darfix: Data analysis for dark-field X-ray microscopy

Júlia Garriga Ferrer

European Synchrotron Radiation Facility

julia.garriga@esrf.fr

May 7, 2021

Introduction

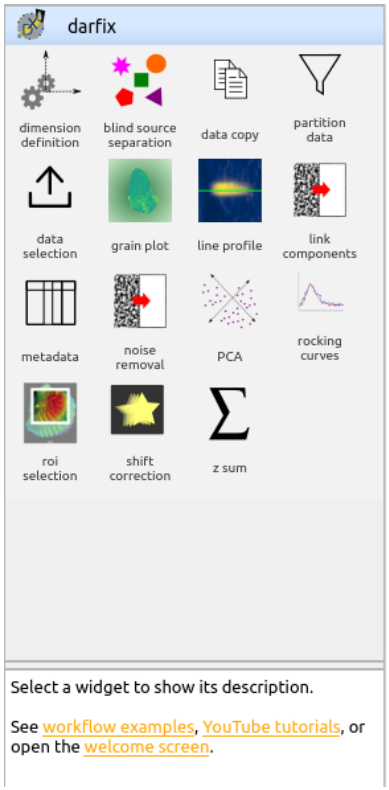
Objective: provide a set of data processing and visualization tools for dark-field X-ray microscopy.
In particular, cover the needs in ID06 for the analysis of their data.

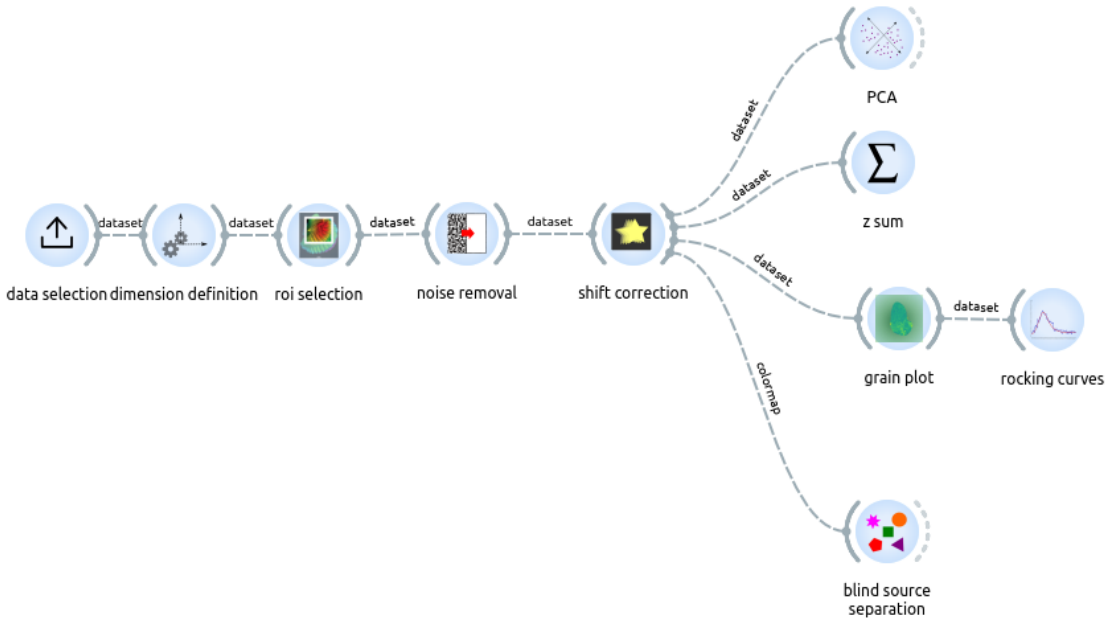
Structure

- Data selection: images taken at different motor positions.
- Preprocessing of the data: techniques like noise removal, region of interest and shift correction.
- Operations for the analysis of the images: rocking curves imaging, mosaicity maps, blind source separation.

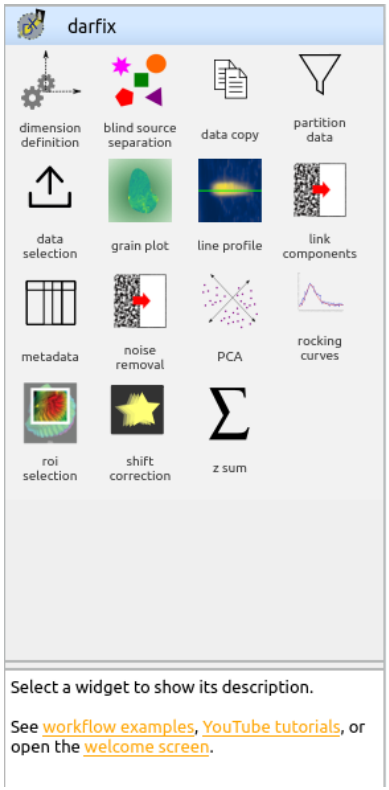
(!) At the end of the presentation a demonstration of darfix will be made.

