Integrating Structural Biology
Instruct-ULTRA

WP8 – Increasing the quality and integrity of structural data and metadata and increasing open data sharing

Lead Beneficiary: P8 - CSIC

Leader: Jose Maria Carazo (P8 – CSIC)

Deliverable: D8.3 Develop enhanced deposition of structural data to database

Contractual delivery date: 31 December 2020

Actual delivery date: 4 December 2020

Authors of this deliverable: Pablo Conesa (P8 – CSIC)

Project objective

Develop an easy to install Scipion plugin to enhance and almost automatize the deposition of structural data to public accessible databases

Executive summary

In order to correctly curate data from Cryo Electron Microscopy experiments according to FAIR (Findable, Accessible, Interoperable, Reproducible) principles, there needed to be a simple method for researchers to harvest, record and deposit meta data relating to image processing steps. A new mechanism has been set up to enable output from the Scipion processing plugin into a text file. An “empiar depositor” application was developed in order that all relevant information about a protein structure is deposited along with the raw data, such as Principal investigator and Corresponding Author. Finally an improved viewer was designed for EMPIAR to better display protein structures and data fields once deposited. Close collaboration with EMPIAR has contributed to the success of this project.

Horizon 2020 Coordination and Support action Number ID: 731005: Instruct-ULTRA
H2020-INFRADEV-2016-1

1. Introduction

Currently, a typical formal outcome of a cryo electron microscopy (cryoEM) research analysis comprises a final electron density map and a structural model (deposited in EMDB and PDB) and the corresponding scientific article. In the latter, it is very likely to find that there will be some paragraphs describing the image processing steps, in all cases, far from what would be a FAIR (Findable, Accessible, Interoperable, Reproducible) practice. In most of the cases, even though there is a public database for depositing cryoEM RAW data (EMPIAR), only a small percentage of authors are actually sharing the RAW data publicly (less than 10%, even for globally critical data such as those related to Covid19 research). There are many factors that need to be addressed to increase this proportion, but one of them is to streamline deposition, even automate it, so that this process would not be perceived as a burden.

It is the purpose of this work to reduce the effort in making public depositions of cryoEM data, and at the same time fill the existing gap in terms of reporting “FAIRer” details about all the steps to go from RAW microscope movies to a final density map.

1.1. CryoEM image processing

Obtaining a final electron density map, starting from the raw data acquired in a transmission electron microscope (TEM), implies, in the **simplest cases**, the application of tens of steps. There are many variables that can make this ideal case worse, like having a complex specimen, or a flexible one, or a small one. There are other crucial steps that affect the quality of the raw data, like the sample preparation required to place your specimen in a TEM, as well as the acquisition itself in complex TEM, which can be unstable and very sensitive to many factors that can worsen the images acquired.

