





and



<http://www.silx.org/>

HDF5 European Workshop for science and industry, 18th of September 2019, ESRF.

silx contributors: T. Vincent; V. Valls; H. Payno; J. Kieffer; V. A. Solé; P. Paleo; D. Naudet; G. Communie; P. Knobel; M. Retegan; M. Rovezzi; P. Kenter; H. Fangohr; @UUSim; V. Favre-Nicolin; F. E. Picca; C. Nemoz; W. De Nolf; @schooft; T. Coutinho; J. Kotanski; C. J. Wright; A. Campbell

- Initial goals:
 - Share maintenance of common features across Python-based applications/libraries (PyMca, pyFAI,...):
 - share/reuse code,
 - documented,
 - tested.
 - Make features available for others
 - Provide tools to support the migration/use of HDF5

- Features:
 - I/O:
 - Reading SPEC and images
 - HDF5 conversion helpers
 - Data processing, including GPU-accelerated routines (using OpenCL):
 - SIFT
 - (Filtered) Back projection
 - Median filter,...
 - Qt widgets for:
 - Browsing HDF5-like data
 - Visualizing and interacting with data (1D, 2D, 3D)

- Python package (with Cython, C, C++, OpenCL, GLSL code)
- License: MIT
- Linux, Windows, macOS
- Last version: v0.11.0:
- Install through: pip, conda, debian
- Resources:
 - <http://www.silx.org/>
 - <https://github.com/silx-kit/silx>
 - Documentation: <http://www.silx.org/doc/silx/latest/>



and



Conversion to HDF5

- Context:
 - ESRF legacy acquisition format: SPEC file + EDF (or other format) images
 - Moving towards HDF5/NeXus
- Transition period:
 - Data in SPEC+images format
 - “Future[/present]-proof” software needs to read HDF5

Conversion to HDF5: Tool

- Command line tool: `silx convert`

- Concatenation of all SPEC files in the current directory:

```
silx convert *.dat -o all_SPEC.h5
```

- Add a list of images (supported by fabio) into an existing HDF5 file:

```
silx convert ch09_mca_0005_0000_%d.edf -o archive.h5::/ch09_mca_0005_0000_0008 -m a  
--compression
```

Conversion to HDF5: API

Python package: `silx.io`

- File conversion functions:

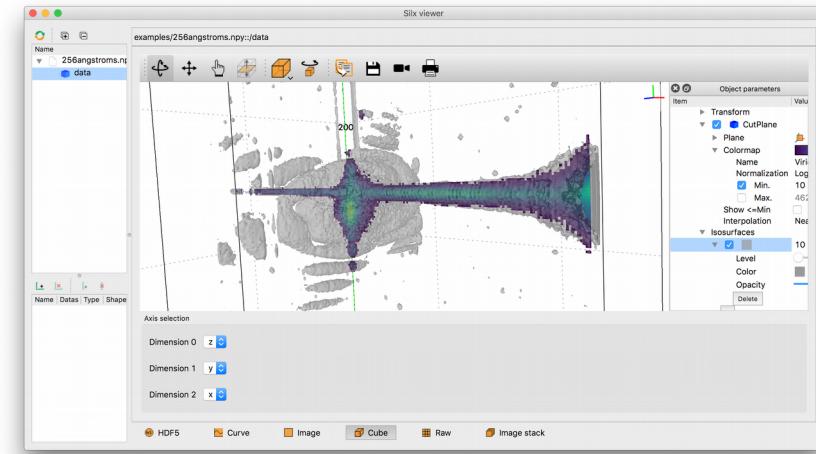
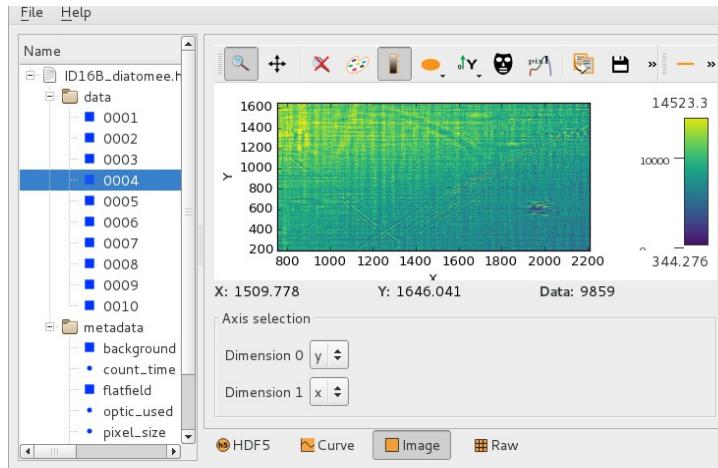
```
silx.io.convert.write_to_h5 # or convert
```

