STREAMLINE

WP4 Build Capacity

D4.2

Report on workflows for online data reduction and analysis for four techniques used at ESRF beamlines

Expected date 14/10/2022

STREAMLINE has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 870313.

PROJECT DETAILS

PROJECT ACRONYM	PROJECT TITLE
STREAMLINE	**Sustainable research at micro and nano X-ray beamlines**
GRANT AGREEMENT NO:	THEME
870313	**H2020-INFRADEV-2018-2020 Development and long-term sustainability of new pan-European research infrastructures**
START DATE	
15/11/2019	

DELIVERABLE DETAILS

WORK PACKAGE ID	EXPECTED DATE
WP4	**14/11/2020**
WORK PACKAGE TITLE	DELIVERABLE TITLE
Build Capacity	**Report on workflows for online data reduction and analysis for four techniques used at ESRF beamlines**
WORK PACKAGE LEADER	DELIVERABLE DESCRIPTION
Andy Götz	**Report on workflows for online data reduction and analysis for four techniques used at ESRF beamlines - Task 4.4**
DELIVERABLE ID	PERSON RESPONSIBLE FOR THE DELIVERABLE
D4.2	**Andy Götz**

NATURE

☒ **R- Report** ☐ **P - Prototype** ☐ **D - Demonstrator** ☐ **O - Other**

DISSEMINATION LEVEL

☒ **P - Public**
☐ **PP- Restricted to other programme participants & EC:**
☐ **RE – Restricted to a group**
☐ **CO – Confidential, only for members of the consortium**

REPORT DETAILS

VERSION	DATE	NUMBER OF PAGES
1.0	**14/11/2022**	**15**
DELIVERABLE REPORT AUTHOR(S)	FOR MORE INFO PLEASE CONTACT	
Wout de Nolf, Olof Svensson, Giannis Koumotsos, Henri Payno, Andy Götz	**Andy Götz**	

STATUS

☐ **Template** ☐ **Draft**
☒ **Final** ☒ **Released to the EC**

Contents

Introduction

When implementing online data processing and beamline automation, the specifics of the calculations or automation steps need to be implemented case by case, but the following features are common for all use cases:

- **Reproducibility, reusability and data provenance**: the process should be fully described in a ***recipe*** that can be stored and transferred in different ways. This allows experts to create the recipe and anyone can execute it:
  - scientists that are not experts in X-ray technique
  - the acquisition control system
  - reviewers or readers of scientific publications
- **Collaboration**: the process described by a recipe should be ***modular*** so that experts in different fields can combine their knowledge that create the recipe
- **Traceability**: process execution needs to be ***monitored*** and the ***history*** of all executions needs to be recorded
- **Different time and place**: the same recipe can be used for online and offline data processing, executed ***locally*** or ***remote*** (e.g. computer cluster)

The ***workflow*** approach fits these needs where the recipe describes a workflow of tasks with data transfer between the tasks and links that can be conditional (i.e. links that are followed when pre-defined conditions are met). The tasks are most often written in python. For beamline automation, workflows with loops and conditional links are common.

The aim of this task was to develop four (or more) workflows within Streamline. The Ewoks workflow system, new workflows and the adaption of existing workflows (use cases #1, #4 and #5) described below were developed within Streamline.

ESRF Workflow System (ewoks)

Many *workflow management systems* exist (https://s.apache.org/existing-workflow-systems) each providing different features and having different strengths and weaknesses. With *ewoks* (https://doi.org/10.5281/zenodo.6075054) we created a meta workflow system which allows developers to *implement tasks* and scientists to *create workflows*, independent of the workflow management system. This means workflow creation is independent of the execution environment (locally, parallelized, distributed) and the way the user interacts with and visualises workflows. (command line, desktop GUI or web GUI).

Currently ewoks supports the following third-party workflow management systems

- *dask*: execution engine which supports parallel and distributed execution (i.e. cluster) (https://www.dask.org/)
- *orange3*: provides a desktop GUI to create and execute workflows (https://orangedatamining.com/)
- *pypushflow*: execution engine which supports cyclic workflows and conditional links (https://pypushflow.readthedocs.io/en/latest/)
- *celery*: trigger workflows remotely (job scheduling) (https://docs.celeryq.dev/en/stable/)

Ewoks itself provides

- a common definition to describe workflows (based on *networkx*, supporting different formats such as json and yaml)
- a common way to describe and implement tasks for ewoks workflows
- sequential execution of directed cyclic graphs (i.e. no loops or conditional links)
- web support to create and execute workflows (React frontend with a REST backend)
- start workflows on a SLURM cluster using the SLURM REST API
- event mechanism to monitor and visualise the execution of a workflow, regardless of how and where it is executed

Ewoks provides the following interfaces to create and execute workflows

- Python API: execute workflows (locally or remote), dynamic in-memory workflow creation (useful for data-parallel workflows where the side of the workflow depends on the side of the data)
- Command Line API: execute workflows locally or remote
- REST API: store workflows, execute workflows in the backend or remote, websocket for workflow execution monitoring
- Desktop GUI: create and execute workflows
- Web GUI: create, execute and monitor workflows

The web GUI was developed in order to provide access and fully exploit the ewoks infrastructure. Its main purpose is to give the user the ability to visualise and edit workflows as graphs on a canvas. The GUI can handle complicated graphs with subgraphs and conditional links. The user can easily inspect, build and parameterize ewoks graphs in a feature-rich graphical environment. A printscreen of the workflow editing interface is presented in Figure 1.