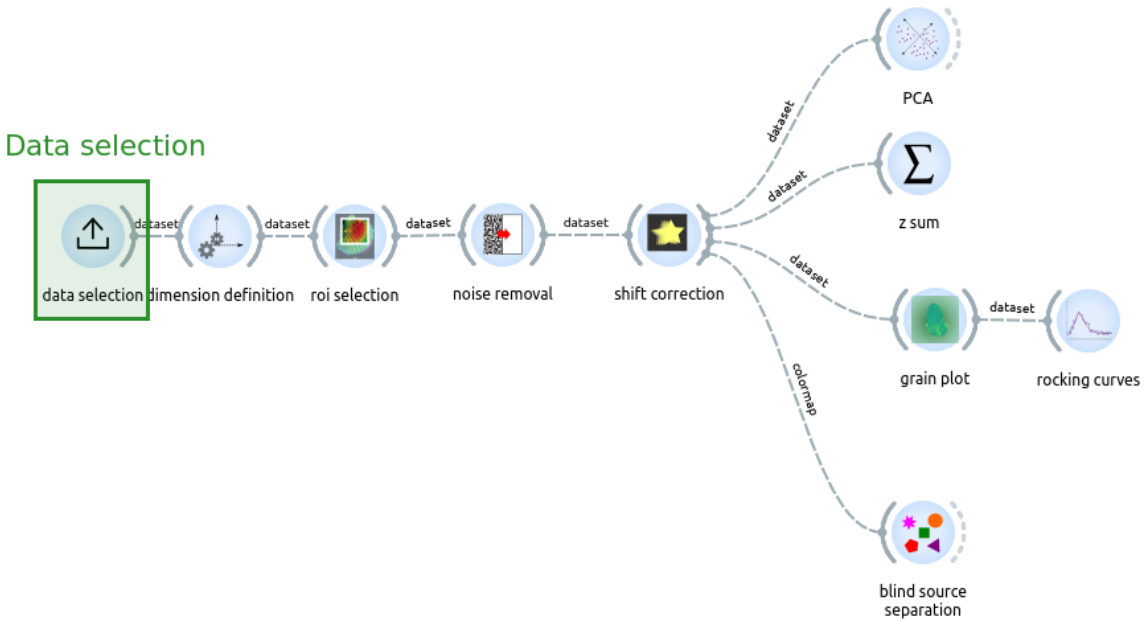
Orange workflow example



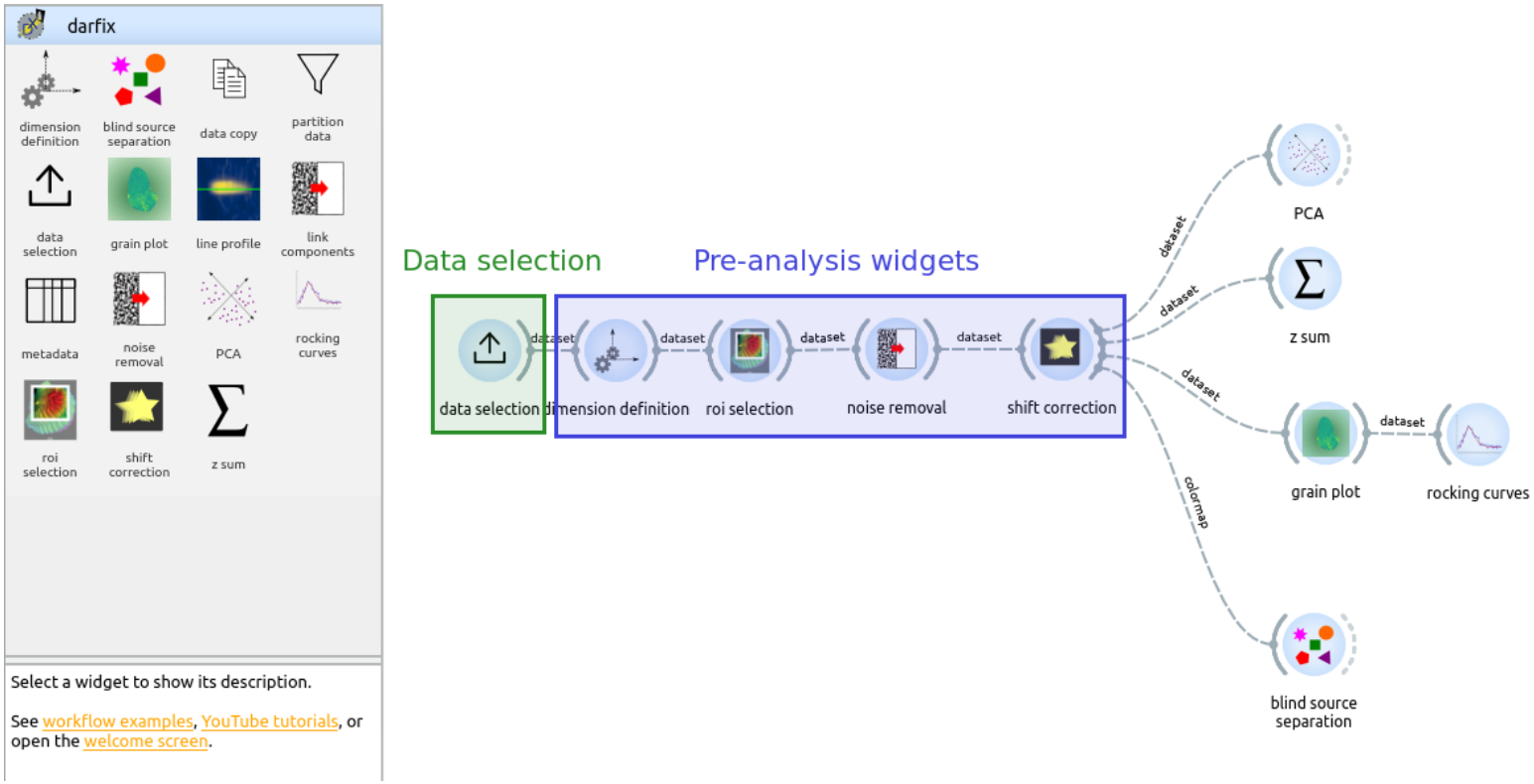


Orange workflow example

