IBEX - THE NEW EPICS BASED INSTRUMENT CONTROL SYSTEM AT THE ISIS PULSED NEUTRON AND MUON SOURCE

M. J. Clarke, F. A. Akeroyd, K. Baker, G. Howells, D. Keymer, K. J. Knowles,
C. Moreton-Smith, D. E. Oram, ISIS, STFC Rutherford Appleton Laboratory, Oxon, UK
M. Bell, I.A. Bush, R.F. Nelson, K. Ward, K. Woods, Tessella, Abingdon, Oxon, UK

Abstract

Instrument control at ISIS [1] is in the process of migrating from a mainly locally developed system to an EPICS [2] based system. The new control system, called IBEX, was initially used during commissioning of a new instrument prior to a long maintenance shutdown. This first usage has provided valuable feedback and significant progress has been made on enhancing the system during the facility maintenance period in preparation for the move onto production use. Areas that will be of particular interest to scientists in the future will be linking feedback from live data analysis with instrument control and also providing a simple and powerful scripting interface for facility users. In this paper we will cover the architecture and design of the new control system, our choices of technologies, how the system has evolved following initial use, and our plans for moving forward.

BACKGROUND

The ISIS pulsed neutron and muon source has been providing world-class science since the mid-1980s and currently has over thirty beamline instruments producing world-leading research. The software and hardware used to control these beamline instruments plays a significant part in the productivity of ISIS.

The current instrument control system, known as the SECI system, consists of LabVIEW [3] drivers for equipment control alongside in-house developed software for scripting, data collection and experiment management, etc. This control system has performed well for many years; however, the construction of new beamline instruments and the refurbishment of older instruments has led to significantly more complex experiments becoming possible and, as a result, are pushing beyond the boundaries for which the current control system was designed. With this in mind, and with the prospect of even more complex instruments being built in the future, it was decided to create a new control system based on EPICS.

HARDWARE ARCHITECTURE

The IBEX control system runs on a Windows 7 x64 Virtual Machine (VM) which is hosted on a rack-mounted Windows 2012 server. Each beamline instrument has its own independent server. Windows was chosen for a number of reasons, such as:

- The controls team have significant experience and expertise with running control systems on Windows
- ISIS has an IT support group that can assist with installation and maintenance
- Easier interfacing to existing site infrastructure (e.g. data archiving, system backup/replication)
- Can use existing LabVIEW drivers more easily
- Ability to do a phased migration, running either in parallel or a mixed IBEX/SECI system

The beamline instrument server and equipment are usually placed on the general ISIS network; however, network sensitive equipment can be placed on a private network. This private network is implemented using a Virtual Local Area Network (VLAN) which can be routed back to the main computer room. As the virtual machines are replicated to the computer room it is possible to rapidly restore and run an instrument from there in the event of beamline server failure.

Most beamline and sample environment equipment in use on ISIS instruments are controlled via serial (RS232) communication. For ease of access and flexibility MOXA [4] NPort serial device servers are used to expose the serial devices to the network.

SOFTWARE ARCHITECTURE

A system based on EPICS lends itself naturally to a client-server style architecture and this is how IBEX has been implemented. Figure 1 shows the basic flow of information relating to the experiment; the dashed line indicates the separation between the client and server side components. Where possible the system uses EPICS Channel Access (CA) for communication between the client and server side.

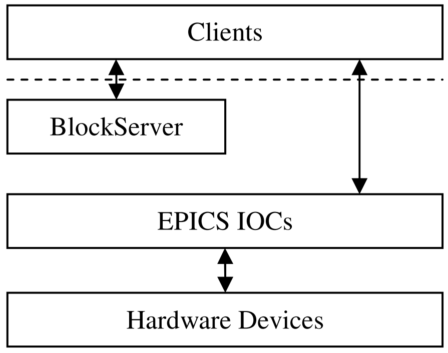


Figure 1: The flow of experiment information for a beamline instrument.

EPICS Input/Output Controllers (IOCs) are the standard way of communicating with the devices used on

ISBN 978-3-95450-148-9

the instruments. The procServ [5] package is used to host the IOCs as it provides telnet access to the IOC console. The conserver program [6] is used to provide remote access to these consoles for diagnostic and support purposes and the procServControl [7] package adds support for starting and stopping the IOC remotely via Channel Access.