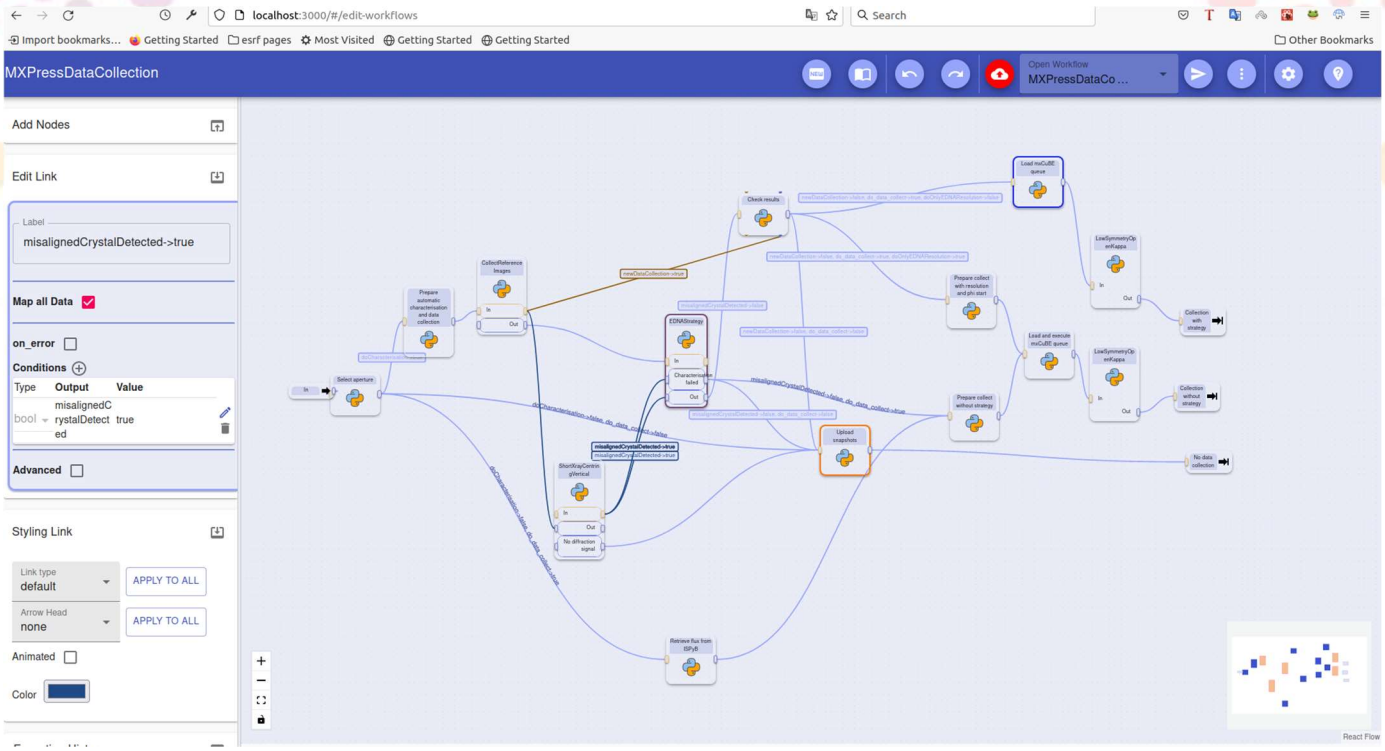

Figure 1: Ewoks workflow editing interface.

Alongside building workflows, the GUI provides the ability to launch the execution of workflows through a REST API. To achieve this an environment is under development for monitoring workflows that are being executed or having finished execution for investigating results and errors. Figure 2 presents the interface for monitoring executed workflows.

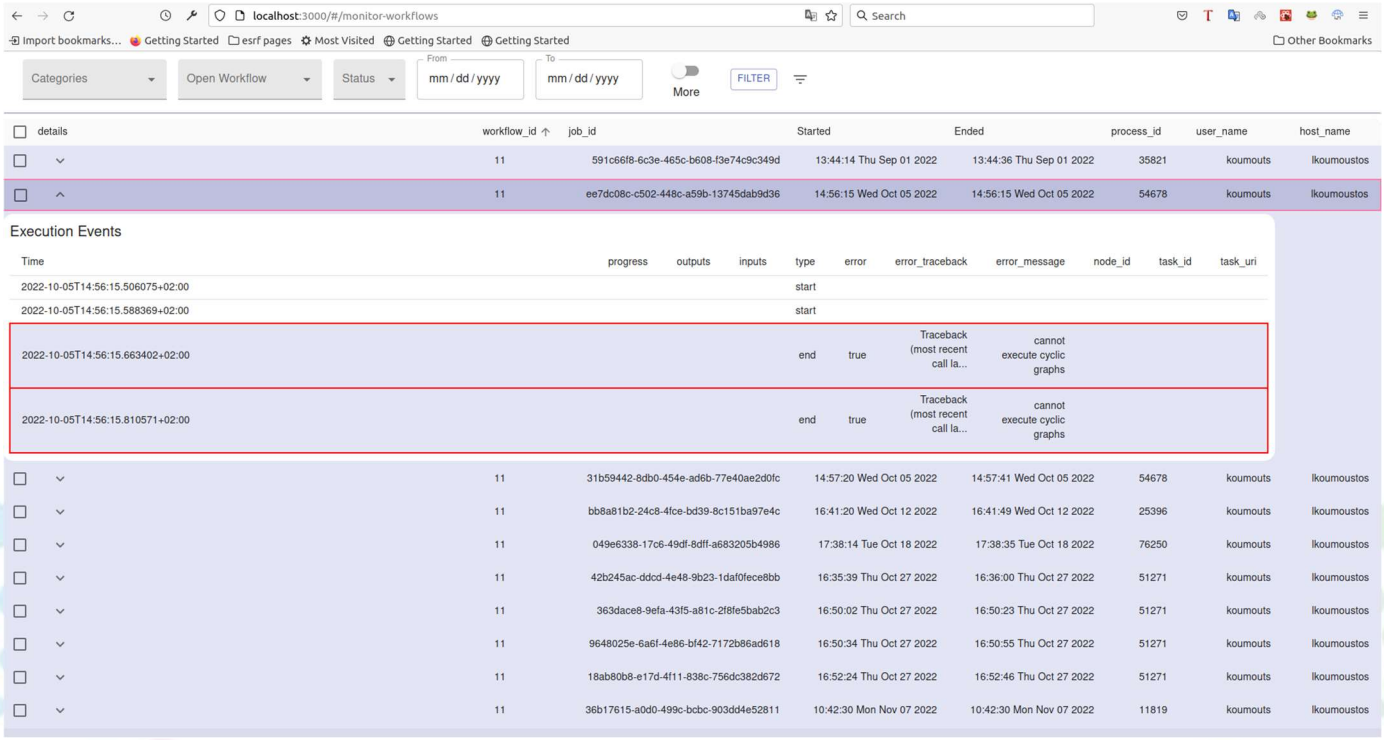

Figure 2: Ewoks interface for monitoring executed workflows.

