EPICS – FUTURE PLANS

Matthias Clausen, DESY, Hamburg
Martin Richard Kraimer, ANL, Argonne, Illinois
Jeffrey Owen Hill, LANL, Los Alamos, New Mexico
Kay-Uwe Kasemir, ORNL, Oak Ridge, Tennessee
Timo Korhonen, PSI, Villigen
Leo Bob Dalesio, BNL, Upton, NY

Abstract

Over the last two decades EPICS has evolved from a basic set of control applications created for the Ground Test Accelerator to a rich and reliable control system framework installed in more than 120 locations worldwide. The continuous development of EPICS is supported by the worldwide collaboration and coordinated by a set of major laboratories. This procedure ensures continuous quality checking and thus leads to stable production versions. The clear separation of the robust core software on the Input Output Controllers (IOC'S) from the channel access protocol and the applications running on workstations and servers allows nearly independent software developments on all three levels.

This paper will describe the new developments on the IOC side, which will increase robustness by adding redundancy, and improve the management and the functionality. This includes the vision of a new Java-based IOC. The support for new data types will bring more flexibility to the channel access protocol. New developments on the application side are clearly indicating that Java and Eclipse (e.g. Control System Studio – CSS, XAL and others) will form the basis for many future applications.

STATUS

Even though – or just because the EPICS core software has reached a mature state, it is time to think about the long term future plans for EPICS. The core software is running in several thousand IOCs around the world and has proven to be at least as reliable as commercial products. IOCs are running in several instances for several years continuously. The collaboration is growing continuously and every year new installations and institutions decide to choose EPICS as their 'control work horse'.

In order to keep the software up to date and meeting state of the art requirements, there was an attempt in 2003/4 to discuss the future of EPICS. Three meetings were carried out at Rutherford, Tsukuba and Santa Fe under the title 'EPICS-2010'. As a result a list of desired features was composed. In contrast to the mutual agreement that it would be important to work on these subjects there was no straight answer how these developments could be funded. Several unsuccessful attempts were made to apply for funding dedicated to EPICS-2010 developments. As a result one can state that the main source for new developments is new projects. These kinds of developments are driven by local requirements rather than global design specifications. In the following we will describe that this evolutionary approach can still play an important role for the future of EPICS.

IOC CORE SOFTWARE

Development work continues in EPICS Base, although the pace of change has slowed due to lack of effort, the increasing maturity of the code-base and a desire to maintain an easy upward migration path for existing users and installations. Marty Kraimer's departure from Argonne has also reduced the effort available for new development within the existing code-base. In contrast however the number of Operating Systems and OS/Processor combinations that EPICS supports has increased, and still more targets are expected.

The 3.14 branch of our CVS repository contains a number of minor enhancements made since the R3.14.9 release in February: Soft timing synchronization for RTEMS IOCs, a new VAL keyword in CALC expressions, and improved error reporting from dbLoadRecords being the main items of note.

We maintain a list of enhancements which could be implemented in the existing code-base with relatively little effort, as well as lists of longer-term development projects. The APS will be hosting a week-long Codeathon (Coding marathon) in the spring of 2008 for core developers to try and get some of the smaller items completed.

We are also hoping to have a visiting PhD student do some work on automating our regression test suite in the next few months, which if successful will make the tests much quicker to run and the release process easier to manage.

The EPICS Extensions programs distributed by the APS have all now been converted to use the 3.14 build rules, and we hope to have a new Windows Extensions installer built against Base version 3.14.9 very soon.

Driver Support

The Asyn Driver IOC support module is another package that has progressed quite a lot recently, mainly due to the efforts of Mark Rivers. Asyn provides standardized and cross-platform communications interfaces over RS232, RS485, TCP/IP, UDP/IP, GPIB and VXI-11 protocols, and can also be used for register-based synchronous I/O devices as well as for asynchronous message-based I/O. EPICS Device support written for the old devCommonGpib package can be used with Asyn, and for new devices the Streams module from the Swiss Light Source provides an easy way to support moderately complex protocols over any Asyn port. Asyn can be used to control PLCs using various flavors of MODbus as well as via DirectNet, and all these communications modules are portable to any Operating System that supports EPICS and the necessary interfaces.

