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-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].

† odd.oyvind.andreassen@cern.ch

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].

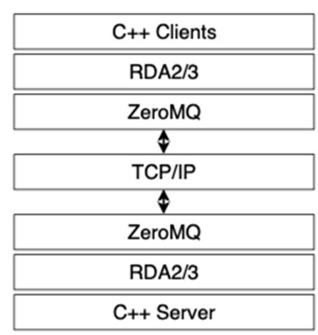

Figure 1: CMW architecture.

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).

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA026

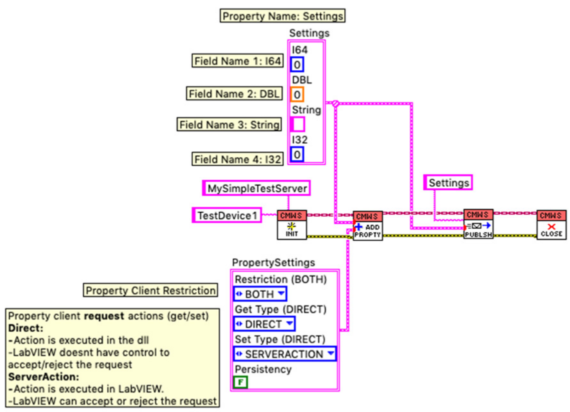

Figure 2: RADE CMW Server Wrapper example in LabVIEW.

The architecture (Fig. 3) is the same as for the CMW stack with an added C++ client wrapper for both the Server and Client libraries. The library has been ported to both CentOS, OpenEmbedded, Pharlap and Windows. And is compatible with LabVIEW version 2010 and onwards.

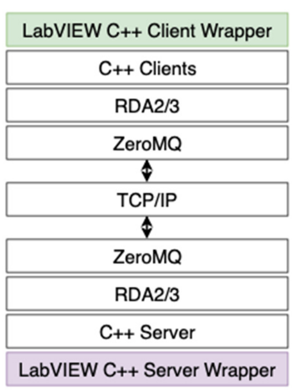

Figure 3: LabVIEW CMW Server and Client wrapper architecture.

This integration makes it possible to implement the server stack with the standard drag and drop development in LabVIEW and the developers does not have to leave the IDE to implement the server.

TIMING

CERN's General Machine Timing (GMT) system guarantees that all accelerators in the complex act as a networked particle production facility by coordinating and synchronizing their activities. The system is based on multi-drop RS-485 networks piloted by CTG cards. On the receiving side, different modules react to messages on the network by arming counters, producing pulses on their front panel connectors or interrupts to synchronize the different front-end computers to events happening in the accelerators. The new generation of receiving modules makes use of FPGA technology and a special PLL to recover a stable 40 MHz reference from the 500 kb/s messages [8].

GMT Timing in RADE

Due to the extensive documentation and memory-based hardware design of the CERN timing library and NI-VISA drivers, the integration of the GMT timing receiver in the RADE framework was straight forward. By knowing the hardware addresses and interrupts, we could map the CTRp card and make use of it on PCI or PXI based systems (Fig. 4).

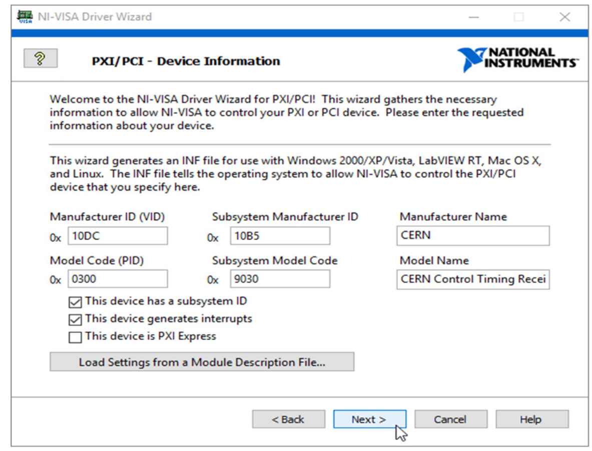

Figure 4: NI-VISA configuration manager.

As for the RADE LabVIEW interface, we modeled it after the CMW server interface with the same typical "Open, Use, Close" approach used in most LabVIEW hardware interfaces (Fig. 5).