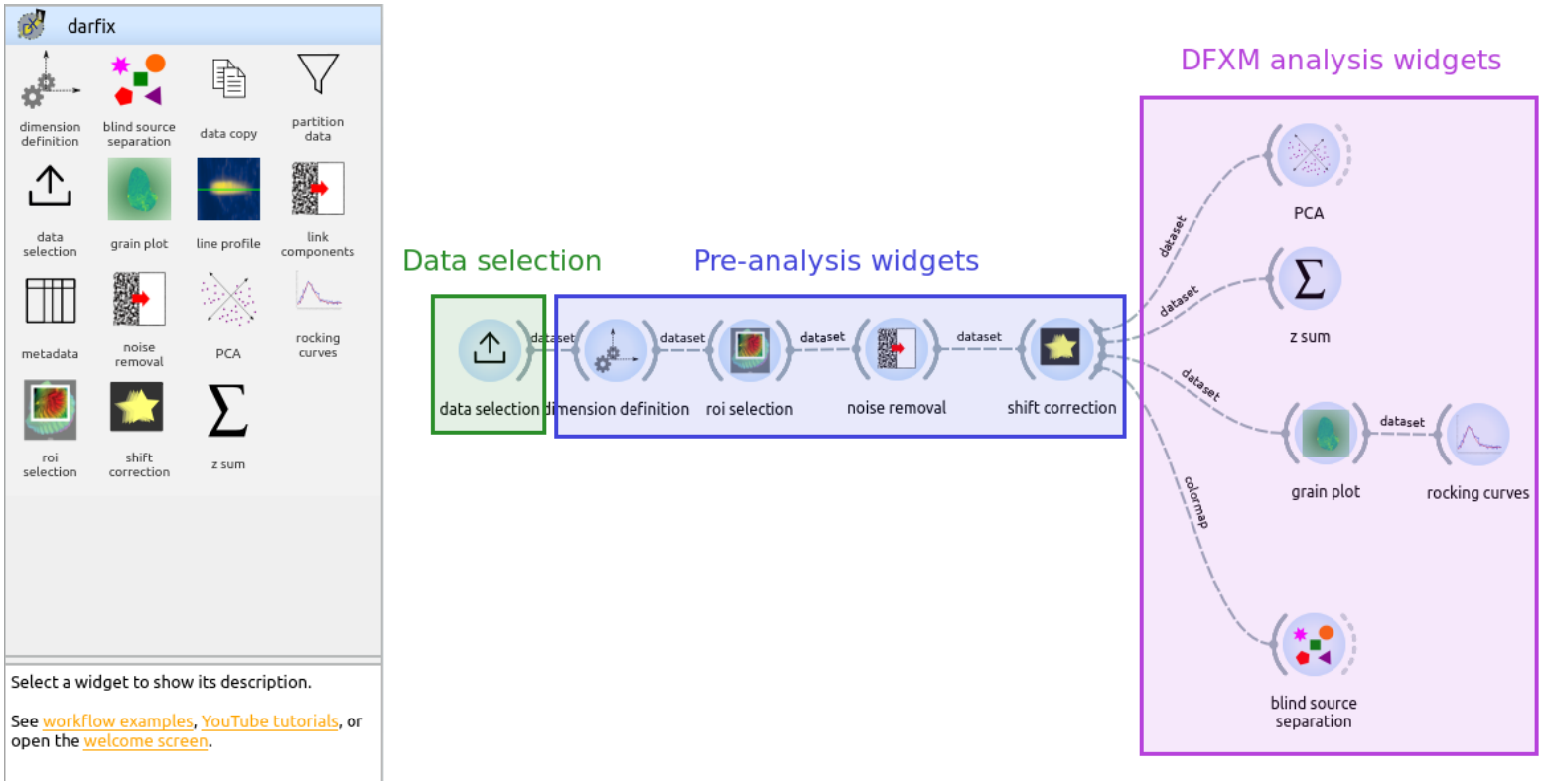




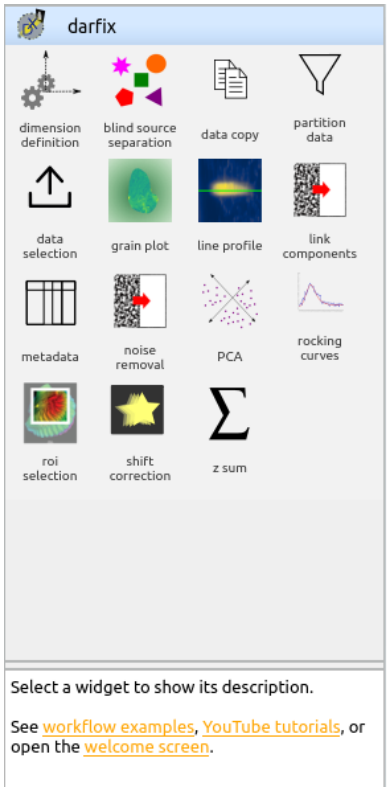
Orange workflow example

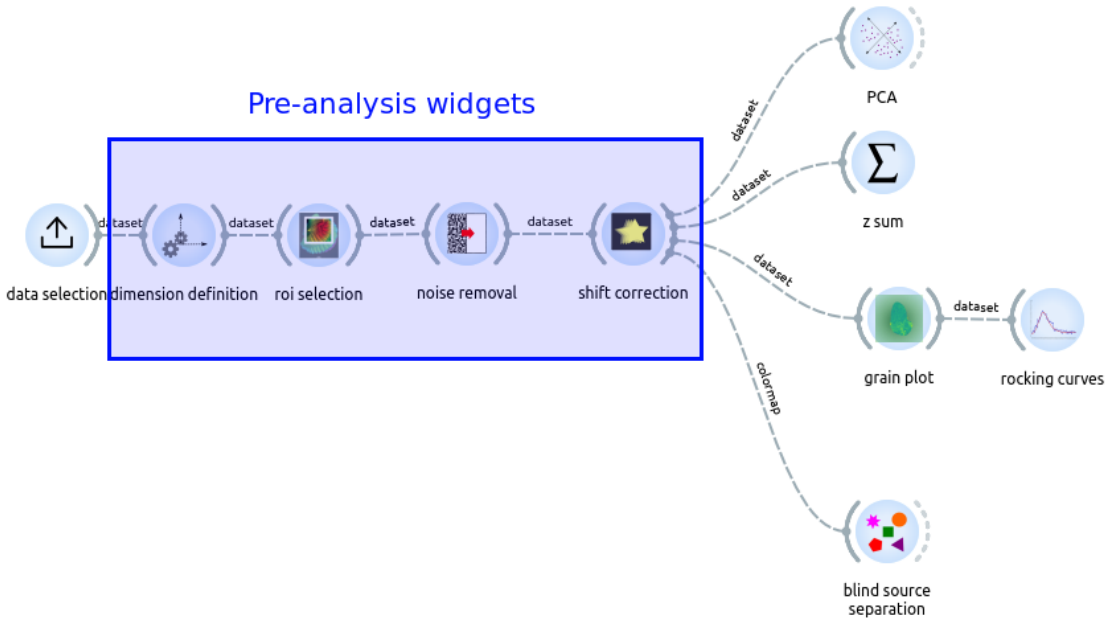


