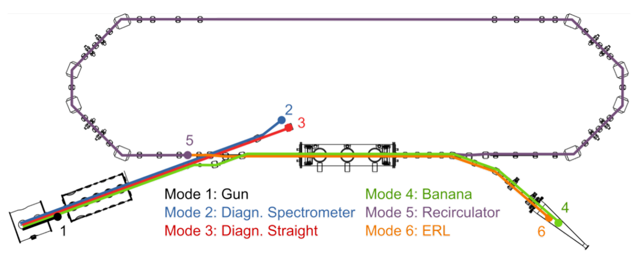


Figure 1: Basic bERLinPro layout and planned operation modes.

DEVICE INTEGRATION

The source part, consisting of an SRF photo electron gun, has already been set up in GunLab [4], where precision control of the laser guide system and the timing is presently in the works. With the beginning of installation in the bERLinPro bunker, integration of major functional blocks (e.g. laser) will be realized by remote control of 3rd party subsystems.

In March 2018 the first vacuum components for the *banana* path have been delivered, pumps and sensors are made available and are logged in the already running archiver [5] as they are installed. Similarly RF power conditioning and cryo system surveillance of cold compressors, warm vacuum pumps and the module feed boxes are well known and progress smoothly.

The various sub components of the booster and linac cryo modules are about to be addressed. At this point competing requirements for BESSY VSR [6] generate synergies, but also challenging working conditions.