The main advantage of the web application is that it makes it a lot easier for scientists and engineers to build their own tasks and dynamically use them in their workflows.

Use case #1 – Structural biology

BES

Automation of MX (Macro-molecular crystallography) beamlines is becoming increasingly more important for synchrotron radiation facilities in general and for the ESRF in particular. The ESRF was the first facility to operate a MX beamline completely automatically in 2014 (MASSIF1, ID30A-1, http://dx.doi.org/10.1107/S1600577515016604). The operation of the other four MX beamlines (ID23-1, ID23-2, ID30A-3 and ID30B) is increasingly being automated, especially during nights and weekends, where automation takes over from the daytime experiment control piloted by experts.

The backbone of the MX beamline automation is a set of workflows called the Beamline Expert System (BES). The fully automatic workflows operating MASSIF 1 and other MX beamlines (https://www.esrf.fr/MXPressWF) are implemented using BES. These workflows not only allow fully automatic operation of the beamlines (http://dx.doi.org/10.1107/S1399004715011918) but also allow data mining from a large set of samples (https://doi.org/10.1107/S2052252519008017) thanks to the workflow database, which automatically records metadata during the experiment, and the MX beamline LIMS ISPyB. BES is also routinely used for interactive workflows including mesh scans, X-ray centring and another dozen workflows currently in production on the MX

beamlines. The MX beamlines typically execute several hundreds of fully automatic and interactive workflows every day of operation.

The BES workflows rely on the ESRF workflow system Ewoks. BES uses the pypushflow workflow engine through the ewoksppf API. Thanks to Ewoks, the workflows on the MX beamlines are now implemented in a similar way to other non-MX beamline workflows which has significantly improved the support and maintenance situation. Figure 3 presents an example MX workflow.

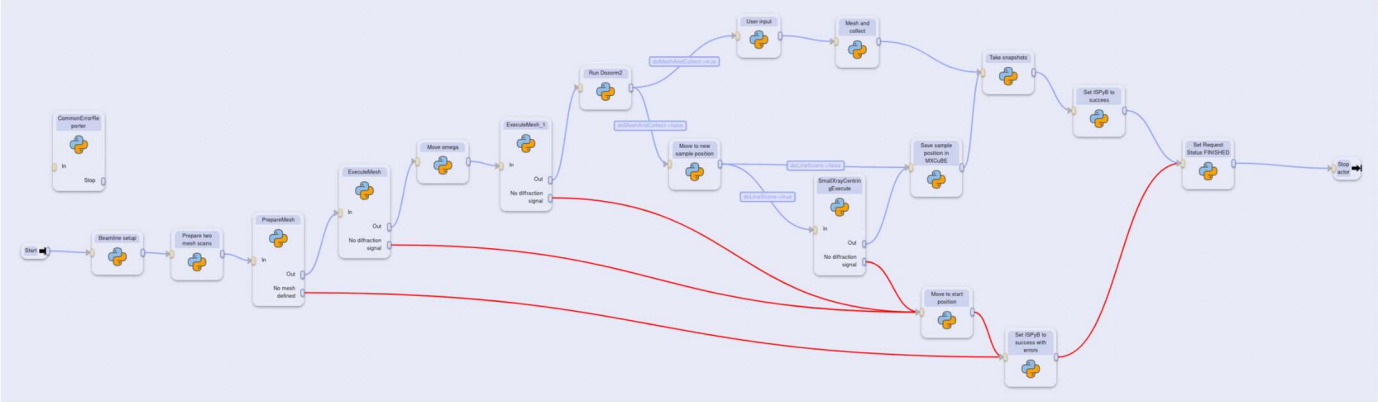

Figure 3: New workflow "Two Mesh Scans" developed using the Ewoks web interface.

Workflow(s)	MX
Beamlines	ID23-1, ID23-2, ID30A3, ID30-B
Users	Beamline scientists + users
When	Automatically or interactively
Input	2D diffraction images (CBF or H5)
Output	Scaled and merged intensities
Trigger	mxcube

Table 1: Summary of MX workflow(s).

CryoEM workflows

Single particle analysis (SPA) in cryo electron microscopy has become a routine structural technique to achieve sub 3 Å resolution. In spite of all the advances in this field, the data collection and image processing steps still suffer from low throughput. If cryo-EM SPA is to become a true complementary technique to X-ray crystallography, the following gaps need to be bridged.

1. Faster speed in data collection: Ongoing developments in hardware and software are resulting in more stable microscopes (less time spent in alignment procedures and equal results from lesser amount of data), better image recording devices (faster and higher DQE) and better data collection software. With these advances, a single SPA data set can be collected in less than 24 hours (currently implemented in only a few facilities worldwide).

2. On-the-fly feedback on the quality of the collected data: The faster the data is collected the more imperative it is to keep up with the image pre-processing which provides indications of the quality of the data that is being collected.

3. Final output in a desirable time frame: The ultimate goal is to achieve sub 3 Å resolution 3D structure of the macromolecule, but currently this is a very time-consuming process due to the overwhelming need for GPU and CPU resources and substantial expertise in cryo-EM image processing. A cryo-EM facility has to wait between 3 months to 3 years to witness the results of their users.

