



HAL
open science

BioImageIT: Integration of image data-management with analysis

Ludovic Leconte, Sylvain Prigent, Léo Maury, Cesar Augusto Valades Cruz,
Jean Salamero, Charles Kervrann

► **To cite this version:**

Ludovic Leconte, Sylvain Prigent, Léo Maury, Cesar Augusto Valades Cruz, Jean Salamero, et al.. BioImageIT: Integration of image data-management with analysis. BioImage Informatics 2021 (poster), Nov 2021, Paris, France. pp.1. hal-03509440

HAL Id: hal-03509440

<https://hal.inria.fr/hal-03509440>

Submitted on 4 Jan 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

BioImage-IT

Integration of data-management with analysis

@BioImageIt

Ludovic Leconte, Sylvain Prigent, Léo Maury César Augusto Valades Cruz, Jean Salamero Charles Kervrann

STED/Serpico team, INRIA-CNRS-Institut Curie

UMR144
75248 Paris

Inria Rennes
35042 Rennes Cedex



Why BioImage-IT?

Currently, there are very few software tools addressing both the data management and analysis.

Problem 1: Data management

- Hard drives: storage without metadata
- Local Servers: storage with poor metadata integration
- ⇒ Data traceability (FAIR principles)

Problem 2: Processing & visualization tools

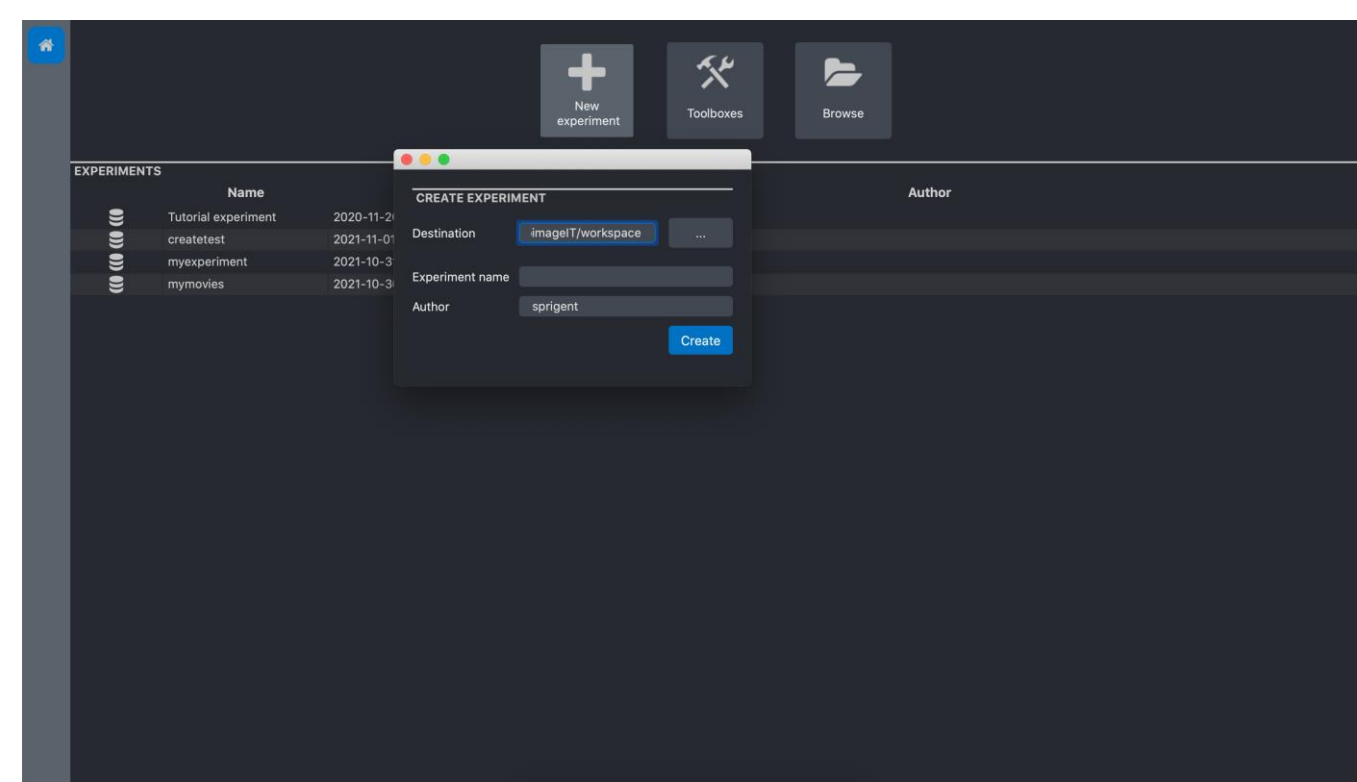
- Many languages
- For different systems (Linux, Apple, Microsoft)
- Needs for Documentations & reference test dataset

