

# ATF Proposal Status Report

## AE124: Simulation-aided Instrument Optimization using Artificial Intelligence and Machine Learning Methods

**Co-PIs:** Maksim Rakitin (DSSI, NSLS-II), Yonghua Du (TES, NSLS-II), Mikhail Fedurin (ATF)

**Team Members:** Thomas Morris, Abigail Giles, Brianna Romasky, Irina Petrushina

2023-03-01

# Team



Maksim Rakitin, NSLS-II



Mikhail Fedurin, ATF



Abigail Giles, NSLS-II



Thomas Morris, NSLS-II



Yonghua Du, NSLS-II

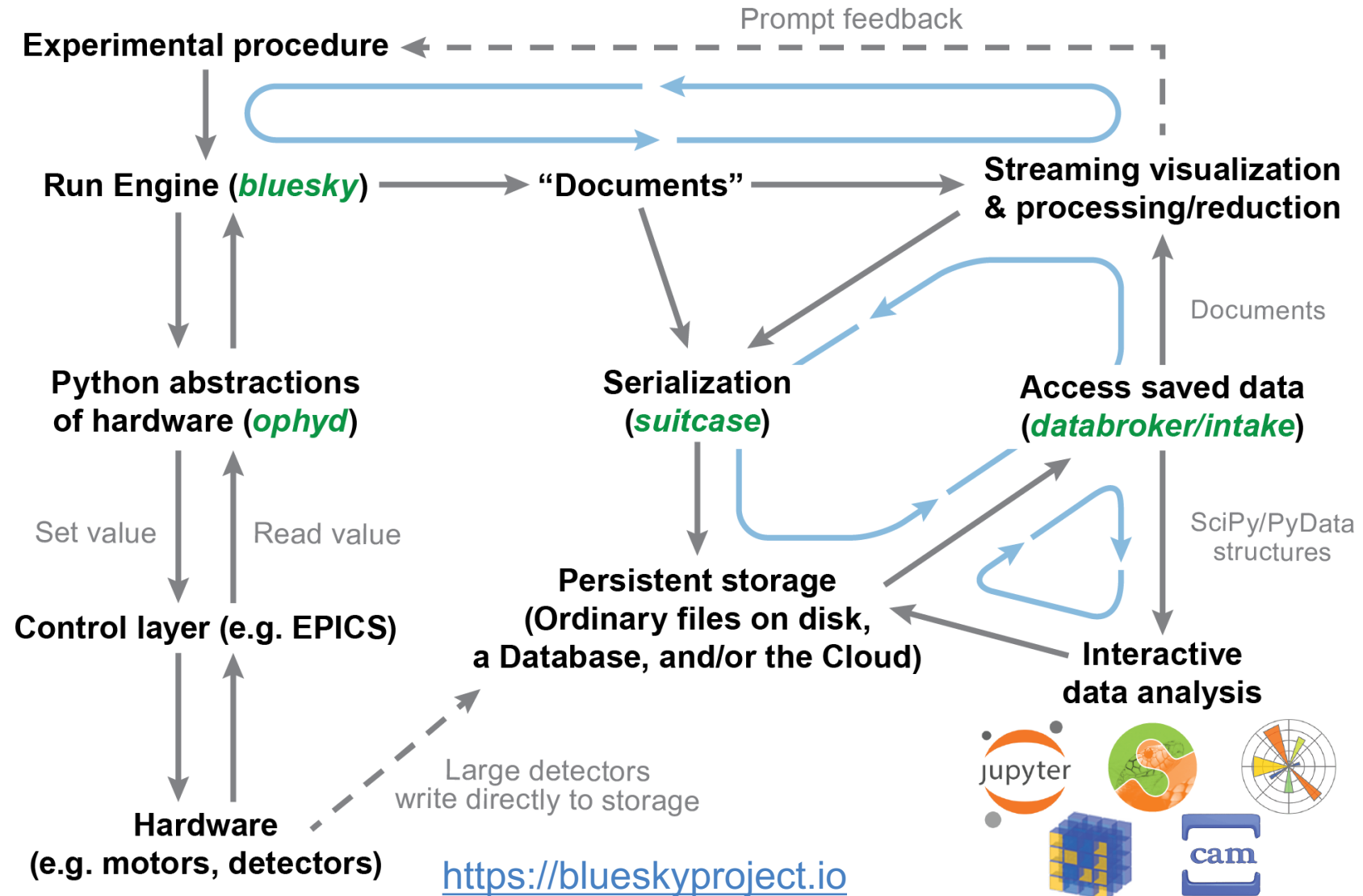


Brianna Romasky, Rutgers/ATF



Irina Petrushina, SBU/ATF

# bluesky Data Collection Ecosystem



# Recent achievements

- Integration of the ATF hardware with the Bluesky data acquisition framework/Ophyd abstractions:
  - Basler cameras integration via Pylon API (ATF production & emulation):
    - <https://github.com/BNL-ATF/ophyd-basler>
  - ATFDB Ophyd support (ATF production & emulation):
    - <https://github.com/BNL-ATF/atfdb>
  - IPython startup files:
    - [https://github.com/BNL-ATF/profile\\_atf](https://github.com/BNL-ATF/profile_atf)
  - MAD-X simulations via Sirepo-Bluesky:
    - <https://nsls-ii.github.io/sirepo-bluesky/notebooks/madx.html>
- Beamline Optimization