Orange workflow example

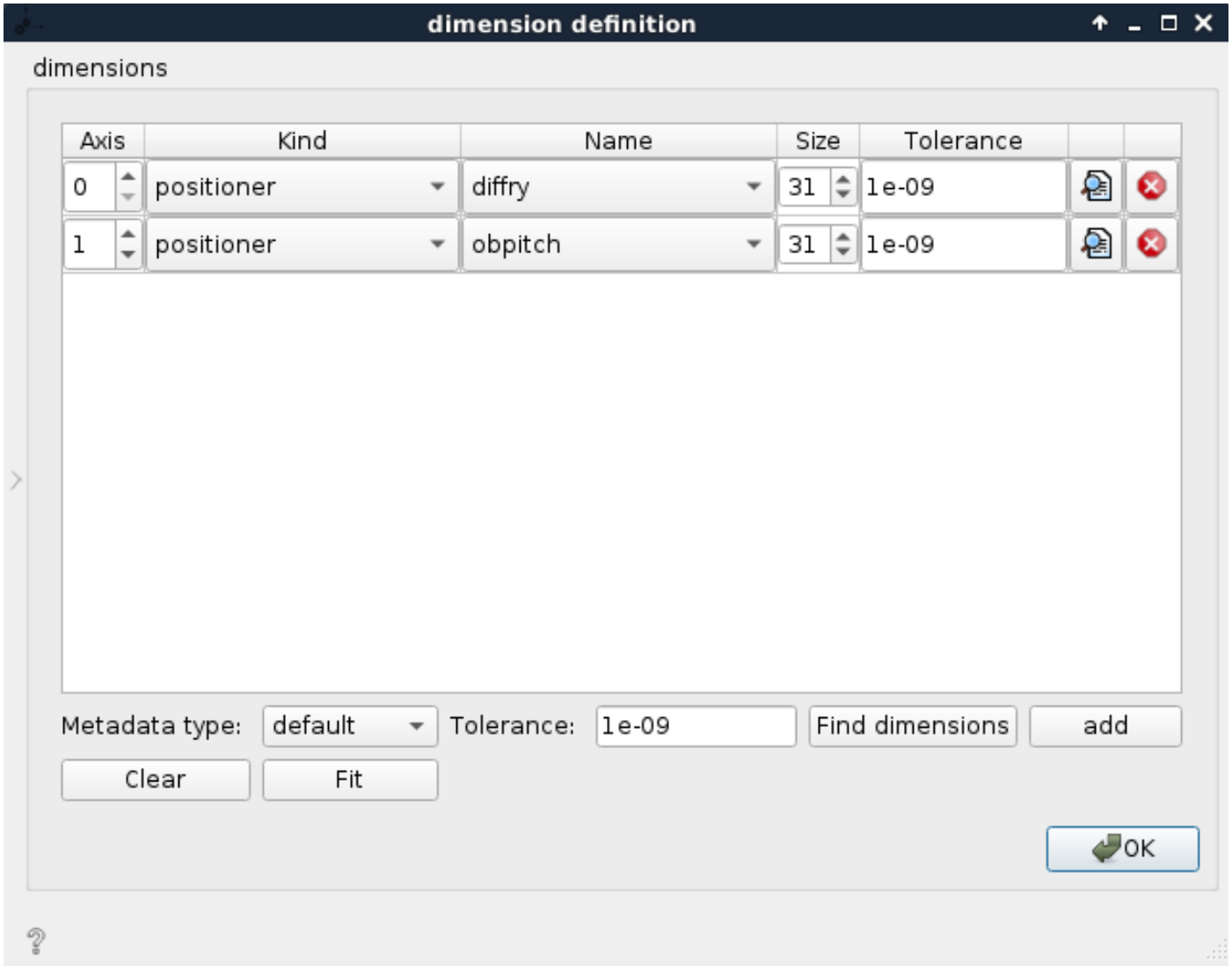


Pre-analysis



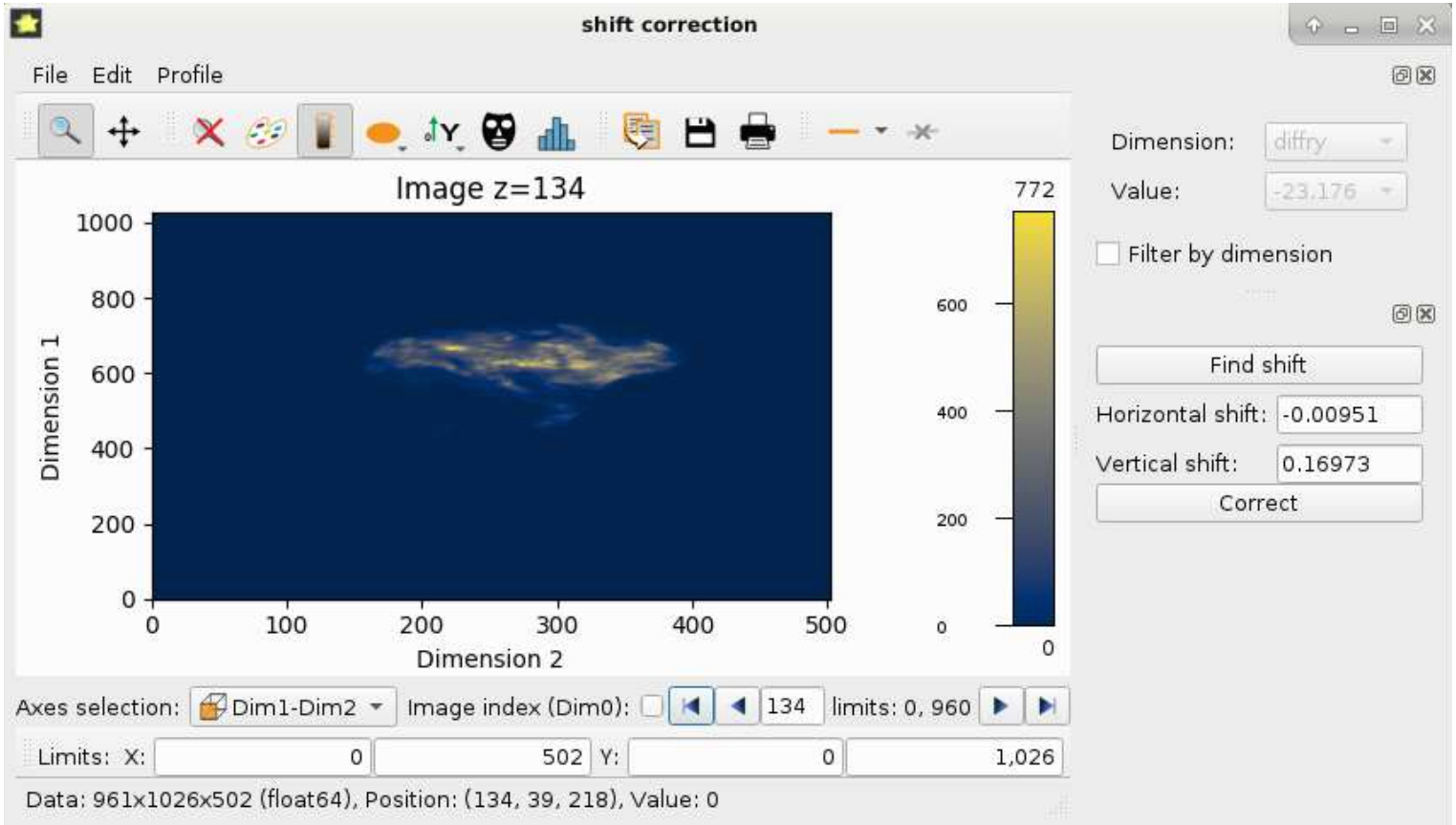


Dimension definition widget



Shift correction

Imperfections in the mechanical alignment can lead to a linear shift through the dataset.