Problem 3: link data to processing & visualization

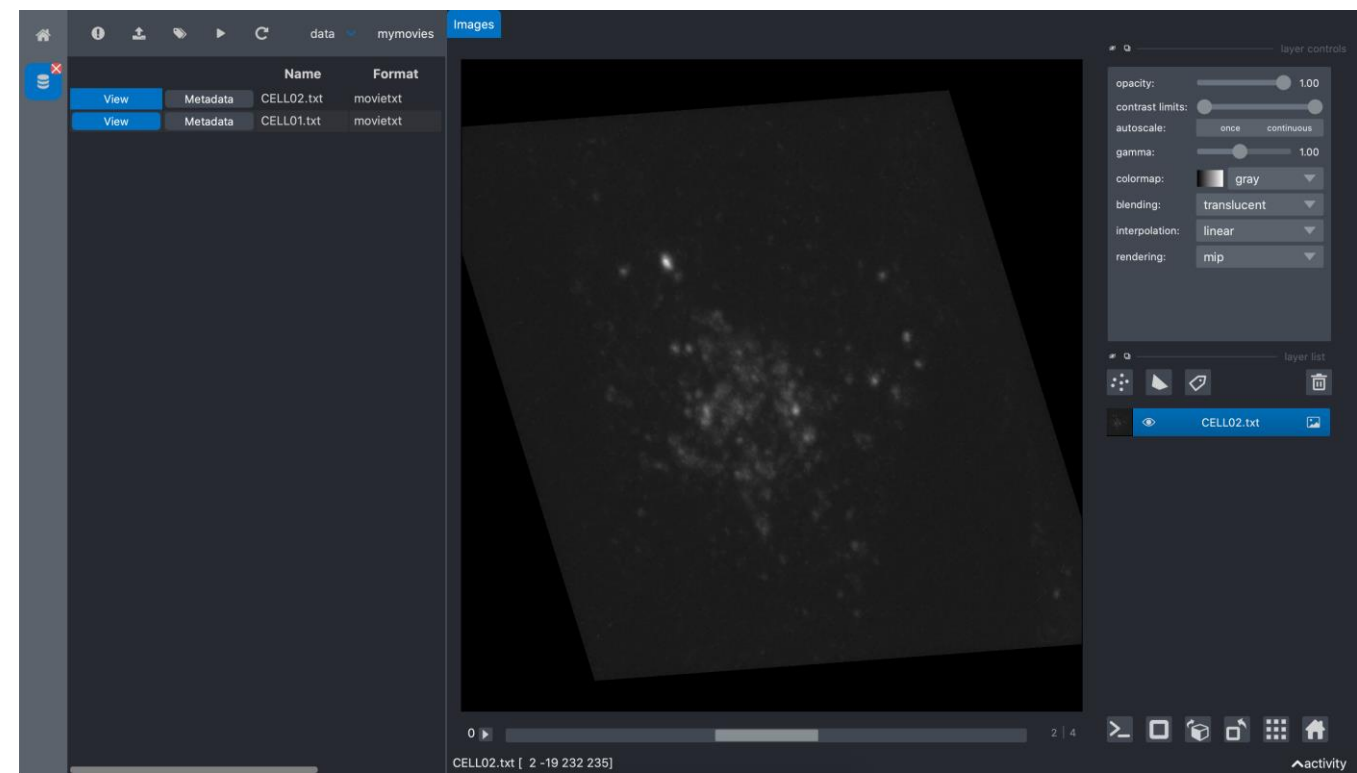
- Manual scripting
- Compatibility of data formats
- Main problem is to keep tracks of each processing steps with data rendering steps

