

# 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

Content from this work may be used under the terms of the CC BY 3.0 licence (© 2019). Any distribution of this work must maintain attribution to the author(s), title of the work, publisher, and DOI.

our newer systems. Planned obsolescence of older hardware systems may remove the need to ever develop IOCs for those systems.

In the specific case of FETS this means that we may choose not to develop IOCs for the older CSS system, while ensuring the CPS system is fully migrated to EPICS.

## IMPLEMENTATION OF THE VSYS-TEM/EPICS BRIDGE

Our existing Vsystem software runs on OpenVMS servers. OpenVMS is not one of the EPICS supported operating systems, and porting to a new OS presents an unknown amount of work. Fortunately, we have already deployed a system written in Python which uses the MQTT messaging protocol [11] and Vsystem's events architecture to transmit all changes in channel values to a MQTT broker. This same system allows the values of channels to be set. This allowed us to develop the EPICS bridge on the supported Linux OS, rather than OpenVMS.

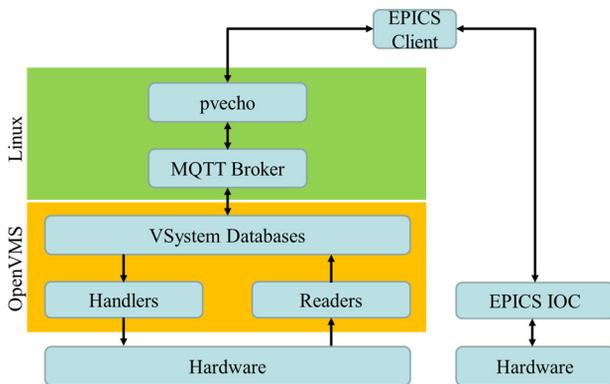


Figure 1: Overview of Vsystem/EPICS bridge pvecho (left), and ordinary IOC (right).

The bridge was written in C++ with EPICS 7, using only the standard libraries and Boost [12] as well as a headers-only MQTT library [13]. Additional attributes describing the channels, such as units, limits, and alarm values were drawn from a CouchDB database [14] developed for another project. The bridge was written to make use of pvDatabaseCPP [15], and thus uses the newer EPICS 4 style process variables. Since we have no legacy IOCs, and there is an easy bridge to EPICS 3 style channel access variables, this approach allows us the maximum design freedom.

The operation of the bridge, called pvecho, is shown in Fig. 1.

## CONCLUSION

Evaluation of a migration of control systems software to EPICS is underway at the ISIS accelerators, beginning with a test deployment on our FETS hardware. Learning lessons from past migrations we are being careful to perform this initial deployment with careful consideration of phased deployment over a number of years, and of decoupling backend and UI upgrades.

## REFERENCES

- [1] J.W.G. Thomason, "The ISIS Spallation Neutron and Muon Source—The first thirty-three years," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, Vol. 917, pp. 61-67, February 2019.
- [2] B. Mannix and T. Gray, "Vista Controls Vsystem at the ISIS Pulsed Neutron Facility", presented at the 10th Int. Conf. on Accelerator and Large Experimental Physics Control Systems (ICALEPCS'07), Knoxville, TN, USA, Oct. 2007, unpublished.
- [3] "Select Intel Itanium Processors and Intel Scalable Memory Buffer, PCN 116733-00, Product Discontinuance, End of Life" (PDF). Intel. January 30, 2019. <http://qdms.intel.com/dm/i.aspx/F65EEA26-13FB-4580-972B-46B75E0AB322/PCN116733-00.pdf>
- [4] Vista Control Systems, Inc. <http://www.vista-control.com>
- [5] EPICS Control System, <https://epics-controls.org/>
- [6] A. J. Kozubal *et al.*, "Run-time Environment and Application Tools for the Ground Test Accelerator Control System," in *Proc. Int. Conf. on Accelerator and Large Experimental Physics Control Systems*, Vancouver, British Columbia, Canada, 1989, pp. 288-291.
- [7] A. Letchford *et al.*, "Status of the RAL Front End Test Stand," in *Proc. 4th Int. Particle Accelerator Conf. (IPAC'13)*, Shanghai, China, May 2013, THPWO086, pp. 3725-3728.
- [8] 1000-1987 - IEEE Standard for an 8-Bit Backplane Interface: STEbus.
- [9] CompactPCI Overview, <https://www.picmg.org/openstandards/compactpci/>
- [10] R. A. Washington, "Migrating to Tiny Core Linux in a Control System", presented at the 17th Int. Conf. on Accelerator and Large Experimental Control Systems (ICALEPCS'19), New York, NY, USA, Oct. 2019, paper WECPL05, this conference.
- [11] MQTT Version 3.1.1 Plus Errata 01, OASIS Standard Incorporating Approved Errata 01, <http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/mqtt-v3.1.1.html>
- [12] Boost, free peer-reviewed portable C++ source libraries, [https://www.boost.org/users/history/version\\_1\\_70\\_0.html](https://www.boost.org/users/history/version_1_70_0.html)
- [13] MQTT client/server for C++14 based on Boost.Asio, [https://github.com/redboltz/mqtt\\_cpp](https://github.com/redboltz/mqtt_cpp)
- [14] Apache CouchDB, document-oriented NoSQL database, <http://couchdb.apache.org/>
- [15] pvDatabaseCPP, <https://github.com/epics-base/pvDatabaseCPP>