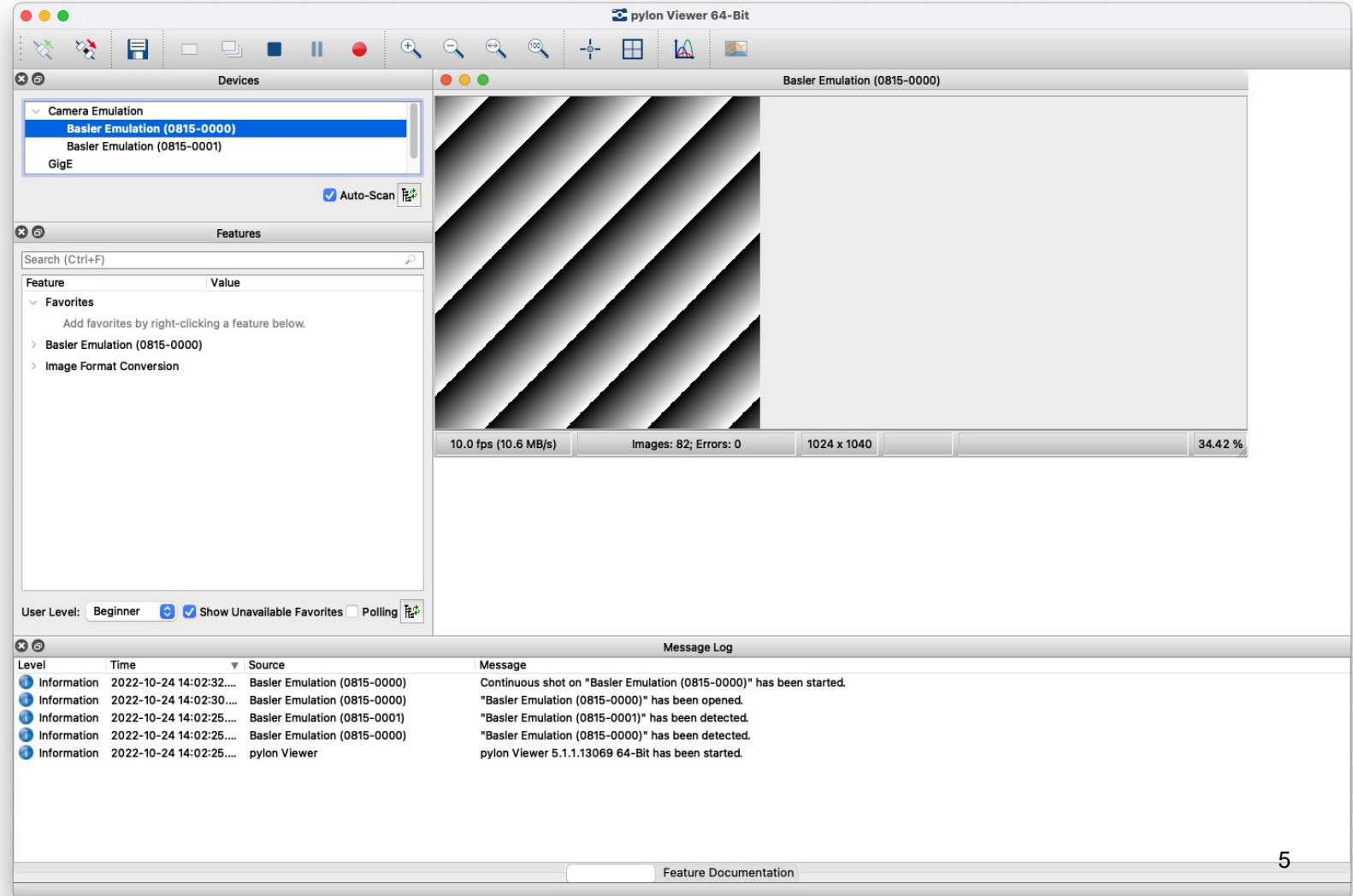
# Basler cameras integration via Pylon API (emulation)

<https://www.baslerweb.com/en/products/basler-pylon-camera-software-suite/>

The Pylon software is available for all major platforms:

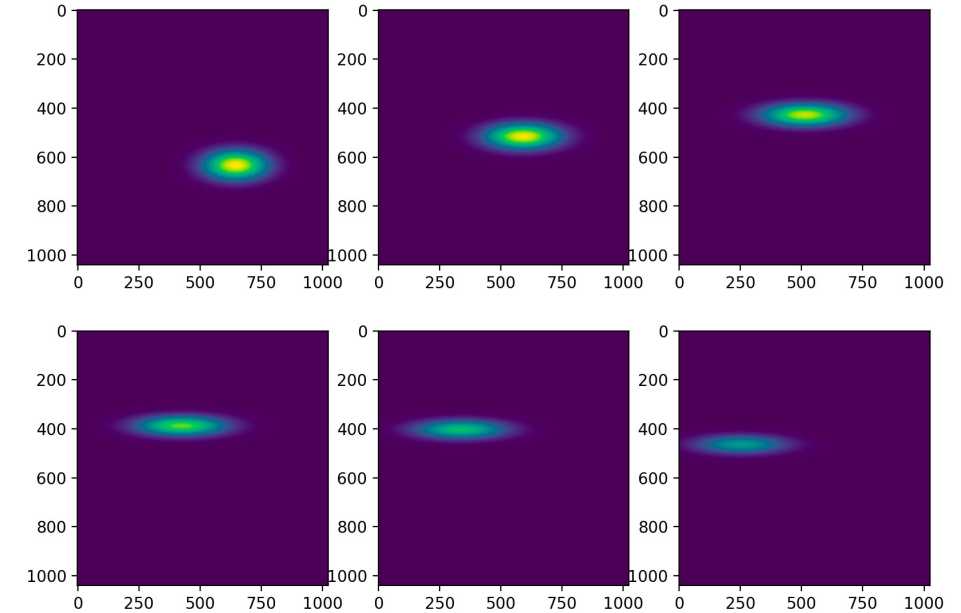
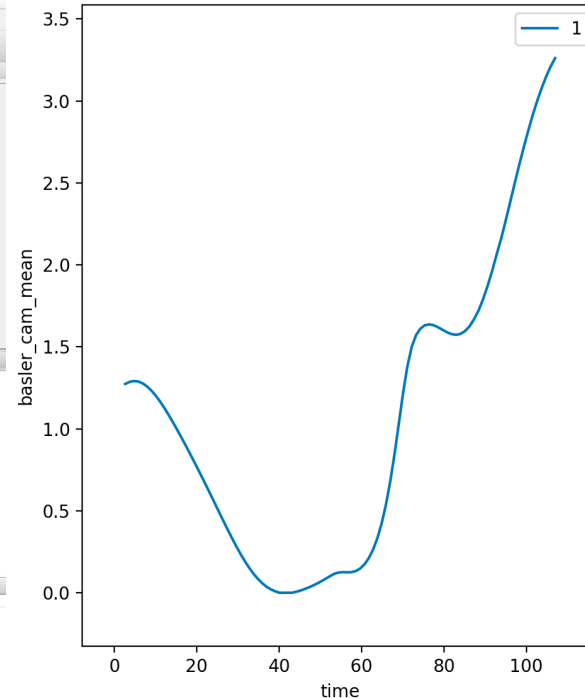
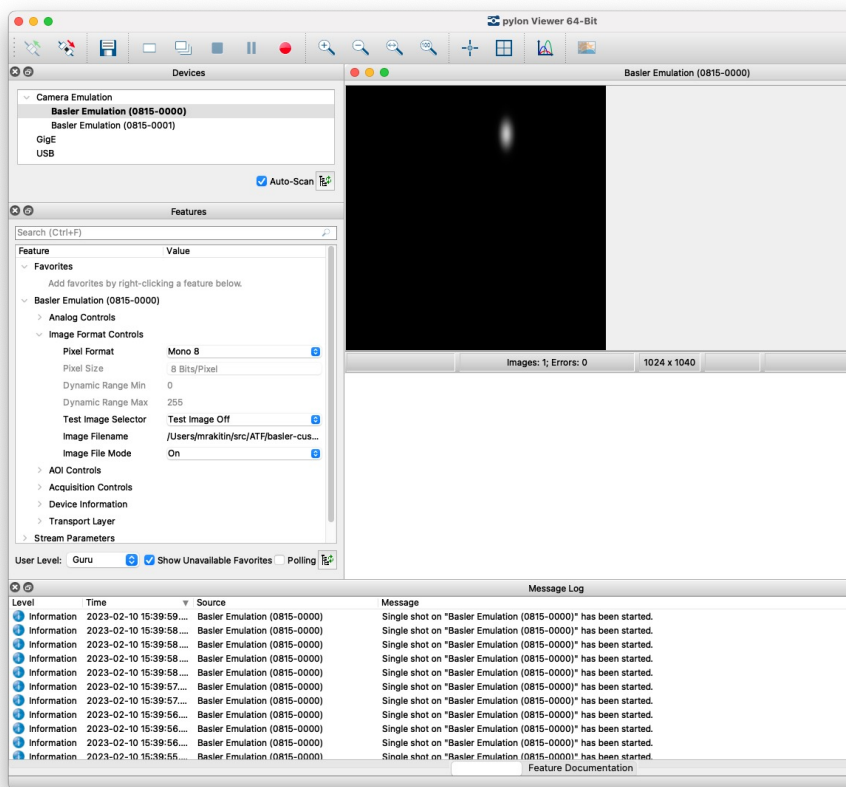
- Linux
- macOS
- Windows