Java IOC

A rich list of ideas has been collected throughout the EPICS 2010 meetings. Some meetings of the EPICS core were taking place to identify items which could make it into next releases of the EPICS core distribution. Funding and restructuring in institutions inhibited to continue these efforts. Fortunately the development of the Java IOC started independent from the core development. 'Fortunately' - because new ideas can be investigated in this development without having to worry about reliability and compatibility in the first place.

Two projects are actually funding Java IOC developments:

- A Small Business Innovative Research grant
- BACnet Interface/ Gateway

The core features of this new development can be summarized as follows:

- Hierarchical database definition
- Implementation the notion of 'Device/ Property/ Characteristics and Commands'
- Decoupling of data processing from the control system protocol
- Decoupling of data processing from data acquisition.

The creation of a table record which includes in the hierarchy a set of motor records, which are composed of a set of binary I/O and analog I/O as well as position outputs and readbacks, provide a demonstration of a hierarchical record. The proof of concept (passing data from a BACnet device to a channel access client) is part of the BACnet project. This will demonstrate the potential of this development. A potential application for this project is to support the data representation for high level physics applications.

Timing Support

Time stamping of EPICS channels is based on the IOC nodes delivering the time, which is naturally the solution that provides the best performance. However, it requires that the IOCs need to keep themselves in the correct time. This has been so far achieved with a custom written package (many platforms do not necessarily have good timekeeping services built-in.) This time service contained all the services for an IOC to maintain its local time in a single monolithic package. The design had several shortcomings: difficult to understand, tailored to a specific system (vxWorks), the functionality was not up to today's standards for heterogeneous IOC nodes and so on.

The main design goals for the new time service are:

- Provide a robust time service that has a well-defined behaviour
- The design should allow adding user's preferred time services (modularity)
- The system should have fallbacks in case some service has problems (reliability)
- Switching between the different time services should be as smooth as possible (monotonously progressing time)

The new time service is mainly based on the General Time idea [1]. In this structure, there is a lightweight time source manager and then a number of time providers. The set of time providers can be configured for each IOC separately. A time provider needs a driver that gets the time from a time service (IOC clock, NTP, timing system, etc.) The time providers are organized like a stack and have the notion of validity.

Redundancy

The demands for high availability (HA) controls are nowadays mainly driven by new light sources which are currently under construction or in the design phase. Availability numbers of 99.8% and more can only be achieved when utilities like power, water or cryogenics (nearly) never fail. Continuous operation of the control systems in charge can only be achieved with redundant font end controllers.

The support to operate a redundant pair of IOCs has been implemented at DESY [4] to meet the requirements for continuous operations of the cryogenic installations for the XFEL.

The implementation consists of three major components:

- The Redundancy Monitor Task
  This is an OS and EPICS independent implementation to supervise the state of so called primary redundancy resources (PRR).
- The Continuous Control Executive
  This task is supervised by the RMT. It's purpose is to continuous synchronize of the two redundant EPICS databases.
- The State Notation Language (SNL)-Executive
  This task synchronizes SNL programs between two IOC's over the dedicated Ethernet link

An XML based protocol has been implemented to monitor and control the RMT and all the registered processes (like CCE, SNL-exec, drivers …) In conjunction with a user interface plugin for CSS it is now possible to monitor and control IOC core functionalities remote.

State Notation Language Support

State notation programs are providing the second level of interaction with the controlled process. Depending on the programmers desire record processing or SNL programs can be used for continuous operations – like supervisory programs. State oriented controls are the domain of SNL programs, like startup-, shutdown- or recovery procedures.