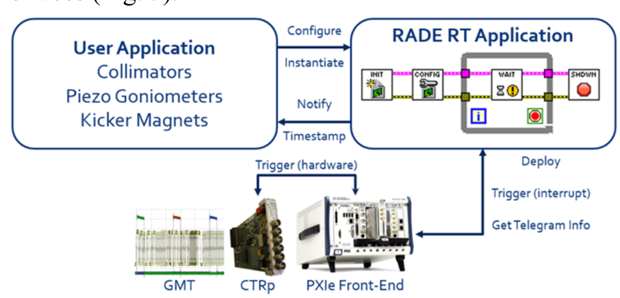

Figure 5: LabVIEW CTRp RADE Driver interface.

DISKLESS BOOT

At CERN, most of the critical front-end computers are booted from memory using the PXE. This avoids relying on the file system integrity on physical storage and makes it easier to distribute and update the control software.

"In computing, the PXE specification describes a standardized client-server environment that boots a software assembly, retrieved from a network, on PXE-enabled clients. On the client side it requires only a PXE-capable network interface controller (NIC), and uses a small set of industry-standard network protocols such as DHCP and TFTP" [9].

The PXE boot capabilities have been brought to PXI systems thanks to a special bios feature specified by the EN-SMM group at CERN and provided by NI. This feature has been extensively tested by the collimator team and has been in place for the LHC collimator LLCS project since

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA026

2007. In addition, the collimator team has developed a configuration manager which makes it possible to effortlessly add and remove PXI systems in to CERN's elaborate network domain [10].

SYSTEMS IN THE FIELD

As a proof of concept to test the RADE framework real time extensions in the field, 3 target projects with different requirements where chosen:

- AWAKE, to test the RADE CMW Server streaming and logging capabilities with precise timing.
- CrystalPiezoGoniometer, to test the RADE CMW Server for non-critical mechatronics control applications during Machine Development (MD) which requires PXE booting.
- TwinEBIS, to test the RADE CMW Server on a distributed mixed platform with the aim of future commercialization and therefore the need to run both inside and outside the CERN infrastructure.

AWAKE

The Advanced Proton Driven Plasma Wakefield Acceleration Experiment (AWAKE) is an accelerator R&D project. It is a proof-of-concept experiment investigating the use of plasma wakefields driven by a proton bunch to accelerate charged particles, to achieve this a powerful pulsed laser is used [11].

The AWAKE acquisition system uses multiple cameras to monitor the specific laser. The system is composed of a PXI crate, up to 12 cameras and a Network Attached Storage (NAS). When an external trigger arrives, the PXI acquires raw images from the cameras which, store them on the NAS and publish the data to the CERN infrastructure via the RADE based CMW server (Fig. 6).

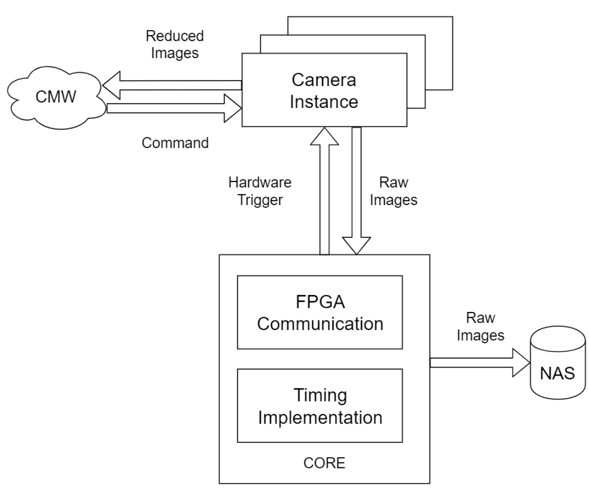

Figure 6: AWAKE communication.

In the AWAKE project CMW is used to both control the server and publish data from the cameras. Each camera feed has its own dedicated CMW device.

There are many benefits to this architecture. The main one is that each camera is an abstract device; the User can send and acquire data directly from the camera without worrying about the underlying structure of the PXI. Instead of having a message broker on the server that relays the messages to the cameras on request, the CMW communication makes it possible to publish data directly from the cameras. As an added value, any user on the CERN network can connect with many different interfaces such as C++, MATLAB (via JAPC), Python and Java. The AWAKE scientist simply needs to subscribe to the image property from any camera and use their dedicated expert interface to analyze the laser data.

