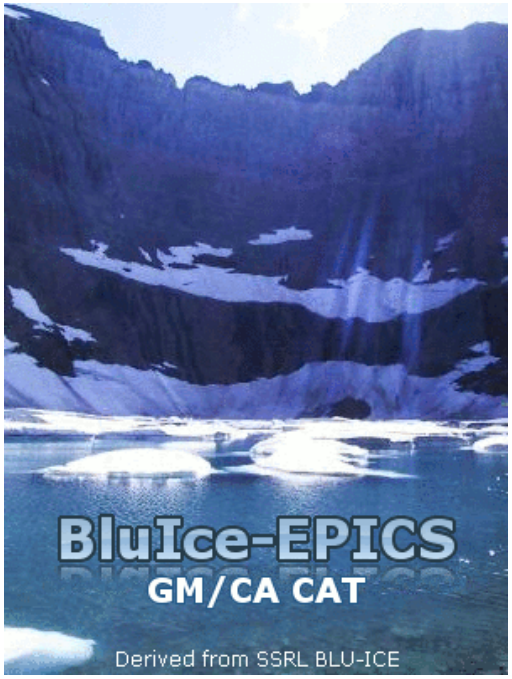
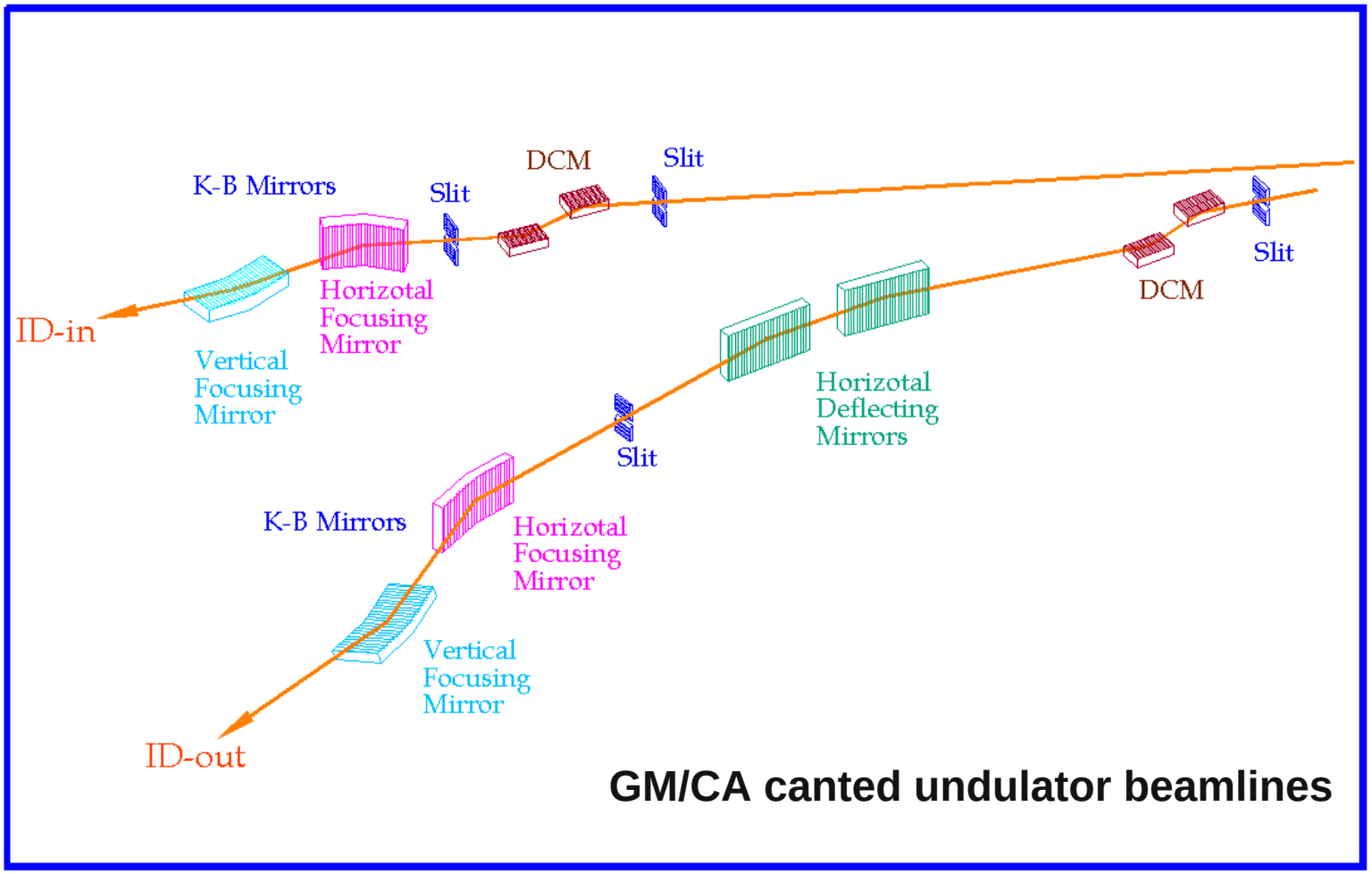


