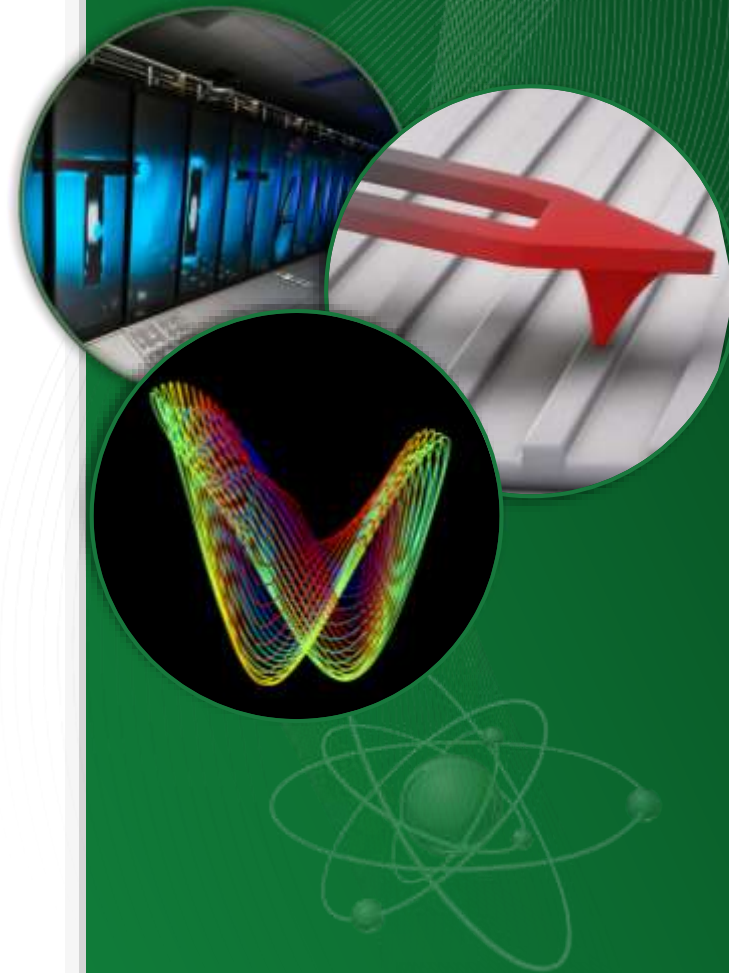


Imaging in the Information Dimension

- Suhas Somnath
- Chris R. Smith
- Stephen Jesse

 **OAK RIDGE** | OAK RIDGE
National Laboratory | LEADERSHIP
COMPUTING FACILITY

 INSTITUTE FOR FUNCTIONAL
IMAGING OF MATERIALS | CENTER FOR
OAK RIDGE NATIONAL LABORATORY | NANOPHASE
MATERIALS SCIENCES



ORNL is managed by UT-Battelle
for the US Department of Energy

Multitude of Instruments



Micro Raman Microscope



Atomic Force
Microscope (AFM)



AFM with Infrared
spectroscopy (AFM-IR)



Scanning
Tunneling
Microscope (STM)



Scanning
Transmission
Electron
Microscope (STEM)



AFM with Raman
spectroscopy

What we wanted



Instrument Tier

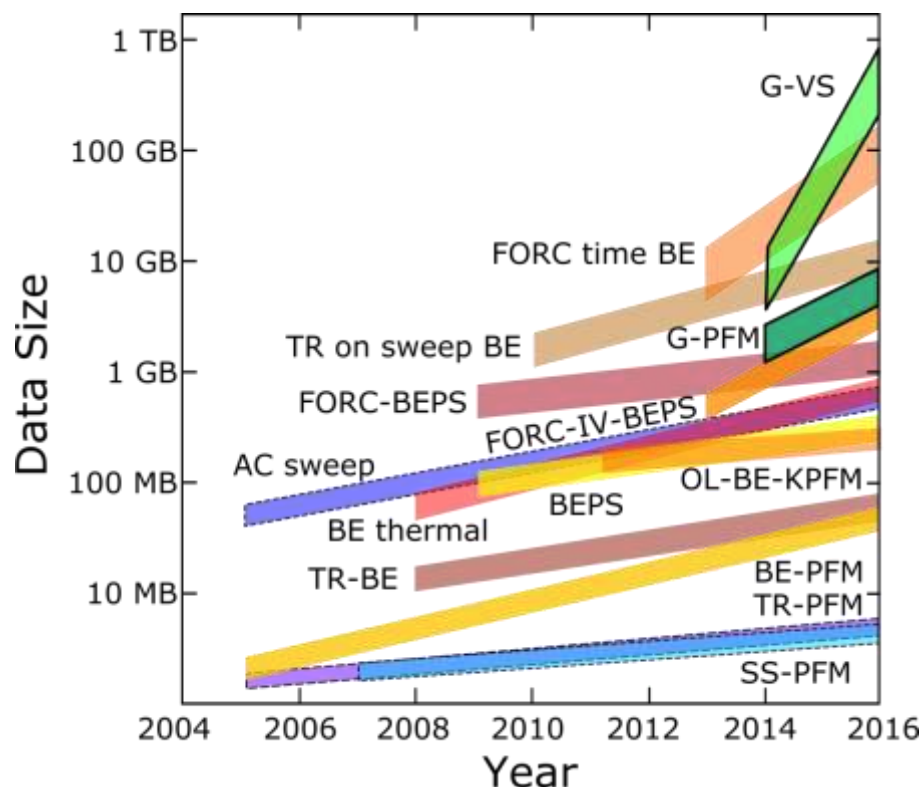
?



Interactive visualization, analysis,
storage on supercomputers

Growing Data Sizes and Dimensionality

Evolution of Scanning Probe Microscopy Data



- Data sizes have grown from ~ 10 MB to ~ 1 TB in 10 years!
- Dimensionality ranges from 1D spectra to 7D hyperspectral datasets
- Cannot use laptops to analyze data

Instrumentation Software Inadequate for Analysis



- Software provided for controlling instruments typically only comes with basic data analysis capabilities
- Integrating user-developed functionality often impossible



Multitude of File Formats

- Proprietary
- Incompatible

.wdf

.ibw

.cdb

.asc

.dm3

.mat

≠



Disjoint & Unorganized Communities



- Clustering
- Fit spectra ...



- Filter Image
- Register Image ...



- Fit Spectra
- SVD Filtering ...

- FFT Filtering
- SVD Filtering ...



- FFT Filtering
- Classify Images ...