The functionality of the SNL package has been continuously improved. It is currently hosted by SLAC. The development support on the other hand is traditionally relatively poor. This will be improved by a new language sensitive editor based on the Eclipse C/C++ Development Tooling (CDT), which is currently under development. The final tool can be loaded as a plug into Eclipse as well as into CSS. It will be available from the DESY CSS-Update site [6].

EPICS Goes Embedded

A good example for developments to run EPICS on small – or embedded hardware is the APS booster magnet upgrade at Argonne for top off operation. It requires new instrumentation to meet increased magnet ramping requirements. To address these requirements, the APS Instrumentation and Controls groups collaborated to design a new control system module called the Mini IOC. The Mini IOC hardware is based on a commercial evaluation board containing an FPGA with embedded processor and built-in interfaces for 128MB of DDR SDRAM and Ethernet. A custom module is used for analog controls and monitors. The PowerPC embedded processor runs an EPICS database built on the VxWorks operating system allowing remote access via Ethernet.

Other developments are targeting to run EPICS on PLC hardware. Since version 3.14 EPICS core has been unbundled from the VxWorks specific system calls. This has increased the flexibility to run EPICS on ‘any’ supported operating system.

CA GATEWAYS

Channel Access (CA) gateways are in operation in many places. They allow separating the controls network from the office network or even public networks. Besides this security aspect gateways also optimize the number of CA connections to the IOCs because several CA clients can share one connection to an individual IOC.

Due to these important functionalities gateways play a growing role in today’s installations. Performance and functionality have been continuously improved over the last years. The availability of this service is key for machine operations in many places. This was the driving force to implement redundancy also for the CA gateways. The development was carried out based on the applications – namely RMT – which are already available for redundant IOC’s [12].

PROTOCOLS

The Channel Access (CA) protocol between the IOCs and the applications has been designed for a reliable, high performance data transport providing the basic support for the data being processed on the IOC. Only basic changes over a ten years time frame make sure that the compatibility between old and new client/ server connections is guaranteed. This is an important factor for sites which have to support a mixture of ‘legacy’ systems running old EPICS versions in parallel with applications running the latest EPICS CA protocol stack.

This conservative approach implies that new features make it only very slowly into new CA releases. One example is the support for flavoured data being developed for the Los Alamos Neutron Science (LANSCE) control system.

Another approach is the integration of new protocols into the EPICS control domain. New developments on the IOC (Java IOC) and on the application level (Control System Studio) will create opportunities in this direction.

Channel Access

Currently, the subscription update event queue in the EPICS server is capable of carrying event payloads consisting always of the channel's value, time stamp, and alarm state. The complexity of the LANSCE macro pulse beam gates requires unique capabilities from the control system - which is currently a hybrid of EPICS and also the original LANSCE Data System. The favoured configuration after an upgrade would be a homogeneous EPICS based system with a tool based approach to the development of modular application programs, but this has evolved new requirements for enhanced capabilities within EPICS. Specifically, EPICS Channel Access (CA) clients need to dynamically specify the LANSCE macro pulse beam gate combinatorial (LANSCE Flavored Data), and the window in time (LANSCE Timed Data), to be sampled when they subscribe. EPICS upgrades fulfilling these requirements, including generic software interfaces accommodating site specific event queue payloads and client specified subscription update filtering expressions are being installed into the EPICS core software so that the needs of LANSCE can be accommodated consistent with the EPICS project and device portable model.

These upgrades will greatly improve the versatility of the original EPICS system expanding its intersection into the domain of data acquisition systems[13].

BASIC APPLICATIONS

Basic applications for archiving, alarming and display have been part of the EPICS framework from the very beginning. Several of these went through various kinds of development stages. New versions of the same kind and/ or completely new functionalities were implemented by different collaboration members.

Archiving

Archiving is a core control system functionality. Several stages of development resulted in the required performance for data storage and retrieval of recent history, using either custom data files or highly specialized relation database setups. Long-term retrieval and maintainability often suffered, and new approaches at TJNAF and ORNL now re-evaluate the more conventional use of relational databases.

