
Diamond synApps workshops

Nick Rees



Motivation

- **synApps workshops came about for a number of reasons**
 - **Need to bring Diamond beamlines team up to speed – none of the DLS beamline controls team have had synchrotron experience and very few had motion control and EPICS experience.**
 - **Discussions at the last EPICS meeting about beamline collaborations**
 - **Marty trying to sell synApps to me**
 - **Collaboration with the APS on motor record**

Attendees

- **Speakers:**
 - Tim Mooney (APS)
 - Mark Rivers (APS)
- **External attendees:**
 - Andreas Balzer (BESSY)
 - Stefan Heim (BESSY)
 - Axel Pohl (BESSY)
 - Markus Willimann (BESSY)
 - David Maden (SLS)
 - Jud Gaudenz (SLS)
 - Juraj Krempasky (SLS)
 - Dennis Armstrong (SLS)
 - Marcel Grunder (SLS)
 - Rok Sabjan (CosyLab)
 - Darrell Nineham (Hytec)
 - Andy Foster (Obs Sci)
- **Diamond Beamline Controls:**
 - Nick Rees
 - Tom Cobb
 - Peter Denison
 - Pete Leicester
 - Ulrik Pedersen
 - Emma Shepherd
 - Malcolm Walters
 - Keith Baker
 - Steve Singleton
- **Diamond Data Acquisition Group:**
 - Fajin Yuan
 - Vasanthi Nagalingam
 - Eric Ren
 - Stuart Robertson
 - Matthew Pearson

Agenda

- **Workshop 1: 5-8 December 2005 (Tim Mooney)**
 - Introduction to synApps, building a beamline
 - Motion control, motor record
 - Command/response, caPutCallback
 - Discussion on how to improve collaborative activities
- **Workshop 2: 20-22 February 2006 (Mark Rivers)**
 - asynDriver
 - Detectors, MCA and Scalar record
 - IDL and Python interfaces to EPICS
- **Most important factors were:**
 - Workshop time spent actually working on problems
 - Discussion time trying to decide the ways forward

EPICS command/response

- **Tim spent a lot of time explaining this to us (thanks Tim!)**
- **Essential for high level control of processes like motion control...**
- **caPutCallback is the EPICS name for command/response 😊**
- **Amazing how much work has been done on this (thanks Tim and Marty), but how poorly it is understood (or maybe the unexpected change in behaviour has broken things). For example:**
 - **Problems with gateway.**
 - **Problems with CAJ.**

asynDriver

- **This is a huge step forward**
- **The concept of a port that that can be addressed by multiple drivers is very flexible.**
- **For example, the Delta Tau low-level asyn port can be used by:**
 - **Generic asynMotor support (for the motor record).**
 - **streamDevice 2.0 (for any arbitrary Delta Tau controller specific function).**
 - **the asynRecord (for interactive command line I/O).**
- **But it is a fairly complex framework**
 - **In some cases we will might end up with something as complex as the database underneath the device layer.**

Collaborative development

- **Hurdles**
 - **Sharing immediate goals** What is the most important thing for me to do (or fund) right now?
 - **Sharing source code repositories**
 - **Licenses**
 - **Different build systems, naming, opi's and colours, client tools**
 - **Quality controls – avoiding repository corruption**
- **Ideas**
 - **Use Sourceforge or similar.**
 - There is now an EPICS project in Sourceforge, and a small amount of code in its subversion repository
 - <http://sourceforge.net/projects/epics/>
 - **Use an OSI licence or get EPICS license OSI certified?**
 - **Try and work towards reducing system conflicts. If not possible, we may have to resort to duplication.**

Developer enhancements - debugging

- **EPICS regression testing**
 - Ability to type 'make test'
 - Check that everything compiles cleanly and the IOC boots properly (although there might be cross compilation problems)
 - Simple functional tests e.g. do a 'caput' and check results against expected using Perl or Python test harnesses.
- **Runtime record query, display and manipulation**
 - Select a record and show all records in the lockset
 - Show values of all the fields
 - Auto save all the fields
- **Graphical database debugger**
 - Set breakpoints, continue
 - Step mode debugger through a lockset

Developer enhancements - VDCT

- **Colour coding/line format distinctions for different link types**
- **Context sensitive help in VDCT**
 - At it simplest - hyperlink to the record reference manual
- **‘On the fly’ link editing with VDCT**
- **Integrating developer tools**
 - switch between graphical and text mode for db edits,
 - marking a record for archiving, setting its alarm group, reboot persistence etc
 - Extensible file format
- **Integrating makeBaseApp functionality into an IDE (Eclipse)**
 - Ken seems to have done this already.
- **Build a library of design patterns in the form of templates that implement specific algorithms.**

Display managers

- **EPICS has too many display managers and not much difference between them.**
- **If we are going to do anything, we should make an existing GUI builder EPICS aware**
 - **Maximum gain for very little pain.**
 - **Much improved GUI editor**
 - **Can be configuration file driven, but simple to extend.**
- **Should be a command line option to make GUI read only.**
- **Dynamically set-able limits**
- **Allow greyed out widgets and fields.**
- **File chooser with long file names.**

Summary

- **Beamlines are where most of the new user requirements are now coming from**
- **We need to address the needs of this community, and leverage the benefits of the collaboration to generate common solutions.**
- **We need to regularly (every EPICS meeting?) draw the requirements together and try and determine the way forward.**
- **The requirements are both in traditional EPICS low level area, and also in high level integration.**
- **synApps is just the start – it is up to us to take it forward.**