

WIR SCHAFFEN WISSEN – HEUTE FÜR MORGEN



Anders Kaestner :: Paul Scherrer Institut

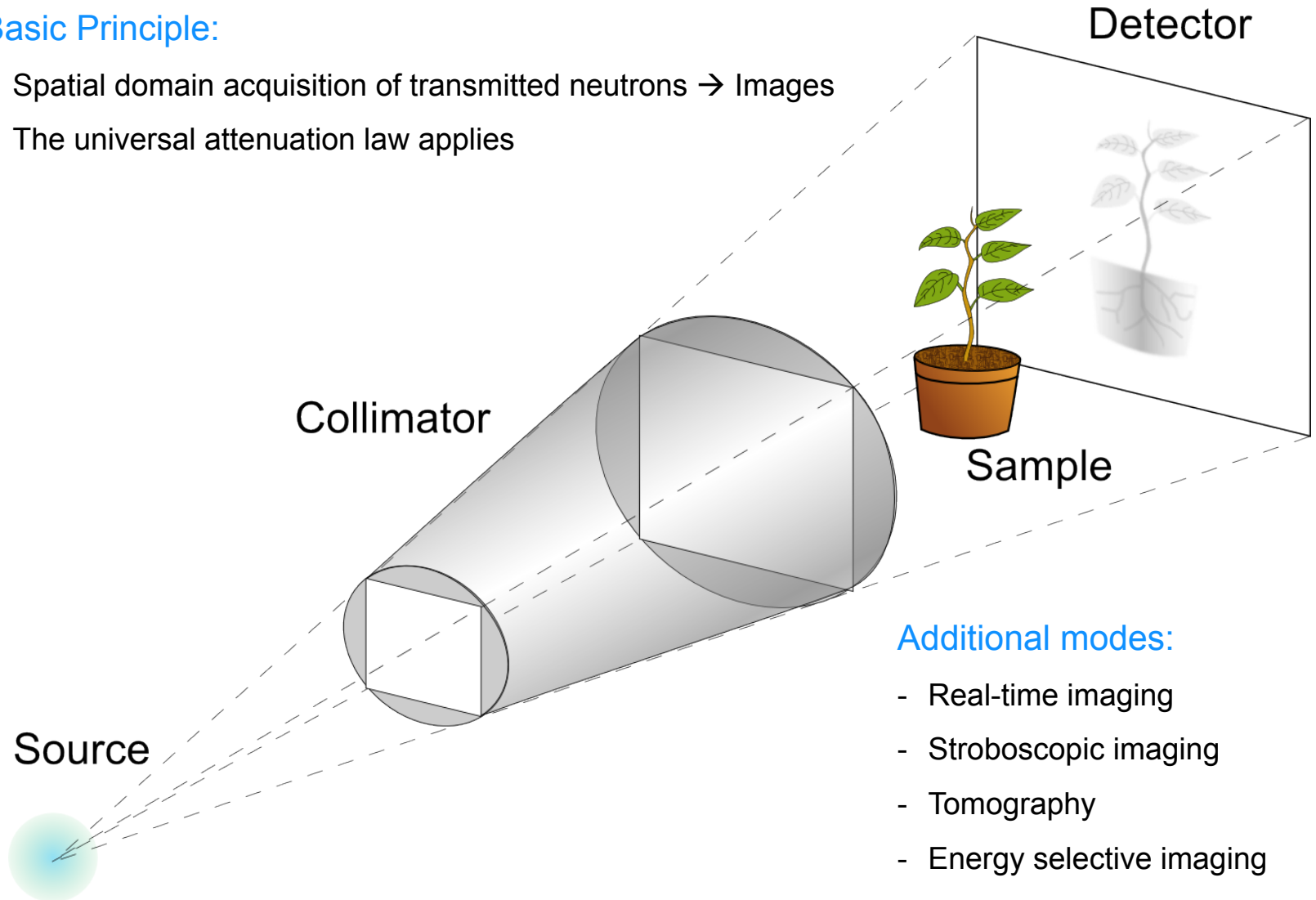
MuhRec and KIPTool under the hood

SINE2020 WP10 1st Coordination meeting, Villigen PSI, CH, April 2016

Neutron imaging

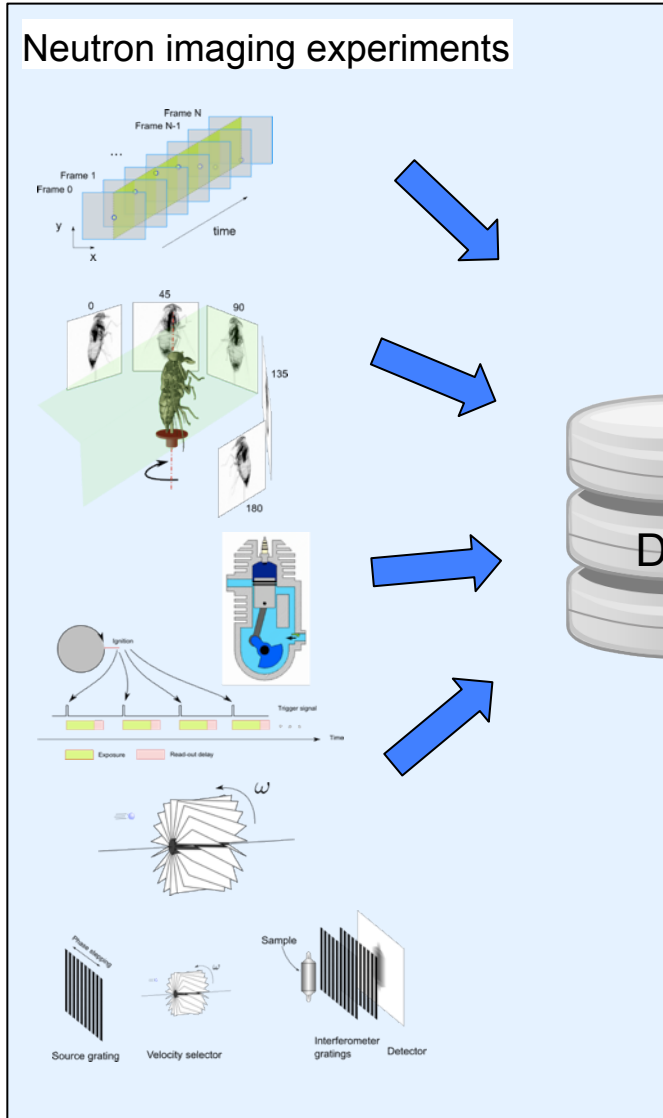
Basic Principle:

- Spatial domain acquisition of transmitted neutrons → Images
- The universal attenuation law applies



Additional modes:

- Real-time imaging
- Stroboscopic imaging
- Tomography
- Energy selective imaging
- Grating interferometry