Archive Displays

The 'DataBrowser' from SNS is integrated into the CSS operator cockpit and based on a common archive interface. This generic approach to visualize archived data together with real-time data is very promising.

Alarming

Record processing on the IOC includes alarm handling as part of the core functionality. All EPICS records calculate alarm settings whenever they get processed. This rich feature is still not used adequately in today's EPICS installations. The standard alarm handler (alh) only handles those alarms which have been configured in its individual configuration file. This way alarms are only monitored if the fields in the record are set to trigger alarms – and – if a client application (typically alh) is monitoring this channel.

The approach to push any triggered alarm from the IOC to the outside is already in operation for long time at the D0 experiment. A generic implementation based on Java Message Services (JMS) is now also available [5]. As one of the basic features it makes sure that any alarm being pushed from an IOC will get written to a relational database. In addition a new set of alarm views is available as part of CSS – from the CSS update site [6]. This includes new tools to filter alarms and to trigger actions like GSM messages.

OPERATOR COCKPIT

Key EPICS operator interface tools are limited to X11/Motif, only few were ported to MS-Windows. Other problems include:

- Different look and feel
- Lack of data transport between tools
- Program design and dwindling number of programmers familiar with the older technologies make changes or extensions impossible.

Control System Studio (CSS)

CSS is meant to be the answer for future operator applications. The development of CSS started in 2005 with a workshop at DESY. Meanwhile version 1.0.0 of the CSS-core been released. The CSS idea is well accepted by a growing group of developers. It's already in production at SNS (DataBrowser) and at DESY (alarm-displays and – configuration). CSS is not limited to user interface applications, as exemplified by headless Java applications that use the CSS core management facilities.

CSS features in short [7]:

- Built on Eclipse which itself is built on OSGi
- Provides core functions like: Logging, XMPP based management, common data types for data exchange and drag and drop, common interfaces to control systems DAL (Data Access Layer) and to archive data AAL (Archive Access Layer).
- Easy extensible by Eclipse compliant plugins.

CSS-Synoptic Display Studio (SDS)

The core components for operator interactions with the controlled process are synoptic displays. All of the existing display applications are written in 'C'/ C++. They are using Motif for their graphics.

The SDS is designed to run in the CSS context as an Eclipse plugin. The core graphic technology is GEF – the graphical editing framework. This framework provides a rich set of functionalities. The most prominent advantage in using GEF is the extensibility to create and add new graphical widgets in the form of Eclipse plugins.

A generic conversion tool has been created to convert existing configuration files into SDS readable XML files. This way the investment into the actual synoptic tool can be saved. SDS is available in version 0.9.9 and about to get ready to go into production.

Web Based Applications

Besides the CSS developments for a rich client platform application (RCP), there are also developments going on to improve the support for Web based applications. Web applications gain importance for remote process controls. Future will tell how Web applications and RCP applications will be used in the EPICS community.

TOOL KITS

MMLT

Matlab Middle Layer Toolkit is a set of Matlab applications for the commissioning and beam studies of 3rd generation lights sources. Developed at LBL and in use at Spear, Diamnod,CLS, and ASP, this toolkit provides an extensible platform for physics applications.

XAL

XAL is a Java framework for developing accelerator physics applications for the commissioning and operation of the Spallation Neutron Source at Oak Ridge. It was designed to be extensible and has evolved to support ongoing accelerator operations. XAL has a growing user base. Being written in Java and configurable by SMF (XAL standard machine format) configuration files, XAL provides an open well accepted standard for accelerator physics applications.[8]

SEAL

The SLAC Eclipse Accelerator Lab (SEAL) [9] is a new development for the LCLS at SLAC. As already documented in the name, SEAL is based on Eclipse. Several XAL applications will be integrated as plugins into SEAL. General purpose (EPICS) tools will be integrated by CSS.