Architecture of BluIce-EPICS data collection software for macromolecular crystallography

Sergey Stepanov

GM/CA CAT at the Advanced Photon Source,
Biosciences Division of
Argonne National Laboratory, USA

GM/CA CAT: Dedicated sector for macromolecular crystallography at APS

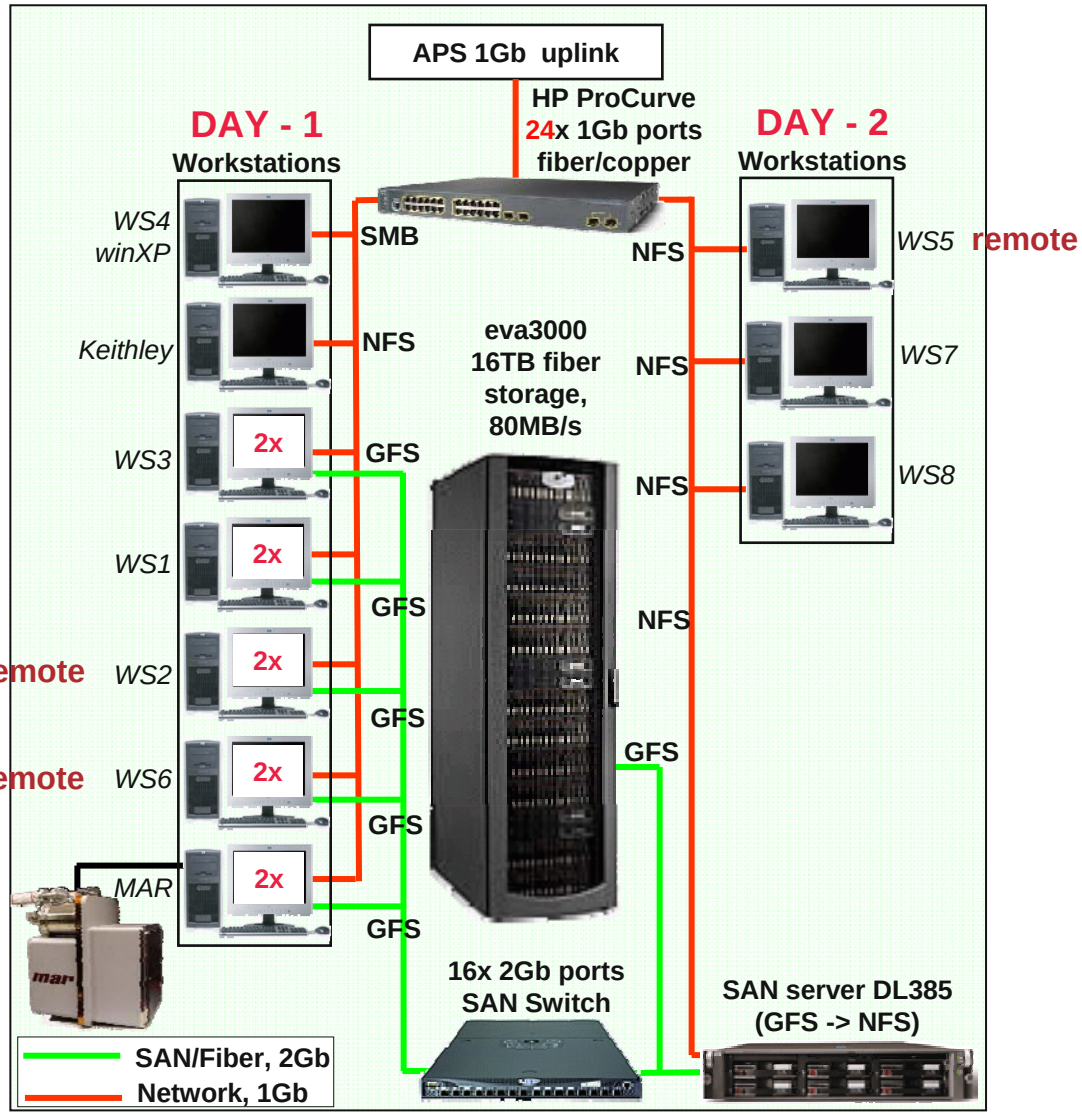


GM/CA canted undulator beamlines

A lot of optics to control !

GM/CA CAT layout: two canted-undulator ID beamlines and one BM beamline at APS Sector 23, all dedicated to macromolecular crystallography.

GM/CA CAT computing environment



All three beamlines have similar computing infrastructure

- Independent subnets, storage & accounts at each beamline
- 16TB fiber channel disk storage arrays at ID lines 8TB at BM
- 1Gb fiber network & 2Gb fiber SAN with global file system, GFS