- Register Images
- Clustering



Cannot Share Code Efficiently

- HIGHLY instrument-specific code
- Different programming languages
- Often licensed / costly software like Matlab
- Most popular sharing method = email!
- No centralized repository

Problems Opportunities in Imaging

1. Closed science
 - a. No traceability for data analysis
 - b. Results not (readily) reproducible
2. Multiple, incompatible, proprietary data formats
3. Disorganized and unorganized communities
4. No proper analysis software
5. Growing data volumes, variety, and dimensionality

The Solution



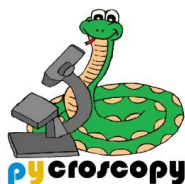
Instrument Tier



Automated, standardized,
modularized data acquisition



Instrument-agnostic, self-describing,
model in HPC-friendly file format



Centralized repository for data
processing, analysis



Interactive visualization + analysis +
storage on supercomputers

Expectation of Data Model

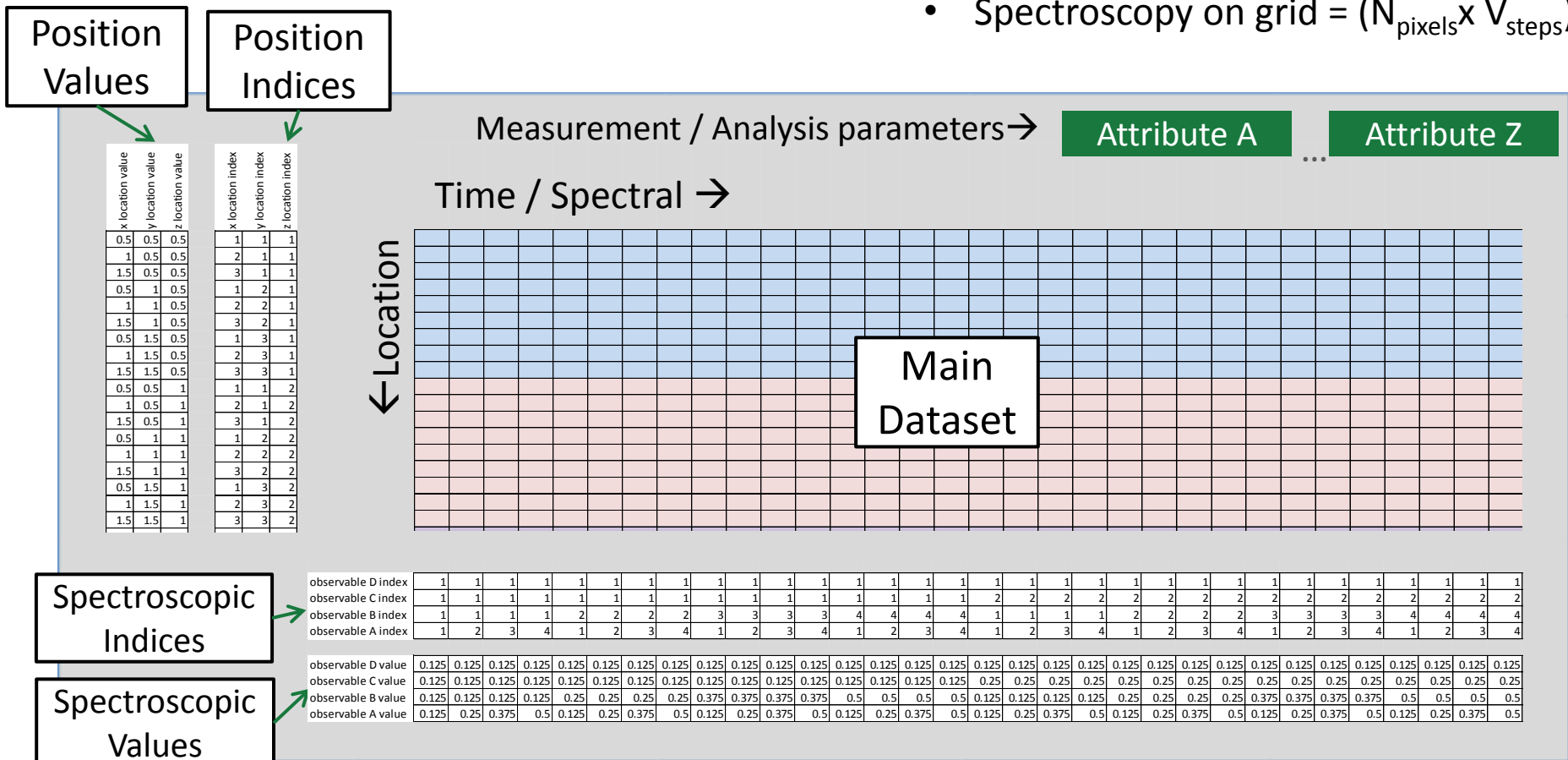
- Accommodate data of different shapes, dimensionalities, precision and sizes.
- Accommodate data without N-dimensional form
 - Compressed sensing / sparse sampling
 - Not all combinations of spectroscopic variables
 - Incomplete experimental data

Universal Imaging and Spectroscopic Data (USID)

- Data stored as 2D matrix of (position x spectral values) regardless of dimensionality
- Ancillary datasets explain the data

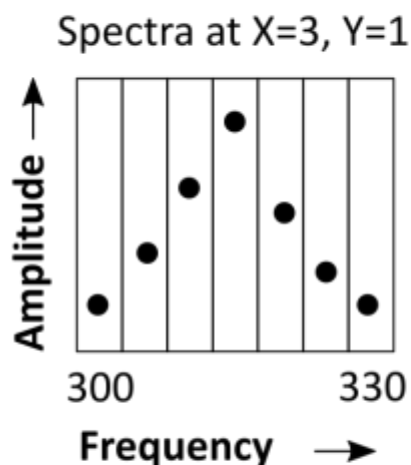
Example data types:

- 2D images = ($N_{\text{pixels}} \times 1$)
- Single spectra = ($1 \times Z_{\text{steps}}$)
- Spectroscopy on grid = ($N_{\text{pixels}} \times V_{\text{steps}}$)



USID – 1D spectra

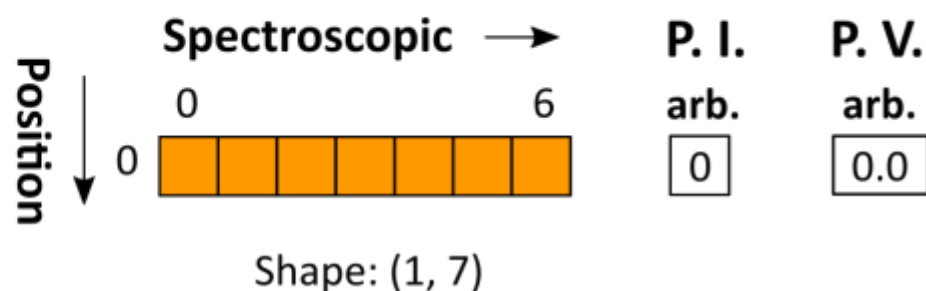
Original N-dimensional form



Shape: (7,)
Quantity: Amplitude
Units: V

=

USID 2-dimensional form



S. I. Frequency

0	1	2	3	4	5	6
---	---	---	---	---	---	---

S. V. Frequency

300	305	310	315	320	325	330
-----	-----	-----	-----	-----	-----	-----

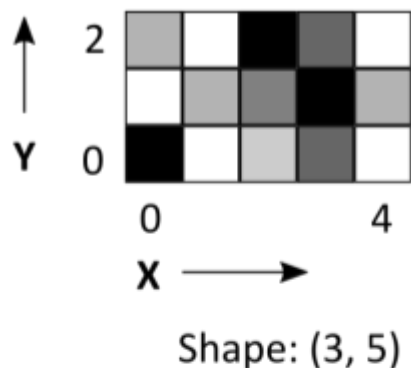
P. I.
arb.

