

Diamond Light Source GUI Strategies



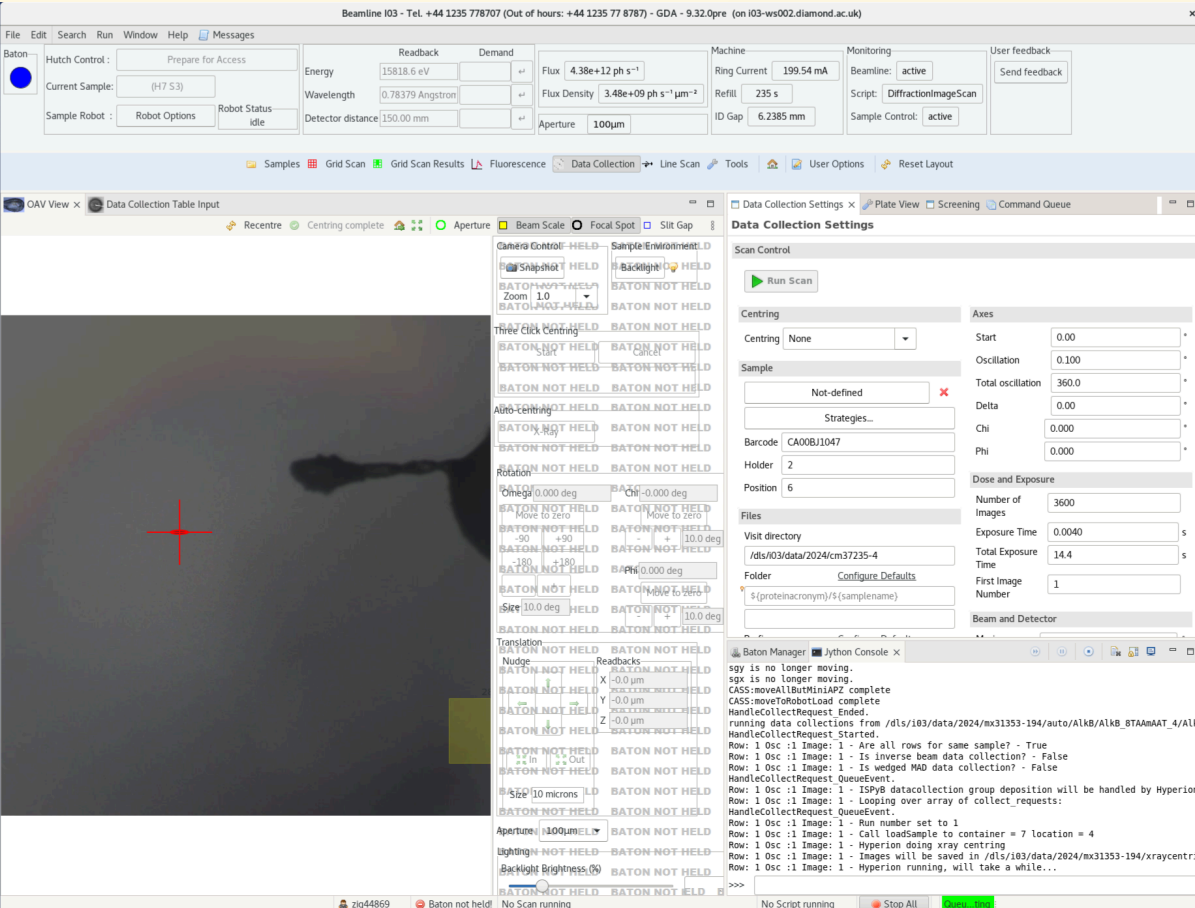
Intro

- Me: MX DAQ perspective - gathered some info from other groups
- DLS has been running for about 25 years with organically developed software
- Upgrading the synchrotron -> upgrading the software:
 - Machine GUI
 - DAQ GUI
 - Analysis / SIMS