RDB-Based Database Generation

There are two different – but not conflicting – methods to handle EPICS configuration databases: The descriptive way and the prescriptive way. Two prominent examples for these approaches are: EPICS-ORA [10] and IRMIS [11].

RDB-based, or at least RDB supported creation and management of EPICS configuration databases is an essential item on the wish list of many senior control system managers. So far only the IRMIS approach has found its place in many EPICS installations. Creating configuration databases from a RDB seems to be much harder then managing these with a RDB.

THE COLLABORATION

A recent poll in the EPICS community has clearly identified the collaboration itself to be one good reason to start and continue to use EPICS. Close contact to the developers and a fast responsive mailing list are essential element of the success story. 'There's no free beer, but always help'. This might be a good description. 'Once I have solved my problems, I will share my solution to help others'… might be another way to describe the work in the collaboration. The whole collaboration works on a voluntary basis. This is undoubtedly the most difficult fact to describe to commercial partners.

INDUSTRY

EPICS has reached the state of a de facto standard for machine controls. Companies provide EPICS drivers for their hardware or design hardware for EPICS applications. The EPICS core has been ported to several platforms by industry. Turn key systems with EPICS support have been purchased in several places.

Companies were formed by former members of the EPICS community. This way it is possible to purchase professional EPICS support.

WISHLIST FOR FUTURE RELEASES

In a limited poll about missing features – or wishes for future releases – the following topics seem to be of general interest:

- Documentation: A 'real' beginners document. An updated record reference doc.
- Code examples for drivers, records, etc.
- Better information within the collaboration about new developments.
- Better support for RTEMS. E.g. a VxWorks to RTEMS converter.
- Dynamic (online) record addition in IOC databases.
- Better VDCT support.
- Better time support
- New display manager: Should be not Motif based; should be able to read all existing configuration file formats.
- Redundancy for IOCs
- Support to store/handle alarms in RDBs.
- Better/ more support to create IOC databases from RDBs.

From this list one can draw the conclusion that some tasks never get done right for all users – like documentation. Some things will stay forever on the list – like better information about new developments or online changes, and some things really get done – like the things in the second part of the list.

OUTLOOK

EPICS has reached a very mature state. This is the keystone for applications like redundant IOCs and IOC core software running on FPGA PPC cores. Continuous improvements keep the code up to date. New developments like the Java IOC and CSS show that maturity and new innovations can coexist. Support and experience in industry is still growing. The EPICS collaboration itself is the most valuable part of the EPICS toolkit.

REFERENCES

[1] D. Thompson, S. Peng,, K. Kasemir, "General Time Proposal",(http://www-srl.slac.stanford.edu/lcls/epics/Talks/ThurAM/PengGeneralTIme.ppt)

[2] Event System by Micro-Research Finland, http://www.mrf.fi

[3] Event Time Driver for General Time, B. Kalantari, EPICS Meeting '07, (http://epics.desy.de/content/e2/e127/e183/e176/EPICS-agenda-2007-11.htm)

[4] Redundancy for EPICS IOCs, M. Clausen *et al.*, this conference, MOPA03

[5] Alarm package load page

[6] CSS Update site at DESY: http://eclipse.desy.de/update

[7] Control System Studio (*CSS*), Jan Hatje *et al.*, this conference MOPB03

[8] Thomas Pelaia, "*XAL* Status", this conference, MOPB02

[9] P. Chu *et al.*, "High Level Application Framework for LCLS", this conference TPPA13

[10] Chr. Gerke *et al.*,"Process Control: Object-Oriented Model for Offline Data", this conference, RPPA12

[11] Don Dohan, "The *IRMIS* Universal Component-Type Model", this conference, TOPA03

[12] Artem Kazakov, "Redundant EPICS IOC in PC-based Unix-like Environment", this conference, TPPA31

[13] EPICS CA Enhancements for LANSCE Timed and Flavored Data, J. Hill, this conference, WPPA24