

Status and Design of the Advanced Photon Source Control System

W. McDowell, M. Knott, F. Lenkszus, M. Kraimer, N. Arnold, R. Daly

Advanced Photon Source, Argonne National Laboratory

9700 South Cass Avenue Argonne, Illinois 60439

Abstract

This paper presents the current status of the Advanced Photon Source (APS) control system. It will discuss the design decisions which led us to use industrial standards and collaborations with other laboratories to develop the APS control system. The system uses high performance graphic workstations and the X-windows Graphical User Interface (GUT) at the operator interface level. It connects to VME/VXI-based microprocessors at the field level using TCP/IP protocols over high performance networks. This strategy assures the flexibility and expansibility of the control system. A defined interface between the system components will allow the system to evolve with the direct addition of future, improved equipment and new capabilities.

I. INTRODUCTION

The APS accelerator control system is a distributed system consisting of operator interfaces, a network, and interfaces to hardware.

The operator interface is a UNIX-based workstation with an X-windows graphical user interface. The workstation may be located at any point on the facility network and maintain full functionality. The user has the ability to generate and alter control displays and to access the alarm handler, the archiver, interactive control programs, custom code, and other tools.

The TCP/IP networking protocol has been selected as the underlying protocol for the control system network. TCP/IP is a commercial standard and readily available from network hardware vendors. Its implementation is independent of the particular network medium selected to implement the controls network. In the development environment copper Ethernet is the network medium; however, in the actual implementation a fiber-based system using hub technology will be utilized. The function of the network is to provide a generalized communication path between the host computers, operator workstations, input/output crates, and other hardware that comprise the control system.

The crate or input/output controller (IOC) provides direct control and input/output interfaces for each accelerator subsystem. The standard crate uses either the VME or VXI standards, a Motorola 68040 processor, network communications, and a variety of signal and sub-network interfaces. The 68040 processor provides the crate with the intelligence to allow it to run its software autonomously with respect to all other devices in the system. The software running in the crate hides hardware dependencies from the high-level software running on the workstation. There are approximately 45 crates used in the accelerator control system with plans for about 50

additional crates controlling insertion devices. A real-time operating system, VxWorks, is run in the crate central processing unit (CPU) to provide the basis for the real-time control.

Design challenges for the group implementing the APS controls included protecting investment from rapid obsolescence and designing into the system the ability to quickly incorporate improved equipment and software. The APS approach to the solution to these problems is to use a distributed control system and standards, to provide "tools" to the greatest extent possible instead of custom programming, and to collaborate with other laboratories and groups whenever possible.

II. STANDARDS

The use of standards allows vendor independence, a migration path for future advances in technology, and the use of commercially available "tools" when these tools meet requirements. A modular approach to design supports both hardware and software modularity in crates, workstations, I/O modules, etc. Standards also make it possible to share accelerator control software and collaborate with other laboratories. Before hardware standards, each facility designed and built its own hardware. This custom hardware also required custom software. Today, most new facilities are selecting hardware based upon industrial standards and thus sharing software has become possible.

III. COLLABORATION

The advantages of collaboration are obvious. Collaboration reduces the need to start from scratch when a majority of the system requirements are not unique. Collaboration also takes advantage of existing "tools" that meet standard requirements and has the effect of freeing up staff to work on "non-standard" requirements. EPICS collaborators now include Los Alamos National Laboratory, Lawrence Berkeley Laboratory, and the Superconducting Supercollider Laboratory.

Prior to the Accelerator Control Toolkit Workshop in 1988 [1] the APS controls group had decided to examine existing control systems with the aim of determining if they could be used at APS. After studying several systems we decided to pursue a collaboration with the AT-8 group at Los Alamos National Laboratory (LANL).

APS received the first version of the LANL Ground Test Accelerator Control System (GTACS) software in late 1988, the controls and computing group at Argonne APS, with the cooperation of Los Alamos AT-8, suggested and implemented several changes to the structure of the core software which facilitated the incorporation of many different hardware devices without recompiling the complete code. This new, extensible, version of the code was then named EPICS (Experimental Physics and Industrial Control System) to distinguish it from the original GTACS. GTACS is still in use at Los Alamos and

*Work supported by U.S. DOE Office of Basic Energy Sciences under contract no. W-31-109-ENG-38.

several other sites.