- Same password for Linux & Windows using LDAP
- All computer see the same home directory
- Support for eSata, firewire and USB2.
- Three computers per beamline open for remote access.

What is BluIce-EPICS?

GM/CA CAT offers a graphical user interface application named **BluIce-EPICS** for collecting crystallographic data at our beamlines.

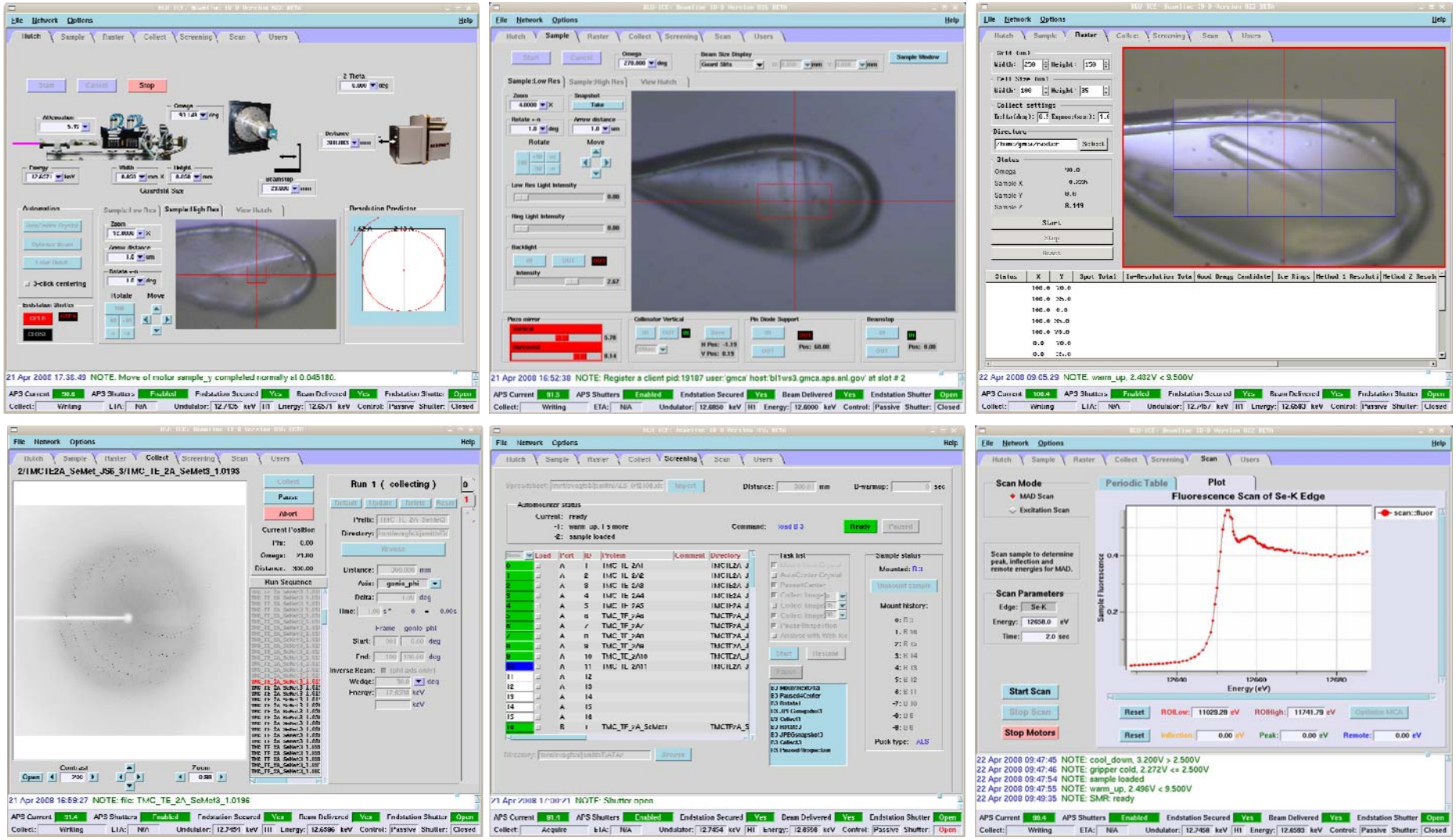
Here name **BluIce** stands for Beam Line Universal Integrated Configuration Environment and descends from a control system of the SSRL SMB beamlines and **EPICS** stands for Experimental Physics and Industrial Control System that is a toolkit for distributed controls widely used at the APS and many other synchrotrons worldwide. Next pages will show how these two are linked.