The BlockServer is the brain of the control system; it coordinates the client and the server side to meet the needs of the type of science being performed on the beamline instrument being used. The BlockServer is responsible for managing the setting up of the system for the particular experiment being performed. The information for each required collection of devices is stored as XML files – these XML files make up what is known in ISIS as a configuration. The configurations contain the following information:

- Which IOCs to start and key IOC macro values
- Which Blocks to create
- Metadata relating to the type of experiment
- Information for configuring the IBEX graphical user interface (such as Block display grouping)

At the most basic level a Block is a scientist friendly alias for an EPICS Process Variable (PV). For example, rather than working with a complicated PV like IN:LARMOR:EUROTHERM:TEMP1 an alias can be created called SampleTemp. This simple science specific name is much more intuitive for a scientist. Blocks are usually created for key experiment parameters and the values are logged for inclusion in the NeXus data file [8] created at the end of the experiment. Block PVs are implemented using aliasing on a PV Gateway [9] which in this instance is running on the loopback interface of the server.

All important PVs on an instrument are automatically logged via the Control System Studio [10] archiver application running on the server. As Blocks are experiment based they are logged in a second instance of the archiver to allow easier adjustment and tailoring of archiving performance. The archiver pushes the log information into a MySQL database [11] on the server. The historical log can be used for monitoring and diagnosing problems and the Blocks log provides information for inclusion in the NeXus data file.

Storing information for configuring the IBEX graphical user interface (GUI) on the server side means that the GUI is instrument independent. This is very convenient for the scientists and support staff as it allows quick switching between different instruments through one GUI. Having a common GUI across all instruments also makes support and maintenance easier.

Although it is not shown on Figure 1 there is also a second PV Gateway on the server which is used to control access to the PVs. Using the CA security built into the Gateway it is possible to restrict access to the PVs on the server; this can be based on user name and/or the machine being connected from. By default, access to the IBEX server is restricted to being read-only, but read-write access is allowed from certain machines.

One of the weaknesses of the SECI system was that there was only one GUI client instance available and that was on the server. This meant that the system could not be viewed easily by multiple people at the same time. IBEX has been developed to allow multiple instances of the client to be run at the same time from any PC on the ISIS network. By default these clients will be read-only but the intention is to develop a baton handling system to allow clients to request to have sole read-write access for an appropriate amount of time.

Also not shown on Figure 1 is the messaging system which allows messages from the IOCs and BlockServer to be passed to the IBEX GUI. The messaging system is part of Control System Studio and is based on the Apache ActiveMQ implementation of the Java Messaging System (JMS) [12]. A new log server process replaces the usual IOC log program and routes IOC messages into this JMS system. A historical record of the messages is stored on the server inside the MySQL database and these historical messages can be searched via the IBEX GUI.

One of the challenges for a control system at a user facility like ISIS is incorporating new bespoke equipment quickly; for example, a user from a university may bring a prototype piece of sample environment equipment to use on an instrument. To handle these use cases there are different technologies which have been developed at ISIS for handling LabVIEW and other simple serial devices.

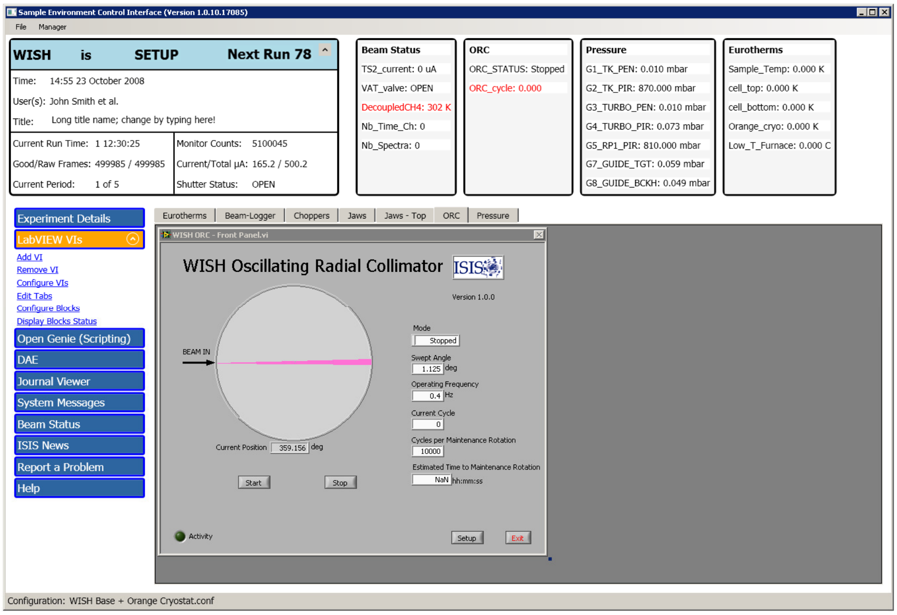
As the SECI system was based on LabVIEW it was desirable to be able to quickly enable existing drivers into the EPICS system. For this, a driver based on Asyn [13] was created, called lvDCOM [14], which communicates with LabVIEW via DCOM [15]. The creation of the lvDCOM driver is mostly automated and because the LabVIEW driver requires no modification it is quick and simple to create an IOC which wraps the LabVIEW driver. It is reasonably common for bespoke equipment to have a LabVIEW driver, thus lvDCOM can be used in those cases. For more complex situations it is possible to use the NetShrVar driver [16] to access National Instrument network shared variables [17] instead.