0

P. V.
arb.

0.0

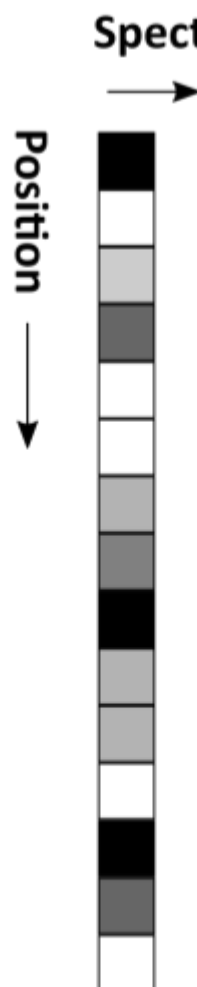
USID – 2D Image



Original
N-D
form

Quantity: Intensity
Units: arb. units

=



S. I. arb.

S. V. arb.

P. I.

X	Y
0	0
1	0
2	0
3	0
4	0
0	1
1	1
2	1
3	1
4	1
0	2
1	2
2	2
3	2
4	2

P. V.

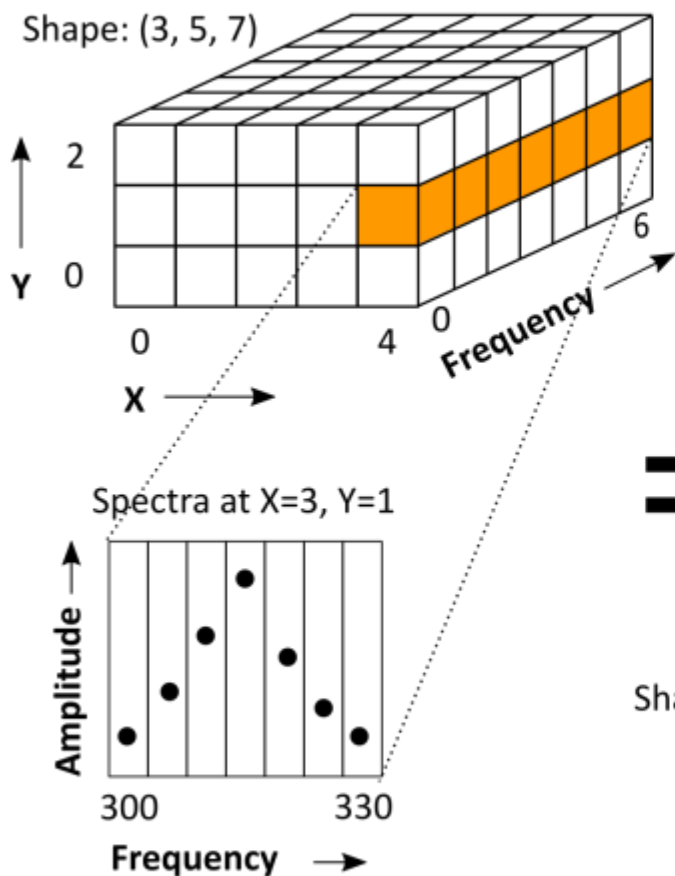
X	Y
-250	0
-125	0
0	0
125	0
250	0
-250	3.5
-125	3.5
0	3.5
125	3.5
250	3.5
-250	7
-125	7
0	7
125	7
250	7

Shape: (15, 1)

USID
2D
form

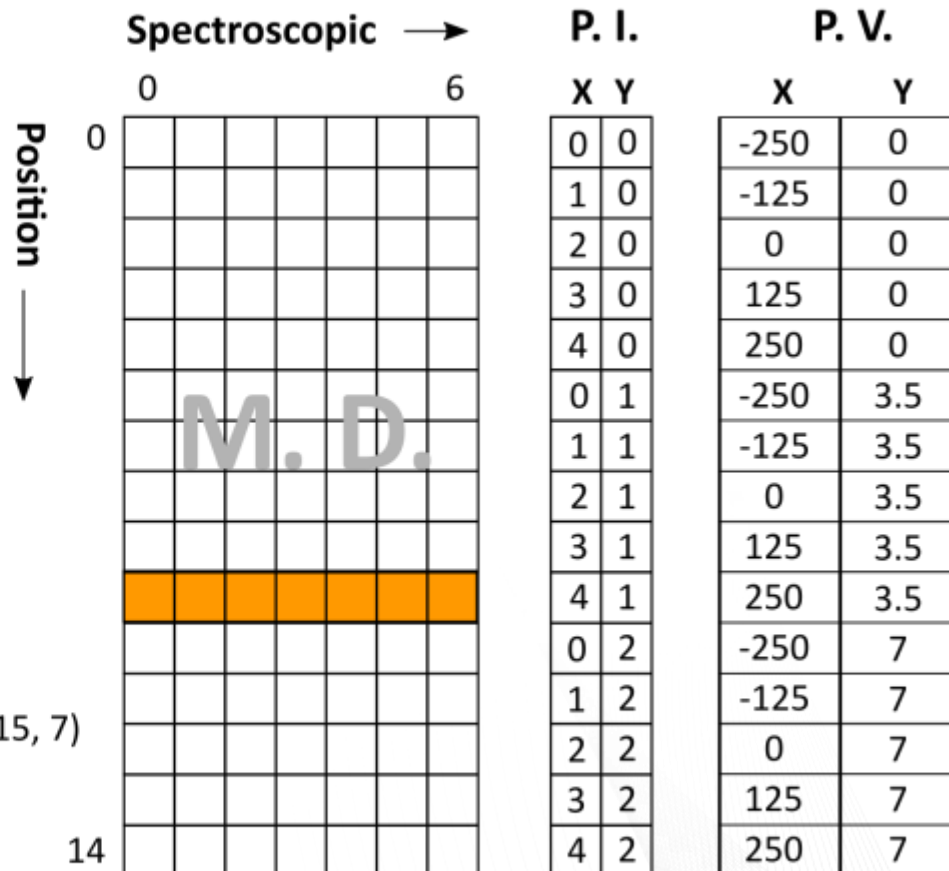
USID – Spectra on Grid (3D)