Registration required, but it's free; we use v5.x.



# Basler cameras integration via Pylon API (realistic emulation)

<https://github.com/BNL-ATF/ophyd-basler> \*



Selected images (`data[:, : 4] [:, 6]`) extracted using a Databroker API (the second row's sequence is from left to right).

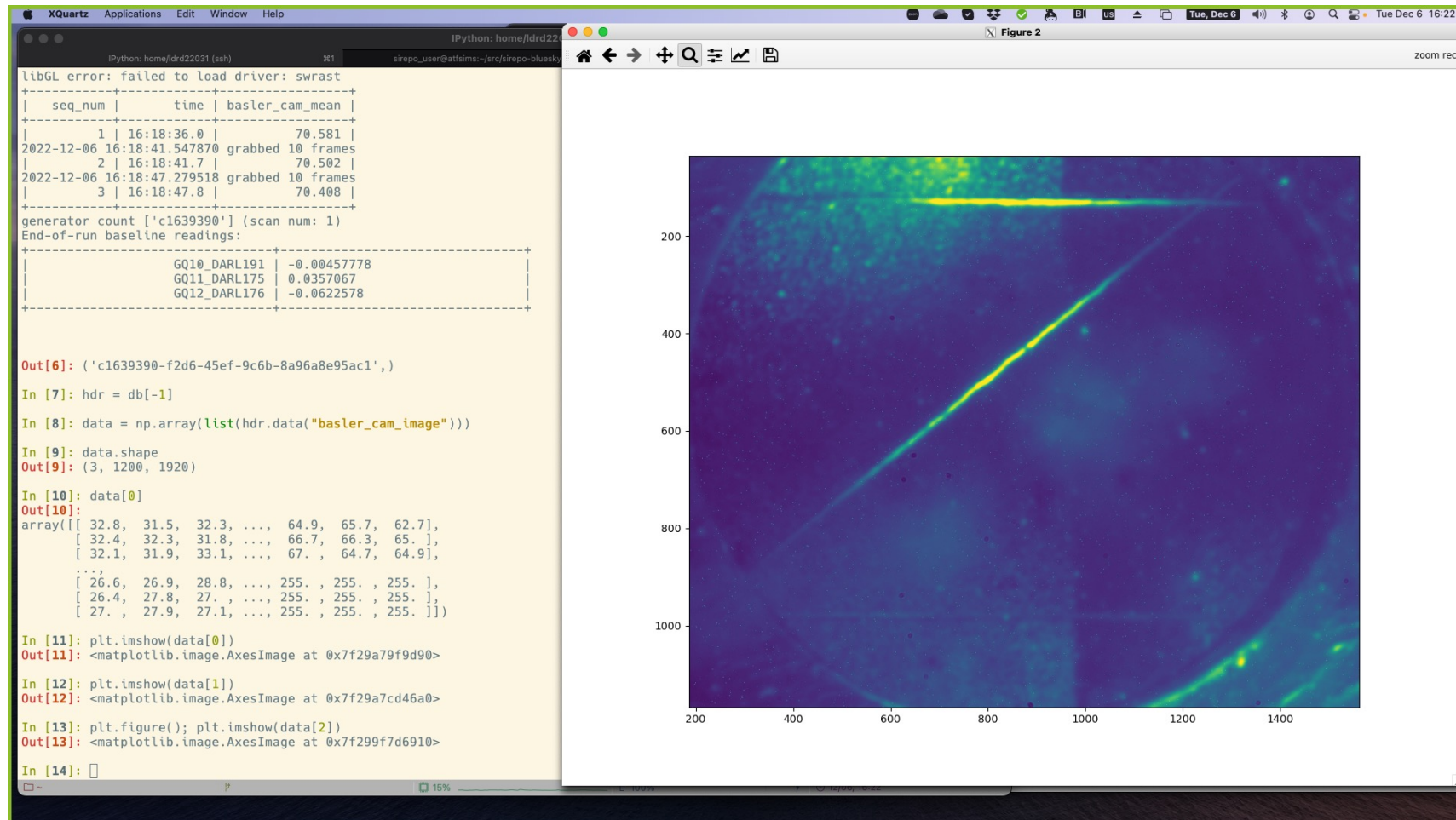
Live plot produced during a Bluesky "count" plan displaying the averaged signal from the emulated Basler camera with predefined images over the course of 100 individual counts.

