Background and suggestions for presentations and discussions at the first 2018 ISPyB-MXCuBE meeting

This document is a follow-up to the Proposed Agenda for the forthcoming ISPyB-MXCuBE meeting (30th January to 2nd February) circulated at the end of 2017. I realise that most of you know all the material I am going to survey much better than I do, but I am only outlining it here for the purpose of providing a context and a backdrop for presentations in Sessions 2 and 3 of each meeting as well as for the various discussions, be they plenary or within Committees.

The agenda templates proposed for both the ISPyB and the MXCuBE meetings put the examination of "scientific capabilities", present and future, at their centre, so I thought it would be useful to write down a common baseline to define the scope of that expression.

Background for Sessions 2: Evolving scientific capabilities

After the recent CCP4 Study Weekend, whose agenda can be perused at

https://eventbooking.stfc.ac.uk/news-events/ccp4-study-weekend-2018-384?agenda=1 ,

the whole MX community – and not just beamline scientists and software developers at synchrotrons – will at last have become aware of the remarkable advances in MX data collection and data analysis methods, based on various combinations of instrumental and computational innovations.

Broadly speaking, they go beyond the well-established and already highly (sometimes fully) automated paradigm of "one dataset from one big crystal per loop in a big beam" towards experiments involving numerous *composite samples*, each consisting in turn of numerous small crystals that need to be located and then "shot" using small X-ray beams to produce large numbers of (very) partial datasets. The implementation of this new paradigm has had a wide-ranging impact on the functionalities required from MXCuBE and ISPyB as well as from auto-processing facilities at synchrotrons. The move towards multi-crystal and serial data collection has been making automation and integration even more crucial than for macro-crystals, and this has been the driving force behind many of the recent developments.

ISPyB aspects

As ISPyB was originally conceived and developed as a sample-tracking ("logistical") facility, and was subsequently extended into a tool for enabling users to visualise the datasets and processing results associated with each tracked sample, it is involved both before and after MXCuBE:

- before MXCuBE, as an optional conduit for user requirements or input information in the form of a Diffraction Plan (typically specifying expected or desired resolution, known space group and cell, beam size to be used, target redundancy and completeness, experiment type, etc.) that is passed on to operators or to workflows in order to guide decisions made in planning and/or executing the data collection and in evaluating whether the processing results meet the user's requirements;
- after MXCuBE, as an optional conduit for user-supplied parameters intended to direct the auto-processing (or auto-reprocessing) of the collected datasets, and especially as a vehicle to access the collected datasets, view their auto-processing results, and examine various summaries of these.

For the new-style experiments involving composite samples, the association between samples and results becomes much more complex: one has to draw a distinction between a "logistical sample" corresponding to a unit of sample tracking (or a unit of action of a sample-changing robot) and a "diffraction sample" (a unit lump of crystalline matter delivered into the beam and giving rise to a – possibly very thin – wedge of rotation data).

- The identification of distinct diffraction samples within a logistical sample may be totally pre-formatted (e.g. at most one crystal per drop per well in an *in situ* plate screening experiment, or at most one micro-crystal per micro-well on a high-density grid), or totally dynamic (i.e. taking place during the experiment itself, e.g. through X-ray rastering of a blob of LCP containing micro-crystals), with intermediate cases such as multiple crystals per CrystalDirect sample support, or per drop per well in a plate.
- In contrast to this tree-like expansion of the list of diffraction samples required to keep track of all the distinct diffraction datasets collected from a given logistical sample, auto-processing (esp. for serial experiments) will give rise to groupings of individual sparse datasets that are not predictable *a priori*, and from which corresponding merged datasets will be produced. This auto-processing may even be integrated into the data collection workflow (see e.g. the DA+ paper https://doi.org/10.1107/S1600577517014503) to help the user in estimating how far the experiment is from having assembled a dataset of sufficient completeness and quality.

MXCuBE aspects

Experiments on composite samples have been, and continue to be, the focus of many creative exploratory developments leading to a diversity of implementations, but certain "unit processes" involved in data collection have clearly emerged:

- Optical pre-centring of a region of interest (ROI) on the sample holder to minimise the extent of the areas that need to be examined with X-rays (automated e.g. on MASSIF-1);
- Mesh/grid/raster scans in one or more orientation(s), with or without Omega rotation;
- simultaneous fast on-line analysis to detect Bragg spots and creation of a "heat map" of the ROI;
- conversion of the peaks in that heat map, by reference to a specified threshold, into a crystal list;
- collection of a thin wedge of data from each selected crystal;

- subsequent processing of these wedges.

There are some important variants of this general template:

- Mesh & Collect (ESRF) and Serial Helical Line Scan (PETRA-III) protocols use simultaneous Omega rotation and raster scanning, then assemble the already collected images into datasets according to the heat map; the various datasets, typically pooled from several sample holders, are then processed, clustered and merged using e.g. Hierarchical Cluster Analysis or a Genetic Algorithm approach;
- in DA+ the collection of rotation images (by CY+) is restricted to the list of crystals selected from the heat map, but processing takes place concurrently with collection, with repeated clustering of accumulated datasets followed by merging for each cluster.

Auto-processing aspects

The simplest and most widespread use of auto-processing is off-line, after all the images of a conventional dataset have been collected. Even here, nothing is standing still and new analyses (e.g. that performed by STARANISO as part of autoPROC) need to be accommodated: new statistics need to be archived and displayed, and full detailed results including numerous graphs and pictures linked to an html file need to be made accessible to the user through ISPyB.

Some synchrotrons provide user input facilities for driving the re-processing of datasets but not their first auto-processing. This can be wasteful, and an extension of user input facilities could help make the first auto-processing maximally useful (instead of being throw-away).

The need for multi-crystal and serial experiments to invoke auto-processing not just in post-collection mode, but in a manner tightly integrated with the experiment itself in order to monitor its progress, prompts the necessity to relocate auto-processing in a space of its own.

Relevance to Sessions 3

Having surveyed in Session 2 where the science-driven "bleeding edges" are [there may be others than those I have outlined!], and where the current versions of MXCuBE and ISPyB stand in their ability to support them or move on to the next steps, some questions naturally arise regarding both collaborations.

- How have the current capabilities been achieved and what role have the collaborations played in achieving them? Have they been developed in a coordinated manner, by using shared components, or has each site carried out its own implementation?
- How can these capabilities be consolidated and disseminated most effectively within the collaborations? Are there good mechanisms in place for forward planning and for developments to be shared, combined, and spread to other sites?
- How can the internal organisation of the software and the mode of operation of the collaborations be strengthened so as to approach the challenges of emerging new types of experiments more effectively? Can we do better at avoiding duplication, combining efforts and streamlining the development process?

- How can the two collaborations be more closely coordinated, seeing that many of the new developments call for an ever increasing degree of integration between their respective domains?
- Where does auto-processing belong, and can it be formalised in some way that would allow shared development and multi-site deployment?