EPICS DIRECTIONS TO ACCOMODATE LARGE PROJECTS AND INCORPORATE NEW TECHNOLOGY

L. Dalesio, J. Hill, LANL, Los Alamos, New Mexico, USA
R. Chesnut, H. Shoaee, SLAC, Stanford, California, USA

Abstract

EPICS will have two major projects that will push the next phase of development. The NLC will push scalability by requiring support for an order of magnitude more channels that existing facilities. SNS will be pushing to incorporate the latest technology within a large collaboration. As these projects have selected EPICS in the design phase, we have the opportunity to provide improvements in ease of use, control of degraded modes, and new functionality.

1 INTRODUCTION

The Experimental Physics and Industrial Control System (EPICS)[R] has been developed in collaboration with a multitude of other laboratories over the last ten years. Although there are over one hundred projects with an EPICS license, there is no direct funding to support EPICS. The toolkit is only modified by project directed, project funded work. Large projects that make an early decision to use EPICS create an opportunity for major advances. The recent decisions by SNS [1] and the Next Linear Collider (NLC) provide an opportunity to make advances in the technology. NLC is estimated to have over one million process variables and demanding requirements for both automatic beam steering and data archiving. The SNS requires the reliability of a production machine and the ability to support a project that is being developed over five different laboratories. This paper will cover the changes that are planned to improve productivity, provide data to everyone involved in these projects, and overcome some of the existing limitations.

2 IMPROVING PRODUCTIVITY

The labor required to implement a control system for NLC is estimated at 500 person-years. Although research budgets are shrinking, the complexity and scope of control systems is not being reduced, so we must work as efficiently as possible. Improved productivity will be accomplished using these methods: 1) reduce the time it takes to become productive EPICS users 2) apply some of the powerful commercial tools that are available to define and maintain the EPICS database, 3) use commercial products for front-end controllers that are more familiar to the equipment engineers, and 4) develop new tools that provide the same functions more efficiently.

2.1 Become Effective EPICS Users

The EPICS learning curve is very steep. Through experience, it is estimated that the learning curve is between three and six months. Experienced programmers that are also control engineers tend to be at the short end of the scale. Several years ago, it was extremely difficult to start up with EPICS, as the installation required a UNIX expert and the tools were poorly documented. Now the installation is simplified and documentation exists for all major components. It is difficult to locate applicable documentation, but there are two guides to EPICS to support navigation [2][3]. The most effective manner to get started, is to attend an EPICS training class or to work in a group where there are already EPICS experts working. This year, in addition to the Particle Accelerator School, Los Alamos held four training sessions and KEK held a training session. With training, the time to become a productive EPICS user is reduced to under a month.

2.2 Tools Used to Define EPICS Databases

Configuring the EPICS database is the method used to interface to the instrumentation. The database can be as simple as a data acquisition engine or as complex as a steady state control loop or complex interlock. The EPICS database is loaded into the front-end controllers at boot time as ASCII. Effective use of existing commercial tools can greatly improve productivity when configuring the EPCIS database. These ASCII files are typically produced using either a Relational Database or a schematic capture program. The relational database has several advantages when defining configuration parameters like hardware address, conversion, alarm limits and monitor dead bands for large number of process variables. The ability to enter the same field values for classes of process variables simplifies data entry. The Sequential Query Language (SQL) provides excellent tools for producing reports including signal lists that aid in point to point check out. Integrating wire list data gives an even more complete tool for integration and maintenance. BESSY II and NLC both provide papers on using a relational database for accelerator controls [4][5].

A relational database does not provide a reasonable tool for viewing the complex data and execution paths that can be present in an EPICS database. For visualizing these relationships, it is better to use a schematic editor. With a schematic editor, one can define the data passing and record processing portions of the database, and allow a process engineer to view schematics to understand the application. As more than one database can be loaded and repeat channels have the newest field take precedence, both databases can be loaded sequentially into an IOC. The obvious draw back is that the data is kept in two places. As long as the data is not overlapping, this is a minimal inconvenience. Using these two tools in combination provides a highly efficient way to configure, document, and maintain the EPICS database.

2.3 Commercial Products for Front-end Controllers

There are many programs available to configure process control systems. Many of these are well known to engineers involved in a project and can be readily used to implement some subsystems. With the application of the portable channel access server, these tools can be treated as front-end controllers in an EPICS system. At LEDA, there are portable server interfaces to Labview, Intellution, ActiveX, and IDL. In addition, PLC controlled subsystems have been integrated through the EPICS driver interface for the Modicon and Allen-Bradley PLCs. There are several issues to consider when using these solutions: data hiding, maintenance, ease of integration, event synchronization, and the late arrival of new requirements. These issues are weighed against the productivity gains of having engineers and physicists work in environments that are familiar to them. A final alternative is to have the system developed in one of these technologies with plans from the beginning to convert them into an EPICS environment at integration time. Judicious use of these technologies can yield significant savings just as well as a poorly planned use of these technologies can cause large additional costs. Some study of this is done at a poster presented at this conference [6]. The portable channel access server [7] has enabled us to use these commercial technologies and integrate them into an EPICS environment.

2.4 New Tools to Improve Productivity

Three areas are seeing very active tool development; state machines, archiving and display generators. The current implementation of state control is the State Notation Language. Changes to the current SNL include entry actions, exit actions, a graphical tool to create SNL [8] and a port onto the UNIX platform. Two new approaches to state transitions are being investigated. One uses Finite State Queuing Tool [9] and the other makes use of a commercial package called Control Shell from Real Time Innovations (http://www.rti.com/). In each of these cases, complex transition diagrams are made easier to express. In addition, the new tools also promise to reduce the time necessary to debug, by providing a more intuitive debug capability.