\* Development of the ophyd-basler interface was led by T. Morris.



# Basler cameras integration via Pylon API (ATF production)

<https://github.com/BNL-ATF/ophyd-basler>



# ATFDB Ophyd support (ATF production & emulation)



[https://www.vista-control.com/about\\_vista.html](https://www.vista-control.com/about_vista.html)

```
class FrameGrabber(Device):
```

```
    db = "FRAME3_DB"
    psname = "FGR3"
    xpos = Cpt(
        ATFSignalR0,
        db=db,
        psname=psname,
        read_suffix="RCX;CENTROID_X",
        write_suffix="RCX;CENTROID_X",
        kind="hinted",
    )
```

```
    ypos = Cpt(
        ATFSignalR0,
        db=db,
        psname=psname,
        read_suffix="RCY;CENTROID_Y",
        write_suffix="RCY;CENTROID_Y",
        kind="hinted",
    )
```

```
    xsig = Cpt(
        ATFSignalR0,
        db=db,
        psname=psname,
        read_suffix="RSX;SIGMA_X",
        write_suffix="RSX;SIGMA_X",
        kind="hinted",
    )
```

Summary

Jobs

- build (3.8)
- build (3.9)
- build (3.10)

Run details

- Usage
- Workflow file

```
fg3 = FrameGrabber(name="fg3")
```

<https://github.com/BNL-ATF/atfdb>

Support of ATFDB components

```
build (3.10)
succeeded last week in 52s

Test with pytest 3s
9 pytest -s -vv
10 + pytest -s -vv
11 ===== test session starts =====
12 platform linux -- Python 3.10.9, pytest-7.2.1, pluggy-1.0.0 -- /opt/hostedtoolcache/Python/3.10.9/x64/bin/python
13 cachedir: .pytest_cache
14 rootdir: /home/runner/work/atfdb/atfdb
15 collecting ... collected 3 items
16
17 atfdb/tests/test_atfdb.py::test_socket_read -----
18 2023-01-31 16:08:24.773846
19 [This module is to be imported from Python 3.x scripts.]
20 ATF DB - NOTE: Connecting to database host localhost on port 5000...
21 -----
22 2023-01-31 16:08:24.774472
23 ATF DB - SUCCESS: Successful connection to database host.
24 PASSED
25 atfdb/tests/test_atfdb.py::test_socket_write PASSED
26 atfdb/tests/test_ophyd.py::test_ophyd_atfsignal 0.6066357757671799
27 {'test': {'value': 0.7294965609839984, 'timestamp': 1675199304.7787242}}
28 0.35779519670907023
29 PASSED
30 2023-01-31 16:08:26.789995
31 ATF DB - NOTE: Socket closed
32
33
34 ===== 3 passed in 2.79s =====

Check socket server logs 0s
1 ▶ Run set -vxexo pipefail
9 cat /tmp/socket.log
10 + cat /tmp/socket.log
11 2023-01-31T16:08:24.774387 Connection from: ('127.0.0.1', 55750)
12 2023-01-31T16:08:24.774909 from connected user: GETCHIDX 'X' 'RT_DATABASE::PTEN20;RAS;RB_CURRENT_SETPT'
13
14 2023-01-31T16:08:24.774961 reply to client: 85
15 2023-01-31T16:08:24.775046 from connected user: GETRS 'X' 85
16
17 2023-01-31T16:08:24.775076 reply to client: 0.2697867137638703
18 2023-01-31T16:08:24.777218 from connected user: GETCHIDX 'X' 'RT_DATABASE::PTEN20;CDS;SET_CURRENT_SETPT'
19
20 2023-01-31T16:08:24.777247 reply to client: 63
```



# IPython startup files

[https://github.com/BNL-ATF/profile\\_atf](https://github.com/BNL-ATF/profile_atf)