1) Data Management

Referencing a new dataset (new experiment)

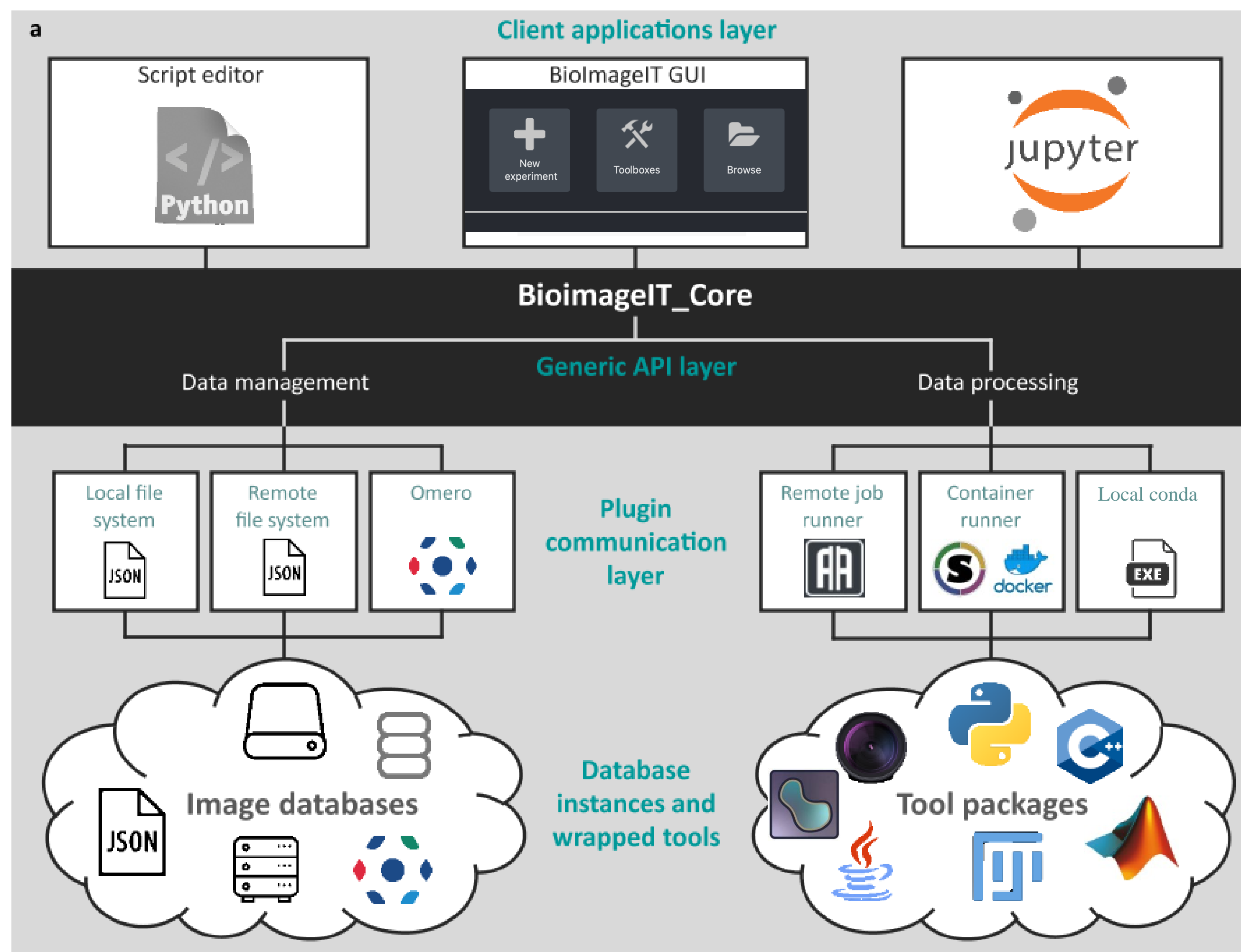