Original N-dimensional form



Quantity: Amplitude
Units: V

USID 2-dimensional Form

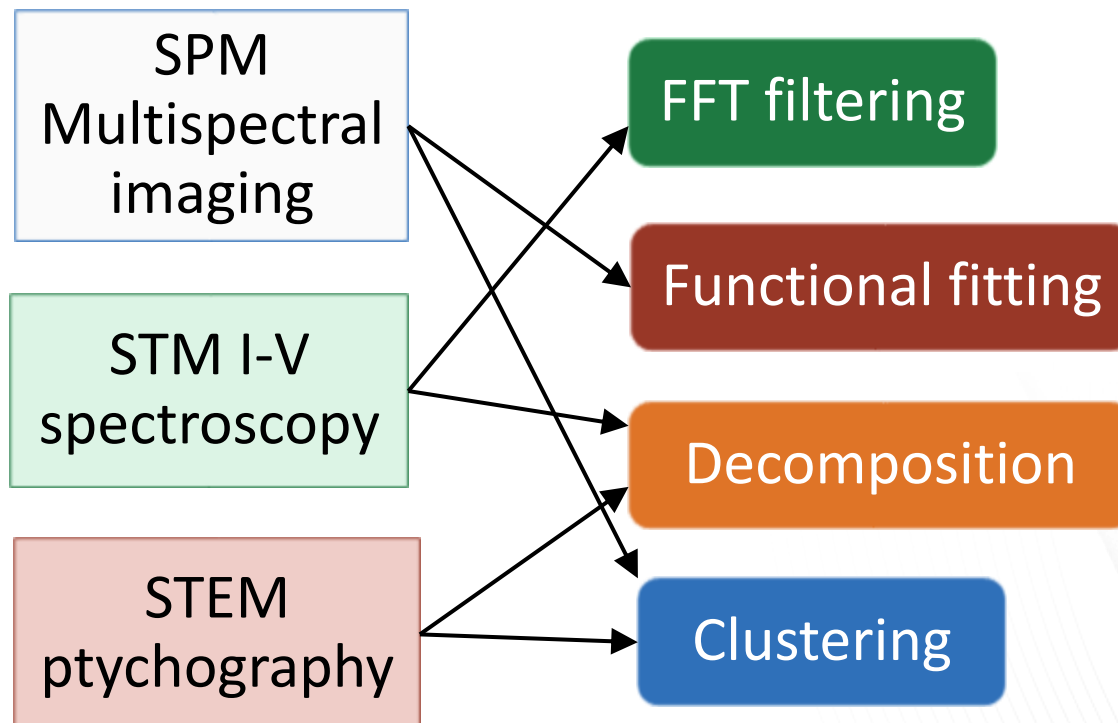


S. I. Frequency [0 1 2 3 4 5 6]

S. V. Frequency [300 305 310 315 320 325 330]

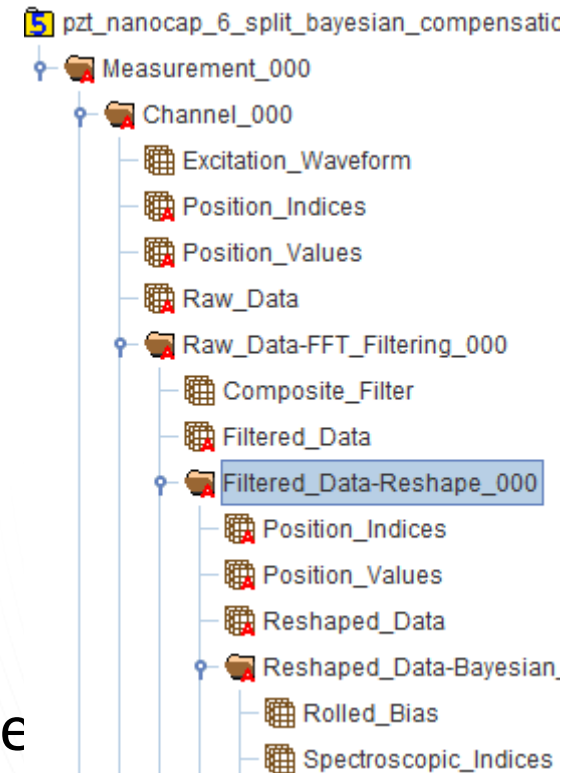
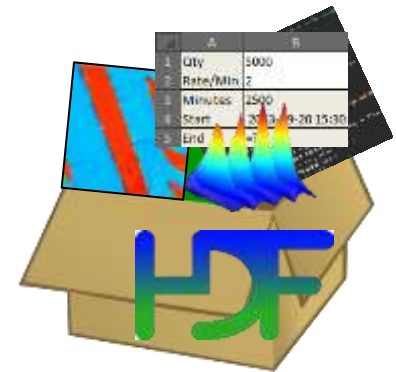
USID - Instrument Agnostic Code

- Instrument-agnostic data allows instrument-agnostic code
- Single version of analysis and processing routine
- Brings multiple scientific communities together



Hierarchical Data Format (HDF5)

- A HDF5 file is a smart container
 - Capable of storing multidimensional datasets, Images, text, measurement parameters, etc.
 - Contents organized like traditional folders and files
 - **Groups** - Analogous to file folders
 - **Dataset** – 1 to N dimensional data
 - Integer, floating point, complex numbers etc
 - **Attributes** – {Key : value} pairs useful for describing data and experimental parameters, etc.
- Easily accessible – C, C++, python, Java....
- Tree structure + nomenclature + attributes are **records of workflow** applied to dataset
- Parallel read / write, HPC & cloud compatible