BNL-ATF / profile\_atf Public

Edit Pins Unwatch 2 Fork 3 Star 0

Code Issues 1 Pull requests 2 Actions Projects Wiki Security Insights Settings

main 3 branches 0 tags

Go to file Add file Code

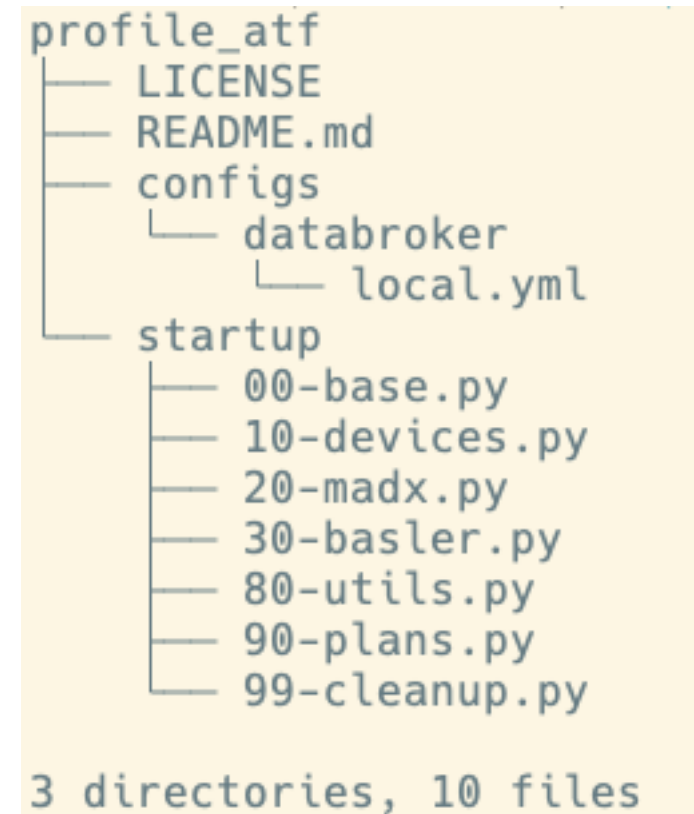
**About**

ATF IPython startup files for data collection via sockets

ipython socket-client bluesky  
pypylon basler-camera ophyd  
sirepo-bluesky

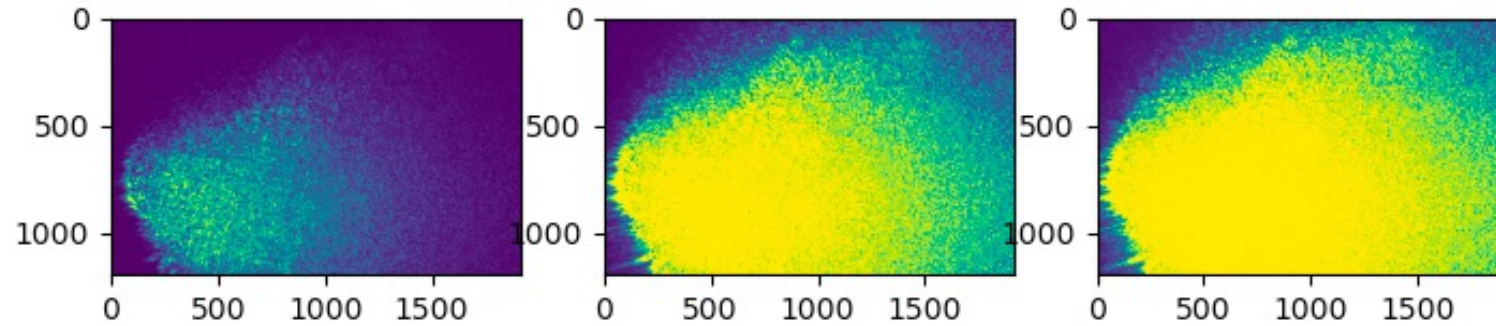
...	BriannaRomasky Merge pull request #10 from BNL-ATF/dev-20230120 ...	6488f2c 2 weeks ago	63 commits
...	.ci	CI: upload artifacts; misc. fixes and docs additions	2 weeks ago
...	.github/workflows	CI: add pre-commit workflow for style checks	2 weeks ago
...	configs/databroker	Add startup files and the atf_db lib	4 months ago
...	startup	Some usability improvements and style fixes	2 weeks ago
...	.gitignore	Update .gitignore	4 months ago
...	.pre-commit-config.yaml	CI: upload artifacts; misc. fixes and docs additions	2 weeks ago
...	LICENSE	Update license holder	4 months ago
...	README.md	CI: upload artifacts; misc. fixes and docs additions	2 weeks ago

Readme  
BSD-3-Clause license  
0 stars  
2 watching  
3 forks

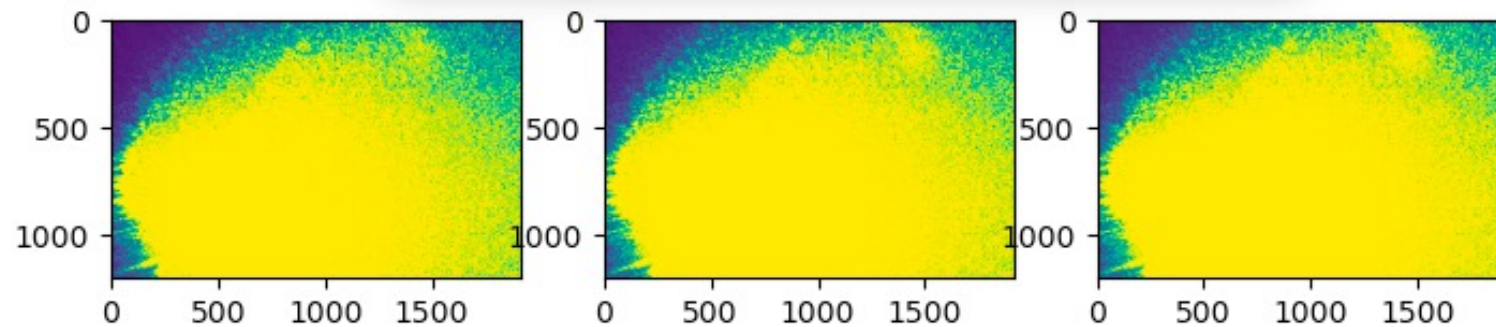


# Scans with Bluesky (controlled over Pylon API & ATFDB)

[https://github.com/BNL-ATF/profile\\_atf](https://github.com/BNL-ATF/profile_atf)



```
RE(bp.scan([GPOP13], HeNe1, 0, 5, 6))
```



Results of the Bluesky scan performed at ATF where the *HeNe* laser intensity was captured by a Basler camera as the laser intensity increased. Exposure time: 0.667 s

# MAD-X simulations via Sirepo-Bluesky

<https://nsls-ii.github.io/sirepo-bluesky/notebooks/madx.html>

```
%run -i ../../../../examples/prepare_flyer_env.py
```

```
import matplotlib.pyplot as plt
```

