

A REVIEW OF OPTIONS FOR THE DIAMOND CONTROL SYSTEM

M.T. Heron, B.G. Martlew[#],

CLRC Daresbury Laboratory, Warrington, Cheshire, WA4 4AD, United Kingdom

Abstract

DIAMOND is a new UK national light source project to replace the SRS at Daresbury. Many alternative control system designs and implementations exist that might be used as a model for the machine and beamline controls on DIAMOND. This paper surveys the recent developments in light source control systems and attempts to identify some of the main advantages and disadvantages of each alternative. Also, an attempt is made to assess the relative resource implications of each of the major options within the context of a modern light source project.

1 INTRODUCTION

DIAMOND is a design for a 3rd generation, 3Gev synchrotron light source based on a 20-cell DBA lattice of about 400m circumference. It uses a full energy booster synchrotron and linac for injection. The spectral output is optimised for high brightness up to 20keV from undulators and high flux up to 50keV from multipole wigglers. The project status is described in [1].

This paper presents the current view of the options for the DIAMOND control system. Many of the issues involved in selecting a model system will revolve around estimates of the required resources, the ease of development and ease of future expansion. The technical differences between competing options, although important, are unlikely to be the deciding factor. After all, the systems described here are all successful control systems and are all capable of meeting most, if not all, of the technical requirements presented in section 2 below.

2 CONTROL SYSTEM REQUIREMENTS

The final design model for the DIAMOND control system has not yet been selected, but several key requirements have already been identified that will help in deciding the most appropriate route to take.

The main requirements are:

- The ability to remotely operate and monitor accelerator systems at a rate sufficient to provide smooth, 'analogue-style' control. This implies update rates of 10 Hz or greater.

- Parameter archive and set facilities for post-mortem analysis of machine operations and for restoring the accelerator to a pre-determined mode.
- An easy to use, versatile Graphical User Interface (GUI). Ideally, this should be a GUI that is already familiar to the scientists and engineers working on the project to reduce the learning time.
- Tight integration with standard software packages. All control system data should be accessible directly from major desktop software packages such as spreadsheets, plotting packages, mathematical analysis tools, etc.
- Access to the control system via the web. As much information as possible should be presented via the web, including live control system displays, documentation, database access and, with appropriate security, limited control facilities.
- Use of a modular I/O system for plant interfacing.
- Use of standards. To reduce the resources required for the design and build phase of the project, as well as increasing future flexibility, maximum use should be made of readily available hardware and software solutions.
- Use of a standard solution for plant protection. A clearly defined strategy for handling plant status and interlocks needs to be included in the design.
- Comprehensive alarm system. This should be configurable to allow the operators and/or system specialists to tailor limits and trends, and to permit a wide variety of automatic fault reports to be generated.
- Storage of all important machine data in a relational database system (RDBMS). Development and overall management of both the control system in particular, and the entire project in general, will benefit from a comprehensive database containing all machine details and configuration data.
- A full set of support servers and services should be provided. These will include high performance file stores, compute servers, centralised printing and backup facilities, etc.
- Ease of expansion. Experience on the SRS has shown that a constant programme of development and expansion will take place throughout the operational life of the facility. The control system should be designed with this in mind.

[#] Email: B.G.Martlew@dl.ac.uk

- The control system should exhibit a high degree of fault tolerance so that one isolated failure will not compromise operation of the entire light source.

3 REVIEW OF RECENT LIGHT SOURCE CONTROL SYSTEMS

Six recent projects have been selected to provide an indication of the current 'state-of-the-art' for light source control systems. The selection has been heavily biased towards European projects because the opportunities for close collaboration on control system design is likely to be greater here than for geographically more diverse laboratories. The APS has also been included in the list because of its major use of, and involvement with, the Experimental Physics and Industrial Control System (EPICS) collaboration, which is one of the main control system toolkits currently available.