The BluIce-EPICS application is a top layer of our distributed control system and it is aimed to be the only application needed by users, although staff has access to a variety of advanced controls that can be launched in parallel with BluIce-EPICS.

The application is executed under Linux OS. In our environment users have access to multiple Linux computers that all see the same home directory.

Multiple copies of BluIce-EPICS can run in parallel on them, but only one instance at a time may be a Master, i.e. have rights to control the hardware.

BluIce-EPICS is a multi-tab GUI: core tabs correspond to typical tasks



Older tabs are in Tcl/Tk, newer tabs are in Java. The plan is to convert all tabs into Java.

Why and how our software is different from SSRL BluIce?

BluIce-EPICS user interface was directly derived from the SSRL BluIce code, but the underlying architecture was completely changed. We had to do that because of several reasons, but all of them were related to the goal of making the control system **fast** and **flexible in adding features**:

-- BluIce did not have drivers for the electronics we planned for our beamlines; so we had to use EPICS framework.

-- BluIce did not expect concurrent controls that are unavoidable in the EPICS world were many applications can talk to the same electronics in parallel.

-- BluIce commands were too high-level and generic (e.g. “move motor to XXX”) while we wanted to have direct access to the advanced features of beamline controllers. For example, we wanted to implement on-the-fly scanning where motor speed needs to be changed and motor position needs to be synchronized with switching data points.