Since its commissioning in November 2017 as an international user facility for cryo-EM SPA, the ESRF cryo-electron microscope CM01 has been providing user service for SPA experiments with a routine data collection taking over 48 hours. The ESRF currently lacks the investment to address point 1 above. We have addressed point 2 by providing automatic pre-processing of data as well as automatic uploading of meta-data to the ISPyB LIMS and automatic archiving of meta-data and data via the ESRF data policy. The automatic pre-processing, upload of meta-data and archiving is implemented using the Scipion framework. Thanks to this Scipion workflow, users have been able to follow the experiment in real time using EXI/ISPyB and have at the end of the experiment access to motion-corrected files that are the starting point for further data processing (Figure 4).

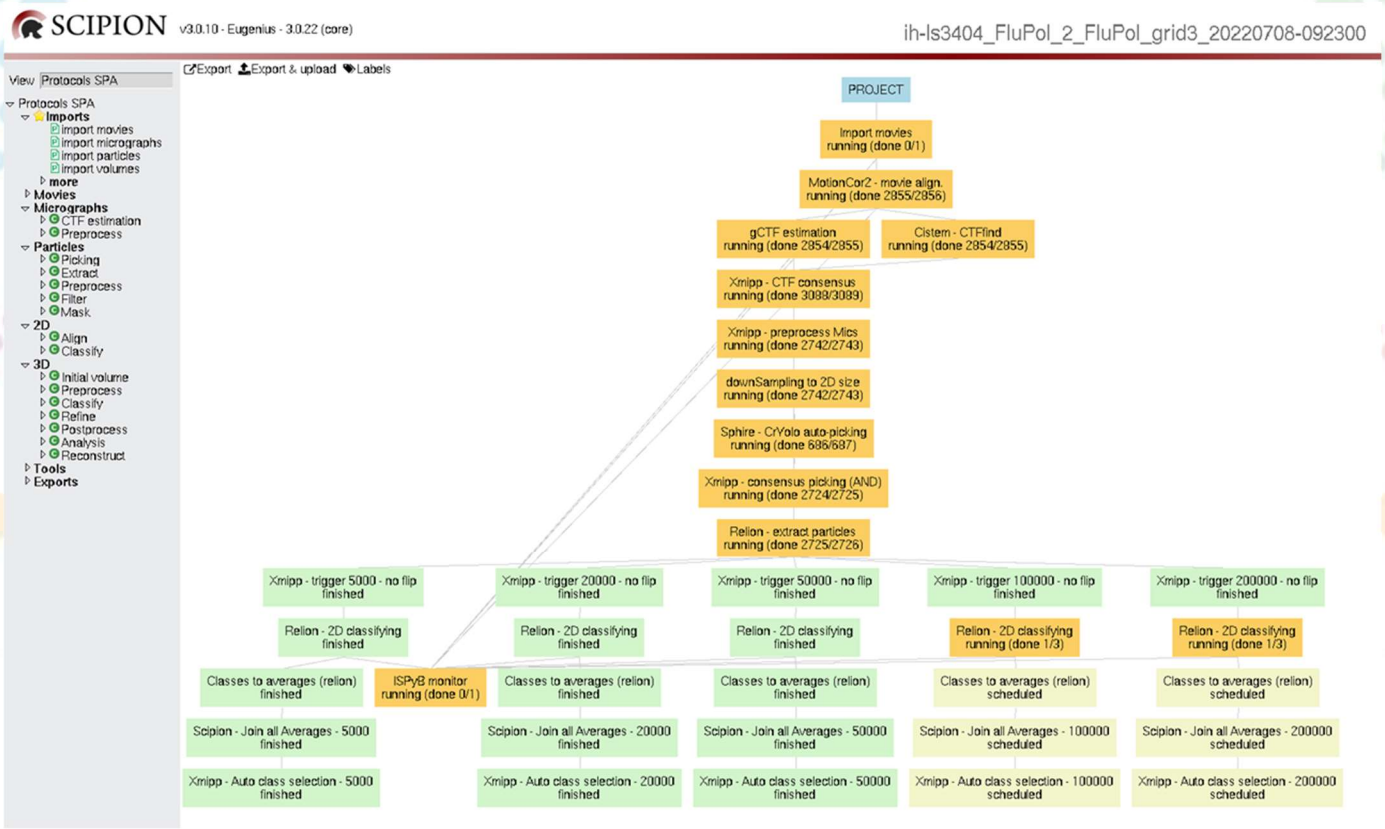

Figure 4: Scipion workflow for automatic 2D classification.

There are two main goals of this workflow:

• **To help the CM01 scientists track each data collection collected and make sure the productivity is kept optimal.** Although the pre-processing of the images gives an indication of the quality of the data collected, the real potential of the data to go to high resolution is indicated by the class averages. Moreover, with the speed of technological advances in the field, it is fair to anticipate faster acquisition modes being implemented for cryo-EM in the near future, in which case, the ESRF would be one step ahead of several facilities by having a pipeline to assess the

sample quality and choose whether the experiment should be continued or abandoned in favour of the next sample. This would make optimal use of beam time at the CryoEM instrument.

• **To test the pipeline for 3D refinement, which is the ultimate aim.** A data collection shown by the 2D workflow to give good results for initial class averages with a small data set would be continued to accumulate sufficient data to achieve optimal 2D class averages, which helps to prepare the input for 3D refinement steps.

Further developments of this workflow will include generation of the initial 3D model and 3D classification and auto refine. If this full pipeline could be implemented, either completely automated or with minimum user input, then a notable application would be for drug screening experiments. This application would be in addition to all the above listed advantages of use to academic users.

