ISPyB developers' meeting
20 April 2022

Participants:

- Alejandro de Maria, Olof Svensson, Mael Gaonach, Stu Fischer (ESRF)
- Karl Levik, James Hall (Diamond)
- Rasmus Fogh, Gerard Bricogne, (Global Phasing)
- Clemente Borges (DESY)
- Alberto Nardella (MaxIV)
- Idrissou Chado , Iquiliz(SOLEIL)
- Gleb Bourenkov, (EMBL Hamburg)
- Marc Armaito Hierro,(ALBA)
- Alexander Dillman (HZB)

Minutes: Rasmus Fogh

Agenda

1. Vote on the coding framework: FastAPI or Flask.
2. Organization of working groups
3. News on the UI (if enough time)

1) Decision on development framework

Stuart Fischer was asked to summarise the pro's and con's of the two alternatives:
Fast-API is a single framework that includes marshalling, and documentation generation. It uses Python standard libraries like Pydantic, and relies on Python typing for type description. FastAPI is organised for asynchronous communication. AdM noted that the biggest risk for FastAPI would be how it developed over a 5-10 year time horizon.

Flask by itself only includes data access; for full functionality you need additional libraries, e.g. RestX (there are other alternatives, but RestX seemd to be the most developed). This means that you need to select the additional libraries and do the work of making sure they work together. Also there are some indications that some libraries may (have to) change. Flask is organised for synchronous communication; this is in principle slower for some use cases, but it depends on the coding how much.

While both solutions are powerful and workable, SF was of the opinion that FastAPI was more maintainable, cleaner and more Pythonic to work with; he particularly appreciated the

use of Python types as opposed to a custom specification language for typing. Other people who had tested the libraries (MG) agreed with this opinion.

After work by SF and MG the FastAPI option is about as advanced as the Flask-RestX option, so there was not much to choose between them on that account. It was pointed out (SF) that the project is currently in a pared-down state, with much development about to begin. This is a point where a change of framework is possible with minimal disruption. AdM pointed out that ESRF needed a functioning SSX application by the summer, so that this window would soon close.

Gerard Bricogne explored what considerations might have led Ivars Karpics to prefer Flask, and whether we ought to consult him on this point. The only answer proposed was that developing the Flask proposal had taken maybe a couple of years of developer discussions, and that FastAPI was relatively new, having only been released in 2018., and so might have been missed

Gleb Bourenkov emphasized that any such change of technology would have to be a very rare event, and queried how long this agreement could be expected to hold, and how this could be guaranteed.

After further discussion it was agreed that either choice ought to work, and that the decision taken now would be permanent, unless an unexpected technical reason arose that would require another change. The amount of work that would shortly be sunk into developing with the chosen framework would serve to discourage all but the most unavoidable technology changes later.

The decision was taken to vote, and the FastAPI framework was selected by six votes (ESRF, ALBA, MAXIV, GPhL, DESY, SOLEIL) against one for Flask-RestX (EMBL-HH)

2) Setting up of working groups.

To advance the project we are setting up working groups for the separate subtasks. Each working group has an institution responsible for coordinating the work, call to meetings, motivate participants, gather site requirements, and have documentation made. Participants are expected to contribute, be it with coding, requirements, or feedback. The working groups will report progress at the regular plenary meetings.

Some institutions are not in a position to promise manpower at the moment, but are joining the working groups and will be contributing as manpower resources shall permit. DLS specifically is not able to make any resource commitments either now or for the future, but will participate as an interested party. The spreadsheet with working group membership is to be found at
https://docs.google.com/spreadsheets/d/1OaPpiqDYdoPrMm8fTnNuRWMzgo6TOfL8jzZdQf

d8eWc/edit#gid=0

Framework WG

Responsible: ESRF.
Members: All participants

Authentication WG

Responsible: ESRF
Members: SOLEIL, MAXIV, DLS
It is noted that the work of implementing authentication in the new framework, using either LDAP or KeyCloak, is mostly done already.

User portal synchronisation WG

Responsible: DESY
Members: All synchrotron sites (i.e. not GPhL).

Shipping WG

Responsible DESY, ESRF, DLS (jointly)
Members: GphL, SOLEIL

EM WG

Responsible: ESRF
Members: DLS

MX WG

Responsible: GPhL, SOLEIL
Members: EMBL, DESY, ESRF, MAXIV, ALBA, DLS

BIOSAXS WG

No participants selected

SSX WG

Responsible: ESRF
Members: EMBL, DESY, DLS, GPhL

Documentation WG

Responsible: GPhL
Members: ESRF, SOLEIL

Xray imaging WG

Responsible: EMBL-HH
Members: None (The technique is only supported at EMBL).

3) User interfaces

Postponed till next meeting because of time constraints

Next Meeting

Thursday 28 April, 0930 European time