- Accessing SPEC and image files (using fabio) with a h5py-like API and a NeXus structure:

```
silx.io.open(filename)
```

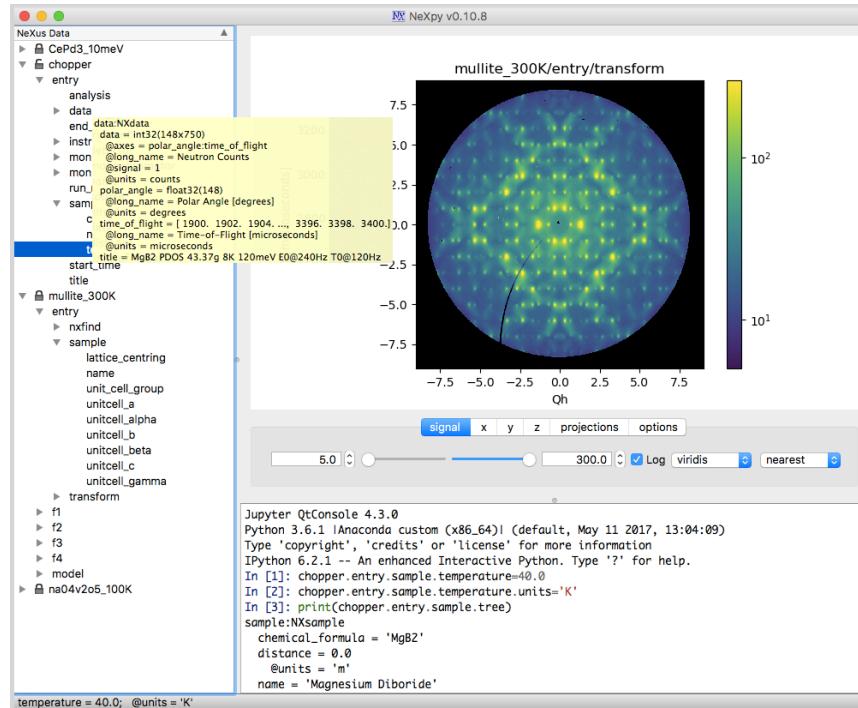
silx view

- HDF5 (and other formats) browsing and viewing
- GUI application (PyQt5/PySide2-based)
- Visualization: matplotlib and OpenGL