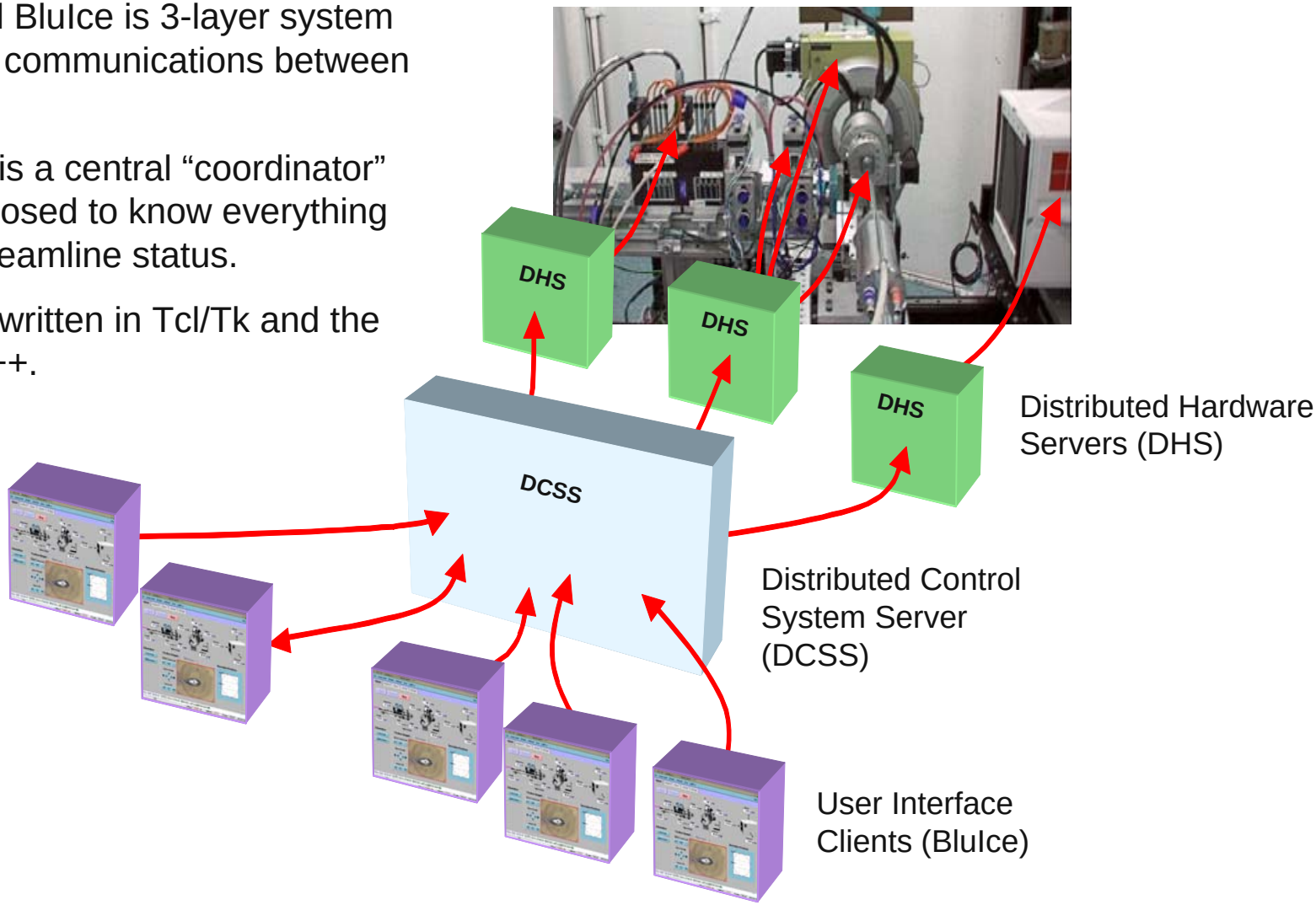
To make controls fast & flexible we converted BluIce from 3-layer system into a single layer EPICS application. Multiple instances of BluIce talk to each other also by means of EPICS mechanisms.

Architecture of the original SSRL BluIce

The original BluIce is 3-layer system with socket communications between the layers.

The DCSS is a central “coordinator” that is supposed to know everything about the beamline status.