3.1 ANKA

The ANKA control system [2] makes much use of modern, distributed object techniques and web technology. Low-cost PCs running Windows NT are used as both operator consoles and as data collection servers. Device interfacing is via the LonWorks field-bus with intelligent nodes and standard I/O modules, again to keep hardware costs to a minimum. The data collection systems utilise TACO-style device servers and communicate with the rest of the control system through CORBA. All applications are being written as Java applets so that a standard web-browser can be used to provide a familiar working environment for the operators.

3.2 APS

The APS control system [3] was one of the originating EPICS systems. It follows the classical standard model with Unix workstations as consoles connected over Ethernet and FDDI to VME crates with Motorola processor boards running VxWorks. It uses sub-nets of BitBus, GPIB and Allen Bradley 1771 to interface to the plant.

3.3 BESSY II

The BESSY II control system [4] is based on the EPICS toolkit with HP workstations running Unix for the consoles.[4] These are connected using Ethernet and ATM to VME crates running VxWorks for the middle layer and in-house CAN-Bus modules as the third layer (the plant interface). Extensive use is made of CAN-bus as the principle interface to the plant. Some use of VME I/O is made at layer 2 for applications requiring high data volume and processing speed and hardware with EPICS support such as timing modules. Application development has used the Self-Describing Data Set (SDDS) toolkit and Tcl/Tk scripting. Oracle has been

selected as a RDBMS to store all configuration data and other associated information.

3.4 ELETTRA

The ELETTRA control system [5] is, again, a three-layer structure with workstations running Unix as the consoles and two lower layers of VME crates running OS-9. The lower two layers are connected using MIL1553 bus and the upper two with Ethernet. The system makes extensive use of a RDBMS for storing parameter and machine settings.

3.5 ESRF

The ESRF control system [6] is based on workstations running Unix and VME crates running OS-9, below which is an in-house field-bus connecting to G64 crates used for the majority of the plant interfacing. Intelligent subsystems are interfaced at the VME crate level.[6] This system has been further developed and is now packaged as the TACO toolkit.

3.6 SLS

The SLS control system [7] is based on EPICS in a two-layer structure with PCs running Linux as consoles and VME crates with PPC processor boards running VxWorks for the plant interface. The majority of the plant is interfaced directly through digital and analogue connections, dispensing with the need for a field-bus layer. Applications are being developed in Java or Tcl/Tk to give an element of platform independence and will make use of an RDBMS. The CDEV interface has been implemented on top of the EPICS API to provide a more uniform and device oriented programming model.

4 MAJOR OPTIONS

4.1 EPICS

The EPICS system [8] has been widely adopted throughout the accelerator community. It has been used as the primary software toolkit on several major light source control systems with great success. The majority of implementations use the traditional approach of VME/VXI-based I/O controllers and Motorola 68K CPUs running the VxWorks operating system and UNIX workstations as operator interfaces. However, a considerable amount of work has been done around the world to port some aspects of the EPICS software to other platforms such as PCs running Windows NT or Linux for consoles and PPC processors for IOCs. Although EPICS provides standard tools for display creation, archiving, alarm handling etc, many users have found these tools to be inadequate and developed in-house alternatives. The big success of EPICS is based on the definition of a standard IOC structure, together with

an extensive library of driver software for a wide range of I/O cards. The CDEV interface is often used in conjunction with EPICS to provide a more object oriented programming interface for application software. Many users of the system report a steep learning curve and the need for significant development resources, but this is balanced by the large installed base and proven ability of this approach.

4.2 TACO/TANGO

The TACO control system, recently re-packaged and renamed as TANGO, is a development of the control system that was originally designed for the ESRF.[9] It is used extensively at the ESRF for both accelerator and beamline controls. It is also used to control the Hartebeesthoek radio telescope in South Africa and certain elements of the design have been adopted for the ANKA control system (see 3.1 above). The core of the design centres around object oriented device servers that communicate with higher levels via SUN ONC/RPC and XDR protocols. The original implementation used OS-9 on 68k CPUs for device servers and UNIX workstations for operator interfaces. More recently, TACO/TANGO has been ported to Linux, Windows NT, LynxOS and VxWorks although not all are fully supported. The core of TACO/TANGO doesn't provide as much functionality as EPICS, but it does benefit from the availability of ports to several alternative platforms.