Reusing legacy UIs for new features

- Many existing GUIs will take a long time to replace
- Buttons from existing Java SWT GUI hooked up to new DAQ routines
- Feedback from EPICS is already integrated, but feedback from the experiment requires more work

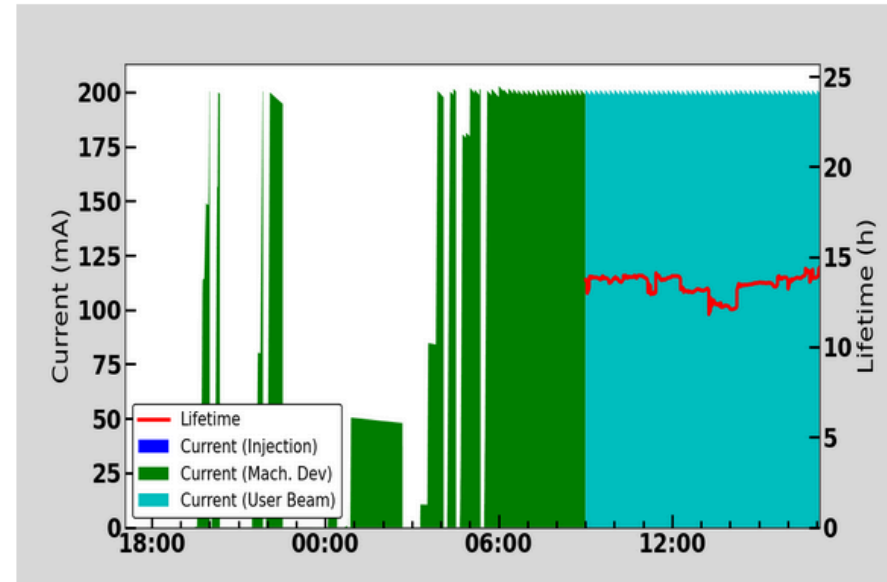




Web GUIs

- Have been exploring technologies for the last year or two
- Started with machine GUI
 - Developing a web gui but in parallel to using Phoebus
 - The same specification file can generate both interfaces

Storage Ring Status (day view)



Lifetime (hours)

14.38

Current (mA)

199.72

Last Fill

18 Sep 05:36

F/back Status

Fast FB On

Mode

User

Top-up Status

Running

Fill Pattern

900 Bunches

Operator

Asim Younus

Coordinator

Doug Scott

Messages from operations

User Beam time

200mA , 900 bunches



Web GUIs - Standards

- Settled on React + Material UI
- Taking some time to define best practices for our projects (UX/accessibility, security, code quality)
- We will implement a template project (like our python copier template) which comes with linting, CI, publishing, auth (?) to lighten the burden
- Work has begun on a reusable component library to get a consistent look and feel



Experiment control

- DAQ core team:
 - Strategy is API-led: can develop both local and web GUIs
 - Interested in autogenerating forms from JSON/GraphQL schemas to give initial GUIs, which can be manually tweaked with feedback from beamline staff
- MX team:
 - Starting by reimplementing some GUIs that scientists have made by hand in EDM



Issues we have faced so far

- Performance of many open subscriptions to PVs is an issue
 - Now looking at PVWS (<https://github.com/ornl-epics/pvws>) for this
- Few if any of us have web dev experience - React is new syntax, there are new best practices to learn
- Dependency management is a pain, but not a dealbreaker

Takeaways we would like from this meeting

- To see what others are doing in case we have missed something important!
- Some collated questions from people at DLS:
 - What tools are other organisations using to develop applications with separate concerns, like different science areas. Are they going for a more generic approach, or using technologies such as federated components to allow different departments to build their own web pages, that'll then get included in a larger app?
 - Does anyone already have web UIs for experiment control in production, and what hurdles did they overcome for that?
 - Is there interest in a collaboration to share components and viewing tools? Such as viewers for MRC files, tomograms, sample management...
 - What is a good forum for collaboration on UIs? For the Bluesky collaboration we have a Mattermost instance with members from many facilities. Could e.g. lightsources.org host something similar for more generic software collaboration?