Shift correction

Difference in the z-axis sum of the stack before and after applying the shift correction:

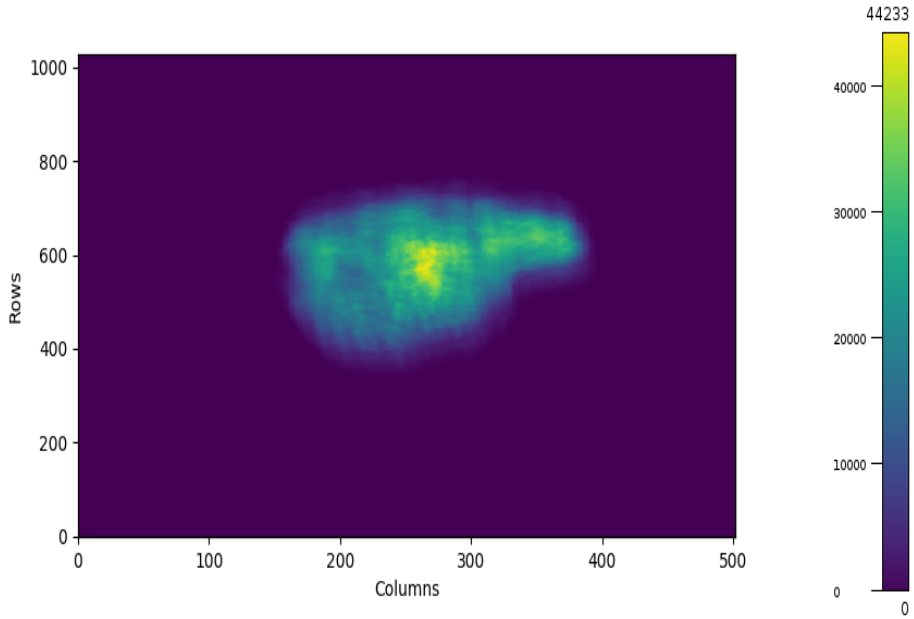


Figure: Before shift correction

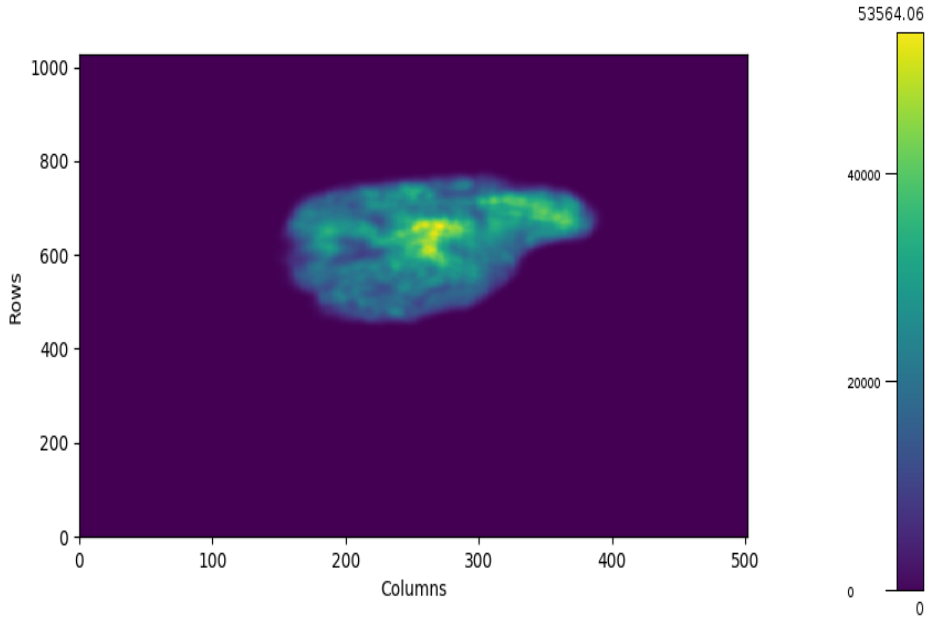
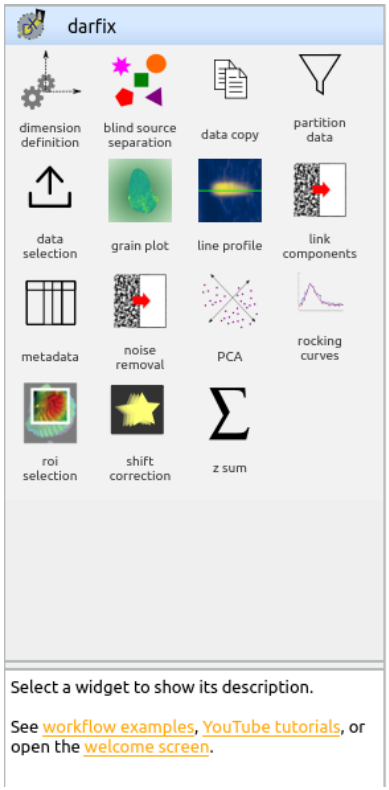
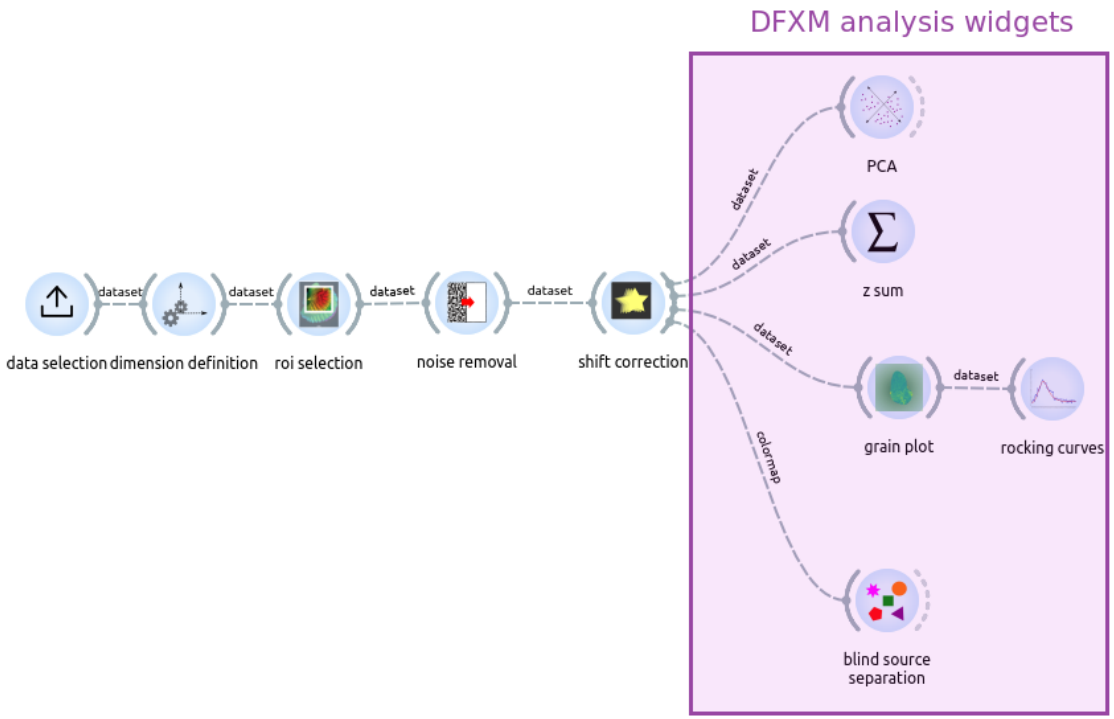