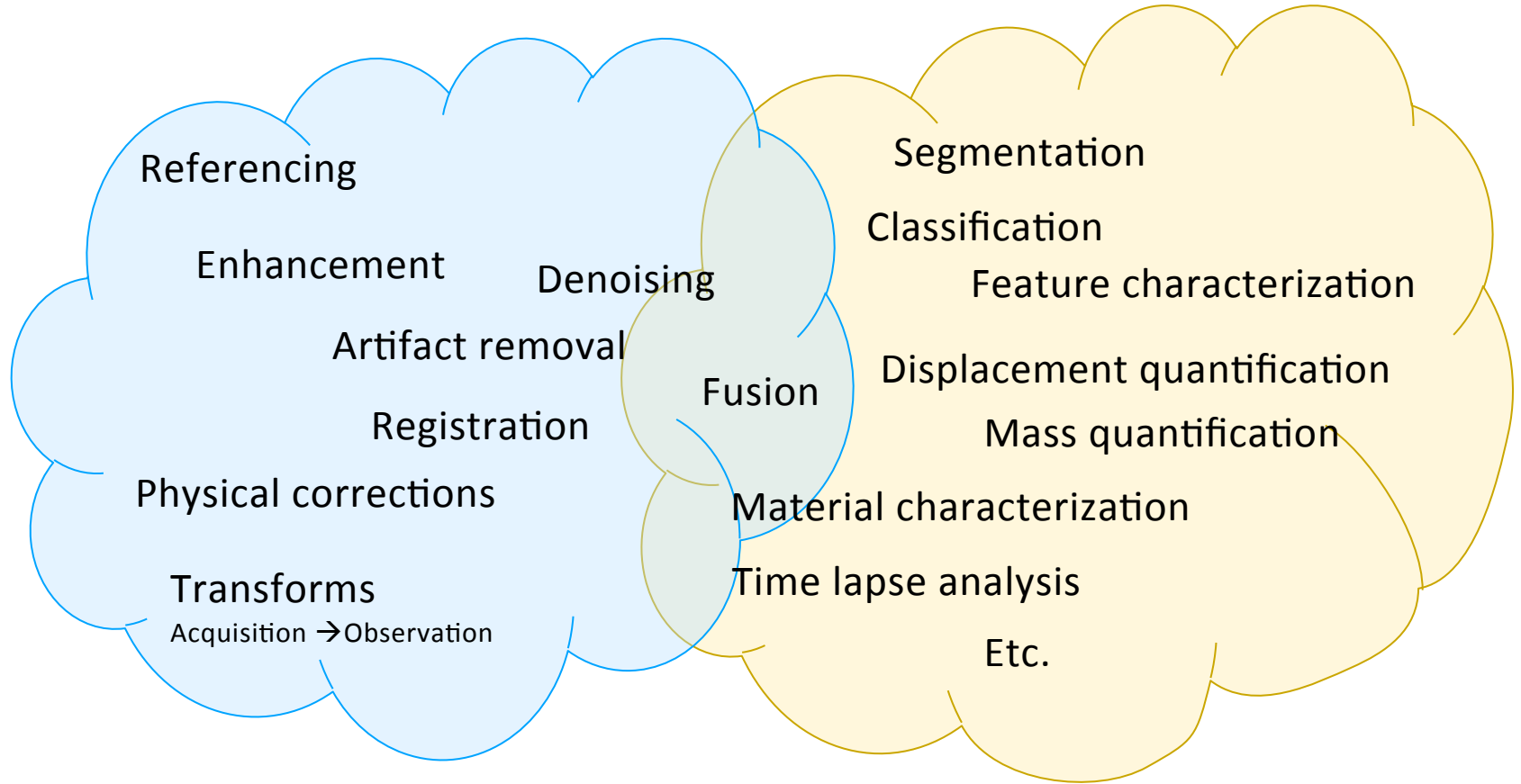


- Pre-processing
 - Noise suppression (first level)
 - Artifact reduction
 - Scattering correction
 - Experiment related processing
 - Interpretation/estimation
 - Tomographic reconstruction
 - Image processing
 - Noise suppression (second level)
 - Segmentation
-
- Visualization
Modeling

Data processing tasks for imaging

Image preparation

Image analysis






Instrument dependent

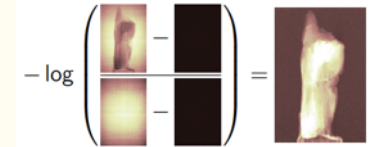
Application dependent

Which tools are used today?

Most users




Projection processing

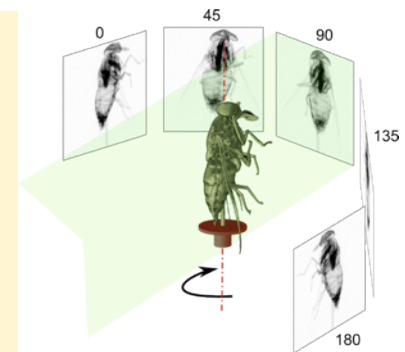
- ImageJ/FIJI
-  QNI scattering correction (Hassanein, 2006)
-  nGI Tool
-  Matlab, IDL
- Python

$$-\log \left(\begin{array}{c|c} \text{Image} & \text{Mask} \\ \hline \text{Image} & \text{Mask} \end{array} \right) = \text{Result}$$