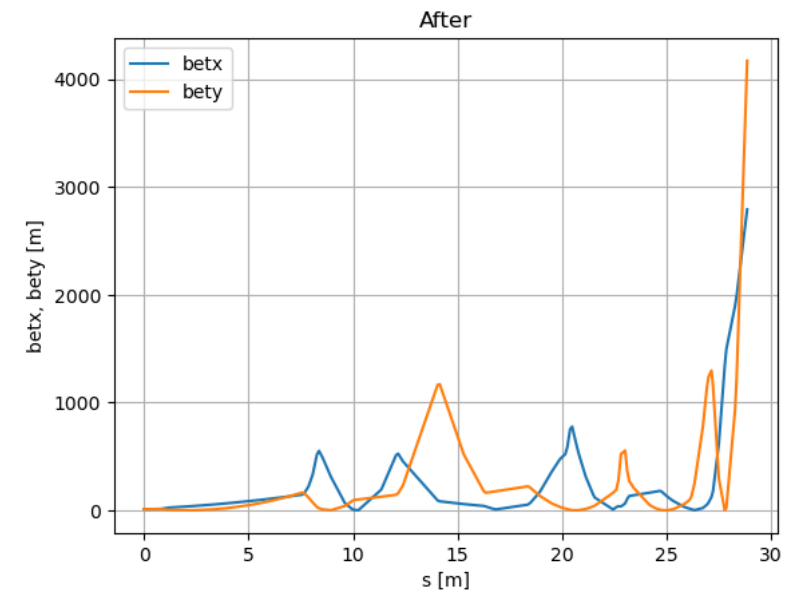
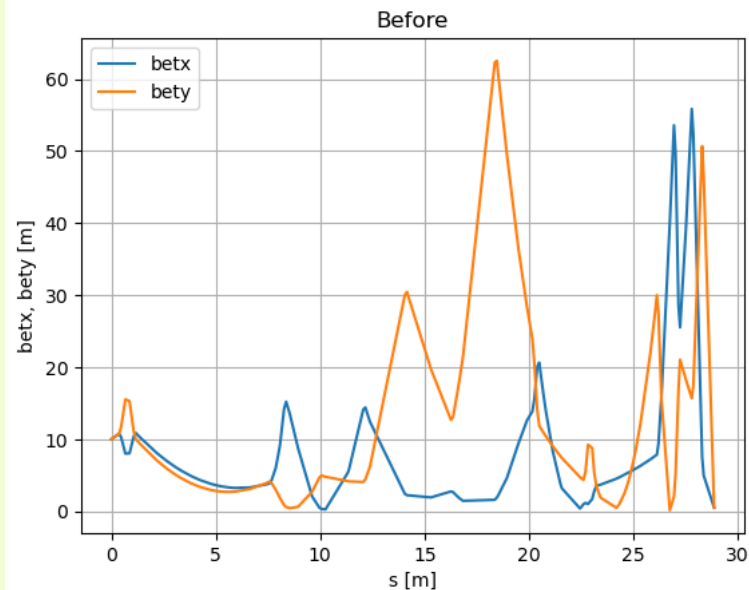
```
from sirepo_bluesky.sirepo_bluesky import SirepoBluesky
from sirepo_bluesky.madx_flyer import MADXFlyer
from sirepo_bluesky.sirepo_ophyd import create_classes
```

```
connection = SirepoBluesky("http://localhost:8000")
```

```
data, schema = connection.auth("madx", "00000002")
classes, objects = create_classes(connection.data,
                                  connection=connection,
                                  extra_model_fields=["rpnVariables", "commands"])
globals().update(**objects)
```

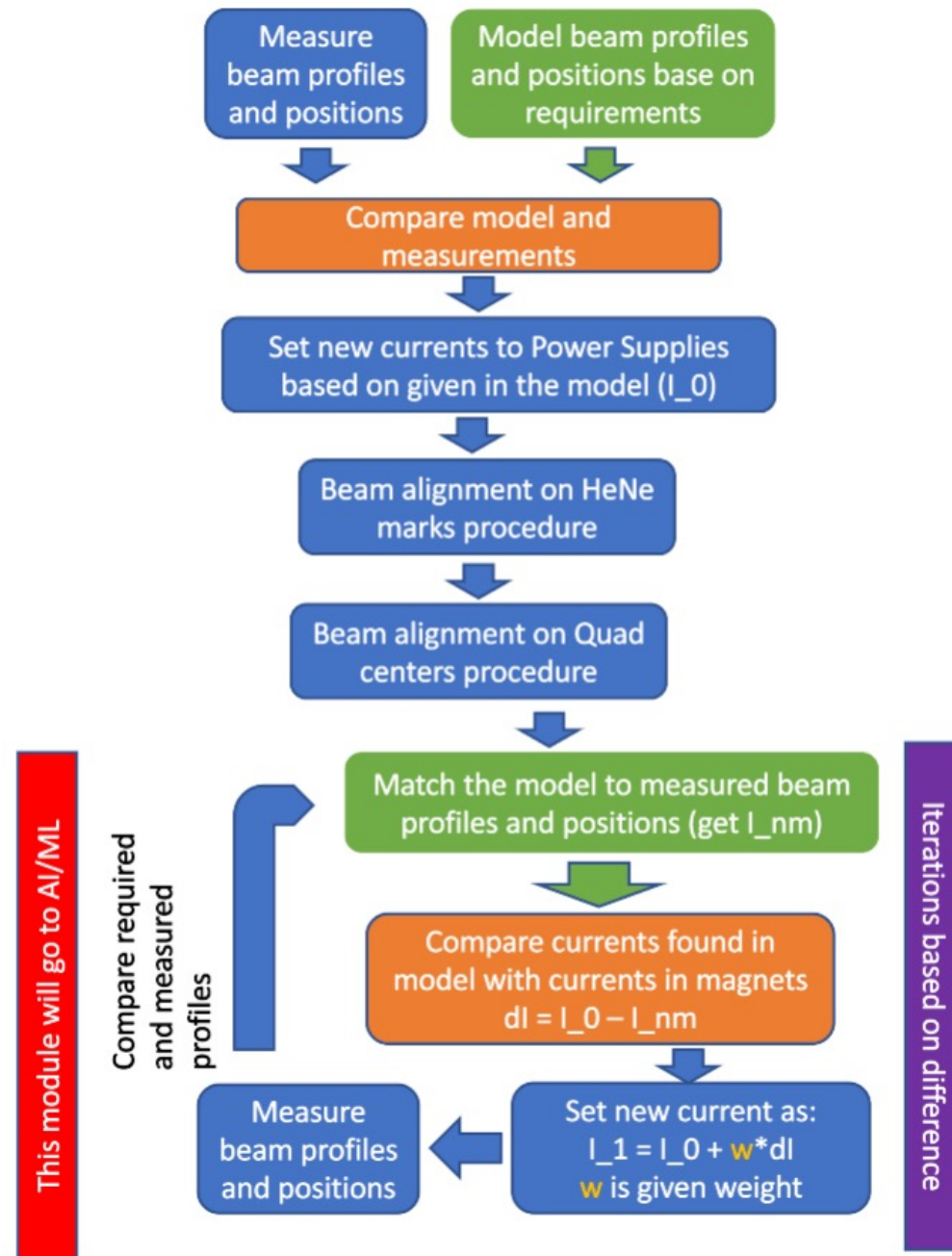
```
madx_flyer = MADXFlyer(connection=connection,
                       root_dir="/tmp/sirepo-bluesky-data",
                       report="elementAnimation250-20")
```

```
(uid1,) = RE(bp.fly([madx_flyer]))
hdr1 = db[uid1]
tbl1 = hdr1.table(stream_name="madx_flyer", fill=True)
print(tbl1)
```