Figure: After shift correction

DFXM analysis





Mosaicity map

The mosaicity map is an hsv image that has the COM of the first motor as hue and the COM of the second motor as saturation.

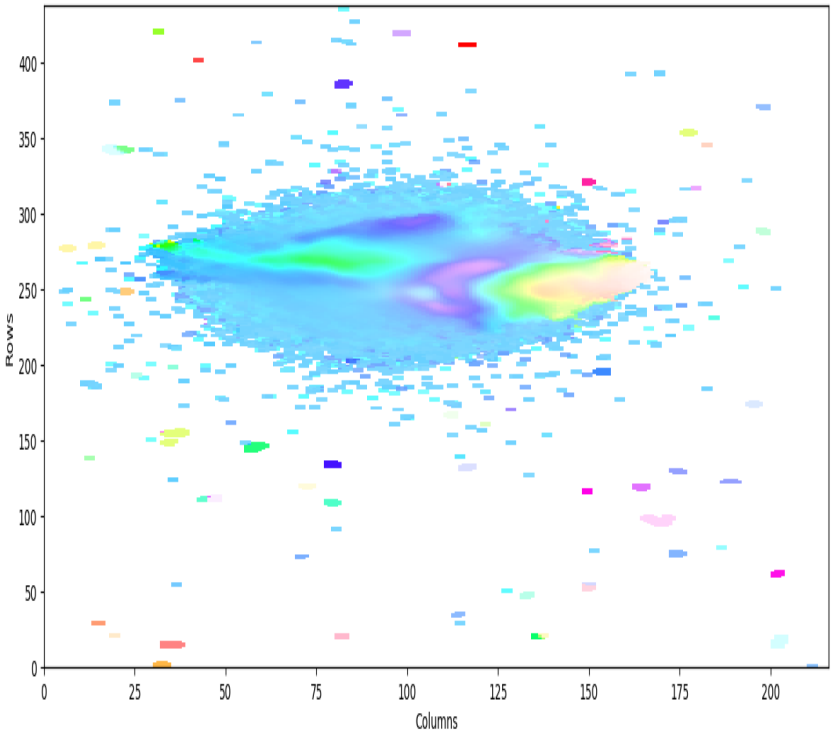


Figure: Mosaicity map

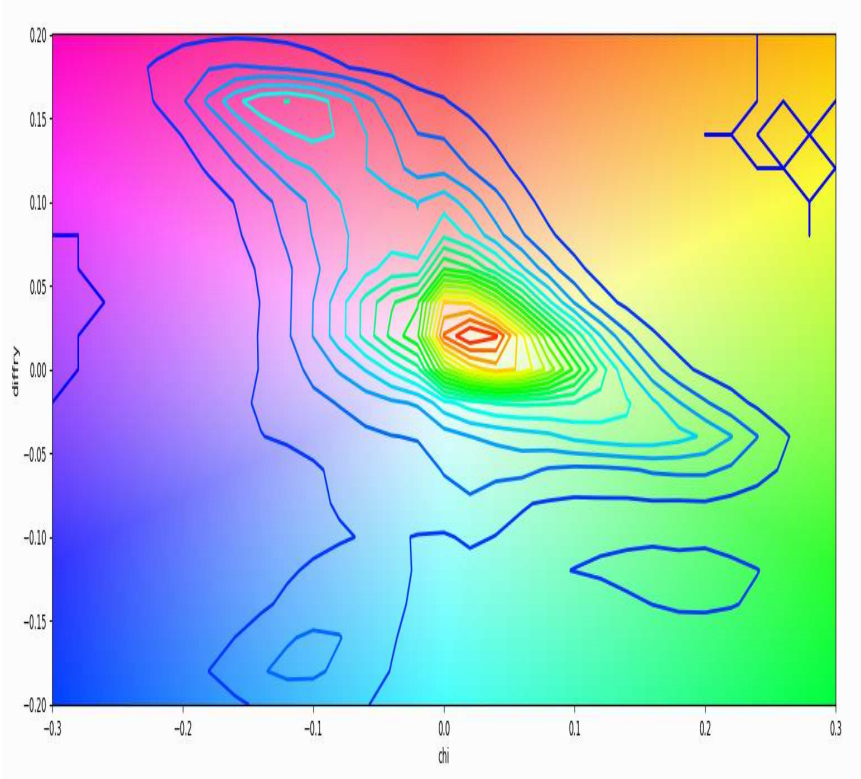


Figure: Orientation distribution contour map

Jupyter example

Dataset

```
[27]: import glob
      import os
      import numpy
      from darfix.core.dataset import Dataset

      dataset = Dataset(_dir="/home/julia/Documents/tests/darfix/reduced_strain")
```

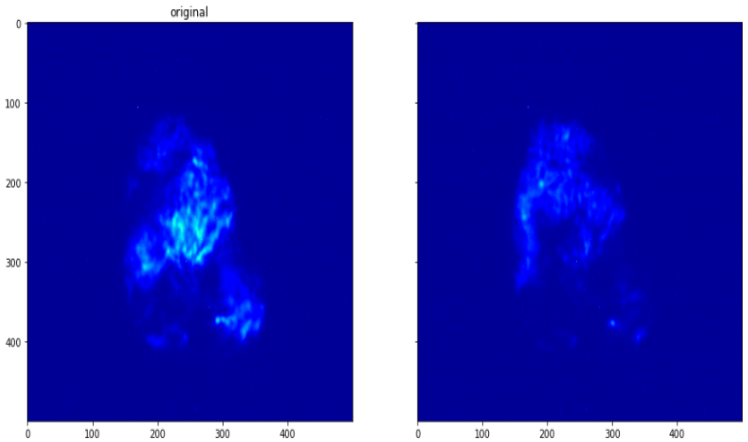
ROI

```
[11]: from darfix.core import roi

      roi_dataset = reshaped_dataset.apply_roi(origin=[900, 750], size=[500, 500])
      # dataset.empty_frames = roi.apply_3D_ROI(dataset.empty_frames, [317, 250], [1170, 991])
      # dataset.dark_frames = roi.apply_3D_ROI(dataset.dark_frames, [317, 250], [1150, 991])
```

```
Applying roi |████████████████████████████████████████████████████████████████| 100.0%