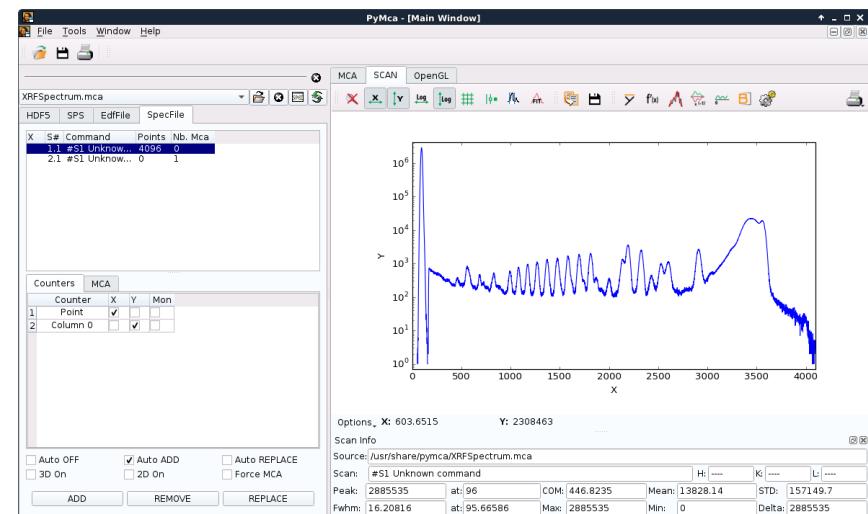


Other HDF5/NeXus viewer

NeXpy

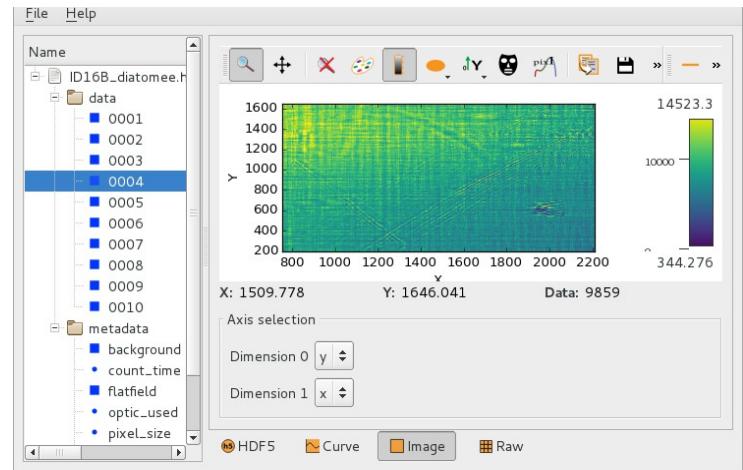


PyMca



silx view

- Visualization of data as:
 - Array
 - Curve
 - Color-mapped image
 - 3D field: isosurface and color-mapped cut plane
- Support of 2D/3D complex data
- Support of nD datasets by selecting:
 - The kind of visualization
 - The slicing, i.e., viewed axes and positions along other axes
- Tools: mask, profile, curve fitting