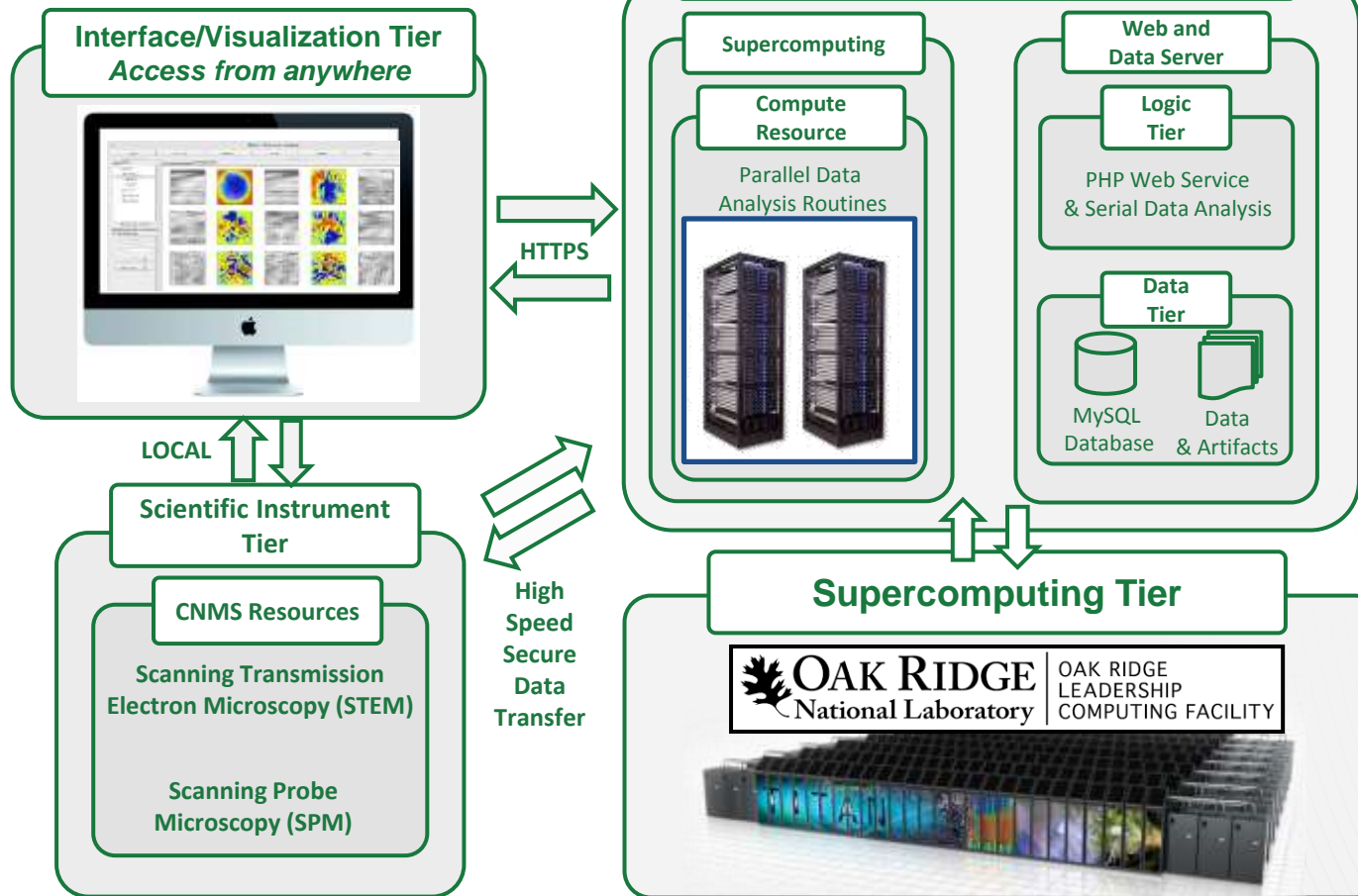


Expectation from Software

- Easy to learn and understand
- Strong support-base
- Established community standard
- Straightforward to implement and maintain
- Optimized libraries for scientific and numeric algorithms
- Access to existing imaging related packages
- Free
- Scalable to multiple CPU cores + distributed computing

(Purely) Programmer-Driven Solution

Software connecting scientific instruments to supercomputers



- **Successes:**

- Easy to use – Point-click
- Fast – on super-computers

- **Shortcomings:**

- Very long development cycle
- Very expensive
- Brittle (points of failure)
- Scientists had no control!!

Python for Scientific Research

Very easy to learn + code

Numerous, **powerful**
libraries for science



NumPy



SciPy



- Facilitates innovation
- More robust code
- Improved adoption of new methods / standards
- Accelerates scientific progress

Cross-
platform



scalable



Established standard for:

- Microscopy
- Microbiology
- Deep learning
- Data science
- Neutron science
- More!

Strong user
community



stackoverflow

All for a princely sum of **\$0!**

See Jake Vanderplas' Pycon 2017 Keynote talk:
<http://www.youtube.com/watch?v=ZyjCqQEUa8o&t=19m20s>

(2) Software Packages

- Written in Python
- Open source & free
- Written by scientists
- Data centric
- Instrument-independent data model in HDF5
- Instrument-independent analysis algorithms
 - Reusable across scientific domains

pyUSID



Software Organization

pycroscopy

I/O

- Data translators (proprietary formats to HDF5)

Visualization

- Plotting utilities
- Jupyter widgets

Simulation

- AFM Force-distance ...

Analysis

- Physical model specific
- Fitting to model, etc.
- Physics based regression

Processing

- Physical model agnostic
- Image filtering, registration,
- Multivariate analysis

pyUSID

- HDF5 file i/o operations
- Base data processing,
- visualization...

Software Organization

Science / data
analytics
applications



File / data tools

pyUSID

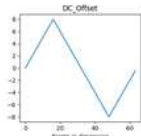
Well documented

Beginner topics

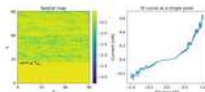
To learn how to use pyUSID, Please go through the following documents in the recommended order:



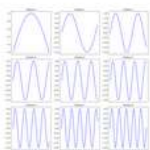
01. Primer to HDF5 and h5py



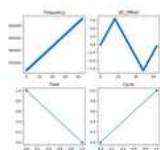
02. The USIDataset



03. Translation and the NumpyTranslator



04. Plotting utilities



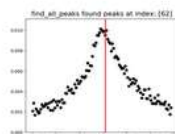
05. Utilities for reading h5USID files

Intermediate topics

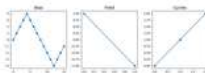
To learn how to write to h5USID files, write data processing classes, or adding functionality to pyUSID, go through these additional documents in the recommended order: Those interested in contributing to pyUSID are encouraged to read our [guidelines for contributing code](#)



06. Utilities for handling



07. Speed up



08. Utilities that assist

```
reshape_to_Ndims(h5_main, h5_pos=None, h5_spec=None, get_labels=False, verbose=False, sort_dims=False) \[source\]
```

Reshape the input 2D matrix to be N-dimensions based on the position and spectroscopic datasets.

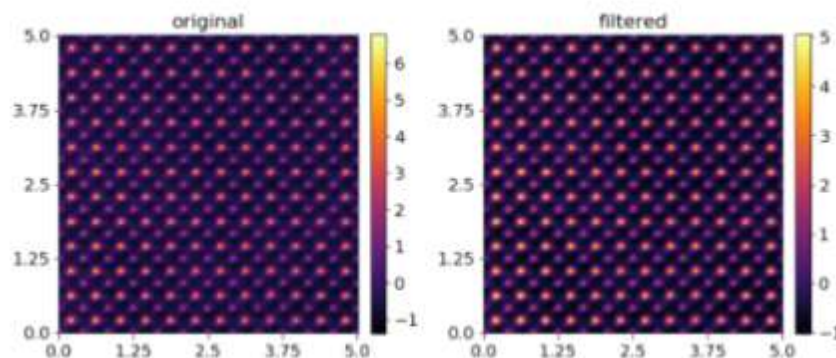
- Parameters:
- `h5_main` (*HDF5 Dataset*) – 2D data to be reshaped
 - `h5_pos` (*HDF5 Dataset, optional*) – Position indices corresponding to rows in `h5_main`
 - `h5_spec` (*HDF5 Dataset, optional*) – Spectroscopic indices corresponding to columns in `h5_main`
 - `get_labels` (*bool, optional*) – Whether or not to return the dimension labels. Default `False`
 - `verbose` (*bool, optional*) – Whether or not to print debugging statements
 - `sort_dims` (*bool*) – If `True`, the data is sorted so that the dimensions are in order from fastest to slowest. If `False`, the data is kept in the original order. If `get_labels` is also `True`, the labels are sorted as well.

- Returns:
- `ds_Nd` (*N-D numpy array*) – N dimensional numpy array arranged as [positions slowest to fastest, spectroscopic slowest to fastest]

To view the filter ('fft2'). Remember necessary to use the transform. Also the inverse transform symmetric about the result in the inverse times smaller than kept.