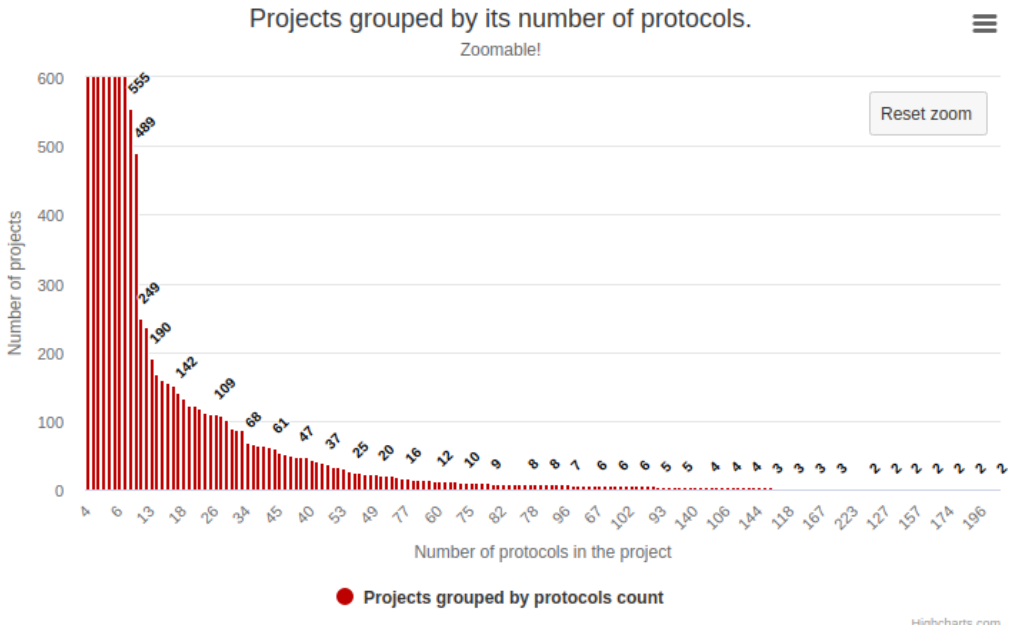

Figure 1. Zoomed plot of projects grouped by its number of protocols. Most Scipion projects reported falls below 20 steps, but there are many cases where project could reach over 100s steps.

Each of these steps are associated with anything from a few parameters to tens of them in the more complex cases, and each of them will very likely affect the intermediate result this step produces. Therefore, knowing what these parameters are essential to understand how

the final map was produced. As a very rough estimate, a 10-step pipeline may have 100 associated parameters (10 parameters per step) that affect the final result obtained.

2. Introduction to Scipion

Scipion is an image processing framework, recently pluginised, that integrates most commonly used cryoEM image processing software like RELION, Cryosparcs, Motioncor2, Xmipp, etc. It organises all the steps into a project structure where it clearly presents and stores all the steps with the corresponding parameters (see Figures 2 and 3).

