

MXCuBE Developers meeting, 19/3/2018

DRAFT

Present (virtually): Jordi Andreu, Michael Hellmig, Gleb Bourenkov, Ivars Karpics, Roberto, Martin Savko, Marcus Oscarsson, Antonia Beteva, Pavel Palau, Mikel Eguiraun, Peter Keller, Rasmus Fogh, Gerard Bricogne.

1. Minutes of last meeting, and matters arising

The minutes were approved.

2. Permission to review and accept PRs

It was agreed that every site ought to have a person with repository write rights to merge and accept Pull Requests; those interested should ask to be given permission from e.g. Matias

The general code review situation was discussed. It was noted that the Qt4 version, with fewer highly active committers, found it harder to get merge approval from other groups, and that in general more participation in code review would be helpful. No changes to current policies were agreed, notably it is still not allowed to accept you own pull requests. If necessary, these policies can be discussed again on another occasion.

RF nagged people to stick to PEP-8 and other style guidelines, including avoiding ‘import *’, not modifying input parameters (collections and dictionaries) in functions, ensuring agreement in calling signature for overriding functions, The current policy was reconfirmed, to wit that changes in this direction should be made, but gradually as files had to be modified for other reasons.

3. MXCuBE refactoring

The current state of discussion was as reflected on Github issues. It was agreed that the interface functions that had been presented by MXCuBE3 were not sufficient as a starting point; more information was needed on the working of the different MXCuBE implementations. It is necessary to make a proper interface specification, to include a description of what each function ‘means’ and what context it should appear in, so that a back-end developer has enough information to implement each function locally. Currently we do not even know how many bricks are involved in implementing the various functionalities, or what they do. For e.g. a function like ‘mount sample’ would have to document the list of operations involved: lower detector cover, move detector out, actually mount sample, move detector back in, lift detector cover, ... To this end, each site should describe the operations that their MXCuBE implementation needs, and how they match the interface specification being developed.

Initially, it is proposed to start with the rough division of MXCuBE presented by GB in Issue #139:

1. Diffractometer
2. Sample changer
3. data collection
4. basic meta-devices : detector distance, energy, transmission
5. queue
6. ISPYB interface
7. XRF and energy scans
8. External exp. Control

The list may well need expansion later – MO proposed having separate treatment of characterisation, std data collection, mesh scans, helical scans, ...

ACTION: All groups should begin by making their own description of diffractometer (1) and sample changer (2) as understood and implemented by them. The format is free – even email can be used – but as much as possible contributions should be put in the github issues, where they are easier to track. ESRF and EMBL will start with their respective implementations, and other groups are invited to join in.

For the next meeting (early April) we want to have a first overview of the organisation and implementation of diffractometer and sample changer across different sites. This should give us a starting point for discussing what problems we need to deal with for the refactoring, how much resource is needed to provide a complete interface specification, and how the work could be organised and distributed.

4. Any Other Business

None

Next Meeting

Next meeting is in early April, and a Doodle poll has been issued. **ACTION:** RF to send out reminder.