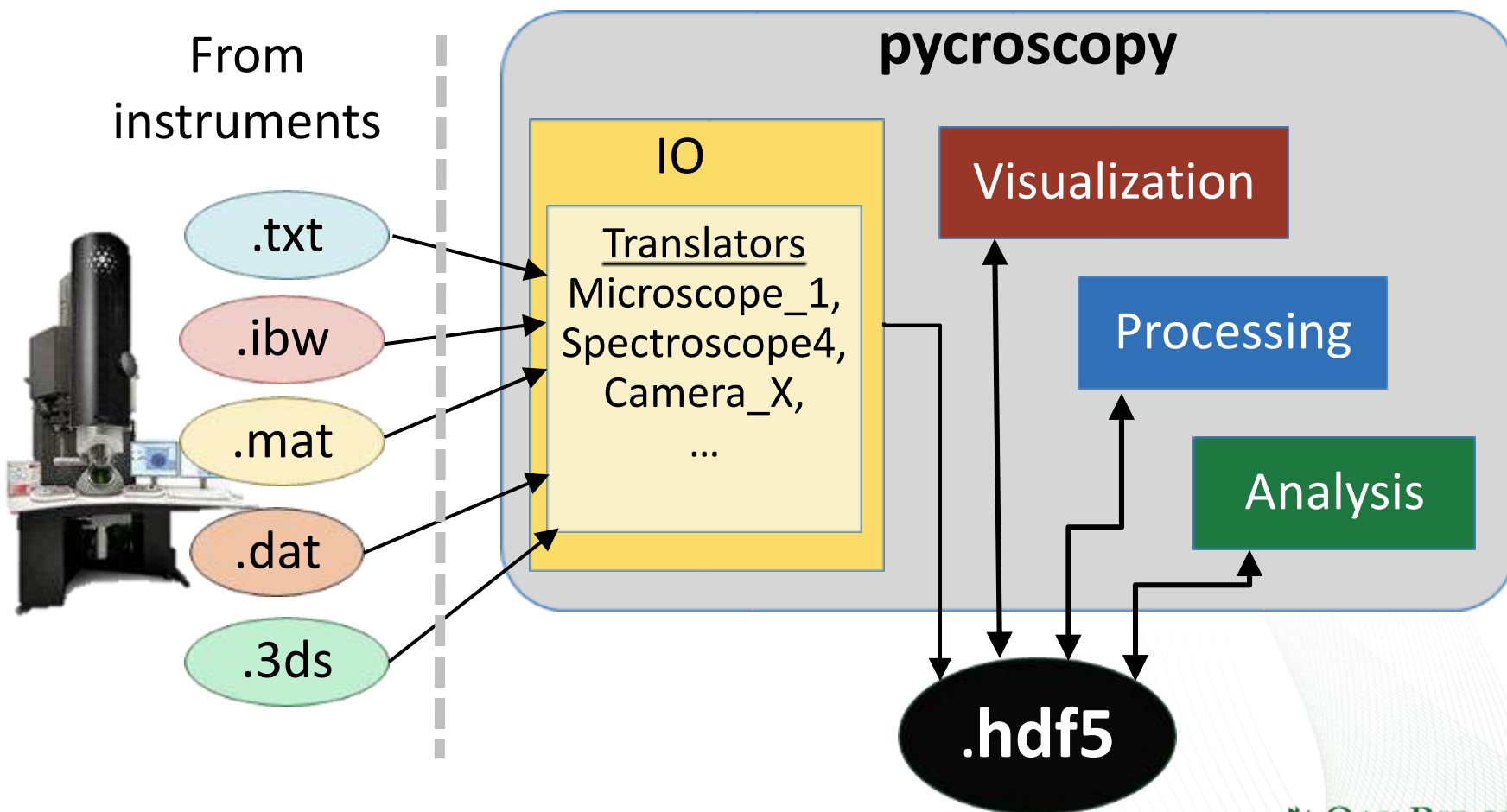
```
image_filter  
image_filter
```

```
fig, axes = plt.subplots(ncols=2, figsize=(10, 5))  
for axis, img, title in zip(axes, [image_raw, image_filtered], ['original', 'filtered']):  
    _ = px.plot_utils.plot_map(axis, img, cmap=plt.cm.inferno,  
                              x_size=x_edge_length, y_size=y_edge_length, num_ticks=5)  
    axis.set_title(title)  
fig.tight_layout()
```



Entering the USID Ecosystem

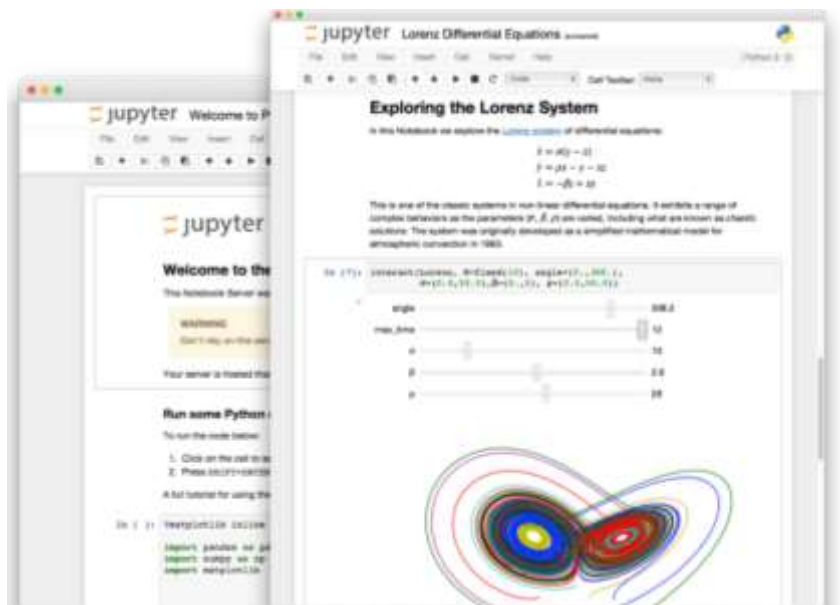
- hdf5 file is the hub for all operations
- Analysis, processing, visualization available after translation to .hdf5