Some users



CT reconstruction

-  Octopus
- GridRec (+IDL GUI )
-  MuhRec
- Astra, TomoPy, TomoJ etc

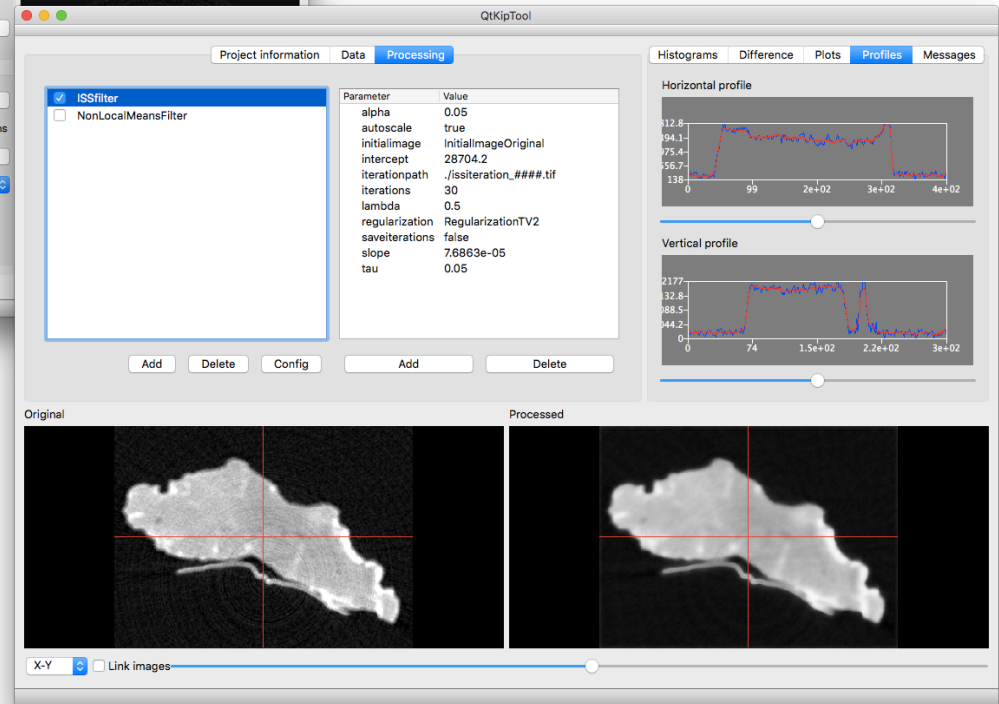
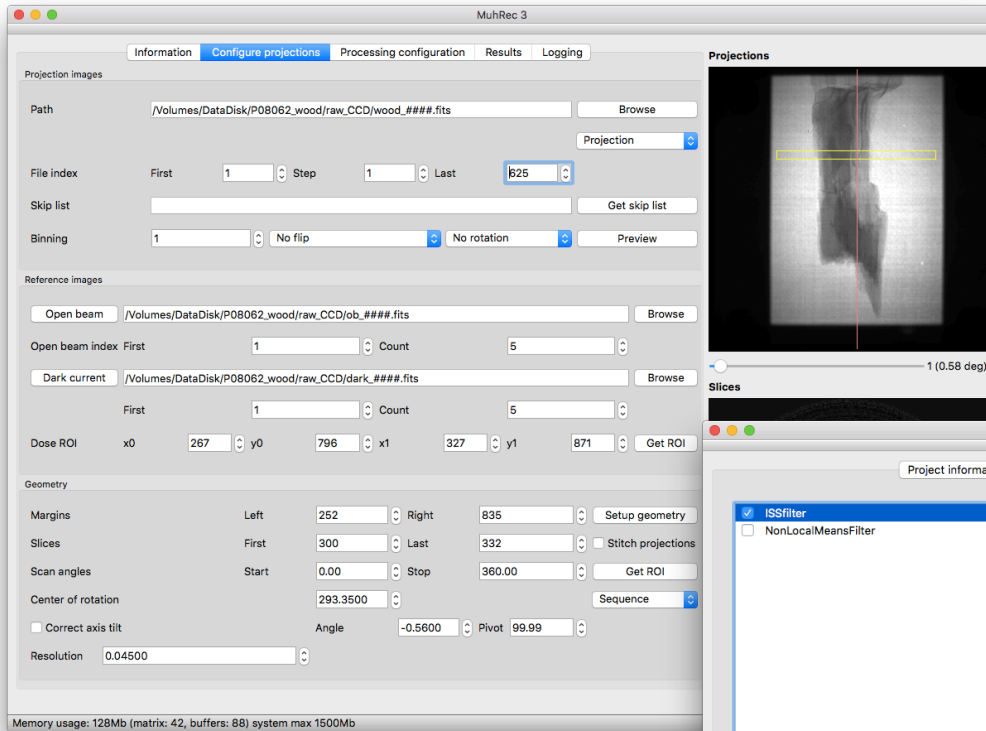


Few users

3D processing, analysis, and visualization

-  VG Studio, Aviso
- ParaView, Drishti, MeVisLab, KNIME, VisTrails
-  KipTool (image processing for special applications)
- Matlab