SDTest is a simple generic serial solution based on the EPICS StreamDevice [18]. With knowledge of the command strings to send to the device and how to interpret the corresponding response a StreamDevice driver can be quickly created as part of a configuration.

THE IBEX CLIENT

For IBEX it was decided to create a GUI based on Control System Studio. Control System Studio was chosen as it was the most feature complete of the modern EPICS GUI implementations. Control System Studio itself is based on the Eclipse Rich Client Platform (RCP) framework [19].

The overall design of the IBEX GUI was based on the GUI of the SECI system (see Figure 2) as the ISIS scientists and users liked this design.



WISH is SETUP Next Run 78

Time: 14:55 23 October 2008

User(s): John Smith et al.

Title: Long title name; change by typing here!

Current Run Time: 1 12:30:25

Good/Raw Frames: 499985 / 499985

Current Period: 1 of 5

Monitor Counts: 5100045

Current/Total μA: 165.2 / 500.2

Shutter Status: OPEN

Beam Status

TS2_current: 0 uA

VAT_valve: OPEN

DecoupledCH4: 302 K

Nb_Time_Ch: 0

Nb_Spectra: 0

ORC

ORC_STATUS: Stopped

ORC_cycle: 0.000

Pressure

G1_TK_PEN: 0.010 mbar

G2_TK_PIR: 870.000 mbar

G3_TURBO_PEN: 0.010 mbar

G4_TURBO_PIR: 0.073 mbar

G5_RP1_PIR: 810.000 mbar

G7_GUIDE_TGT: 0.059 mbar

G8_GUIDE_BCKH: 0.049 mbar

Eurotherms

Sample_Temp: 0.000 K

cell_top: 0.000 K

cell_bottom: 0.000 K

Orange_cryo: 0.000 K

Low_T_Furnace: 0.000 C

WISH Oscillating Radial Collimator

Version 1.0.0

Mode: Stopped

Swept Angle: 1.125 deg

Operating Frequency: 0.4 Hz

Current Cycle: 0

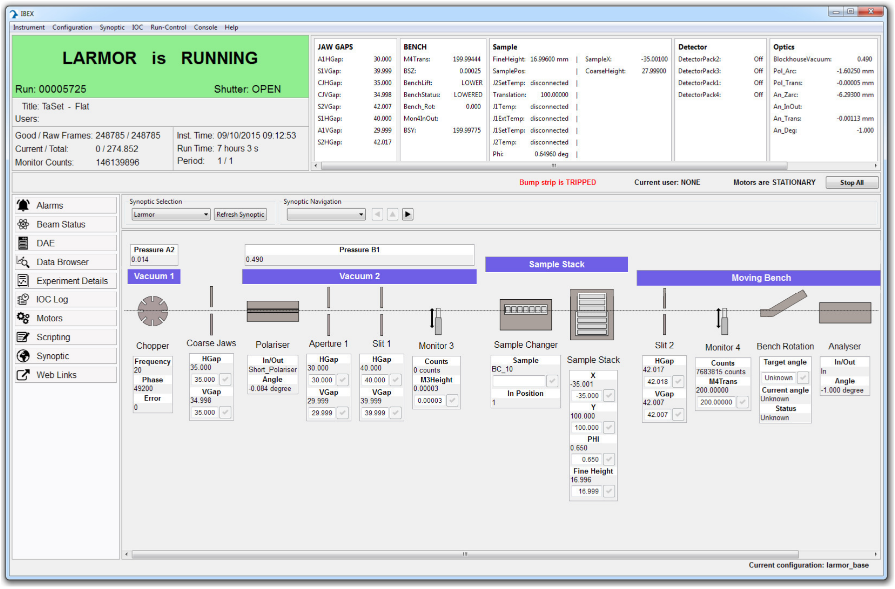
Cycles per Maintenance Rotation: 10000

Estimated Time to Maintenance Rotation: NaN hh:mm:ss

Current Position: 359.156 deg

Configuration: WISH Base + Orange Cryostat.conf

Figure 2: The GUI for the SECI system.



