EVOLUTION OF CONTROL SYSTEM STANDARDS ON THE DIAMOND SYNCHROTRON LIGHT SOURCE

M. T. Heron, T. Cobb, R. Mercado, N. Rees, I. Uzun, K. Wilkinson, Diamond Light Source, Oxfordshire, UK.

Abstract

Control system standards for the Diamond synchrotron light source were initially developed in 2003. They were largely based on Linux, EPICS and VME and were applied fairly consistently across the three accelerators and first twenty photon beamlines. With funding for further photon beamlines in 2011 the opportunity was taken to redefine the standards to be largely based on Linux, EPICS, PC's and Ethernet. The developments associated with this will be presented, together with solutions being developed for requirements that fall outside the standards.

INTRODUCTION

Diamond Light Source [1] is a third-generation 3 GeV synchrotron light source based on a 24-cell double-bend achromatic lattice of 561m circumference. The photon output is optimised for high brightness from undulators and high flux from multi-pole wigglers. The accelerators and first phase of seven photon beamline were constructed from 2002 to 2007; a second phase of fifteen photon beamlines from 2006 to 2012; and a third phase of ten photon beamlines was approved in 2011 with construction due to finish in 2017-8.

Established control systems standard have been applied consistently across the accelerators and phases 1 and 2 of photon beamlines. This gives inherent advantages in the sharing of knowledge, reuse of software components and minimises hardware variants. Since the start of the operation of Diamond, in 2007, software of these operational control systems has been systematically up lifted to enable a relatively recent set of software components and tools to be used.

With approval of the third phase of photon beamlines, and after nearly ten years of deploying the original control systems standards, it was considered timely to define new standards for control system hardware and software.

RECAP ON THE ORGINAL DIAMOND CONTROL SYSTEM

The Diamond Control System [2] uses the EPICS toolkit [3] and provides a high degree of integration of the underlying technical systems. On the accelerators this includes all power converters, most diagnostics, vacuum systems, the machine protection system, insertion devices, RF amplifiers, girder alignment, front-ends and personnel safety system. For each of the photon beamlines, experiment stations and instruments this includes all optical elements (motion), machine protection system, vacuums, diagnostics, personnel safety system and detectors.

In the original Diamond control system, equipment is largely interfaced through a range of generic VME I/O based on VME IP carriers, IP modules, transition cards and plant interface modules, see Fig. 1. For motion control, the OMS VME58 is used for straightforward applications on the accelerators, whereas for the photon beamlines for synchronous control of multiple axes, the Delta Tau PMAC controller is largely used. Programmable Logic Controllers (PLCs) from Omron are used for interlocking and control, e.g. for vacuum valves, whilst for high-end process control applications, such as the Linac, the RF cavities and the cryo-plant controls, the Siemens S7 series of PLCs are used.

Client side tools use the standard EPICS tools for display panels (EDM), archiving (Channel Archiver), alarm management (ALH) and restoring system state (BURT and Save\Restore). In addition, where client side processing is required, tools have been developed based on Python and QT. Matlab, Matlab Middle Layer [4] and Accelerator Toolbox [4] and associated tools are used for physics based optimisation of the accelerators. For station scientists and visiting users Diamond has standardised on a combination of EPICS and GDA [5] for experiment control and data acquisition. EPICS provides the low-level interface to the hardware and user interface functionality for engineering type operations, whilst GDA provides the science-based interface for the station scientists and visiting users. The exceptions to this are where commercial instrumentation forms the basis of the experimental station. Client side tools are largely unchanged in the developments being reported here.

CORE SOFTWARE AND OPERATING SYSTEM EVOLUTION

Core Software Components

The initial machine control system was predominantly deployed with EPICS 3.13.9, (3.14 for the Libera BPMs and soft IOCs) as this was the stable version of EPICS in 2003 and so provided a stable platform for in-house work and for external suppliers delivering turn-key systems. The beamline control systems which started in development in 2005 adopted EPICS version 3.14.8.2. During the period 2008 to 2011, the accelerator and beamline control systems were upgraded to 3.14.11. The difference in processes between 3.13 and 3.14 and environment meant that code changes were required for the accelerator control systems and so required considerable testing. The majority of control systems are

ISBN 978-3-95450-139-7

now at EPICS version 3.14.11 with the next upgrade underway to 3.14.12.3. The policy is to try and keep all systems on a common version of EPICS base, core components and modules.