Argonne, with the cooperation of Los Alamos, has taken a lead role in developing and applying EPICS tools. Los Alamos has taken the role of coordinating the development efforts between laboratories as well as maintaining and improving sections of the system software. The Advanced Photon Source control system has been completely developed using EPICS tools. The original applications at APS were the radio frequency (RF) test stand and the linac test stand. Today the linac itself is being installed and is completely controlled by the EPICS system. As a further example of the APS commitment to EPICS, it has been decided that EPICS will be the system used to develop and run all ANL-developed experimental beamlines and front-end devices at the APS.

Continuing development of the software is foreseen. APS plans to improve the system by adding new functionality when required by applications. We expect other laboratories will also add functionality to the system.

IV. GENERAL DESIGN FEATURES OF EPICS

EPICS-based control systems include operator interfaces which consist of multi-screen UNIX workstations using a graphical user interface based on the X-windows Motif model. The selection of a workstation implementation based on X-windows for the operator interface allows field instruments, video tools and the interface for controls and data acquisition, and the software development effort to take place in the same environment. It is now possible to locate consoles in offices, labs, seminars, and wherever they are required in the field. The operator can select by list, diagram, flow chart, and map as well as the more standard text-based selection menus. The workstation supports external software interfaces including: Wingz, Nodal, Mathematica, accelerator modeling codes, Fortran, C, and any application software a physicist may write. Auxiliary displays using X-window terminals and projection and large screen displays in the main control room and seminar rooms are easily implemented using commercial equipment.

V. SPECIFIC EPICS FEATURES

EPICS provides a number of tools for creating a control system. This minimizes the need for custom coding and helps ensure uniform operator interfaces.

An arbitrary number of IOCs and operator interfaces (OPIs) can be supported and, as long as the network is not saturated, no single bottleneck is present. If a single IOC becomes saturated, its functions can be spread over several IOCs. Rather than running all applications on a single host, the applications can be spread over many OPIs.

A. EPICS Core Software

EPICS consists of a set of core software and a set of optional components. The core software, i.e. the components of EPICS without which EPICS would not function, are: channel access-client and server software, database scanners-monitors, the database configuration tool (DCT), and the source/release control. All other software components are optional.

B. Channel Access

Channel access provides network-transparent access to IOC databases. It is based on a client-server model. Each IOC provides a channel access server which is willing to establish communication with an arbitrary number of clients. Channel access client services are available on both OPIs and IOCs. A client can communicate with an arbitrary number of servers. [2]

C. Database

The heart of an IOC is a memory-resident database together with various memory-resident structures describing the contents of the database. EPICS supports a large and extensible set of record types, e.g. ai (Analog Input), ao (Analog Output), etc. Each record type has a fixed set of fields. Some fields are common to all record types and others are specific to particular record types. Every record has a record name and every field has a field name. The first field of every database record holds the record name, which must be unique across all IOCs attached to the same TCP/IP subnet. A number of data structures are provided so that the database can be accessed efficiently. Because they access the database via database access routines, most software components do not need to be aware of these structures.

D. Database Access

With the exception of record and device support, all access to the database is via the channel or database access routines. Database scanning is the mechanism for deciding when to process a record. Four types of scanning are possible: periodic, event, I/O event, and passive.

A request can be made to process a record periodically; a number of time intervals are supported. Event scanning is based on the posting of an event by any IOC software component. The I/O event scanning system processes records based on external interrupts. An IOC device driver interrupt routine must be available to accept the external interrupts.

E. APS-Developed EPICS Tools

The Motif-based Display Editor/Manager Tool (MEDM) allows staff to easily create, configure, and modify displays and then activate and connect displays to actual hardware. The display editor portion of MEDM provides process connections via an interactive display editor/control. Editing/drawing commands are used to build displays with little or no custom programming required to build applications. Process connections to the data are entered as process variable name and field name; no other knowledge of the hardware configuration is required. This tool allows the designer to modify any aspect of the display element, as well as copy, align, and group display elements. Standard drawing tools can be used to create backgrounds which can then be imported into MEDM via GIF files. The alarm handler provides the ability to display alarm messages in a hierarchical manner. The backup and restore tool (BURT) stores and retrieves data from selected channels. A database configuration tool (DCT) allows the application designer to configure the run-time database. The knob manager (KM) allows the operator to attach physical knobs to process variables.

Third party software now interfaced to EPICS via channel access calls include Wingz, Nodal, DevTest, Mathematica and an interpretative version of C. An automatic cavity conditioning system has been developed using a Mathematica routine running under EPICS.