A new GUI for easy start and control of the automatic processing has been developed. Through it the user can launch on demand new tasks for the celery workers to execute as well as inspect the execution process. It provides dynamic forms with all needed parameters depending on the selected task. It was developed within the Daiquiri platform (https://ui.gitlab-pages.esrf.fr/daiquiri/) which is a web based framework for beamline control and data acquisition. A screenshot of the Cryocube interface as part of Daiquiri is given in Figure 5.

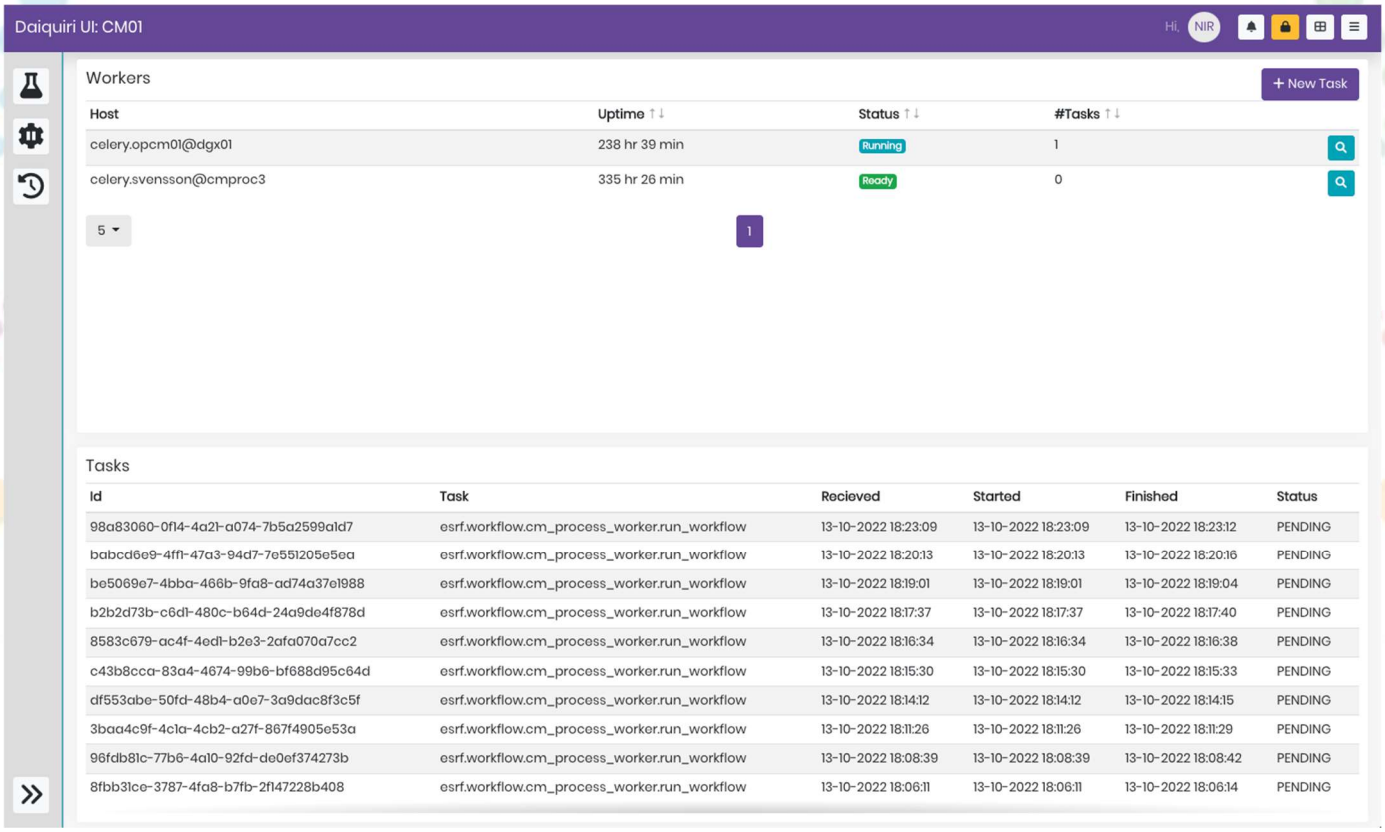

Figure 5: CryoCube UI screenshot.

Workflow	cryoem
Beamlines	CM01
Users	Beamline scientists + users
When	Manually after each acquisition
Input	Movies (TIFF)
Output	Motion corrected micrographs, CTF, picked particles, 2D classification
Trigger	daiquiri

Table 2: Summary of cryoem workflow.

Use case #2 – Powder diffraction

Powder diffraction experiments often require data reduction to obtain powder diffractograms that can be further analysed to obtain structural and quantitative information about the sample. Currently three such workflows are running

- ID31: a data reduction workflow is triggered automatically for scans that include a diffraction camera. The workflow includes normalisation and azimuthal integration with *pyfai* (https://doi.org/10.5281/zenodo.5946928).
- ID22: data conversion (HDF5 to ASCII) and data reduction workflows are triggered actively by the acquisition macros. The data reduction workflow includes rebinning with *multianalyzer* (https://zenodo.org/record/7223895) of data taken with multiple analyzer crystals and a large area detector.
- High-throughput powder diffraction (in use at ID31): a mobile system has been developed to fully automate the measurement of thousands of powders in a matter of hours. The acquisition includes the triggering of a data reduction workflow. The workflow includes calibration, normalisation and azimuthal integration with *pyfai* (https://doi.org/10.5281/zenodo.5946928).