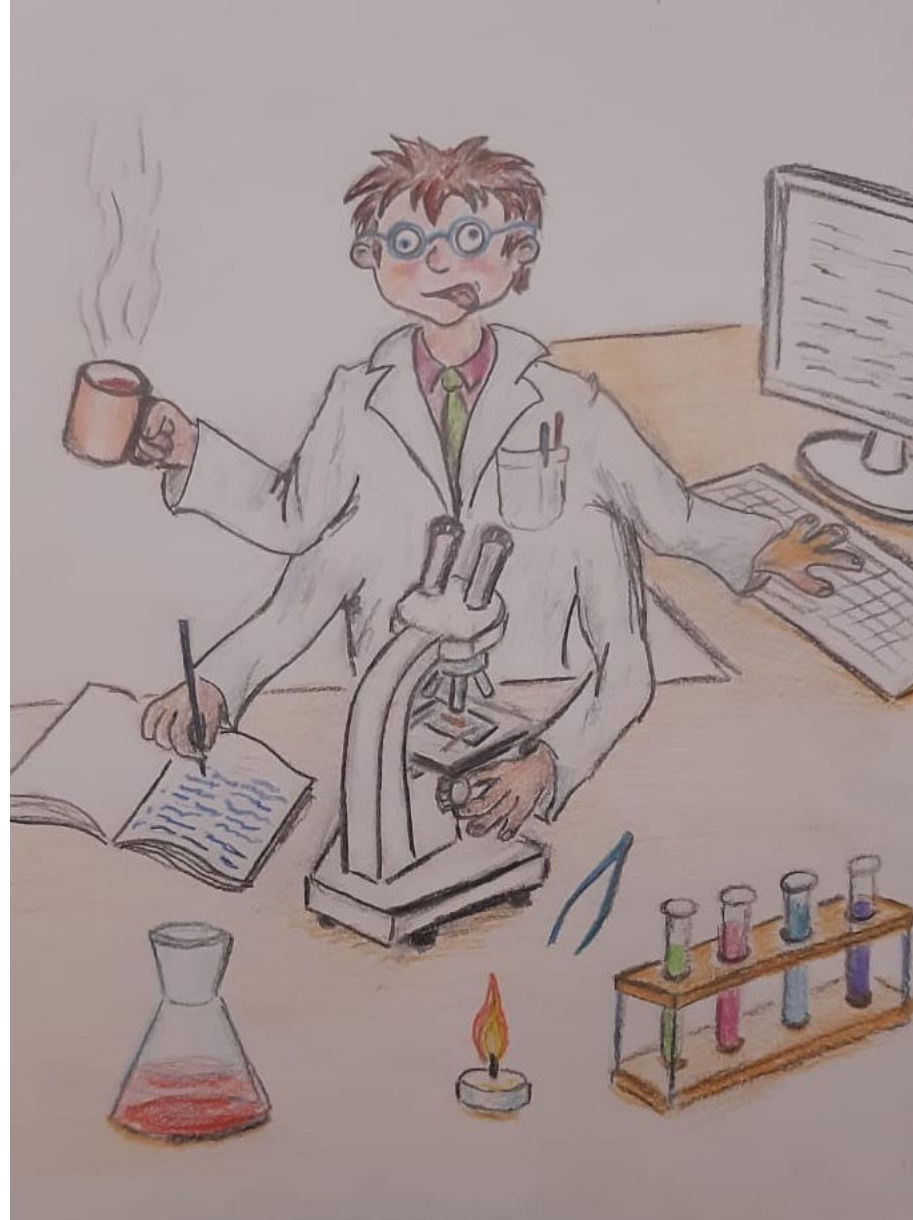


# Mix:

- \* Vista
- \* Basler
- \* Sirepo/MAD-X

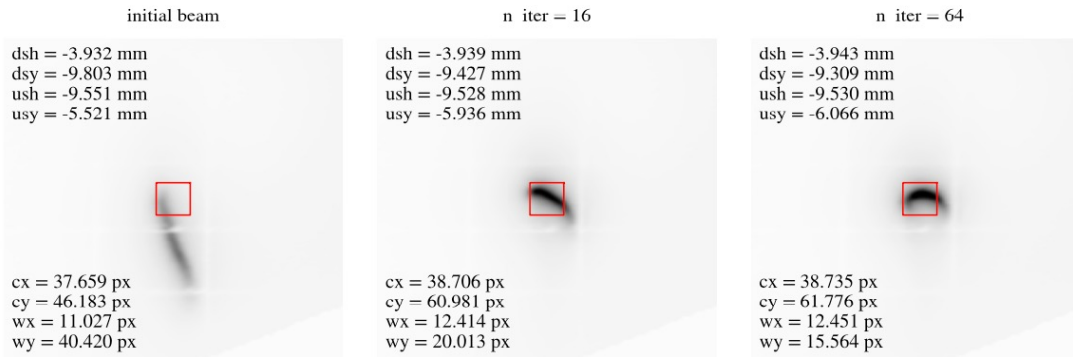
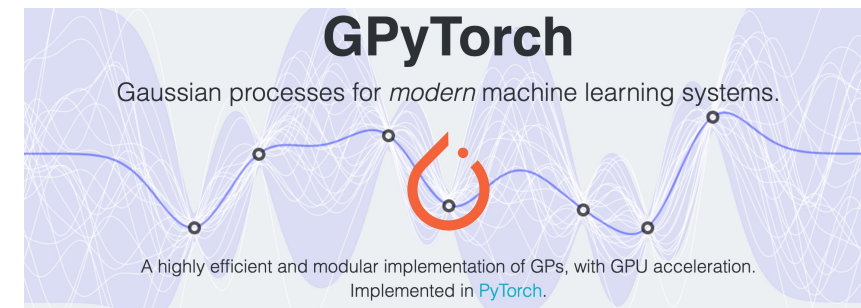