Operating System and Environment

Having standardised on Linux for development the initial version used from 2003 was Red Hat version 9, which was used purely for development. Red Hat Enterprise Linux 3 was used from 2004 for initial operations followed by Enterprise Linux 4 in 2005, Enterprise Linux 5 in 2008 and Enterprise Linux 6, 64bit version, in 2011. Coincident with each operating system uplift, a new version of EPICS Base, Extensions and external modules have been taken along with other dependent software components. While Enterprise Linux provides excellent out-of-the-box support and package management, some packages lag behind what is needed, for example Python, for which a current stable version is deployed separately to the native OS version.

While most development and operational client systems are Linux-based, IOCs predominantly run under Linux or VxWorks. Inevitably there are a few Windows systems, which come about where only Windows drivers are available for a given piece of hardware. In order to ensure quality of deployment, all applications are built on a build-and-deployment-server, thereby ensuring consistency of the tool chain for the build of all operational systems.

EPICS control systems are built around EPICS core IOC components, support modules – non installation specific extensions for example Stream Device – and installation specific extensions; for example, particular device or driver support. While a particular control system is only built for the processor and operating system it is targeted for, the core EPICS components and all support modules are now (largely) built for each combination of processor, operating system and version of EPICS. This is at the expense of build time and disk space. However, while a supported module may be developed and tested under one operating system and hardware combination, it is unlikely to have been tested to the same level on every combination of hardware. While there have been attempts to establish a continuous build and integration environment for EPICS modules at Diamomd, this has not been successful to date.

HARDWARE EVOLUTION

The new control systems standard is based on EPICS on 1U x86 PCs running and embedded version of Red Hat Enterprise Linux with real-time extensions. These IOCs are located within equipment they control, and so are not regarded as “soft IOCs”. All instrumentation is then connected to the IOC by network connections. The new architecture is shown in Fig. 1.

Motion Control

For motion control, a standard based on the Delta Tau Geobrick LV Ethernet-based motor controller is used [6]. This provides 8 axes of motion control and comes complete with amplifiers in a 4U rack-mount box. The existing EPICS motor record software, already in use with older VME hardware, was modified to be compatible with this controller. This was realised by adding an ASYN interpose layer, which provides support for the Delta Tau Ethernet TCP/IP packet structure, and so avoids making changes to the existing PMAC motor controller ASYN driver.

Vacuum Instrumentation and other Serial Devices

Vacuum instrumentation (Gauges and Pump Controllers) and other serial devices are interfaced through RS232, RS422 or RS485 serial connections. These are connected to a terminal server located in the instrumentation rack and the terminal server via Ethernet to the IOC. On the IOC most serial devices are handled by the EPICS Stream Device module communicating to the serial interface over virtual serial connections to the terminal server.

Cameras on Ethernet

Diagnostic applications use a range of GigE cameras from AVT (formerly Prosilica). An EPICS IOC using areaDetector [7] is used to control, process and store images from up to ten cameras, together with an ffmpegServer [8] for visualisation. AreaDetector is a modular system of EPICS drivers and plug-ins that can be "rewired" at run time, allowing a flexible image processing chain to be set up. Plug-ins for controlling the camera, providing statistics on the images that are produced, filtering them and writing them to disk are included with areaDetector. FfmpegServer is a Diamond-produced plug-in that compresses a stream of images to mjpg and serves them over http.

Programmable Logic Controllers

Interlocking and protection of equipment is realised with Omron CJ1 PLCs. These are interfaced to the IOC using Ethernet and the FINS [9] protocol. The PLC optionally uses remote I/O modules called SmartSlice [10] which are located in the beamline optics and experiment hutches. The SmartSlice remote I/O comprises a Communications Unit and a number of I/O Units providing digital I/O, analogue I/O, temperature, counter and positioning interfaces. The I/O Units communicate with the host PLC over a private Ethernet connection running the PROFINET protocol. PROFINET [11] provides flexibility so that it is simple to configure additional I/O modules.