Instant control of dataset integration (use Napari)



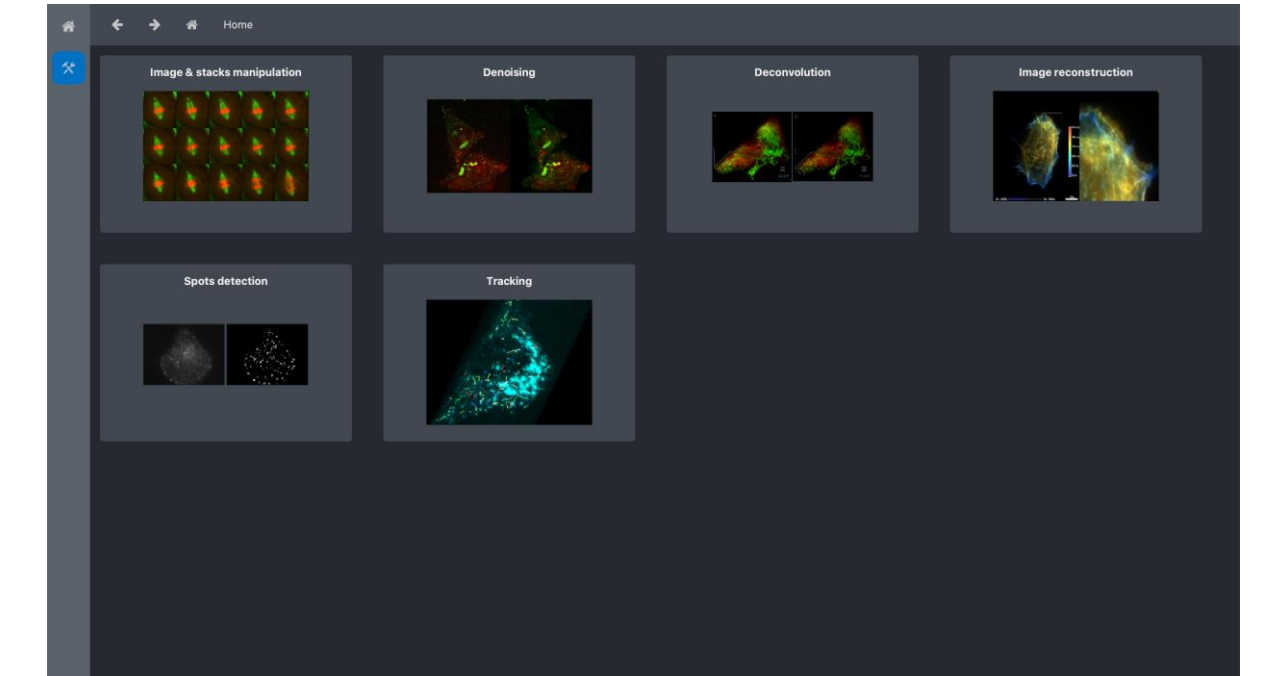
..... (SEE 2 or 3)