LARMOR is RUNNING

Run: 00005725 Shutter: OPEN

Title: TaSet - Flat

Users:

Good / Raw Frames: 248785 / 248785

Current / Total: 0 / 274.852

Monitor Counts: 146139896

Inst. Time: 09/10/2015 09:12:53

Run Time: 7 hours 3 s

Period: 1 / 1

JAW GAPS

A1HGap: 30.000

S1VGap: 39.999

CJHGap: 35.000

CJVGap: 34.998

S2VGap: 42.007

S1HGap: 40.000

A1VGap: 29.999

S2HGap: 42.017

BENCH

M4Trans: 199.99444

BSZ: 0.00025

BenchLift: LOWER

BenchStatus: LOWERED

Bench_Rot: 0.000

Mon4InOut:

BSY: 199.99775

Sample

FineHeight: 16.99600 mm | SampleX: -35.00100

SamplePos: | CoarseHeight: 27.99900

J2SetTemp: disconnected |

Translation: 100.00000 |

J1Temp: disconnected |

J1ExtTemp: disconnected |

J1SetTemp: disconnected |

J2Temp: disconnected |

Phi: 0.64960 deg |

Detector

DetectorPack2: Off

DetectorPack3: Off

DetectorPack1: Off

DetectorPack4: Off

Optics

BlockhouseVacuum: 0.490

Pol_Arc: -1.60250 mm

Pol_Trans: -0.00005 mm

An_Zarc: -6.29300 mm

An_InOut:

An_Trans: -0.00113 mm

An_Deg: -1.000

Bump strip is TRIPPED   Current user: NONE   Motors are STATIONARY

Pressure A2: 0.014

Pressure B1: 0.490

Chopper: Frequency 20; Phase 49200; Error 0

Coarse Jaws: HGap 35.000 [35.000]; VGap 34.998 [35.000]

Polariser: In/Out Short_Polariser; Angle -0.084 degree

Aperture 1: HGap 30.000 [30.000]; VGap 29.999 [29.999]

Slit 1: HGap 40.000 [40.000]; VGap 39.999 [39.999]

Monitor 3: Counts 0 counts; M3Height 0.00003 [0.00003]

Sample Changer: Sample BC_10; In Position 1

Sample Stack: X -35.001 [-35.000]; Y 100.000 [100.000]; PHI 0.650 [0.650]; Fine Height 16.996 [16.999]

Slit 2: HGap 42.017 [42.018]; VGap 42.007 [42.007]

Monitor 4: Counts 7683815 counts; M4Trans 200.00000 [200.00000]

Bench Rotation: Target angle [Unknown]; Current angle Unknown; Status Unknown

Analyser: In/Out In; Angle -1.000 degree

Current configuration: larmor_base

Figure 3: The new IBEX GUI showing an instrument synoptic.

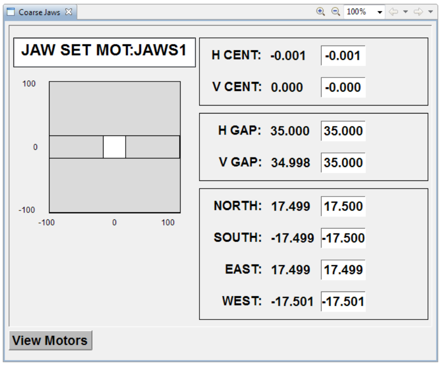
The key features of the design that needed to be reproduced were:

- The data collection dashboard in the top-left corner
- The Block values displayed in groups in the top-right corner
- The navigation bar on the left-hand side

To achieve this it was necessary to create a screen layout very different from a typical Control System Studio screen. ISIS specific views were created and some of the GUI features that are provided by Control System Studio needed to be either hidden or disabled. The current IBEX GUI is shown in Figure 3. New key features required for the IBEX GUI were a synoptic, a data browser and an embedded Python console for scripting.

The synoptic (as shown in Figure 3) provides an overview of the status of the instrument in terms of the layout of the instrument and key values. By selecting an item on the synoptic it will open a window showing more information about that particular item. For this Control System Studio's BOY screens are used, an example of which is shown in Figure 4. An instrument may have multiple synoptics, either for different experiments or for showing various areas of the beamline separately. The layout information for the synoptics is served and managed by the BlockServer which is one of the ways the IBEX GUI is kept instrument independent. The GUI also has a graphical synoptic editor which allows the scientists to create new synoptics or modify existing ones.