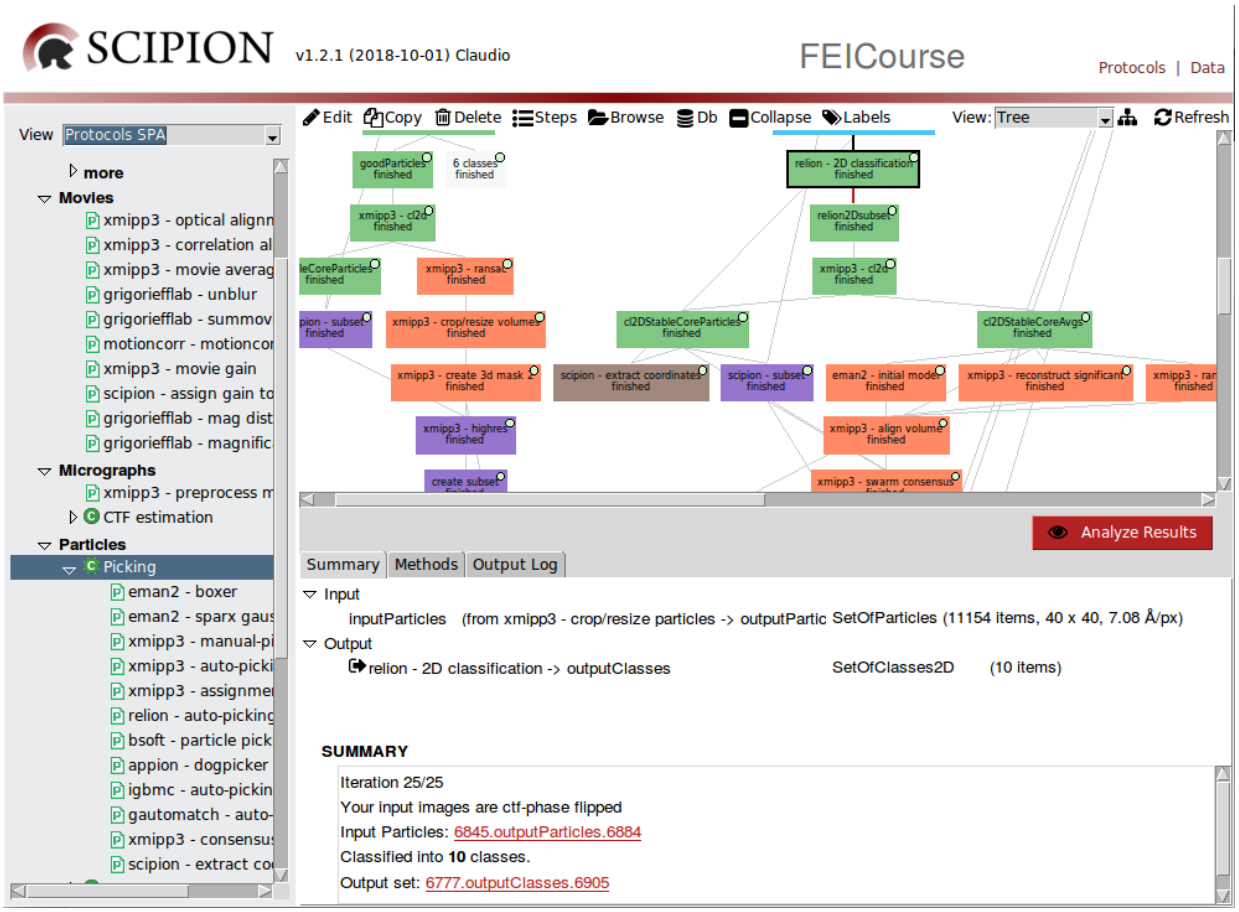

Figure 2. Scipion graphical user interface depicting its 3 common panels: Available steps (left tree), workflow (top), summary and logs tabs (bottom).

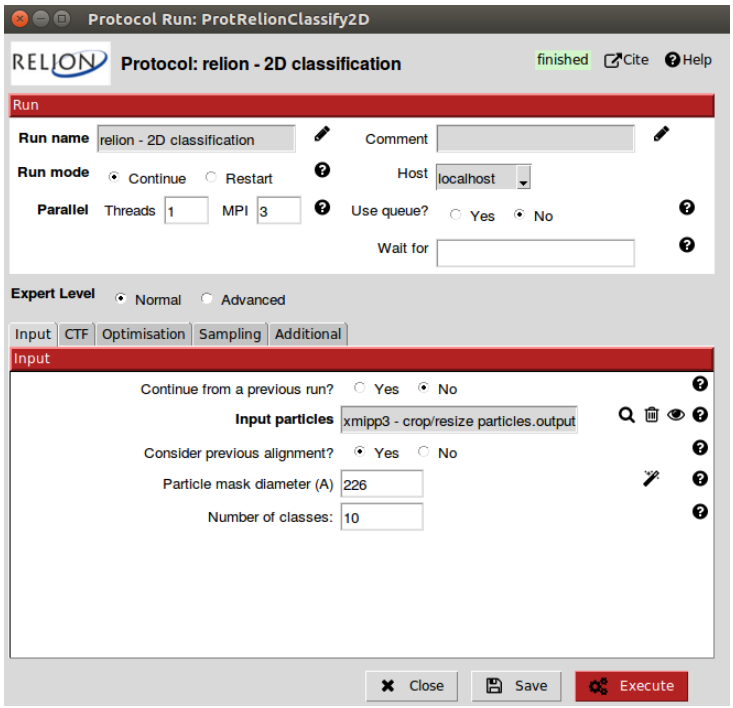

Figure 3. Example of one of the steps from where all the parameters are shown and grouped in tabs. Note that "Input" tab is the one active with 4 parameters but there are more in the rest of the tabs for this particular case.

3. Exporting/Importing with data associated

In the context of this project, P8 has worked on exporting the relevant data contained in a single Scipion project to a text file. Among the options available to easily persist data in text files, the JSON format was chosen. It can be natively read and written by python and it is widely used in web front-ends.

At the beginning of this project, there was a basic exporting functionality covering mostly local exporting/importing functionality not associated with any data at all. During this project the export process was allowed to be associated with the raw data, so any future "import" action smoothly links the JSON content (steps) with the raw data.

```
[
    {
        "object.className": "ProtImportMovies",
        "object.id": "6576",
        "object.label": "import movies",
        "importFrom": 0,
        "filesPath": "/home/pablo/extra/data/10200/",
        "filesPattern": "*.tif",
        "copyFiles": false,
        "haveDataBeenPhaseFlipped": false,
        "acquisitionWizard": null,
        "voltage": 300.0,
        "sphericalAberration": 2.7,
        "amplitudeContrast": 0.1,
        "magnification": 50000,
        "samplingRateMode": 0,
        "samplingRate": 0.814,
        "scannedPixelSize": 7.0,
        "doseInitial": 0.0,
        "dosePerFrame": 47.0,
        "gainFile": "/home/pablo/extra/data/10200/CountRef_26_000_Oct04_16.13.54.mrc",
        ...
    },
    {
        "object.className": "ProtMotionCorr",
        "object.id": "6638",
        "object.label": "motioncorr - movie alignment",
        "gpuList": "0",
        "doSaveAveMic": true,
        "useAlignToSum": true,
        "doSaveMovie": false,
        "doComputePSD": false,
        "doComputeMicThumbnail": false,
        "extraProtocolParams": "",
        "doApplyDoseFilter": true,
        "patchX": 5,
        "patchY": 5,
        "patchOverlap": 0,
        "inputMovies": "6576.outputMovies"
        ...
    },
    {
        "object.className": "CistemProtCTFFind",
```


