py-ispyb Working Group - User Portal Sync

Responsible: DESY
Participants: ESRF, ALBA, Diamond Light Source
20 June 2022 15:00

Participants: Clemente Borges (DESY), Karl Levik (DLS), James Hall (DLS), Alejandro de Maria (ESRF), Maël Gaonach (ESRF), Stu Fischer (ESRF)
Marjolaine Bodin (ESRF), Alexander Dillmanm (HZB)

Agenda

1. Show the current User Portal Sync JSON schema
2. Specification of synchronization per entity
    a. Which fields to use as external unique identifier to match against existing entries in DB
        i. **Person**
            1. siteId (Person primary key from User Portal) - might be deprecated
            2. login
            3. externalID
        ii. **Laboratory**
            1. laboratoryExtPk (Laboratory primary key from User Portal)
            2. Or a combination of name, city and country
        iii. **Proposal**
            1. proposalCode and proposalNumber or
                a. Elettra doesn't have proposalcode
            2. externalId
        iv. **ProposalHasPerson**
            1. personId and
            2. proposalId
        v. **Session**
            1. expSessionPk (Session primary key from User portal - might be deprecated later) or
            2. externalId
        vi. **SessionHasPerson**
            1. sessionId and
            2. personId
        vii. **Protein**
            1. proposalID and

2. Protein acronym
3. externaiIId

viii. **LabContact**

1. cardName and
2. proposalId

3. What about the **BeamLineSetupId** and its relation with the BeamlineSetup Table within the BLSession table?. Currently it has a foreign key constraint. Is this managed by MXCuBE? Perhaps we can exclude the BeamLineSetupId field for the import/sync process.
    a. It may be nullable
    b. ESRF will check if they are syncing anything from the User Portal.
4. Which group or permission will be able to execute the user portal sync endpoint
    a. To be discussed
5. Discuss sqlalchemy_to_pydantic. Although automatic schema definition is nice, this may be one of the few times that manually defining the schemas is useful. By manually defining the schema, we can:
    a. Work out which columns are no longer used (and prepare to deprecate / remove, there is a lot of sql cleaning that could be done in the long term). Thus output only columns that are currently useful
        i. Reduce response payload size
    b. Properly document columns for future devs (via Field(title=..., description=...)
    c. Improve validation WordDashSpace > str, etc
        i. This may be still implemented by overriding sqlalchemy_to_pydantic classes.

Notes from the meeting:

Person entity

Regarding the Person entity, it was discussed whether it would make sense to deprecate some of the fields in the DB. For ESRF, currently the login is the unique field used by Keycloak. Stu mentioned, it would make sense to keep the externalId also since some systems could create users on the fly for shipments, etc, which do not necessarily have a login. The externalID field is a Person UUID encoded in DLS. The combination of having login and externalID should be enough. The Person table also has a personUUID field. The potential field candidates to be deprecated are:

- siteID
- personUUID

Session entity

Alejandro mentioned that he will check what the expSessionPk is doing at the ESRF (probably used to generate URL links to sessions in SMIS). In any case, the externalId will be added, and that should be enough. Maybe in future, expSessionPk can be removed and use instead the externalId field (properly encoded).

Regarding the BeamlineSetupId field for the session, ESRF will check if they are synchronizing something from the User Portal. Stu mentioned it may be nullable.

Permission to execute User Portal Sync

Stu mentioned ESRF is currently discussing the creation of service users. Service users will have specific permissions which will be encoded in the JWT tokens. Still have to wait for the outcome of those discussions.

When to execute the User Portal Sync

Karl mentioned that the User Office may also trigger an update on ISPyB via a message broker. In any case, the current sync_proposal endpoint perhaps is flexible enough to trigger the updates. The same endpoint could be potentially called via cron or any other automatic system (i.e. message broker), and also may be triggered via a button within the ISPyB user interface.

Macromolecules for Biosaxs

Marjolaine mentioned that Macromolecules are currently imported from the User Portal into ISPyB for the Biosaxs module. For now, this will not be taken into account.

DB Cleanup (User Portal sync related entities)

Clemente asked when would be appropriate to start a DB cleanup:

1. During the implementation of py-ispyb
2. Waiting until py-ispyb is partially in production and working together with the legacy systems, and later deal with the cleanup.

Perhaps one can start during the development of py-ispyb.

Actions

- Implement the sync by using the externalId field when possible (if the field is already available)