# Multi-dimensional optimization



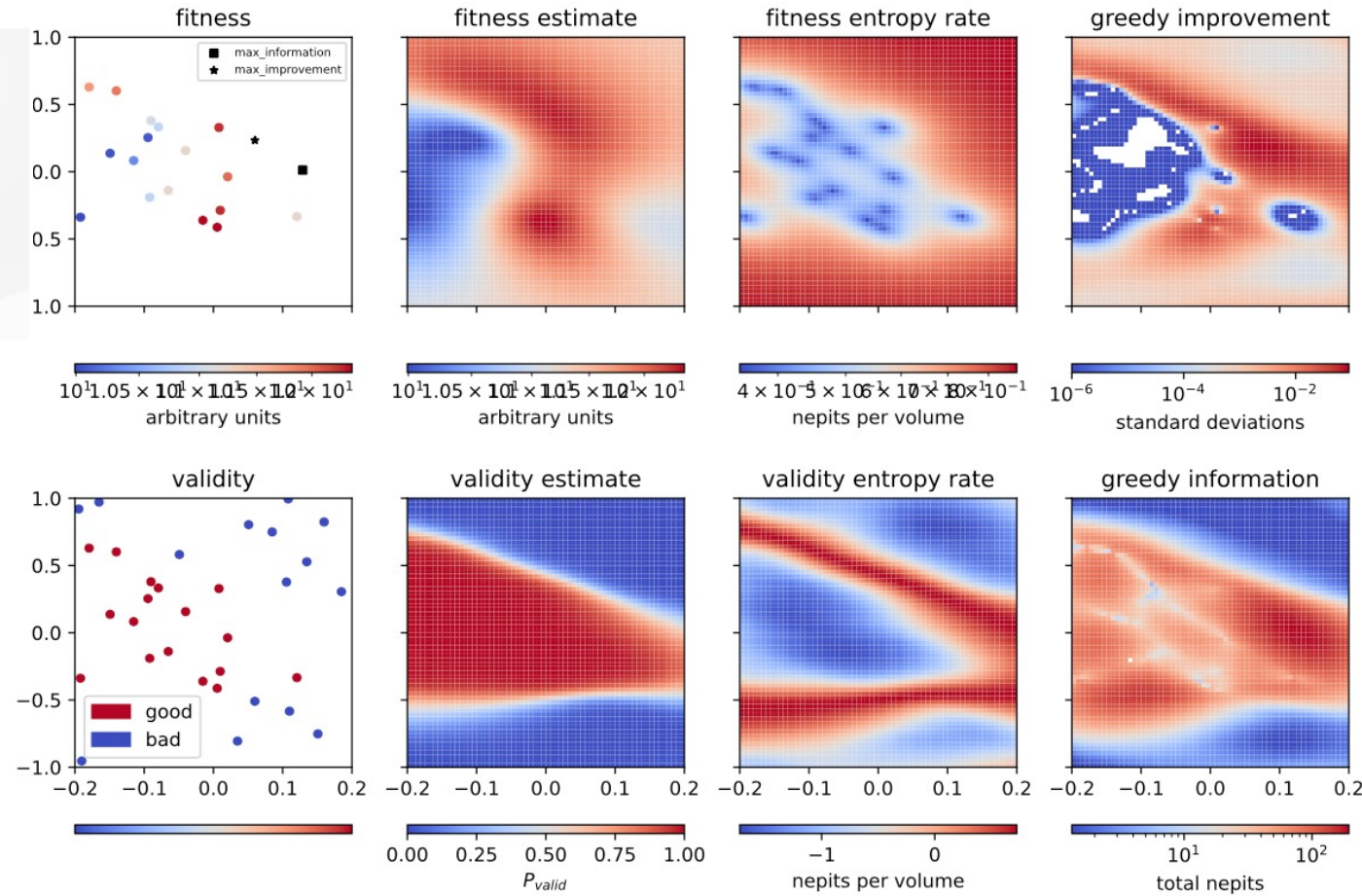


# Beamline Optimization



Auto-alignment of the toroidal mirror at NSLS-II TES. The evolution of the beam shape and position can be seen clearly after 16 and 64 iterations.

T. W. Morris, M. Rakitin, A. Giles, J. Lynch, A. L. Walter, B. Nash, D. Abell, P. Moeller, I. Pogorelov, and N. Goldring "On-the-fly optimization of synchrotron beamlines using machine learning", Proc. SPIE 12222, Optical System Alignment, Tolerancing, and Verification XIV, 122220M (3 October 2022); <https://doi.org/10.1117/12.2644996>



Optimization based on Gaussian Processing (GP)

Sirepo/Shadow3 simulations

# e-Beam requirements

- No special requirements, the team can be using the beam in a “parasitic” mode for collecting the data from ongoing experiments.
- With the active learning, we will need to modify steering magnet parameters (DARL\*)

# Future Plans

- Perform integration and stress testing of the Ophyd abstractions at the ATF beamlines.
- Develop the alignment procedures and integrate them with the MAD-X simulation data.
- Apply the GP optimization routines to the ATF hardware using multi-parameter optimization (8+ dimensions) – a few **2-to-4-hour** periods will be very useful for these studies.