The data browser is a plotting tool developed as part of Control System Studio; it is designed to show historical PV values as a function of time. In the IBEX GUI it is used to plot the history of Blocks as well as standard PV values. The historical data is retrieved from the MySQL database mentioned previously.



JAW SET MOT:JAWS1

H CENT:	-0.001	-0.001
V CENT:	0.000	-0.000
H GAP:	35.000	35.000
V GAP:	34.998	35.000
NORTH:	17.499	17.500
SOUTH:	-17.499	-17.500
EAST:	17.499	17.499
WEST:	-17.501	-17.501

Figure 4: An example BOY screen.

The SECI system used an in-house scripting language called Open GENIE [20] for creating scripts to automate a series of experiments. For the IBEX system Python was chosen as the replacement for Open GENIE because it is easy to learn and many scientists will already be familiar with it. PyDev [21] is a Python integrated development environment and because it is an Eclipse RCP application it was simple to incorporate the PyDev Python command line directly into the IBEX GUI. To enable Python to communicate with the IBEX control system an in-house library was created called genie_python which uses CaChannel [22] for sending and receiving information over Channel Access and provides scientists with a set of useful control commands.

STATUS UPDATE

The IBEX system has been successfully used to commission the LARMOR small angle scattering instrument and is now being used to perform user experiments. The scientists using a prototype IBEX system to commission LARMOR provided invaluable testing time and feedback for the development of the system. Now IBEX is being used for performing user experiments more feedback is being received which will be used to direct further development.

One of the goals for IBEX was for it to be easy to incorporate bespoke equipment quickly. With this in mind lvDCOM, NetShrVar and SDTest were developed. Users on LARMOR have already successfully used SDTest to enable their own specialist equipment to be used within IBEX.

The intention was to develop a system where an instrument could be observed from multiple instances of the GUI at the same time and this has been achieved. Being able to monitor an instrument without disturbing the user has been a very useful feature for both the controls team and the scientists. Overall the IBEX GUI has been well received by the scientists - maintaining the familiar look of the SECI GUI whilst adding powerful new features like the synoptic has proved popular.

The general flexibility of having a system based on EPICS compared to LabVIEW has allowed a more future resilient control system to be developed. Being part of wider collaborations, such as EPICS and Control System Studio, have reduced the amount of development required to create the new control system.

THE FUTURE

The Mantid data analysis framework [23] is widely used at ISIS for analysing data after an experiment but there is a desire amongst the scientists for “intelligent data collection.” An example of this might be stopping data collection when the signal to noise ratio of the data reaches a certain level. As Mantid is Python-aware it is possible to either call genie_python from Mantid or vice versa – this could be a way of providing intelligent data collection. Initial investigations into enabling Mantid to communicate with the control system have proved promising. This is an area which will be explored further.

IBEX is currently being used to commission the new IMAT instrument which presents a new set of challenges. IMAT is an engineering beamline and, thus, is very different to LARMOR; however, this is the kind of challenge IBEX was developed to handle.

After IMAT there will be another new instrument called ZOOM which will also be commissioned using IBEX. Over time the existing instruments running the SECI system will be migrated to IBEX.

REFERENCES

[1] http://www.isis.stfc.ac.uk
[2] http://www.aps.anl.gov/epics/
[3] http://www.ni.com/labview/
[4] http://www.moxa.com
[5] http://sourceforge.net/projects/procserv/
[6] http://www.conserver.com
[7] http://controls.diamond.ac.uk/downloads/support/
[8] http://www.nexusformat.org
[9] http://www.aps.anl.gov/epics/extensions/gateway/
[10] http://controlsystemstudio.org
[11] https://www.mysql.com
[12] http://activemq.apache.org
[13] http://www.aps.anl.gov/epics/modules/soft/asyn/
[14] http://epics.isis.stfc.ac.uk/wiki/LVDCOM
[15] https://msdn.microsoft.com
[16] http://epics.isis.stfc.ac.uk/wiki/NETSHRVAR
[17] http://www.ni.com/white-paper/5484/en/
[18] http://epics.web.psi.ch/software/streamdevice/
[19] https://eclipse.org
[20] http://www.opengenie.org
[21] http://www.pydev.org
[22] https://cachannel.readthedocs.org
[23] http://www.mantidproject.org

ISBN 978-3-95450-148-9