Jupyter Notebooks



Jupyter Notebook

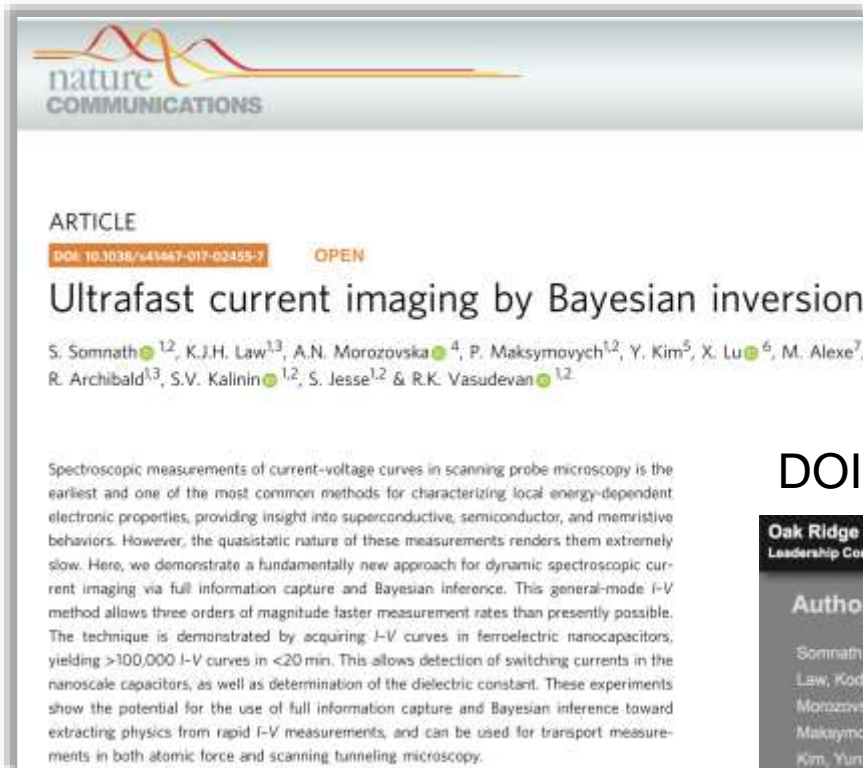


- Interactive documents
- Exploratory programming
- Code
- Text
- Images
- Interactive – slice through data, pan, move, rotate ...

Truly Achieving Open Science, Reproducibility

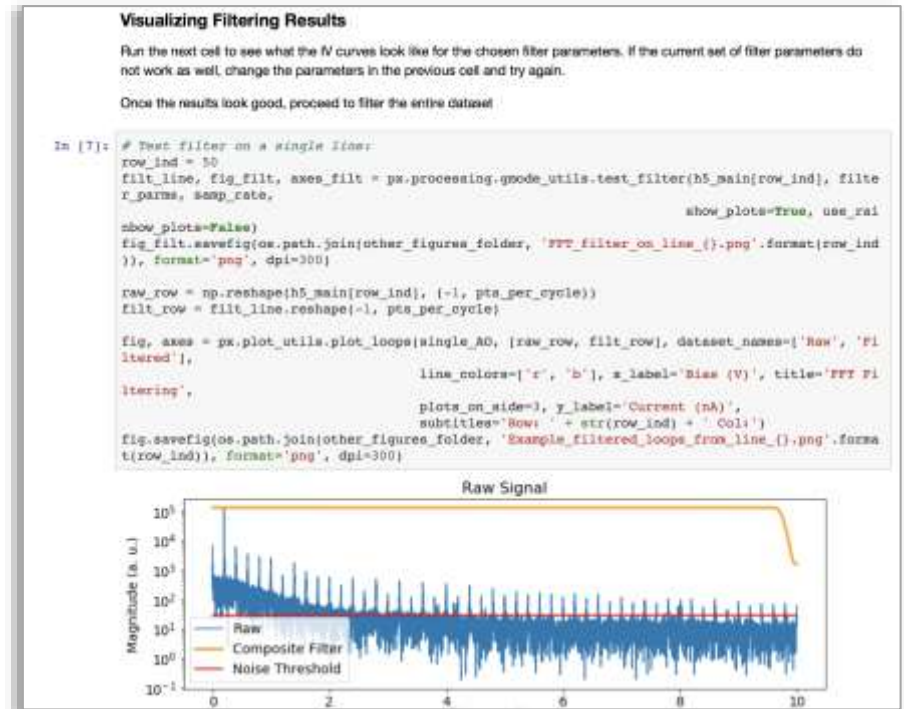
Aim – ALL scientific journal papers accompanied with:

- Jupyter notebook that shows all analysis (raw data → figures).
- Data with DOI number



Suhas Somnath, somnaths@ornl.gov

Jupyter notebook associated with paper



DOI associated with data (raw → paper figures)

Oak Ridge National Laboratory Leadership Computing Facility **10.13139/OLCF/1410993** [Download](#)

Authors

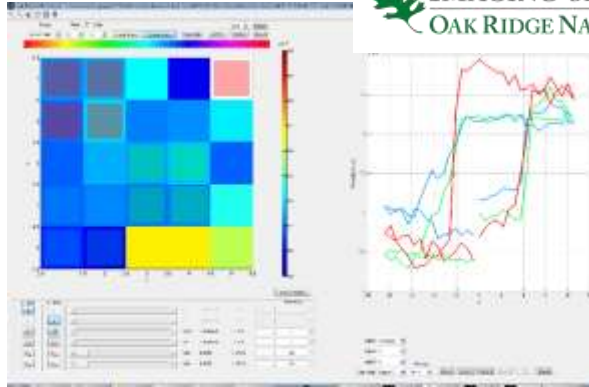
Somnath, Suhas	somnaths@ornl.gov
Law, Kody	lawkj@ornl.gov
Morozovska, Anna	anna.n.morozovska@gmail.com
Maksymovych, Petro	maksymovychp@ornl.gov
Kim, Yurmeek	yloms43@gmail.com
Lu, Xieoli	xliu@xidian.edu.cn
Alexe, Marin	M.Alexe@warwick.ac.uk

Pycroscopy - Supporting User Research

Before 2016

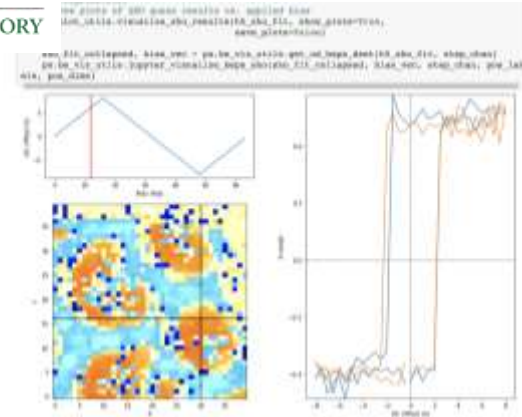


+



INSTITUTE FOR FUNCTIONAL
IMAGING OF MATERIALS
OAK RIDGE NATIONAL LABORATORY

Since 2016



Scripts + complicated, Matlab GUI

Set of simple Jupyter notebooks

Written by dedicated software engineer

Written by material scientists

Not customizable

Completely customizable.

2-3 hours of training before use

Notebooks include instructions. NO training required!