```

```
[12]: plot_comparison(roi_dataset.get_data(13), roi_dataset.get_data(16), "")
```

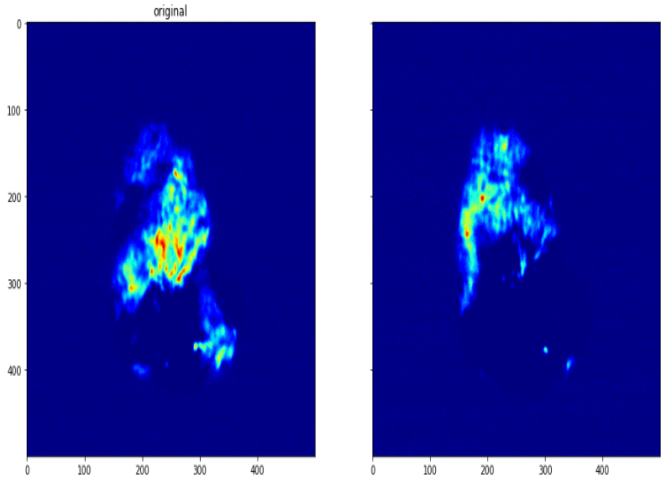


Background subtraction

```
[15]: bs_dataset = roi_dataset.apply_background_subtraction(method="mean")
```

```
Computing mean image |████████████████████████████████████████████████████| 100.0%
Applying background subtraction |████████████████████████████████████████| 100.0%
```

```
[16]: plot_comparison(bs_dataset.get_data(13), bs_dataset.get_data(16), "")
```



Hot Pixel Removal

```
[17]: hp_dataset = bs_dataset.apply_hot_pixel_removal()
```

```
Applying hot pixel removal |████████████████████████████████████████████| 100.0%
```

```
[18]: plot_comparison(hp_dataset.get_data(13), hp_dataset.get_data(16), "")
```

The jupyter notebook can be found in the gitlab repository

The code is open-source and can be downloaded in https://gitlab.esrf.fr/julia.garriga/darfix.

For further questions please e-mail me at julia.garriga@esrf.fr

Thank you for listening!