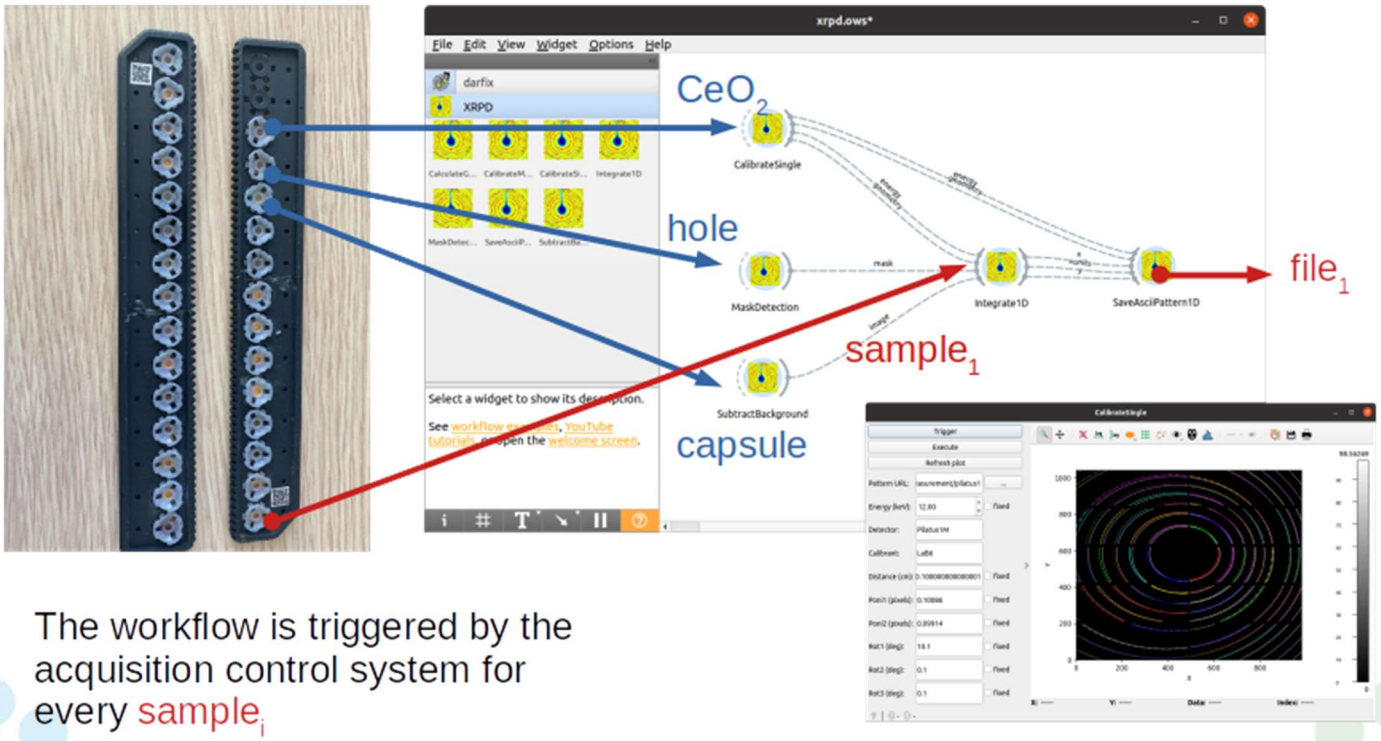

Figure 6: High-throughput powder diffraction workflow.

Workflow	powder
Beamlines	ID22, ID31
Users	Beamline scientists + users
When	Automatically for each sample
Input	2D image (hdf5)
Output	1D spectrum (hdf5)
Trigger	bliss

Table 3: Summary of powder workflow.

Use case #3 – Spectroscopy

In the field of X-ray spectroscopy (e.g. EXAFS and XANES) one acquisition can take up to one hour. To judge the quality of the data being acquired, pre-processing is required. Considering the total time of an acquisition, this should be done during the measurement so that scientists can make decisions without losing valuable beamtime. The workflow (Figure 7) we developed includes normalisation, post-edge background subtraction and XAFS Fourier Transform with *xraylarch* (https://doi.org/10.5281/zenodo.6796308). It is currently operational at BM23.

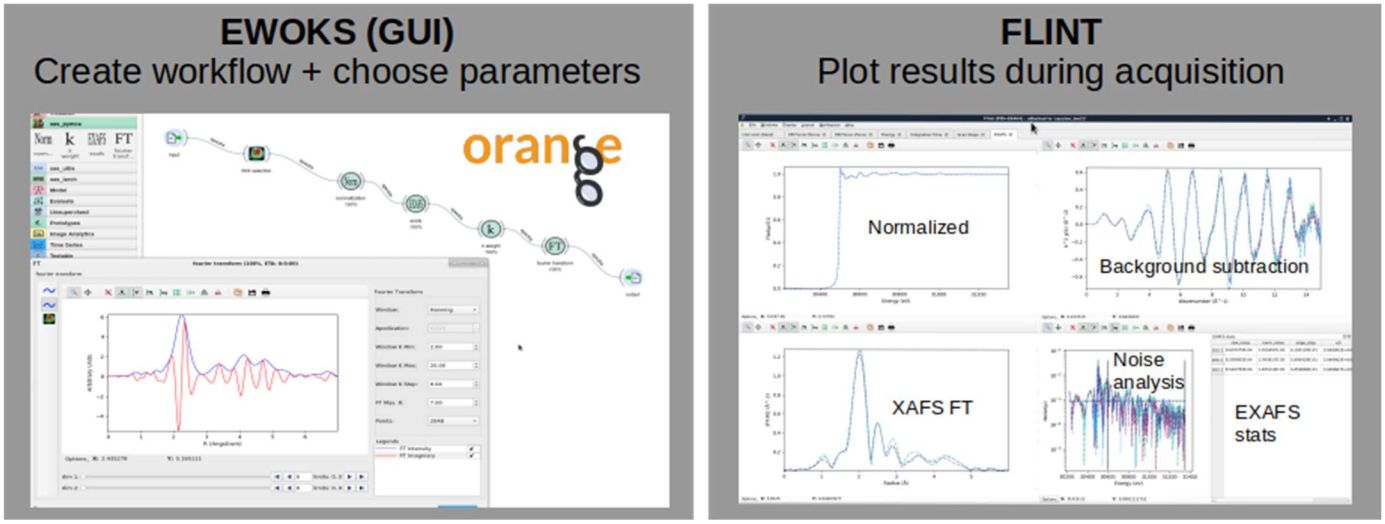

Figure 7. Ewoks workflow of EXAFS plotting during acquisition.