BioImage-IT offers a turnkey solution

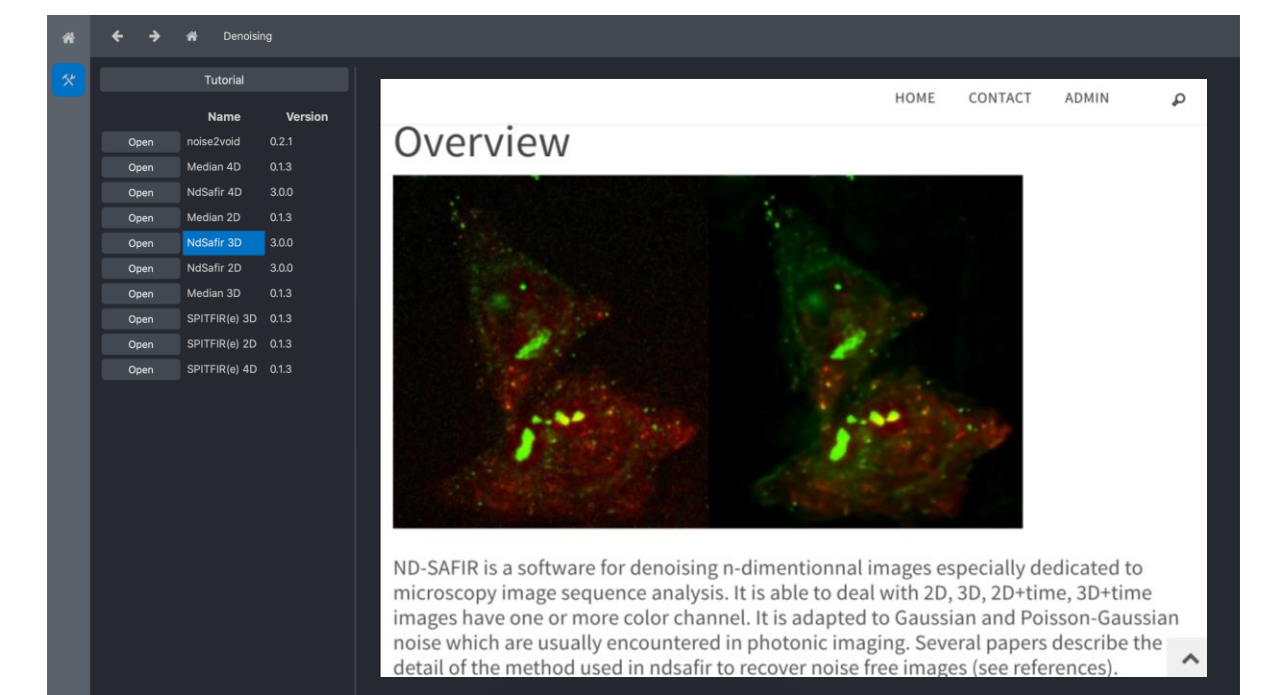


2) User Interface Toolbox

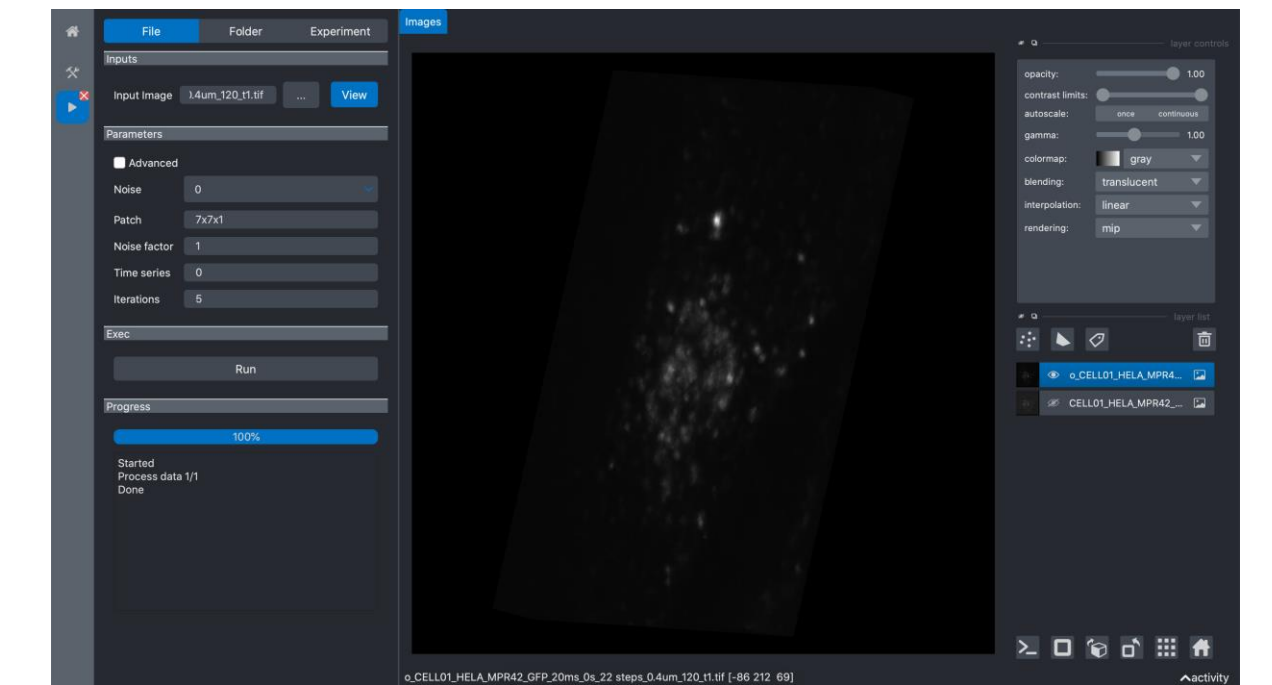
Toolboxes finder



Documentation



Tool runner



3) Developers tools /expert mode

Interact with the python API

```

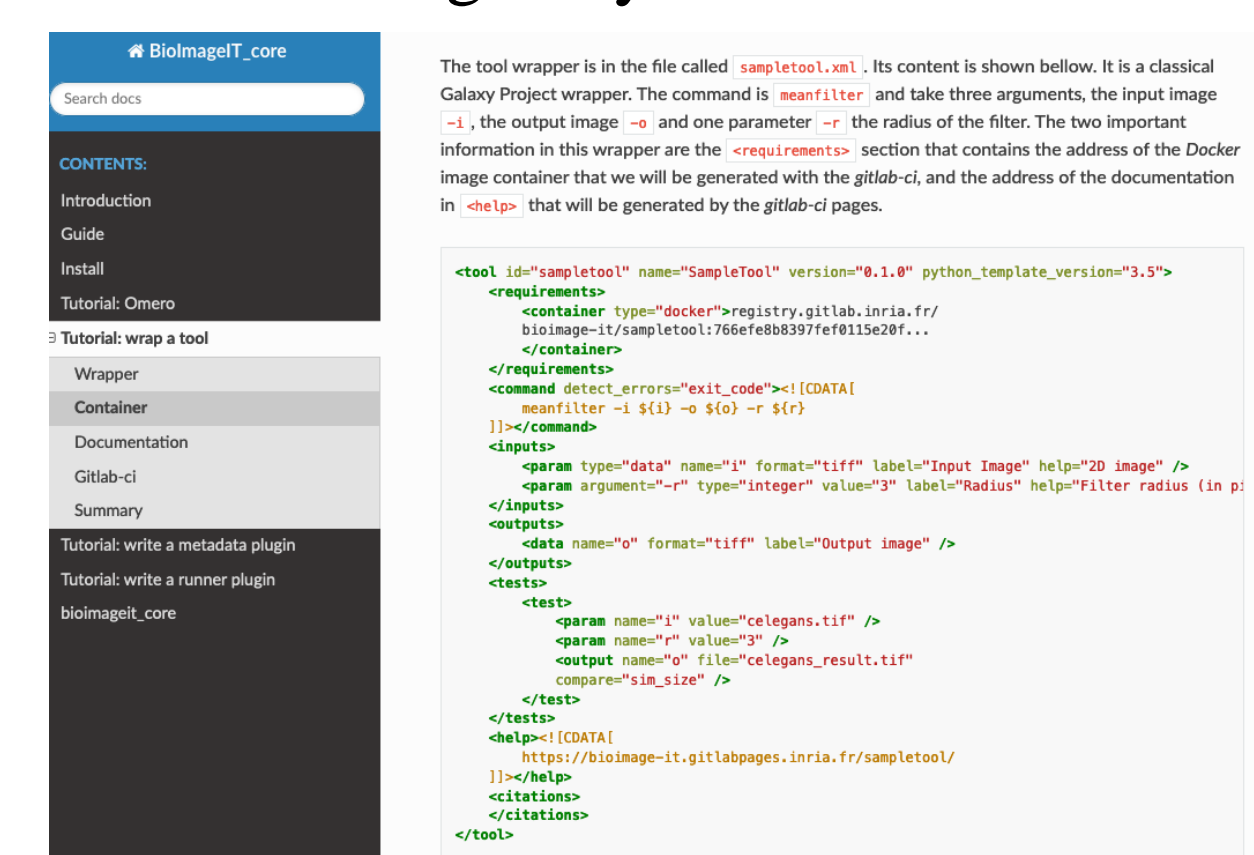
1 from bioimagepy.experiment import Experiment
2
3 # create an experiment
4 my_experiment = Experiment()
5 my_experiment.create(name="myexperiment",
6                     author="Sylvain Prigent",
7                     uri=".")
8
9 print("experiment created")
10
11 # import the data
12 my_experiment.import_dir(dir_uri="./synthetic_data/data/",
13                          filter="*.tif$",
14                          author="Sylvain Prigent",
15                          format="tif",
16                          date="2019-03-17",