4.3 Commercial

There are several commercial products available that can be used to build a control system. Most of them, such as LabView, are intended for systems of limited size and do not scale well when used in a large, distributed arrangement typical with accelerator and light source control systems. To date there is only one serious commercial contender, Vsystem from Vista Controls Inc. [10] Vsystem is of a similar design to EPICS, but has all the advantages (and disadvantages) of commercial standard support and development. Vsystem has been on the market for several years and is more popular for small to medium scale control systems. It has yet to be adopted as the primary control system on any large-scale light source project.

4.4 Other

A fourth alternative would be to adopt one of the less widely used control systems designs, develop the software currently being used on the SRS as Daresbury (based on the ISOLDE control system at CERN [11]) or to develop a new in-house design from scratch. The advantage of the home-brew approach is that the software can be very closely tailored to our exact needs. The disadvantage, however, is that this will inevitably lead to duplication of work already done at other

institutions, increased de-bugging and commissioning time and a lack of overall versatility.

5 CONCLUSIONS

It is possible to build a control system for a light source from any of the options presented. It is quite likely that the final choice will be influenced by some quite arbitrary factors: personal preference, experience with particular hardware platforms, operating systems and development tools. However, a dominating factor will be the level of available resources. It is clear that a lightweight system such as that currently used on the SRS would have a minimum resource requirement to get a basic system up and running. It does this by deferring the implementation of some of the required functionality to a later date. Alternatively, a control system based on EPICS would include most of the functionality from day one and makes the job of the control system user much easier during machine commissioning and early operation. However, as a consequence, it is a more complex system and has a greater resource demand.

The TACO system occupies a middle ground. It can provide a sound basis for a successful control system with lesser resource requirements than for EPICS, but does lack some of the functionality and support.

While the use of a commercial system can reduce the required resource demand, it is not clear that these systems offer significantly more than the 'free' collaborative ventures such as EPICS and TACO.

6 REFERENCES

- [1]A.A. Chesworth, J.A. Clarke, G.S. Dobbins, D.J. Holder, H.L. Owen, M.W. Poole, S.L. Smith, V.P. Suller, A. Wolski, "DIAMOND: A UK National Light Source Project", these proceedings.
- [2]M.Plesko et al, "The Control System for the Accelerator of ANKA", Proc. EPAC98, Stockholm, June 1998.
- [3]M.J.Knott, W.P.McDowell, F.R.Lenkszus, M.R.Kraimer, N.D.Arnold, R.T.Daly, G.R.Gunderson, B.K.Cha, M.D.Anderson., "The Advanced Photon Source Control System", Proc. PAC 91.
- [4]R.Muller et al, "Rapidly Installable High Performance Control System Facilitates BESSY II Commissioning", Proc. EPAC98, Stockholm, June 1998.
- [5]D. Bulfone, "Status and Prospects of the Elettra Control System", Nucl. Instr. and Meth. A 352 (1994), p63.
- [6]W-D. Klotz, "The ESRF Control System; Status and Highlights", Proc. ICALEPCS91, Tsukuba, 1991.
- [7]http://www1.psi.ch/www_sls_hn/controls/
- [8]<http://epics.aps.anl.gov/asd/controls/epics/EpicsDocumentation/WWPages/EpicsFrames.html>
- [9]A. Goetz et al, "TACO: An object oriented control system for PCs running Linux, Windows/NT, OS-9000 or LynxOS", Proc. PCaPAC96, DESY Hamburg, 1996.
- [10]<http://www.vista-control.com/>
- [11]I.Deloose, "Integrating the New Generation of ISOLDE Controls into a Multi-Platform Environment", Proc. PCaPAC96, DESY Hamburg, 1996.