Deployed only on two offline workstations
due to licensing restrictions = queue

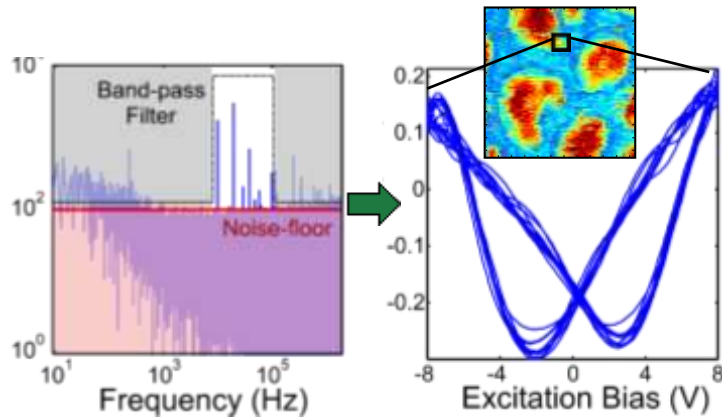
Each user gets VMs with jupyter notebook
server

Will remain on off-line desktops

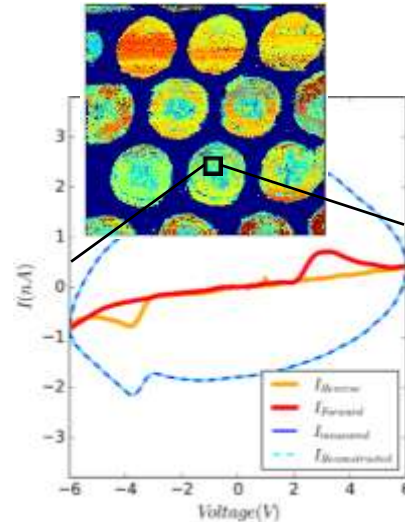
In the process of switching to computations
on clusters

Pycroscopy - Scientific Advancements

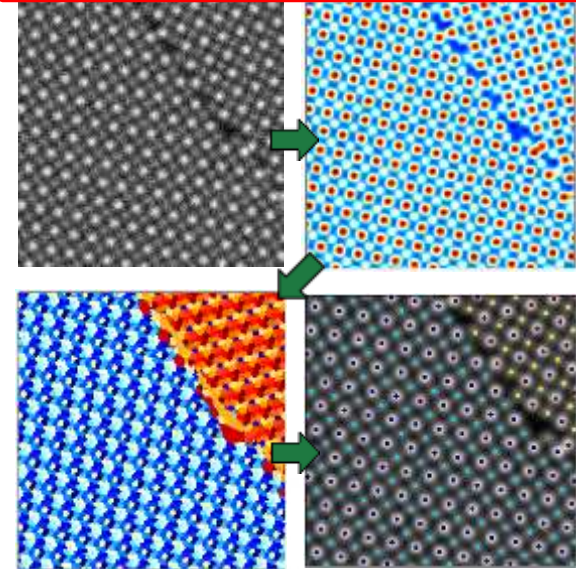
3,500x faster imaging via adaptive signal filtering, linear unmixing of signals



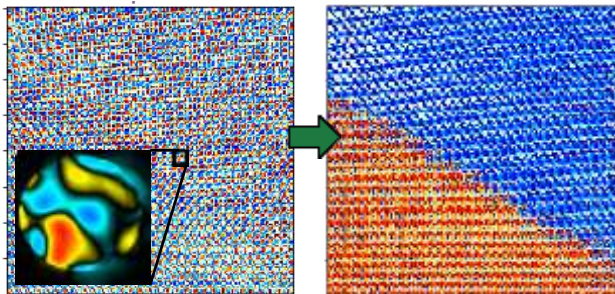
200x faster spectroscopy via Bayesian inference



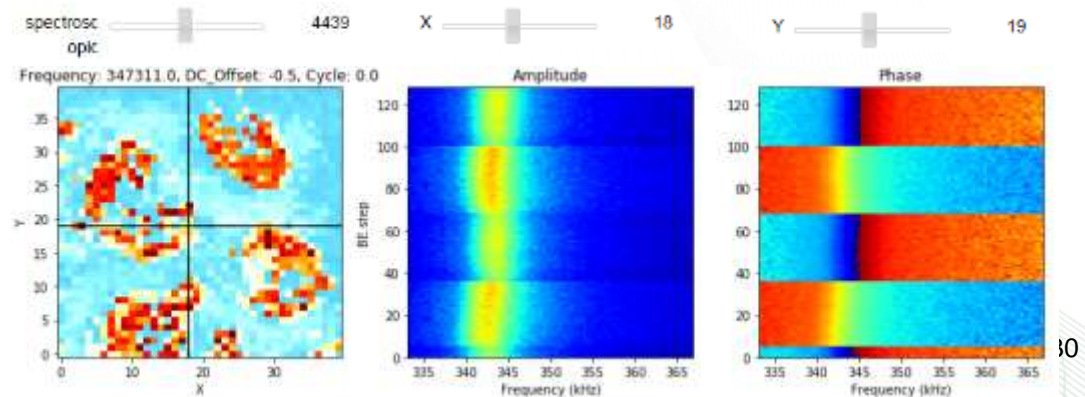
Separating uncorrelated data from correlated data to clean images



Identifying invisible patterns using multivariate analysis



Simplified navigation multidimensional data - users



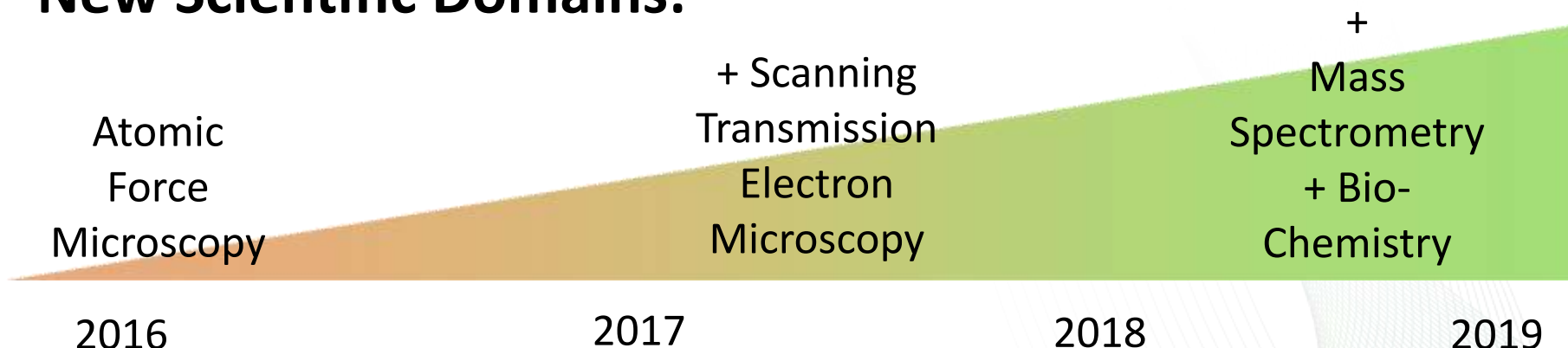
Software Progression

Scaling up Computing:



Emphasis always on ease-of-development instead of raw performance

New Scientific Domains:



Thank you

Questions?



File / data tools related- <https://groups.google.com/forum/#!forum/pyusid>

Science, data analysis, reading proprietary instrument data -
<https://groups.google.com/forum/#!forum/pycroscopy>