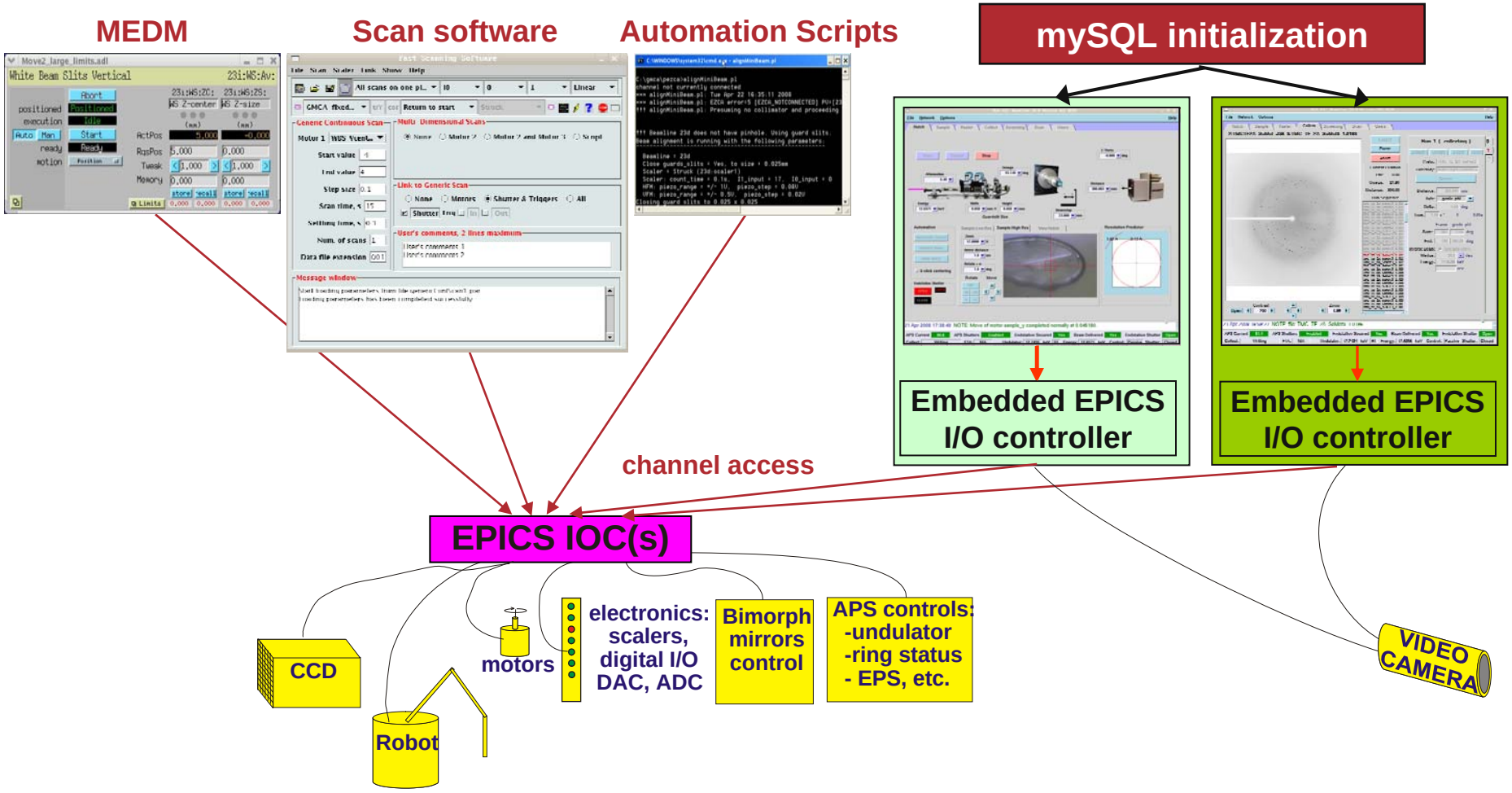
The GUI is written in Tcl/Tk and the rest in C/C++.