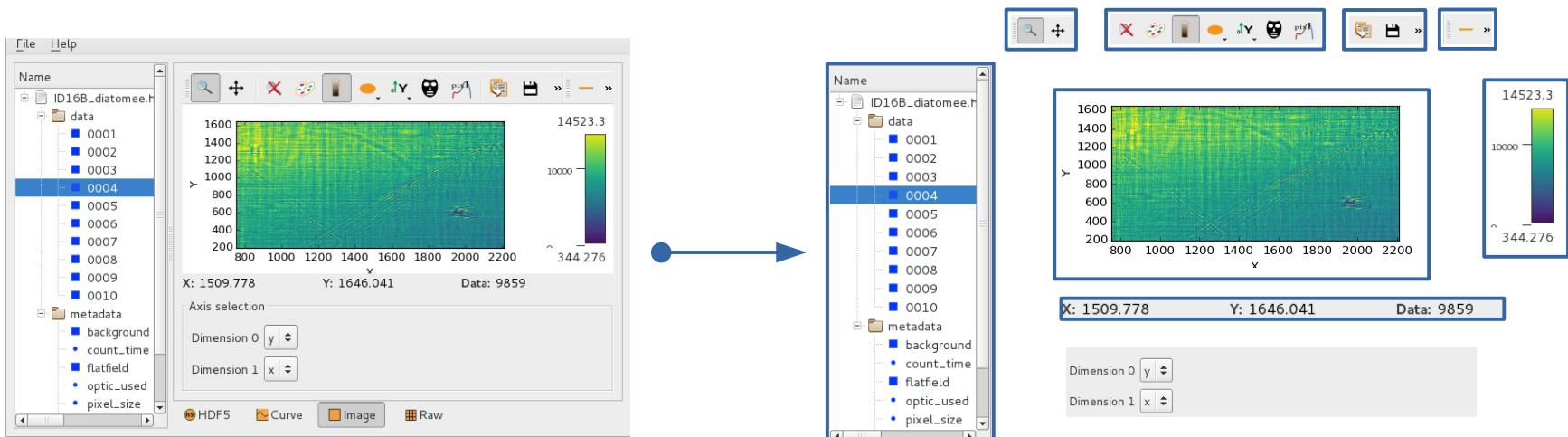


silx view: NeXus support

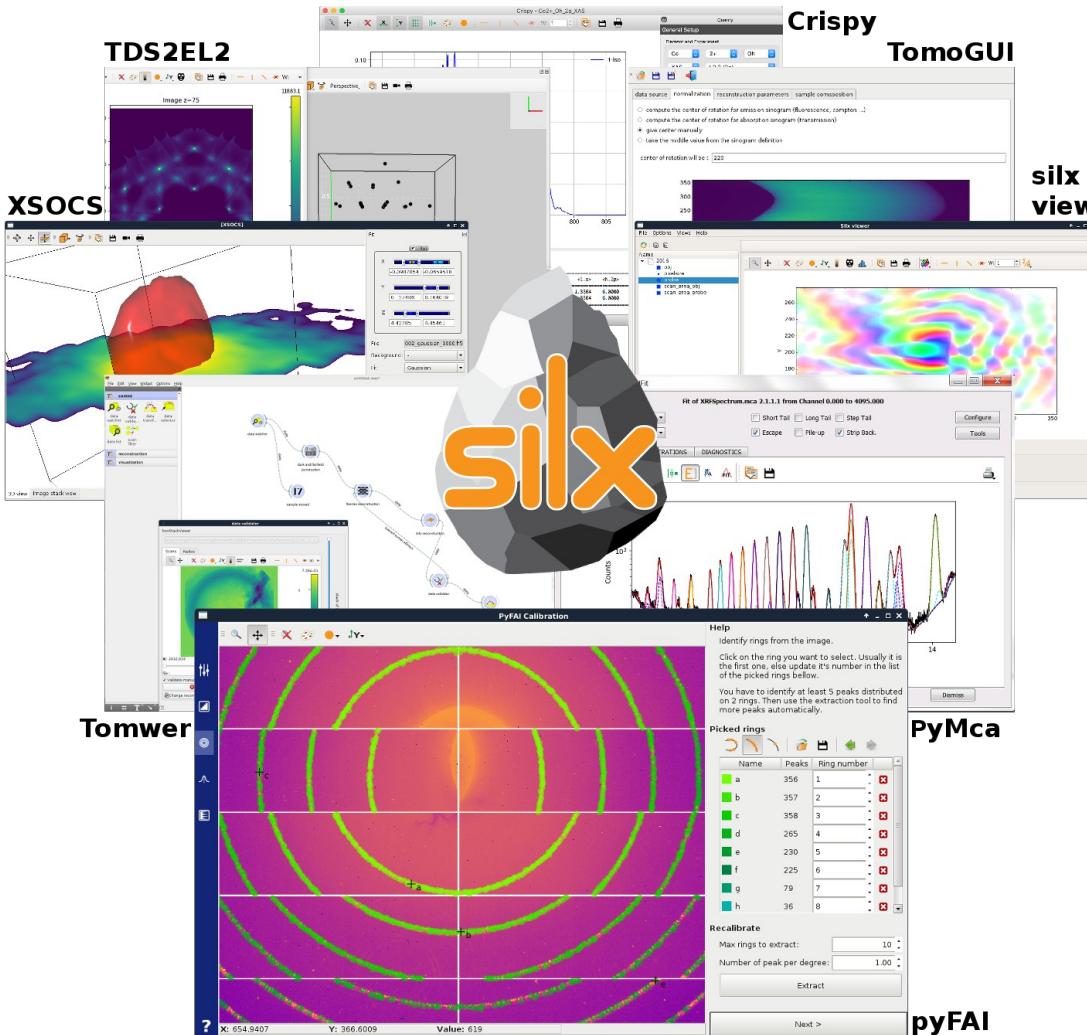
- NeXus-aware:
 - Display @NX_Class in browser
 - Leverage NXData and @default attribute to display a default plot