Workflow	exafs_plot
Beamlines	BM23
Users	Beamline scientists + users
When	Automatically after each scan
Input	EXAFS scan data (memory)
Output	fft of EXAFS spectrum (flint)
Trigger	bliss

Table 4: Summary of exafs_plot workflow.

Use case #4 - Tomography

Tomwer is a library and a set of applications allowing users to treat tomographic data using a workflow. A 'canvas' application is used to create and edit the workflow and settings for each of the steps from a graphical user interface. The graphical editor is based on Orange.

New features based on the ‘ewoks-orange’ library have been added to ensure backward compatibility with EDF format, NXtomo keys upgrade and the projections normalisation.

Some processing (‘darkref’, ‘nabu’ processes, ‘center of rotation research’) have been upgraded to improve usage (Figure 8).

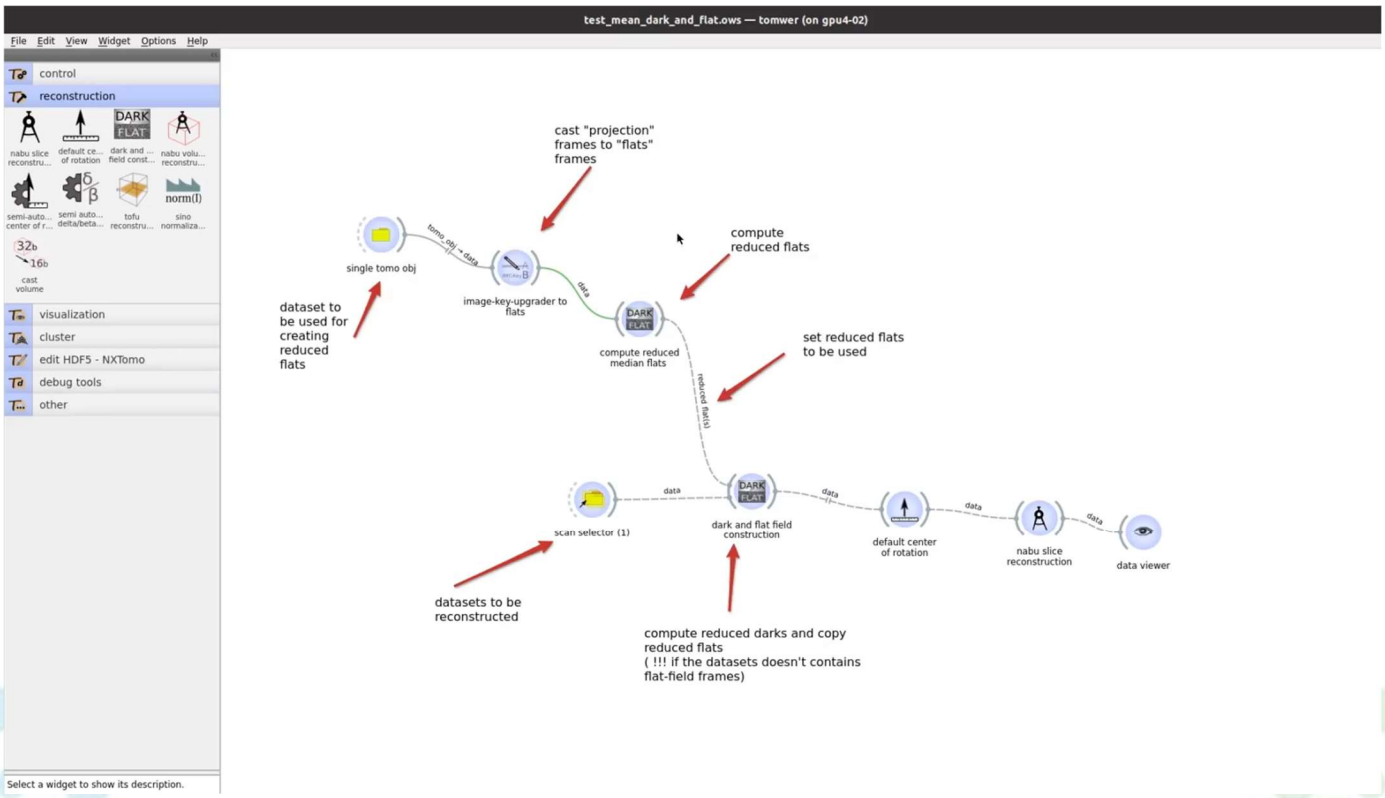

Figure 8. tomwer: workflows to overwrite reduced darks and flats of scans.

An important work has been done on improving handling volumes across the tomotools suite and end it with simplification at the tomwer side and addition of volume related widgets using the ewoks libraries, such as volume cast or volume selection.

Finally job submission to slurm has also been reworked to improve robustness and better distribute the workload to the compute cluster (Figure 9).