Architecture of BluIce-EPICS

Staff beamline automation tasks

BluIce-EPICS for beamline users

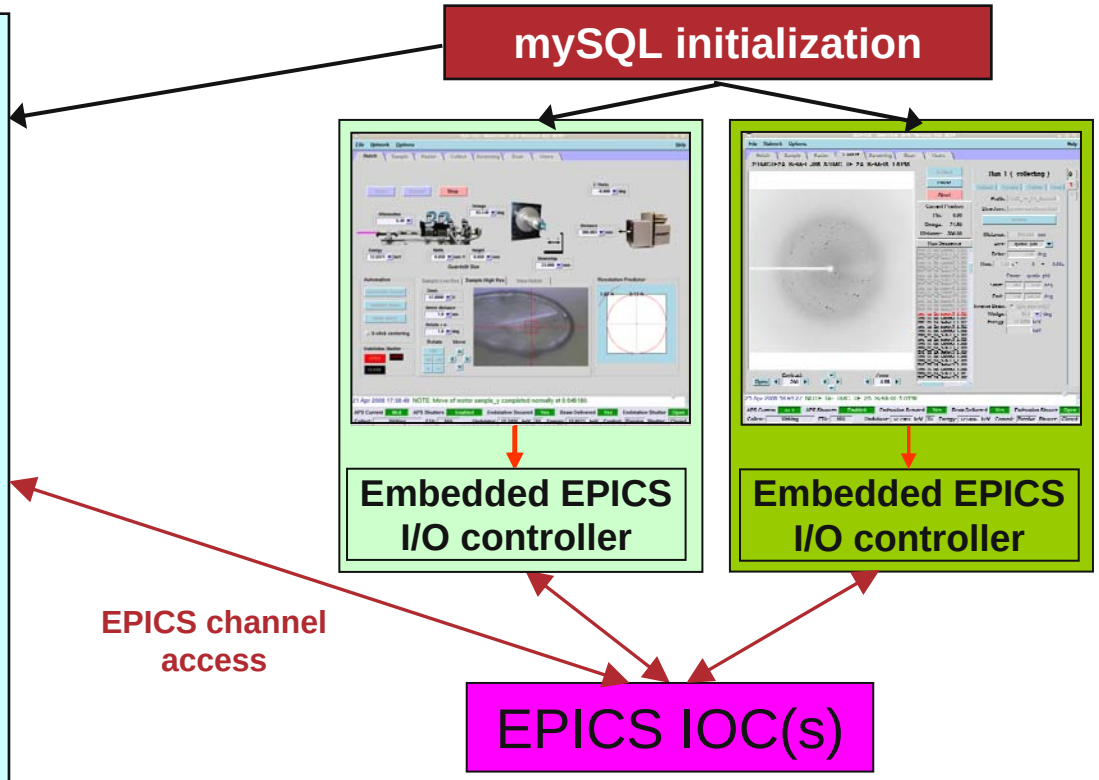


Each BluIce is a single-layer application talking directly to EPICS along with other controls

BluIce-EPICS: outsourcing complex controls to helpers

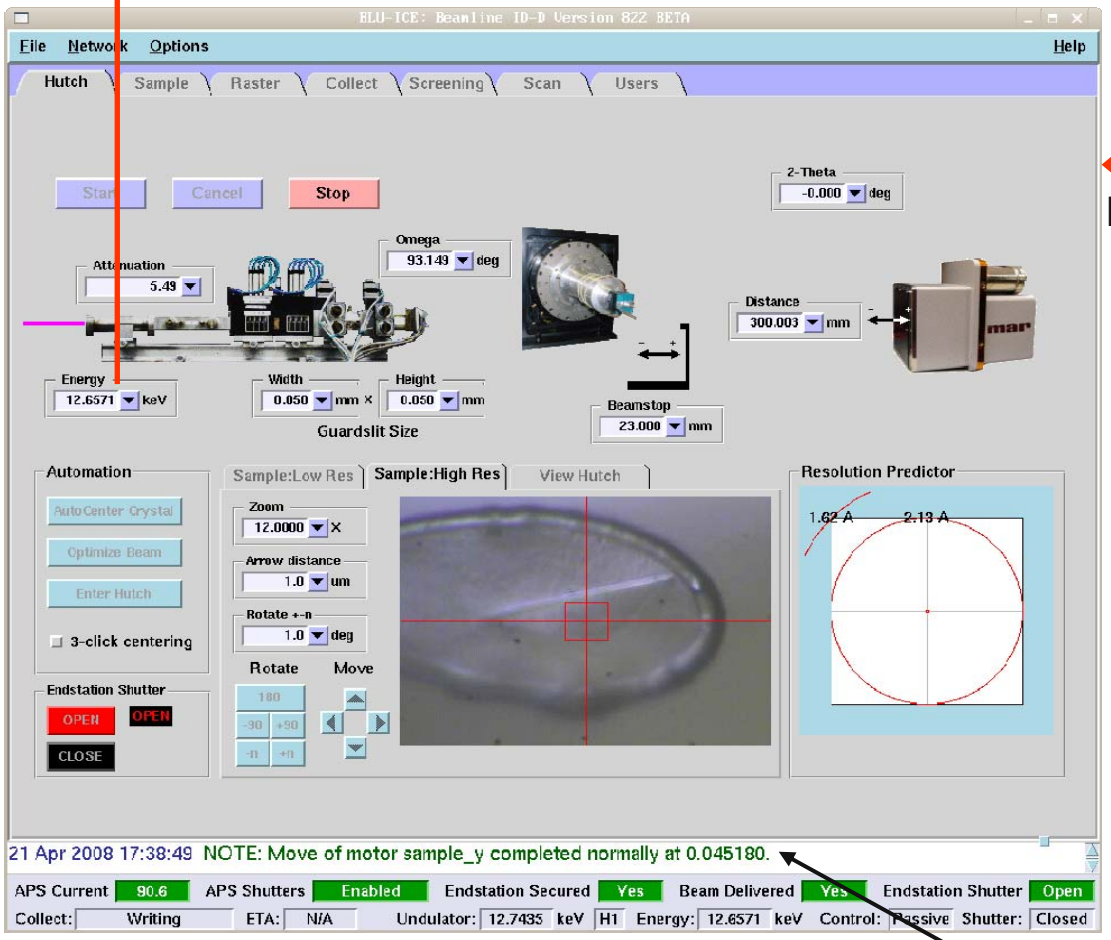
Rack of helpers

- Frame Server (FPE)
- Robot Server
- Point & Click Server
- Keithley Gain Server
- Frame audit
- BL Feedback Server
- Big energy move
- Minibeam alignment
- Sample Autocentering
- On-the-fly Edge Scans
- On-fly Fluor.Rastering
- Fluor.signal tune
- Collimator In/Out