silx view: widgets

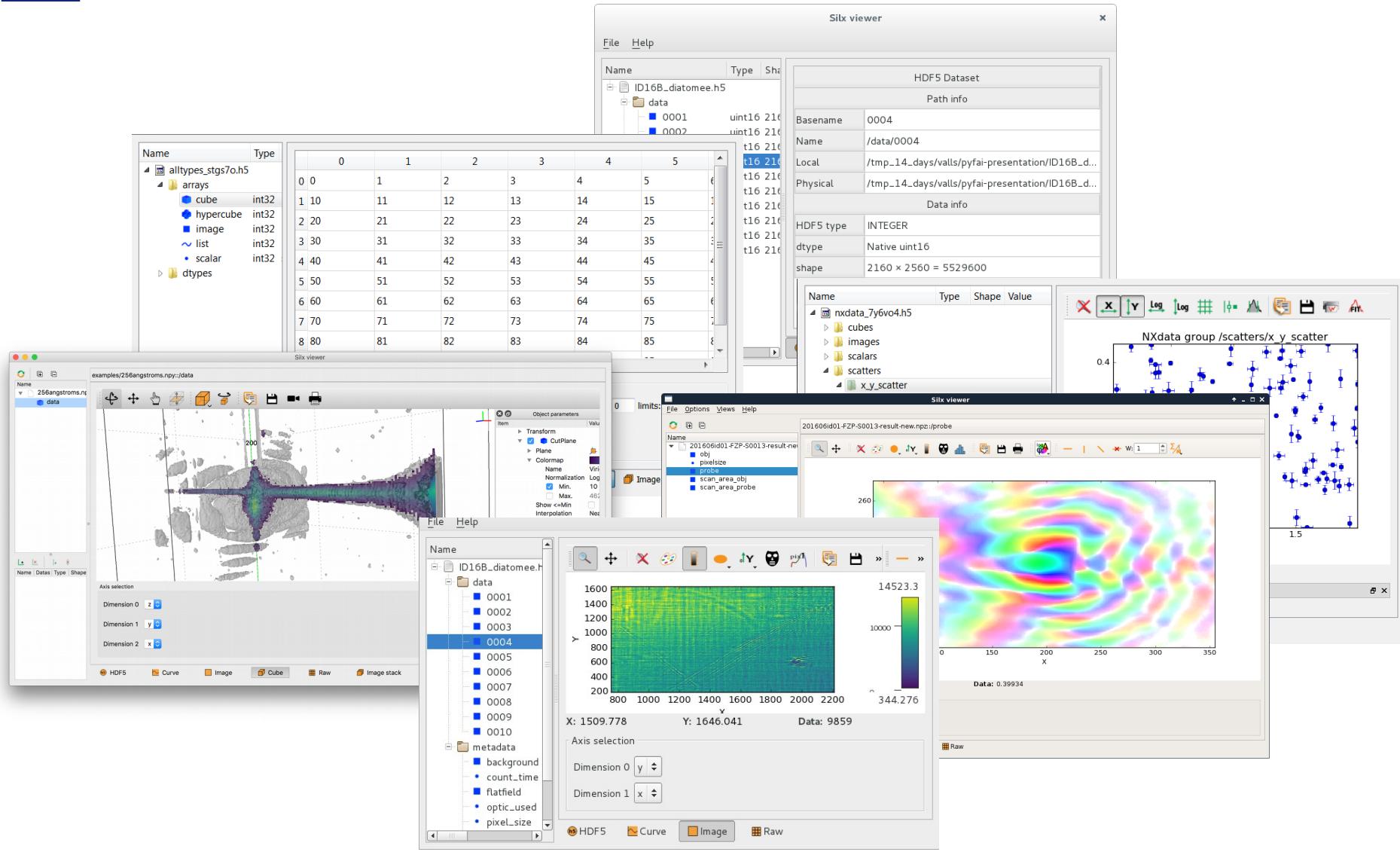
- Widgets used by silx view are available from `silx.gui` for reuse in other applications.



Some inhouse silx-based applications



Demo: silx view





Thanks for your attention!

<http://www.silx.org/>