A new archiving tool is being developed to provide long-term data archiving, retrieval, and viewing [10]. This provides some new capability in archiving, viewing, and retrieval that should improve the capability of the machine physicists.

Several activities are taking place to improve MEDM, DM2K, DM, a new display manager that is easily extendable, and several JAVA display tools. All of the display manager changes are intended to improve capability, require less effort, provide more complex display control, or make the data available on all platforms. In all cases, these new tools will improve the ability of control engineers and physicists to implement, integrate, and maintain their control systems.

3 DATA AVAILABILITY

Most laboratories now support multiple computer architectures and operating systems. Having data available to all of these machines provides an environment where all members of the facility can see the current state of the machine or study its history. The channel access client has made it possible to access process variables into various flavors of UNIX, vxWorks, VMS, Windows98, and WindowsNT for years. The JAVA display editors and a web-based archive data viewer are tools that provide some commonality between windows and UNIX. These new tools mean that data is now available as well as accessible. The development of the PV gateway makes it possible for a large number of clients to connect to the control network with a limited impact. These new developments are providing collaborators with convenient access to their data.

4 OVERCOME EXISTING LIMITATIONS

There are several existing limitations in the communication and database engine, (EPICS core), that will be addressed in this next phase of development. In the database engine, modifications will be made to support online add of new channels, periodic monitors and operation of the database under other operating systems [11]. Channel Access modifications will be made to support longer strings, larger arrays, structured data and a more efficient communication protocol [12]. Changes will be made to allow interlacing channel access and database scan priorities. This will allow high priority control loops to have higher system-wide priority than the scanning of less critical database channels within a single IOC. Finally, a new directory service will need to be developed to support the name resolution for an NLC sized project, while keeping the reliability and discovery aspects of the broadcast method. This list contains the limitations that our community has lived with for the longest time. To make these changes require some serious rearranging of underlying code and would not be possible without the interest of new projects to fund such work.

5 CONCLUSION

The EPICS community continues to improve the technology. Productivity gains are being made by providing better training, making the documentation more useful and available, exploring the more effective use of the existing tools, providing for integration of widely know tools, and developing new tools that are easier to use. By making real-time data and archive data available to any desktop computer through JAVA display managers and web based archive viewers, projects are able to watch their processes from the control room, office or homes. With the opportunities offered by new projects, the current limitations of string and array size, inter-IOC priorities, and other core issues, are scheduled for improvement. The collaboration continues to improve EPICS, a result produced by our members through open dialogue, distributed support, and cooperative development.

REFERENCES

[1] D. Gurd, et.al. "Plans for a Collaboratively Developed Distributed Control System for the Spallation Neutron Source", ICALEPCS 99, MA1I02, Trieste, October 1999. Abstract at: http://www.elettra.trieste.it/ICALEPCS99/proceedings/abstracts/MA1I02.pdf

[2] S.A.Lewis, "Recommended EPICS Documentation", http://csg.lbl.gov/EPICS/RecommendedDocs.html

[3] L.R. Dalesio, "Guide to EPICS", http://mesa53.lanl.gov/lansce8/Epics/epicsguide.html

[4] R. Bakker, et.al., "Significance of a Comprehensive Relational Database System for Modern Accelerator Controls", ICALEPCS 97, Beijing, November 1997. http://www.aps.anl.gov/icalepcs97/paper97/p239.pdf

[5] H. Shoaee, "The Role of a Central Database for Configuration Management", ICALEPCS 99, MD1I01, Trieste, October 1999. Abstract at: http://www.elettra.trieste.it/ICALEPCS99/proceedings/abstracts/MD1I01.pdf

[6] L.R. Dalesio, "Integrating Subsystems from External Sources", ICALEPCS 99, TC1P42, Trieste, October 1999. Abstract at: http://www.elettra.trieste.it/ICALEPCS99/proceedings/abstracts/TC1P42.pdf

[7] J.O. Hill, "A Server Level API for EPICS", ICALEPCS 95, W1A-b, "Chicago, November 1995. Paper at: http://adwww.fnal.gov/www/icalepcs/abstracts/PDF/w1ab.pdf

[8] S. Hunt, "Controls Through Pictures, Graphical Tools for Building Control System Software", ICALEPCS 1999, WD1O05, Trieste, October 1999. Abstract at: http://www.elettra.trieste.it/ICALEPCS99/proceedings/abstracts/WD1O05.pdf

[9] J. Perlas, "Design and Implementation of a Finite State Queuing Tool", ICALEPCS 1999, MC1P64, Trieste, October 1999. Abstract at: http://www.elettra.trieste.it/ICALEPCS99/proceedings/abstracts/MC1P64.pdf

[10] K. Kasemir, "Data Archiving in EPICS", ICALEPCS 99, WC1P61, Trieste, October 1999. Abstract at: http://www.elettra.trieste.it/ICALEPCS99/proceedings/abstracts/WC1P61.pdf

[11] M. Kraimer, "EPICS: Porting IOC Core to Other Operating Systems", ICALEPCS 99, WC1P01, Trieste, October 1999. Abstract at: http://www.elettra.trieste.it/ICALEPCS99/proceedings/abstracts/WC1P01.pdf

[12] J.O. Hill, "Next Generation EPICS Communication Protocols", ICALEPCS 99, MC1P57, Trieste, October 1999. Abstract at: http://www.elettra.trieste.it/ICALEPCS99/proceedings/abstracts/MC1P57.pdf