VI. SYSTEM FEATURES

The equipment interface or input/output controller is based on VME/VXI standard and uses a Motorola 68040 microprocessor for control and a fast, real-time kernel, VxWorks, as the operating system. No local disk drive is required in this system. The software image can be down-loaded from a file server or from on-board PROM. The system also supports task priorities and interrupts as well as custom code. The APS control system will use about 45 such crates. More than twenty VME input-output modules are currently supported. Others will be supported as they become available and desirable.

Most information preprocessing is performed at this level. Sequential and control-loop operations can be performed. In this way, maximum benefit is gained from the many IOC processors operating in parallel.

Hardware I/O can be generalized into several types: analog and binary I/O, stepper motor, and sub-network connections. Hardware records contain information for run-time processing including scaling, engineering units, hardware address, and scanning information. The process database contains parameters such as scan requirements, address of I/O, engineering unit conversions, alarm limits, control parameters, current value, etc. for each defined process variable. Records can be linked together to implement complex algorithms. The database scanner scans database records and updates dynamic fields (i. e. value, alarm status, severity, etc.). Data conversions including linear, piecewise linear, and value-to-state are supported. Alarm checking for limit violations, state violations, hardware communication errors, and alarm severity is supported. Binary and analog I/O are provided through the database. Analog input can be smoothed, and analog output can be either incremental or positional and the output rate of change can be specified. Multi-bit (16 states) and single binary I/Os are supported.

All I/O fields can be connected to a hardware device or another database element (calculator, PID, fan-out to several devices). Timing signals, motion control, and transient digitizers are supported through the database.

Stepper motor records can get desired position/velocity input from user or other data, provide positional or velocity mode operation, provide position feedback through an encoder or another database record, and allow simple closed-loop position maintenance. Limit switches are supported through index or other database records.

General purpose calculations, a PID algorithm, and a signal selector are supported to provide closed-loop control and data filtering. The calculation record supports up to six sets of input, C language expressions, algebraic operations, relational operations, boolean operations, and trigonometric functions. The PID algorithm allows a user interface optional database

connection for setpoint, proportional, integral, and derivative gains; incremental or positional output; and anti-windup and reset on the integral term.

VII. NETWORKS AND SUBNETS

The communication network will be a combination of FDDI, Ethernet over fiber, and Ethernet over copper. Future plans call for the introduction of 100 MB Ethernet and ATM when these protocols have been standardized. The cable plant will be installed using fiber with a hub and spoke topology. All fibers will terminate in the main control room. This technique will allow APS to tailor network bandwidth as required by the application running in a particular IOC.

Subnets can be driven from the IOC. GPIB, RS-232, and Bitbus [3] are presently supported. These subnets allow various instruments and low-cost, few-point interfaces to be connected and controlled. The Bitbus can be used to make remote, multidrop connections to GPIB and other interface subsystems.

VIII. PRESENT STATUS

All seven of the linac IOCs are now installed and have been successfully tested. All subnets have been installed and are being tested. The linac timing system installation and testing is complete. Local consoles, implemented with X-terminals, are now operational in the klystron gallery. File servers and OPIs have been installed in the injector control room to support linac commissioning. An FDDI link between the Building 362 development area and the injector control room has been installed and is operational. This allows continued controls development and supports network growth into the final APS site. Positron accumulator ring (PAR) installations will start in June 1993.

EPICS tools in development include a graphical process-variable links tool which will reduce the complicated process of creating and documenting multi-parameter control processes. A generalized drag-and-drop process has been developed to facilitate the transfer of process-variable names between EPICS tools. The sequencer tool has been ported from the IOC to the OPI using a Posix threads library.

IX. REFERENCES

- [1] Accelerator Automation Application Toolkit Workshop, sponsored by AT Division and MP Division, Los Alamos National Laboratory and CERN, Geneva, 31 October - 4 November, 1988.
- [2] J. O. Hill, "Channel Access: A Software Bus for the LAACS," in *Accelerator and Large Experimental Physics Control Systems*, D. P. Gurd and M. Crowley-Milling, Eds., ICLAEPCS, Vancouver, B. C., Canada, pp. 288-291, 1989.
- [3] G. J. Nawrocki, N. D. Arnold, M. G. Hoffberg, J. R. Winans, S. J. Benes, "An I/O Subnet for the APS Control System, The BITBUS Universal Gateway," these proceedings.