MXCuBE Developer’s Meeting

Whereby, March 2, 2023

Meeting Minutes

Participants

Rasmus Fogh (Global Phasing)

Marcus Oscarsson, Daniele de Sanctis (ESRF)

Annie Heraux (Elettra)

Michael Hellmig (HZB)

Andrey Gruzinov (DESY)

Jacob Oldfield, David Eriksson (ANSTO)

Bo Yi (NSRRC)

WenMing Qui (HEPS)

Agenda

- Status reports

Versioning and release procedures

JO (ANSTO) is still working on testing and writing test cases. It will anyway be a while before there is beam.

AG (DESY) says nothing has changed. He is working on synchronising their local version with the development tip, but the last time was in 220201. The task for now is ‘get something running first’, and synchronise afterwards. AB, MO, and RF all recommend pushing WIP commits as you go and making PRs for finished parts. This will keep others informed, and share useful code, even for site-specific parts.

MH is working on integrating the ESRF gitlab branch. The plan is to adopt the web version for at least one beamline after the summer shutdown, and then move to the github version.

MO notes that with the latest PR the gitlab and github versions should be pretty much identical.

MH finds it hard to move from the Qt to the web interface. MO suggests a practical workshop and proposes it would be an in-person meeting, if possible. Possibly in connection with the May MXCuBE meeting – or even earlier MH seconds the idea.

Bo Yi is working on integrating EPICS. JO Notes that there are abstract classes for this purpose available from LNLS.

Wen Ming is glad to hear that ANSTO is also using EPICS. He has three bamlines and is trying to use Pilatus Lima Tango, but could not get access to the whole data set. Now trying to move to EPICS. Here he can collect data, but has problems accessing the (current) last image. Uses Exporter on microdiff. WM is synchronising to the mxcubecore branch, mostly testing at the moment.

AB notes that you need to choose / deal with the Eiger stream vs filewriter API. There is a REST API to the detector. JO has wrapped detector access in Ophyd object and Soleil is testing the use of the REST API.

MO reports that the ESRF is about to deploy mxcubecore on an additional beamline at the ESRF, possibly ID30B. Staff shortages are likely to get worse now, and it may well be necessary to prioritise the beamlines at first.

AH: Nothing new to report. Elettra is hiring a new person (Alesasandro Olippo, for end March) to work on ISPyB, but also on MXCUBE. AH wants to put Braggy on the production sites ASAP; AG notes that Braggy needs a lot of things set up to make it work; it can be used with Qt. There is a lack of a ‘follow mode’, so you may need to explicitly call for displaying every nth image. MO suggests one might use ADXV, even if this is less powerful than Albula.

2. Roadmap

MO proposes that we should make a release (i.e. synchronise and copy develop to master) for the May meeting.

Possible targets:

- Code quality (MO)
- Queue organisation with/without data classes (RF)
- Standardisation of signals (MO, RF seconds).

JO has a PR with initial base class test code.

It may soon be time to turn on more features of the black code checker – but not quite yet, and we should plan for choosing the right time. .

3. Yaml v. XML

RF sums up the basic difference: The XML configuration allows adding new keywords in configuration files and accessing them from hardware objects. This gives a great flexibility, but makes it impossible to ensure or know what configuration options are available or how they should be accessed across beamlines, The yaml loader deliberately insisted that all configuration options must be defined in the HardwareObject code, making the opposite trade-off to sacrifice flexibility for predictability. The costs of this choice may not have been fully clear when the decision was made to accept the yaml loader, and the different philosophies make it very hard to work with a mixture of xml and yaml configuration, particularly since the change-over must be made an entire abstract class at a time. Now we are looking to move from XML to yaml across the board. We need to decide exactly how we want it to work.

Advantages of yaml are that it is a more modern system, less verbose, actually designed for configuration and data exchange (unlike XML), comes with a complete parser, and makes for a good opportunity to clean up some fairly old program internals. In general terms the meeting is in favour of moving over to yaml. JO would like to see an example of a yaml implementation of an abstract class. MO proposes that all HardwareObject-type attributes should be (forced to be) defined in the code, whereas data and parameters should be given in and data classes, defined in (and validated against) schemas, one schema per hardware object type. Apparently there is some mutual support between yaml and data classes, and schemas are extensible for subclasses. The proposal is nodded through, none dissenting.

AB mentions the possibility of using TOML instead of YAML – TOML seems to be more thoroughly integrated in Python. The possibility is left open. There is a possibility to start on this as soon as possible, preferably before the May MXCuBE meeting. RF volunteer to look at code to make phase-in XML-YAML easier (**ACTION: RF**).

Any Other Business

DE and JO would like to get rid of gevent and replace it with something more modern (e.g. asyncio). RF would welcome this development, but begs to remember that the result would have to work also with the Qt interface.

MO notes that we need to have a session on Epics/Ophyd at the next MXCuBE meeting

RF reports having a problem with the automatic code coverage test failing. JO is aware of a particular problem with the test container under Linux that has this effect and will look into it. **ACTION: JO**.

Next Meeting

MO to send out Doodle poll