Figure 4. Excerpt of a JSON file exported by Scipion. Note that paths are absolute and this was a problem to move around the JSON file and the associated data. Implemented functionality corrects this for smooth export and import options. Note also file is intentionally invalid/truncated to be able depict its content in a small image.

4. Facilitating the deposition

Since there is already a public database for depositing cryoEM raw data, work was undertaken to make these depositions as easy as possible from Scipion software. As a plugin framework, the logic approach was to create a plugin to deal with the depositions. The code can be found here https://github.com/scipion-em/scipion-em-empiar and is publicly available and regularly updated at https://pypi.org/project/scipion-em-empiar/, which makes it available from Scipion plugin manager.

P8 have also worked on what the “empiar depositor” (see Figure 5), adding all parameters that a regular EMPAIR deposition offers, like author’s information, that previously weren’t available in a Scipion project.

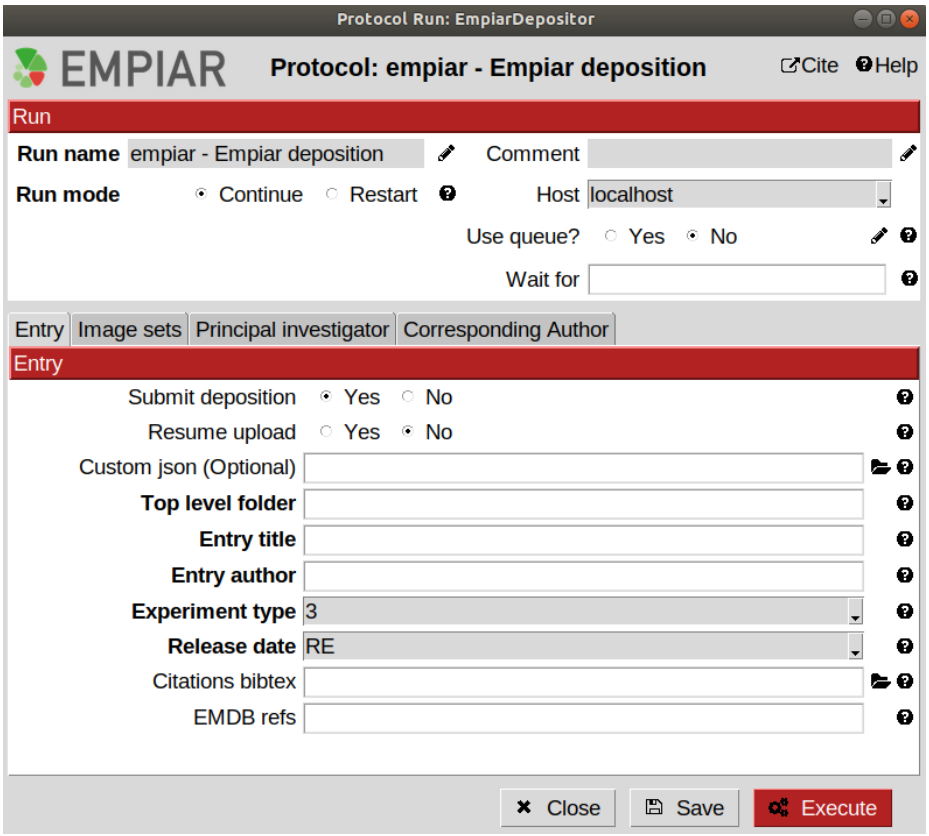

Figure 5. EMPIAR depositor form with all available parameters distributed in tabs: Entry, Image sets, Principal investigator, Corresponding Author.