Another benefit of the CMW publications is that all the data published can automatically be saved in the CERN logging database through CALS without any additional development [2].

CrystalPiezoGoniometer

Collimators are used in accelerators to strip the beam of its halo (particles that stray too far from the beam center), thus keeping the beam clean, reducing the chance of quenches and keeping the equipment safe. This is currently done by controlling jaws to intercept the halo. The Crystal Piezo Goniometer project has tested a new type of collimator which uses bent crystals to channel beam halo particles to an absorber, thus reducing quench risk and providing a less intrusive collimation process (Fig. 7) [12].

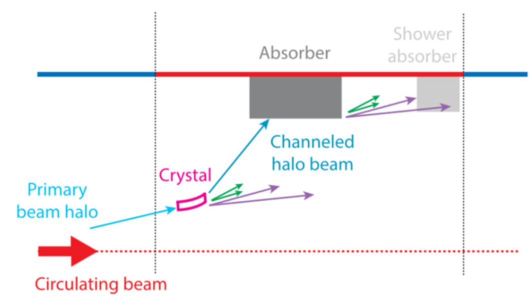

Figure 7: Crystal collimation.

The systems are based on the COTS National Instruments (NI) PXI platform, making use of real-time and FPGA programming to reach the required time constraints. In order to integrate these devices into the beamline of the LHC and its control systems, several challenges needed to be met.

The CPG PXI system is booted using PXE technology, however the file size of the images for the project were too big for the standard TFTP protocol (FAT12 limited to ~29MB). With the help of NI, we tested and validated new images able to TFTP larger files (up to ~96MB), thus overcoming a hurdle that prevented the CPG from being properly integrated into the larger infrastructure [10, 13].

Once booted, the systems expose their configuration to the operators through the RADE CMW Server library. This development has allowed the software on the PXI to directly host the necessary services, to publish data and listen for configuration change. In addition, it eliminates the need for additional hardware [4, 5].

TwinEBIS

The name TwinEBIS is derived from two parts. EBIS is an acronym for Electron Beam Ion Source, which means it

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA026

creates ions with an electron beam. Twin comes from the fact it is a copy or a twin of an already existing EBIS

The goal of TwinEBIS is to research a more affordable and technologically advanced carbon cancer treatment accelerator, reducing the physical size and cost of the accelerator. Ultimately the accelerator should fit in existing hospital x-ray rooms, making groundbreaking cancer therapy available for all [14].

The control system is based on COTS National Instruments (NI) PXI and cRIO platform with the aim of reusing the lab prototype commercially in the future. This means that all the hardware and software used has to function both inside and outside the CERN environment without too many dependencies on the existing infrastructure, and by using the new RDA3 based middleware, this would be possible [15].

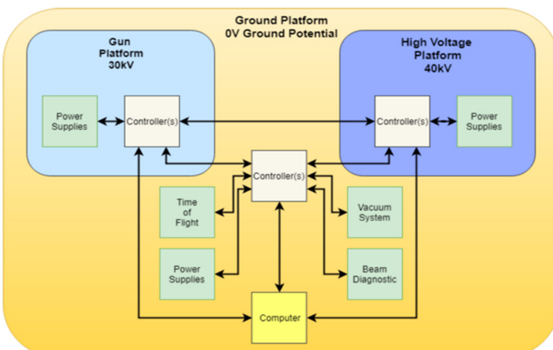

Figure 8: TwinEBIS Control system architecture.

The control system operates on three different platforms with a large potential difference. The voltage is 40kV for the High Voltage Platform, 30kV for the Gun platform and 0V for the ground platform (Fig. 8). Each controller, located on the different platforms, intercommunicate via CMW, and an effort was made to port the RADE CMW libraries to both the Pharlap based PXI and cRIO platform. A benefit of this has been that standard ethernet based fiberoptic links could be used to bind and synchronize the different controllers and at the same time be controlled both with localized expert LabVIEW based applications and more general multi-purpose CERN control tools [2, 4, 5, 15].