17                          copy_data=True)
18
19 # show the experiment directory
20 list_files("./myexperiment")
    
```

Pipeline

```

1 from bioimagepy.experiment import Experiment
2 from bioimagepy.pipeline import Pipeline
3
4 ConfigAccess("../bioimagepy/config_sample.json")
5 my_experiment = Experiment("../userdata/myexperiment")
6 pipeline = Pipeline(my_experiment)
7
8 process1 = pipeline.add_process(ProcessAccess().get('svdeconv2d_v0.1.0'))
9 process1.set_parameters('sigma', 2, 'regularization', 2, 'weighting', 0.1, 'method', 'SV')
10 process1.set_dataset_name('deconv')
11 process1.add_input('I', 'data')
12 process1.run()
13
14 process2 = pipeline.add_process(threshold_particles)
15 process2.set_parameters('threshold', 'Default dark')
16 process2.set_dataset_name('particles')
17 process2.add_input('input', 'deconv')
18 process2.run()
19
20 process3 = pipeline.add_process(ProcessAccess().get('Wilcoxon_v1.0.0'))
21 process3.set_dataset_name('Wilcoxon')
22 process3.add_input('I', 'particles')
23 process3.add_input('y', 'particles')
24 process3.add_input('y', 'Population=population1', 'count')
25 process3.run()
    
```

Integrate your own tools



Example of pipeline for Lattice Light Sheet Microscopy (Using Expert Mode)

LLSM acquisition

Excitation Detection

Objective Orientation

Z stack via sample stage

Skewed raw data

g

MIP 10 μm

h

MIP 10 μm

i

3D 5 μm

2 μm

Conclusion

- Usable for regular users of image tools and for expert Analystist
- Very easy to integrate data management tools (omero, json...) for data tracking
- Integrate your own tools
- Viewing data via Napari

Ressources and website

[Installation and view tutorials](#)