The empiar depositor can submit not only cryoEM movies but other formats available, like micrographs or particles accepted by EMPIAR. Due to the large size of cryoEM raw data, this step will require long execution times for data transfer and the whole process is subject to typical network failures in this context. A resumable option was implemented so the submission can be resumed in case errors occurs.

5. Early depositions from facilities

EMPIAR was designed to accept depositions associated with EMDB entries. This procedure assumes the author already has an electron density map which was being submitted at the end of his/her research, normally associated to a journal publication. It is in the context of Instruct and Instruct facilities that there was the desire to explore the possibility of making the depositions earlier, at acquisition time, when there isn't yet a final density map to be associated with the raw data. The reasoning behind is that in this way it could be assured that 100% of the data collected at Instruct Centers in the course of Instruct access projects would be FAIR from the beginning (of course, considering all pertinent embargo periods).

P8 contacted the EMPIAR team to collaborate and allow EMPIAR to accept "early" submissions, to which they kindly agreed. Not only this, EMPIAR also accepted an embargo period for the data that is longer that the maximum 1 year, increasing it to 3 years to match Instruct data policies.

A new use case was proposed to EMPIAR team that also implied adapting their service to accommodate a "submission ownership transfer" mechanism. Instruct facilities are supposed to make the submission at the beginning of the acquisition with the 3 years embargo period,

but once the facility user, after some time, deposits the density map, the association has to be done. This mechanism has since been designed by the EMPIAR team based on our proposal.

5.1 Acknowledgements

P8 want to publicly acknowledge the effort and quick response EMPIAR team in reacting to their suggestions.

6. Visualisation of the workflow

The deposited json file, once submitted to EMPIAR, is recognised by the server as a workflow file, and shown in the EMPIAR entry page. However, just listing the file did not allow for a quick inspection.

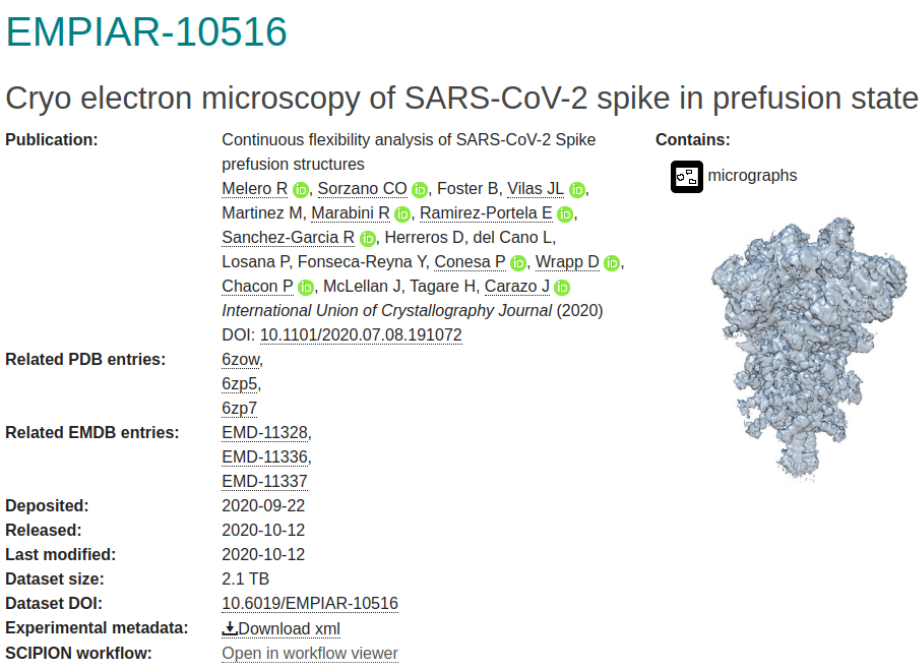

Figure 6. Sample of an EMPIAR entry with a workflow file provided (see “SCIPION workflow” at the bottom).

P8 has developed a workflow viewer based on HTML/Javascript technology using recent web components framework to easily display the content of the workflows in a web page with little effort. The code can be seen at https://github.com/I2PC/web-workflow-viewer/ This viewer is been used already by the EMPIAR data base.

7. Future work

P8 is currently extending their json file to describe the outputs of each of the steps. This description, in many cases, is enriched with thumbnails of images extracted from the Scipion project. The thumbnails also have to be deposited at EMPIAR, and the viewer requires further modification to expose those thumbnails.