ADCs, DACs and Digital IO

To interface ADCs, DACs and digital I/O directly to the IOCs, a range of I/O modules from Beckhoff

ISBN 978-3-95450-139-7

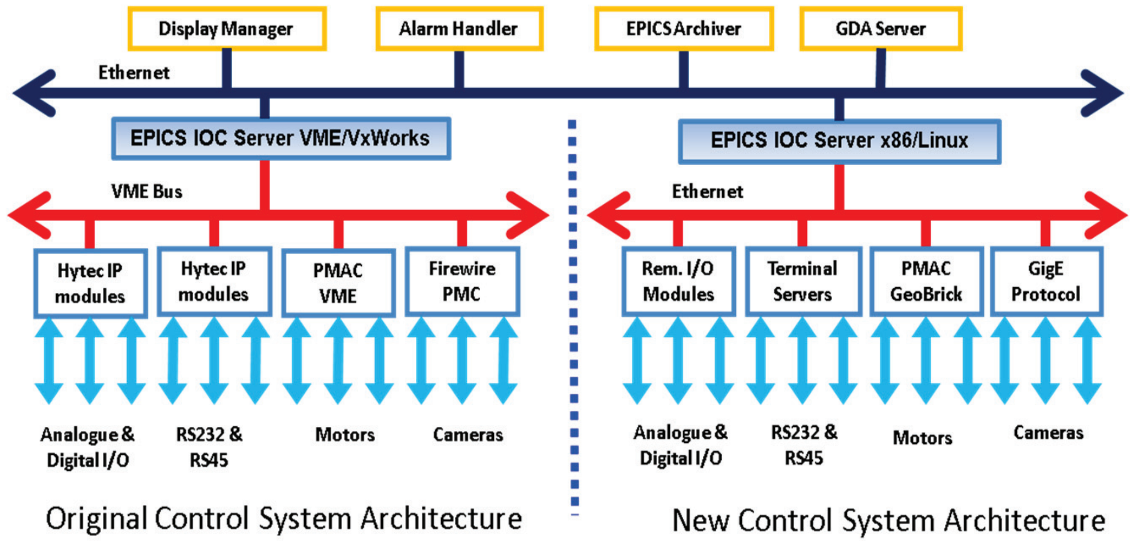

Figure 1: Original and New Control System Architectures.

Automation has been selected. These use EtherCAT [12], an industrial Ethernet-based fieldbus system. This I/O is used for all non-interlocking type applications, and provides lower latency from the plant to the IOC than the PLC solution and so enables accurate time stamping of the signal. It further minimises the number of I/O points in the PLC-based interlocking system and so minimises the need for changes to the PLC code which necessitate revalidation of the interlock logic.

The EtherCAT protocol provides low-latency data transfer from the I/O modules into the host computer. It operates on the principle of a master that communicates with slaves using EtherCAT telegrams that are passed around each node and back to the master. The EtherCAT master uses standard Ethernet controller hardware and a software implementation of the EtherCAT functionality, whilst the slaves use a custom slave controller.

The custom interface implements a Fieldbus Memory Management Unit (FMMU), which allows the mapping of logical addresses in the telegram to physical ones within the slave. This processing occurs on the fly as one slave passes the telegram through to the next slave, introducing a delay of a few nanoseconds. Slaves also automatically close a communication ring when the outgoing Ethernet link (downstream section) is not connected, by returning the telegram to the master back through the chain of slaves.

Although the protocol can operate with other Ethernet-based services and protocols on the same physical network, the Diamond Remote I/O solution adopts strict segregation of the EtherCAT bus.

Timing Signals

The Diamond timing system is applied across the accelerators and beamlines. It is based on the event system from Micro Research Finland [13] with a central event generator and event receivers embedded in the IOCs to decode events as physicals signals, EPICS events or interrupts. The timing system also provides time stamps for EPICS record processing. To support this functionality in the new architecture, a PCIe version of the Event Receiver module has been developed. This makes time stamp information and soft events available in the PC-based IOC and brings out the decoded signals on a 1U interface panel.

