

MXCuBEDevelopers' meeting

June 10, 2022

Participants :

Marcus Oscarsson, Olof Svensson, Antonia Beteva(ESRF)
Rasmus Fogh (Global Phasing)
Michael Hellmig (HZB)
Bo Yi (NSRCC)
Jacob Oldfield, Nicolas L (Australian Synchrotron)
Merghdad Yazdi (MAXIV)
Roeland de Boer (ALBA)
Lelicia, Francisco (LNLS)

Apologies: Martin Savko (SOLEIL)

Feedback on half-yearly meeting

Positively received, few comments. JO was happy to see MXCuBE4 work.

AbstractDetector and units for beam position

The units (in this and other cases such as energy) must fit across multiple external programs. Most modern programs use pixels for beam positions, but the use of mm goes back to MOSFILM, which is why ISPyB uses this unit. Image headers use pixels. One reason for using pixels is that it is a natural unit for pixel detectors, and that detector sizes can be given in integer pixels. Any detector that did not use module gaps that were an integer multiple of the pixel size would break the current model and require specification of each panel separately.

Various possibilities were discussed: Specifying the unit as part of the class (con: bad for uniformity); Using a unit library (con: a bit cumbersome for a situation where we are only dealing with program-internal data transfer); specifying alternative functions with `_mm` and pixel suffixes as wrapper for a function that took a unit input parameter (con: duplication). In the end it was agreed to stick to the current definitions in `AbstractDetector`, to with using pixels for width, height, and beam position, and mm for radius and `outer_radius`. If desired, people would be free to propose a change later.

Feedback from working groups

Little to report. RF reported good progress in implementing the GPhL workflow on MASSIF1, being called from the MXPRESS workflow.

AB reports that the AbstractDiffractometer pull request should be essentially complete, and a merge would be proposed soon.

Topics for next code camps

The discussion started with asking new members for their proposals.

JO reports that the Australian Synchrotron is very much at the integration stage, using a hacksaw to match MXCuBE with existing code and libraries. JO is very interested in using Bluesky for writing workflows.

This combined with other points into a wide-ranging discussion for a '**workflows and procedures**' code camp topic. There were mentions of how Bluesky would overlap with the HardwareObject layer, of considering the celery based processing Hardware Object, of making a native workflow system based on the ESRF EWOCSS library, of using celery as a replacement for XMLRPCServer and Py4J), of how this work would fit with the AbstractProcedures work already done, of interfacing with Ophid, and on establishing clear divisions and interfaces to allow for the use of (many) new and external libraries.

Two specific points:

- On changing over to celery the meeting is waiting for some promised prototype work from Peter Keller.
- On Integration with Bluesky, JO proposed that he could organise a demo on the Australian Synchrotron work on using Bluesky and generally how they are integrating with MXCuBE – which was very positively received.

Testing and integration were discussed as another possible topic for a code camp. OS raised the problem that errors tend to be swallowed up, so that debugging them becomes very complicated. JO proposed using Black as a pre-commit action. The project is in principle already following Black, but execution is not automatic. MO pointed out that the automatic Continuous Integration code that had been used previously had fallen into disuse, in part because of licensing problems, and that a renewed effort in CI and writing more tests would be desirable.

The possibility of releases on Kybernetes, v. Docker, v. Podman were discussed, and there was interest in considering a Kybernetes release. JO currently has a docker release, which is about 6 months behind the develop branch. LNLS (Brazil) also uses Docker. There was interest in using Podman instead, though this would lose the use of Docker Compose.

Next meeting

The next meeting will be around July 21 – MO to make a Doodle poll. So far it has not been possible to select a fixed date / weekday for all future meetings.

On the topic of code camps there seemed to be interest in having one in 2-3 months, probably virtual, in part as preparation for an in-person code camp in connection with the December half-yearly meeting. The format was to be determined, possibly as two half days, seeing that it was becoming ever more important to facilitate participation from all time zones.