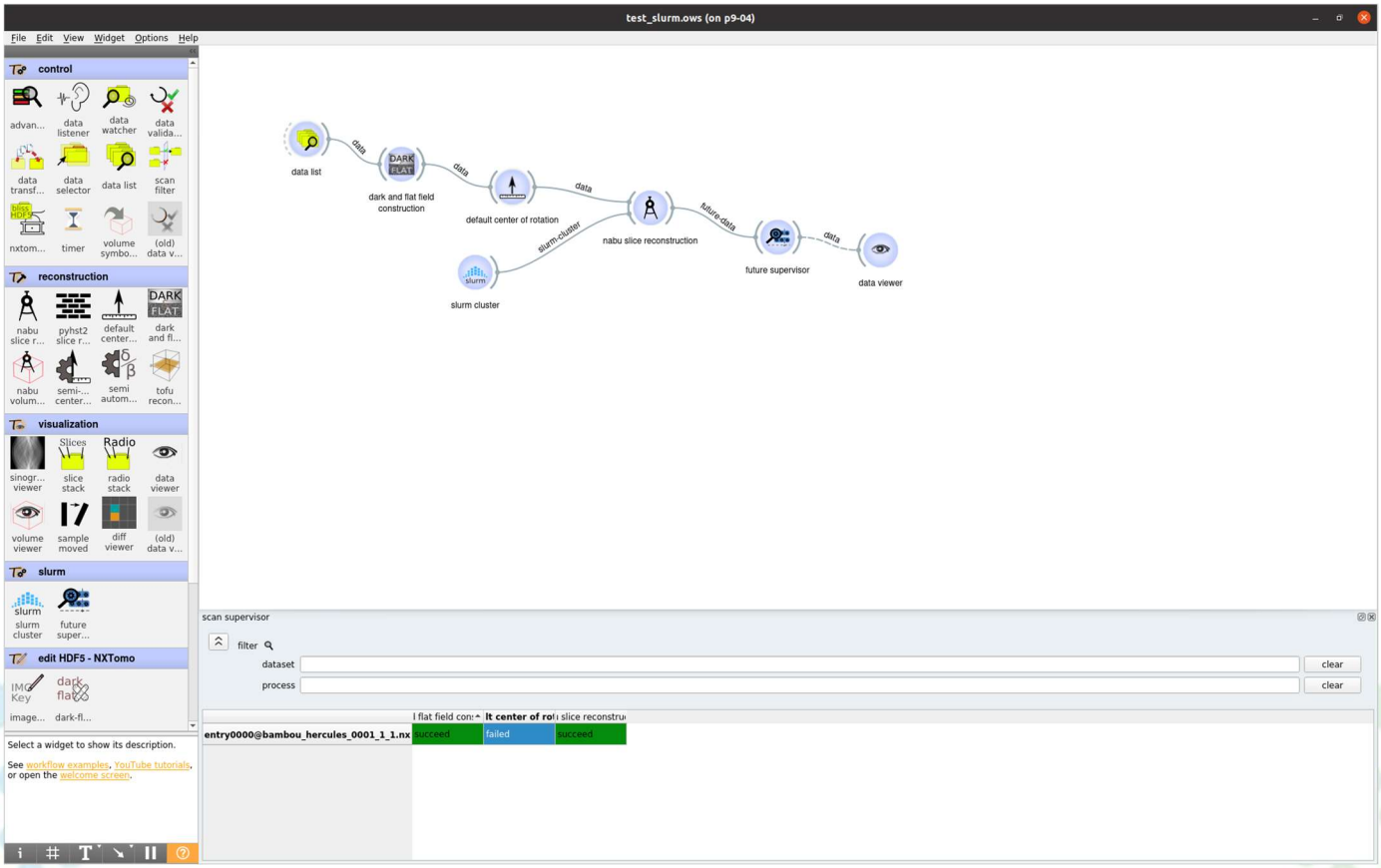

Figure 9: tower: workflow with some nabu job submitted to slurm.

Tomwer and other tomotools have been recently deployed on BM18 in addition to being in production on the ID11, ID16b, ID17, ID19, BM18 and BM05 beamlines.

A series of tutorials has been made to train users how to use tomwer. They are available here: http://www.edna-site.org/pub/doc/tomwer/video/canvas/. In the future, they could also be shared on the pan-training.org site.

Workflow	tomwer
Beamlines	BM05, ID11, ID16B, ID17, BM18, ID19
Users	Beamline scientists + users
When	Daily
Input	raw 2d images (hdf5 or EDF)
Output	sinogram 2d image (hdf5) reconstructed 3d volume (tiff)
Trigger	bliss / user

Table 5: Summary of tomwer workflow.

Use case #5 - Dark-field microscopy

darfix (https://doi.org/10.48550/arXiv.2205.05494) is a Python library for the analysis of dark-field X-ray microscopy data. It provides ewoks tasks and widgets for the ewoks orange3 desktop GUI (Figure 10).

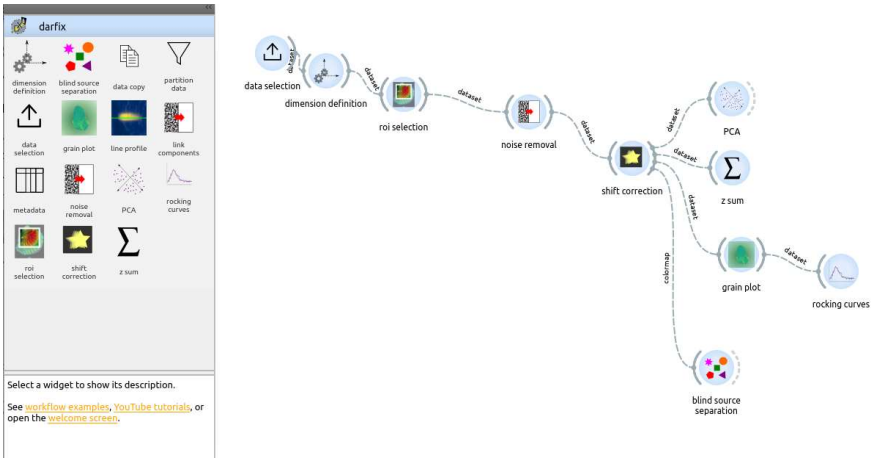

Figure 10: darfix, the workflow for DFXM.

Workflow	**darfix**
Beamlines	**ID06**
Users	**Beamline scientists + users**
When	**Daily**
Input	**raw 2d images (hdf5)**
Output	**3d grain volumes**
Trigger	**user**

Table 6: Summary of darfix workflow.