BEYOND THE STANDARD SOLUTION

While the new control system standard meets greater than 90% of the control system requirements there are some which fall outside the capability.

There is some functionality, for example, scalers and a time frame generator for experiment control for which there remain only VME solutions. There are also detectors and diagnostics instruments where the performance requirement exceeds what is capable through the standard solution. In some cases, for example, the Excalibur detector [14] which generates continuous data at a rate of 600MB/sec, then the control and readout is realised as a bespoke EPICS solution under Linux with direct writing of the data to a parallel file system. For some commercial detectors, for example the PCO Edge [15] camera, for which there are only Windows drivers, then the IOCs for control and readout are realised on Windows.

Where the acquisition rate or latency requirements exceed what is realisable with the standard solution and in particular what is practical in software, then a solution has been realised in a FPGA; examples include RF phase detector and photon beam position monitor acquisition. These have used various Xilinx parts with a UDP stack implemented in VHDL to communicate with a soft IOC. With the advent of Xilinx Zynq [16] system-on-chip components the available CPU power is such that it is now practical that Linux and EPICS can run directly on the FPGA and so this will form the basis of future FPGA based acquisition solutions.

ISBN 978-3-95450-139-7

CONCLUSION

VME hardware has provided an excellent basis for the Diamond control systems and as a standard has had a longevity that many products and standards can only aspire to. As an open standard with multi-vendor support it has provided good hardware interoperability, which is a key aspect to building a distributed control system and subsequently maintaining it. However, control systems interface requirements have largely moved on from analogue and digital signals with increased availability of "intelligent instruments" for which communication based interfaces are increasingly the norm. Despite this, scientific manufacturers still have a long way to go compared to the IT world in terms of plug and play integration. Performance (processing capability, response time and analogue signal resolution) of PLCs has also evolved considerably in the past decade and costs have fallen, such that PLCs can now interface signals that previously went directly to the IOC.

These gains have been at the loss of physical interoperability, loss of cross vendors operability in the case of PLC solutions and possibly reduced life expectancy of hardware (commercial grade components versus those of VME), loss of mechanical standards and functionally and finally often at the loss of the ability to accurately timestamp an acquisition or transition of a signal.

Nevertheless the current generations of control systems in using more commercial hardware and more mainstream operating systems provides great benefits in terms of functionality and cost savings.

REFERENCES

[1] R. P. Walker, "Commissioning and Status of the Diamond Storage Ring", APAC 2007, Indore, India
[2] M. T. Heron, "Implementation, Commissioning and Current Status of the Diamond Light Source Control System", ICALEPCS 2007, Knoxville, USA
[3] http://www.aps.anl.gov/epics
[4] G. Portmann et al, "An Accelerator Control Middle Layer Using Matlab", PAC 2005, Knoxville, USA
[5] P. Gibbons, "GDA and EPICS working in unison for science driven data acquisition and control at Diamond Light Source", ICALEPCS 2011, Grenoble, France
[6] N.P.Rees et.al., "Development of Photon Beamline and Motion Control Software at Diamond Light Source", ICALEPCS 2007, Knoxville, 2007.
[7] http://cars9.uchicago.edu/software/epics/areaDetector.html
[8] http://controls.diamond.ac.uk/downloads/support/ffmpegServer/
[9] OMRON FINS Commands Reference manual (W227-E1-2)
[10] OMRON Slice I/O Units Operation Manual (W455-E1-06)
[11] OMRON GRT1-PNT PROFINET I/O Communication Unit Operation Manual (W13E-EN-01)
[12] ETG. EtherCAT - the Ethernet fieldbus. http://www.ethercat.org/pdf/ethercat_e.pdf .
[13] Y. Chernousko, P. Hamadyk, M. T. Heron, "Review of the Diamond Light Source Timing System", RUPAC 2010, Protvino, Russia, 2010
[14] J. A. Thompson, et al, "Controlling the Excalibur Detector", ICALEPCS 2011, Grenoble, France
[15] http://www.pco.de/categories/scmos-cameras/pcoedge
[16] http://www.xilinx.com/products/silicon-devices/soc/zynq-7000/index.htm

ISBN 978-3-95450-139-7