CEBAF-PR-95-050
DOE/ER/40150--381

The CEBAF Control System[1]

William A. Watson III, Continuous Electron Beam Accelerator Facility,
MS 12H, 12000 Jefferson Av., Newport News, Virginia, 23606, USA

Abstract

CEBAF has recently upgraded its accelerator control system to use EPICS, a control system toolkit being developed by a collaboration among laboratories in the US and Europe. The migration to EPICS has taken place during a year of intense commissioning activity, with new and old control systems operating concurrently. Existing CAMAC hardware was preserved by adding a CAMAC serial highway link to VME; newer hardware developments are now primarily in VME. Software is distributed among three tiers of computers: first, workstations and X terminals for operator interfaces and high level applications; second, VME single board computers for distributed access to hardware and for local control processing (complex sequences, limit checking, some process control); third, embedded processors where needed for faster closed loop operation. In some cases, multiple VME processors transparently access a single serial highway for improved performance. This system has demonstrated the ability to scale EPICS to controlling thousands of devices, including hundreds of embedded processors, with control distributed among dozens of VME processors executing more than 125,000 EPICS database records. To deal with the large size of the control system, CEBAF has integrated an object oriented database, providing data management capabilities for both low level I/O (calibration, alarm limits, etc.) and high level machine modelling (optics properties, etc.). A new callable interface which is control system independent permits access to live EPICS data, data in other Unix processes, and data contained in the object oriented database (extensible to other sources).

INTRODUCTION

The Continuous Electron Beam Accelerator Facility

CEBAF is a 4 GEV electron accelerator in the process of commissioning in Newport News, Virginia, with the first experiments expected to run this summer. The unique features of this facility are its continuous beam and high luminosity -- ideal for experiments requiring large samples of events with minimal accidental coincidence rates.

The accelerator consists of two 0.4 GeV superconducting RF linacs connected by two 180° arcs. Each linac consists of 20 cryomodules, each containing 8 accelerating cavities. Beam is recirculated through the machine for up to 5 passes yielding an energy of 4 GeV. After any pass, the beam may be split and sent to any of the 3 halls, allowing simultaneous operation at the same or different (modulo 20%) energies. Two injectors (one thermionic, one polarized) and a 3 slit chopper will allow different halls to receive different beam intensities and polarization.

Two of the three halls house conventional small solid angle spectrometers, and will use the full beam intensity (200 uA). Hall B will house the CEBAF Large Acceptance Spectrometer (CLAS), which will require beam currents 3 or 4 orders of magnitude lower. In Halls A and C, parity violation experiments will require measurements accurate to a part in 10^7. This flexibility in beam delivery and constraints upon beam stability (both current and polarization) place complex demands upon the control system.

The control systems for both the accelerator and the experimental facilities are based upon EPICS -- Experimental Physics and Industrial Control System. [1] EPICS was selected as a replacement for the original control software when problems with scaling to the full machine were encountered nearly two years ago. The following discussion will describe the controls hardware at CEBAF, the use of EPICS in this system, and the higher level software being added above EPICS.

CONTROL SYSTEM ARCHITECTURE

Standard Model

The control system follows what has been referred to as the "standard model": a client-server system consisting of a collection of Unix workstations and X-terminals connected by a network to multiple servers running device control software. At CEBAF the network is a switched ethernet, which allows simple scaling to high bandwidths as needed. The server machines are VME single board computers running the EPICS real-time database. The client machines are HP workstations configured as two clusters for redundancy (Figure 1).

Conversion to EPICS

EPICS was selected as a replacement for the original CEBAF control system (TACL) when problems were encountered scaling it to over 25,000 control points. [2] The switch to EPICS was accomplished incrementally during machine commissioning, starting with the linacs and arcs a little over a year ago, and ending with the injector (except the gun) this past winter. The gun will be converted following an upgrade to its control hardware this summer.

The two systems were operated concurrently for much of the year, with information being exchanged between the systems. This co-existence was made easier by the fact that both EPICS and TACL are name based control systems -- applications address parameters in the machine by the name of the parameter, and not its hardware address.

[1] Work supported by the Department of Energy, contract DE-AC05-84ER40150. Work performed by the various groups at CEBAF and within the EPICS community.

for unlimited asynchronous replies (in the case of monitoring a value or an alarm status). Clients connect to servers by broadcasting the name of the desired channel (record.field), and broken connections are automatically re-established when a server becomes available again. Several hundred connections per second may be made to a single server, and monitoring several thousand changes each second produces a negligible load on a workstation (6% on an HP 715/50 for 2000 values/sec).

Algorithms

Control algorithms may be implemented via a number of techniques within EPICS. Database records, including subroutine and calc records, may be linked together to form an algorithm, with data transferred from one record to another. Most low level applications use this technique. More complex algorithms may in addition use a state machine sequencer, using a special language and compiler to facilitate this approach. A sequence runs as a channel access client, and may access both local and remote databases as well as any other resources on the VME system. High performance algorithms are implemented as tasks on the IOC, controlled and monitored through database records. This is the approach used for CEBAF's beam position monitors and fast feedback systems. [8] Finally, Unix applications (typically in C or C++) may interact with the EPICS database through channel access.

Utilities

EPICS includes several main general purpose client programs. (1) a save/restore utility, which includes basic check before restore and save-only capabilities; (2) general purpose operator interfaces (one X based, the other Motif); (3) an alarm manager to present alarm status organized into trees of arbitrary depth; (4) an archiver utility supporting 3 styles of data acquisition: (i) periodic sampling, (ii) record on significant change, and (iii) event driven sampling. In the third mode, a change in one channel can initiate recording of values for a set of other channels.