CONCLUSION

The proof of concept implementation of the CMW server and CERN GMT timing has been successfully implemented on both the PXI and cRIO platforms for all the test systems. The implementation has proven to be robust, scalable and performant, and has significantly reduced the development time for NI based systems requiring full system integration in the CERN accelerator domain. The use of the RADE framework has made it possible to implement and integrate all the code without the need of any development outside the LabVIEW environment. In addition, the BIOS successfully in operation for the LHC collimators since 2007 has been modified to allow the download of bigger software images.

FUTURE PLANS

In the future we are planning on adding support for automatic deployment and registration of the CMW based devices in the new CERN Control System Database (CCDB) interface and adapt the CMW server to the new FESA DevServer environment. In addition, we are also considering to add LabVIEW RT support for the new CERN White Rabbit (WR) timing system. A proof of concept has been implemented for the NI-cRIO platform and a 1588 HA WR compatible receiver is being tested in the lab.

REFERENCES

[1] B. Frammery, "The LHC Control System", in *Proc. ICALEPCS'05*, Geneva, Switzerland, Oct. 2005, paper I1_001.

[2] J. Cuperus, R. Billen, and M. Lelaizant, "The Configuration Database for the CERN Accelerator Control System", in *Proc. ICALEPCS'03*, Gyeongju, Korea, Oct. 2003, paper WE114, pp. 309-311.

[3] M. Karuza *et al.*, "The CAST Experiment: Status Report", in *Proc. 13th Patras Workshop on Axions, WIMPs and WISPs – Axion-WIMP 2017*, Thessaloniki, Greece, May 2017, pp. 39-42. doi:0.3204/DESY-PROC-2017-02/karuzamarin

[4] O. O. Andreassen, D. Kudryavtsev, A. Raimondo, A. Rijllart, S. Shaipov, and R. Sorokoletov, "The LabVIEW RADE Framework Distributed Architecture", in *Proc. ICALEPCS'11*, Grenoble, France, Oct. 2011, paper WEMAU003, pp. 658-661.

[5] A. Dworak *et. al.*, "The new CERN Controls Middleware" *J. Phys.: Conf. Ser.*, vol. 396, p. 012017. doi:10.1088/1742-6596/396/1/012017

[6] K. Foraz, "LS1 General Planning and Strategy for the LHC and LHC Injectors," Chamonix Workshop, Chamonix, France, 2012.

[7] K. Kostro, J. Andersson, F. Di Maio, S. Jensen, and N. Trofimov, "The Controls Middleware (CMW) at CERN - Status and Usage", in *Proc. ICALEPCS'03*, Gyeongju, Korea, Oct. 2003, paper WE201, pp. 318-321.

[8] J. Serrano, P. Álvarez, D. Domínguez, and J. Lewis, "Nanosecond Level UTC Timing Generation and Stamping in CERN's LHC", in *Proc. ICALEPCS'03*, Gyeongju, Korea, Oct. 2003, paper MP533, pp. 119-121.

[9] Wikipedia, "Preboot eXecution Environment", https://en.wikipedia.org/wiki/Preboot_Execution_Environment

[10] A. Masi, M. Donze, and R. Losito, "Reliability review of the LHC Collimators Low Level Control System," in *Proc. 12th IFAC Symposium on Large Scale Systems: Theory and Applications*, vol. 43, no. 8, 2010, pp. 200-205. doi:10.3182/20100712-3-FR-2020.00034

[11] E.Gschwendtner *et. al.*, "AWAKE, The Advanced Proton Driven Plasma Wakefield Acceleration Experiment at CERN", *Nuclear Instruments and Methods in Physics*, vol. 829, Sep. 2016, pp. 76-82. doi: 10.1016/j.nima.2016.02.026

[12] R. Rossi *et al.*, "Crystal Collimation During the LHC Energy Ramp", CERN-ACC-NOTE-2018-0053.

ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-WEPHA026

[13] Wikipedia, "Trivial File Transfer Protocol", https://en.wikipedia.org/wiki/Trivial_File_Transfer_Protocol

[14] M. Breitenfeldt *et al.,* "The TwinEBIS setup: Machine description", *Nucl. Instrum. Meth. A*, vol. 856, Jun. 2017, pp. 139-146.

[15] J. Lauener and W. Sliwinski, "How to Design & Implement a Modern Communication Middleware Based on ZeroMQ", in *Proc. ICALEPCS'17*, Barcelona, Spain, Oct. 2017, pp. 45-51. doi:10.18429/JACoW-ICALEPCS2017-MOBPL05