- Helpers include servers and external scripts that interact with BluIce via EPICS PVs; some also share mySQL initialization.
- Each helper is an independent module that can be written in any language without knowledge of the main BluIce application.
- They widen base of developers: use SNL, C, Java, Perl, Tck/Tk…

How BluIce-EPICS helpers work: big energy moves



BluIce: is $|\Delta\theta_B| > 1°$?

No → Drive energy directly in BI

Yes → Call external script

BigEnergyMove script:

Pauses BL intensity feedback server

Changes undulator harmonic and offset, if needed

Changes mirrors lanes, if needed

Changes filters to preserve beam attenuation

Re-centers the beam & restarts intensity feedback server

Reports to BluIce log window & pops-up dialog messages if needed.

BluIce-EPICS software team and software development practices

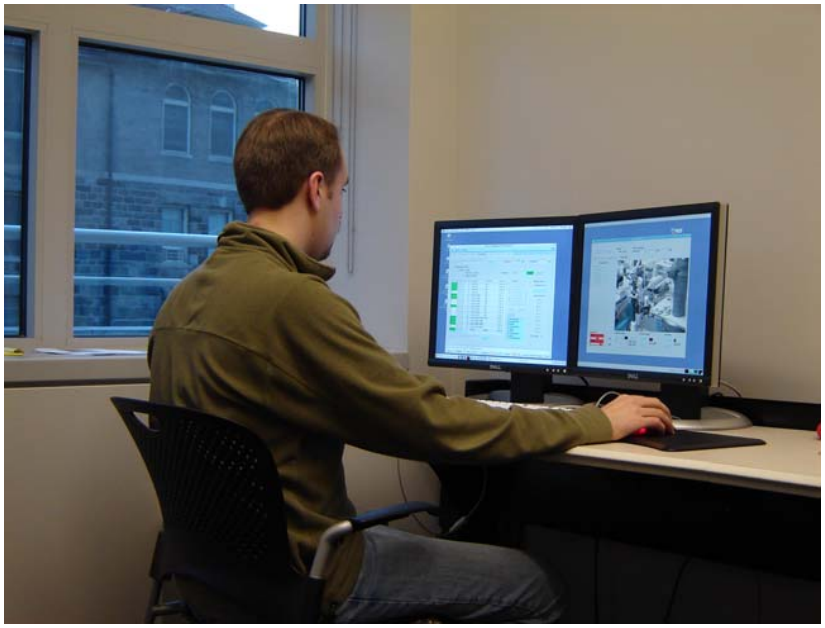
The project started in 2003. Delivering BluIce-EPICS application in tight timeframe and combining development with maintaining high software availability at fully booked beamlines has been tough task!

GM/CA software development team: Mark Hilgart, Oleg Makarov, Sudhir Pothineni, Sergey Stepanov (+ Satish Devarapalli & Alex Urakhchin in the past).

Development practices:

-- collecting suggestions from GM/CA CAT staff crystallographers and end-of-run reports (yes, we do read and analyze your feedback very attentively!)
-- assigning priorities based on occurrences and after weekly group meetings
-- bugs list queue and new features queue
-- typically a week of testing by staff crystallographers before releasing to users
-- BM beamline as test bed and big changes on shutdowns only
-- version control with Subversion software
-- staff documentation on GM/CA group Wiki and video guides for users
-- open source: snapshots at http://www.gmca.anl.gov/bluice-epics/

Remote beamline controls with BluIce-EPICS



Todd Geders using remote controls at U. Michigan

- Same NOMACHINE technology as at many other facilities
- Three computers open per beamline (one for data collection, one for processing & one for data transfer)
- Extra controls for remote operation in BluIce (see snapshot at the bottom of the page)

http://www.gmca.anl.gov

Compliance with Argonne's enhanced security requirements: access is restricted not only to the time of an experiment and specified user name, but also to a list of IP domains from which users work.