There are a wide variety of other general purpose clients including diagnostic utilities, a knob manager, a parameter page display -- with more being written each year.

EPICS also includes graphical and text based database creation tools, and scripts and other tools to facilitate building and managing the databases.

Integration with other software

EPICS has been integrated with a large number of other packages, including tcl/tk, PV-Wave, IDL, Mathematica, WingZ and others. In each of these packages, EPICS variables are accessible by name through channel access, so that channel access has functioned as a limited form of software bus.

CONTROL DEVICE API

One difficulty encountered with EPICS for high level applications is the fact that the implementation details of a low level algorithm are in many cases too visible: the high level application knows the names of the various records and fields. A change in the low level algorithm which adds or deletes records (moving the needed information to other recon invalidate the high level application.

A new layer (cdev, for control device) has been adc above channel access to provide implementation hiding as we. as several additional features. In this new API (defined by a team from all major EPICS sites), all I/O in the system is in the form of messages to devices such as *on* or *off* or *get current*. (Note: cdev builds upon ideas in earlier work done at ANL/ APS [9]). In cdev, a device is a virtual entity potentially spanning multiple servers, and even multiple underlying services such as EPICS channel access, an archive data server, a host-based database, or a legacy control system.

The cdev layer routes messages to the appropriate service (such as channel access) based on the device name and the message. In this way, one can obtain the length of a magnet (from a static database) as easily as the current (from the real-time database). The application program is unaware of the source of the data: if the low level application is changed, it is only necessary to fix the mapping, and all high level applications using that information are correct. An architectural diagram of cdev is shown below. All services at the lowermost layer are dynamically loaded, and new services (interfaces to other systems) may be added without recompiling any other cdev sources (in fact without stopping the running application).

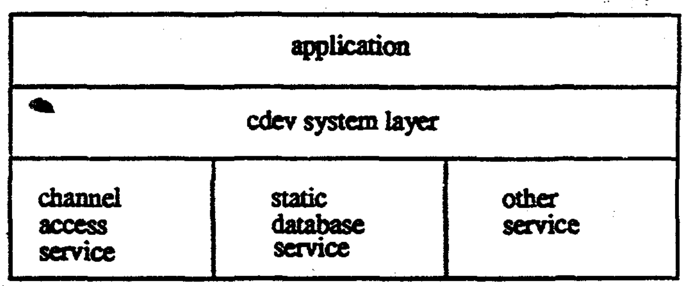

Figure 2: Block diagram of a cdev application with 3 services currently loaded.

An implementation of cdev in C++ has demonstrated that this additional layering introduces on the order of 10% additional I/O processing for both establishing connections and for receiving asynchronous replies. Advantages gained include a device abstraction, implementation hiding, access to a wider variety of data (including a centralized database, described below), wildcard query capability (not present in EPICS), and the ability to treat a collection of devices as a single device.

cdev Applications

A useful cdev demonstration application using tcl/tk for a windowing interface has been written at CEBAF. It allows selecting devices by regular expression, and can read/write/ monitor device attributes.

A more sophisticated cdev application now in development will allow measuring correlations among parameters in the control system. This correlation package takes its inspiration from the SLAC Correlation Plot package, which was designed to "measure anything as a function of anything else".[10] Any number of parameters may be systematically

ACKNOWLEDGEMENTS

The conversion of the CEBAF control system was accomplished by deputy head Karen White and other members of the CEBAF controls department, with considerable debugging assistance from the CEBAF operations crew and technical support by visitors from LANL. Work on the cdev specification was done in collaboration with Claude Saunders and other members of the EPICS collaboration at APS, LBL, LANL, DESY, and Keck; the cdev implementation was done by Jie Chen in the Physics Data Acquisition Group, and by Walt Akers and Danjin Wu in the Controls Department. Development of high level applications was done by Hamid Shoaee (on sabbatical from SLAC), and by Johannes Van Zeijts and other members of the accelerator physics and accelerator performance groups.

REFERENCES

[1] Leo R. Dalesio, et. al., "The Experimental Physics and Industrial Control System Architecture: Past, Present, and Future", *International Conference on Accelerator and Large Experimental Physics Control Systems*, Oct. 1993.

[2] William A. Watson III, et. al., "The CEBAF Accelerator Control System: Migrating from a TACL to an EPICS Based System", op. cit.

[3] Sally Schaffner, et. al., "Device Control at CEBAF", conference.

[4] Leo R. Dalesio, et. al., "The EPICS Architecture", ICALEPCS, 1991.

[5] The web site for EPICS is http://epics.aps.anl.gov.

[6] Jeff Hill, "Channel Access: A Software Bus for the LAACS," ICALEPCS, 1989.

[7] Jeff Hill, private communication.

[8] Mahesh Chowdhary et. al., "A Prototype Fast Feedback System for Energy Lock at CEBAF", this conference.

[9] Claude Saunders, private communication.

[10] L. Hendrickson et. al. "Correlation Plot Facility in the SLC Control System", ICALEPCS, 1991.

[11] Matthew Bickley et. al. "Managing Control Algorithms with an Object-Oriented Database", this conference.

[12] Bruce Bowling et. al., "Integrated On-Line Accelerator Modeling at CEBAF", this conference.

[13] John Ousterhout, "Tcl and the Tk Toolkit" Addison-Wesley, 1994.

[14] Johannes Van Zeijts, private communication.

[15] R. V. Servranckx, User's Guide to the Program DIMAD, SLAC Report 285 UC-28 (A), May 1985.