

# MXCuBE Developers' meeting

November 29 2022

**DRAFT**

## **Participants :**

Marcus Oscarsson, Antonia Beteva, Jean-Baptiste Florial (ESRF)  
Gleb Bourenkov, Marina Nikolova (EMBL-HH)  
Jacob Oldfield, Daniel Eriksson (ANSTO)  
Rasmus Fogh (Global Phasing)  
Andrey Grusinov (DESY)  
Martin Savko (SOLEIL)  
Roberto Borghes (Elettra)  
Nicolas (LNLS)  
Bo Yi (NSRCC)  
Michael Hellmig (HZB)  
Baokongwen, Wenming Quin (SARI)

Minutes: Rasmus Fogh

## **Adapting to mxubecore**

MO presented the slide with questions to be answered in the Status Reports at the Grenoble MXCuBE meeting.

## **Update of Qt user interface and mxubecore upgrade**

The Qt-using branches have fallen far behind at the individual synchrotrons, at the same time as the main develop branch of mxubecore has moved ahead. Catching up is a problem. One particular aspect is that Qt4 is now unsupported, Qt5 is moving out of support, so it would be necessary to consider moving to Qt6.

MO recommends using

'git diff-shortstat local-master .. github-master' to compare versions.

It is queried who are using the newest mxcube3 develop in production. ESRF (who use the web interface) seem to be the only ones. DESY and ANSTO are also on mxcube3, but they do not have it in production yet (ANSTO does not yet have users). Both SOLEIL and EMBL-HH have a mxcube3 branch, but neither is using it in production. Gleb points out that it will need very thorough debugging, there are many deep bugs, and EMBL-HH no longer has any manpower. The deep bugs are typically in the Python layer, e.g., functions with different names in different locations. EMBL-HH is using Python 2.7 and Qt4 in production, with Python3 and Qt5 in the parallel mxcube3 version. EMBL-HH is forced to use their limited resources for an OS upgrade soon, in order to keep maintaining the Python-2.7-based version (alas). SOLEIL have a (non-production) mxcube3 version using Qt5; the upgrade was done starting from a mock installation and gradually changing over to real Hardware Objects, and took ca. two months. At Elettra, one beamline is on an old version of mxcube3, one on a newer mxcube3 (web) version. They, too did the upgrade starting with mockups and using conda. Elettra is facing a long shutdown in the first months of 2023 due to high electricity prices (which, as Gleb notes, would be a good opportunity for upgrading code). RB is the only developer at Elettra at the moment, and the upgrading is frozen, but the experience was 'not so scary'. MH is in a similar situation; production uses a very old pre-mxcube3 Qt version, and a web mxcube3 version is being developed in parallel.

The best road to upgrading is discussed. JO proposes starting with upgrading the hardware objects first, before upgrading the Qt. He also suggests that it should be looked at which abstract hardware objects are not widely used, followed by upgrading those that seem to be hard to adopt. AB points out that with the new standardised abstract classes (AbstractMotor and AbstractNState, particularly) fewer classes are needed. It is agreed that people ought to start and raise an issue, then proposed a partial (WIP) pull request as a way of getting assistance with problems.

AB notes that important information about how mxcube3 should work has not been written down and may not be clear to people who want to adopt it. A written migration route document might help. MO suggests that this could be written collectively as a FAQ by people as they encounter and solve problems. JO wonders if the problem is a lack of trust in the new version, and proposes that more comprehensive tests might reduce the number of bugs and improve trust. Gleb thinks that the main problem is a lack of resources and developer time. AB proposes that we should share configuration files, since their upgrade entails considerable work too, and the meeting agrees. MO thinks we should consider making a roadmap for the upgrade, possibly in conjunction with a code camp in spring to get things started, but the decision is postponed to further discussion at the Grenoble meeting. MO proposes a script to upgrade from xml to yaml configuration files, and

promises to make a start. ESRF has also started writing tests, as has ANSTO. Recent work on linting has reduced the number of changed files from 800 to 112 by disabling some warnings, moving closer the time where the linting can be switched on.

The ESRF is the only place using mxcube in production. They are working off the develop (not master branch), using internal gitlab repositories for the exact production versions. The last change to the master branch was in May, with the 1.0.0 release. New changes are coming into develop now, adding a new more flexible way of handling queue entries, with Pydantic input descriptions, These changes are backwards compatible. The new version is Python 3 only.

MO points out that the agreement had been to make half-yearly releases in the master branch. Gleb, however, opines that a six month release cycle is clearly too short, considering the in-depth testing that is necessary before adoption a branch in production. RF comments that the current situation is rather disconnected from the ideas behind the versioning system: The 'master' branch is not used anywhere, most sites are far behind it, and the 'develop' branch is close to ESRF-specific, much as the then master branch was used pretty much as the in-house branch of EMBL-HH. There is a risk of ending in a situation where one well-resourced site is moving ahead on its own, and the work of keeping different sites synchronised falls squarely on the less well resourced sites. One solution might be to officially support several branches, so that not all sites need to catch up to the tip. This would (JO notes) require additional resources. Projects that support multiple branches generally provides some of the support centrally. JO and MO consider if a scripted upgrade guide would help. Another solution would be to find money to pay for consultants. The idea is well received, but Gleb notes that even this requires local developer resources to incorporate, answer questions, and handle any bugs that turn up.

### Next Meeting

The next meeting will be at the Grenoble in-person MXCuBE meeting.