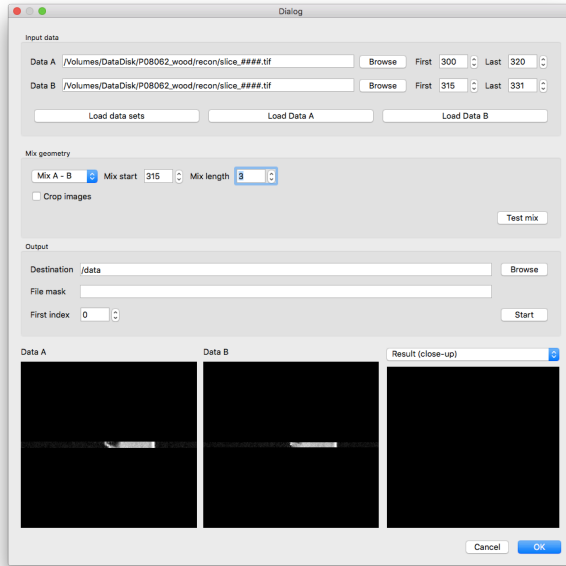
MuhRec: CT reconstructor



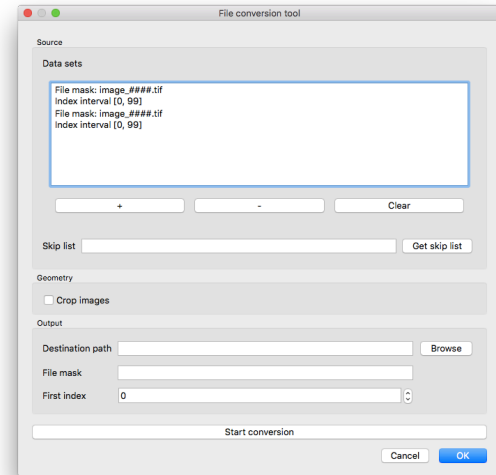
KIPTool: General tool for 3D processing

ImagingTool: A collection of small tools

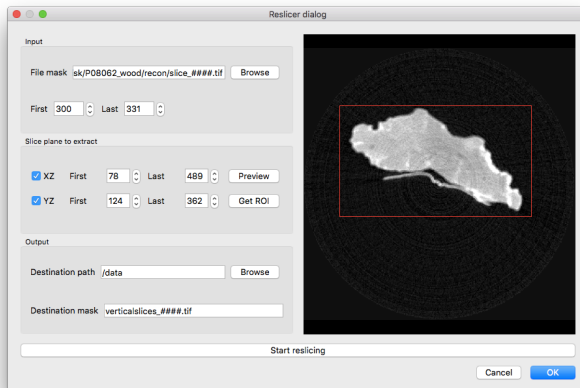
TomoMerge



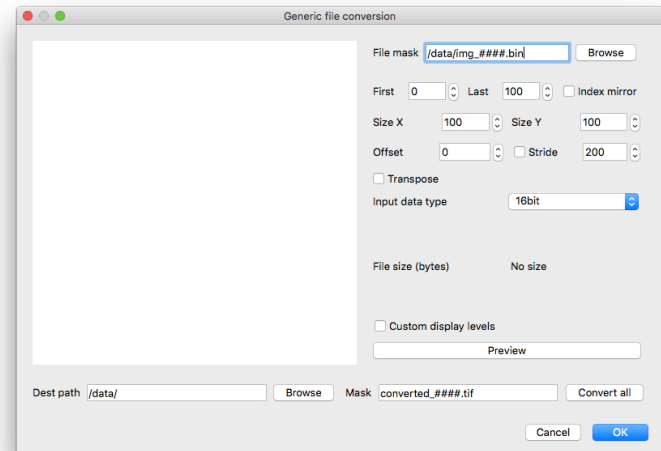
Data renamer

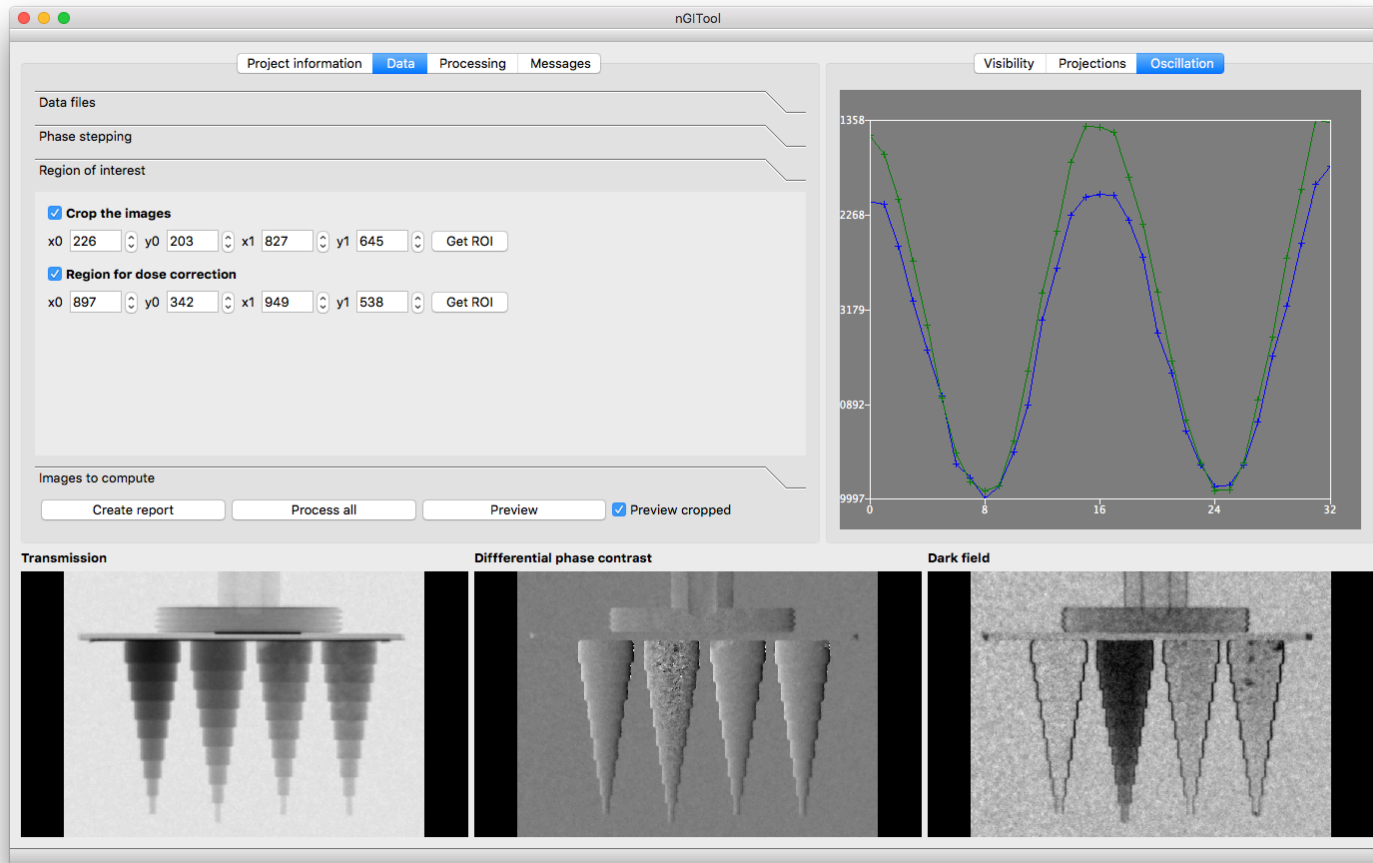


VolumeSlicer



Generic conversion





Efficient processing of phase stepping images from a grating interferometer

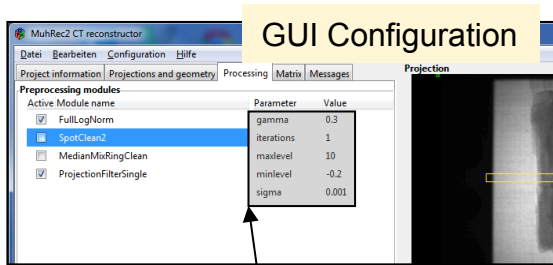
Will only be added to the suite once the algorithms are published

- All applications are available for
 - Windows (64bit), Ubuntu 14.04, MacOS
- GUI is implemented using Qt5.x
- Implementation language C++ (started using C++11)
- Multi-threading
 - Mainly solved using OpenMP (not with Clang on Mac)
 - First tests using C++11 threads
- Development in Qt Creator on all platforms (using QMake)
- No commercial libraries are used
- Until now only few unit tests implemented...
- Limited documentation



The module system

Configuration



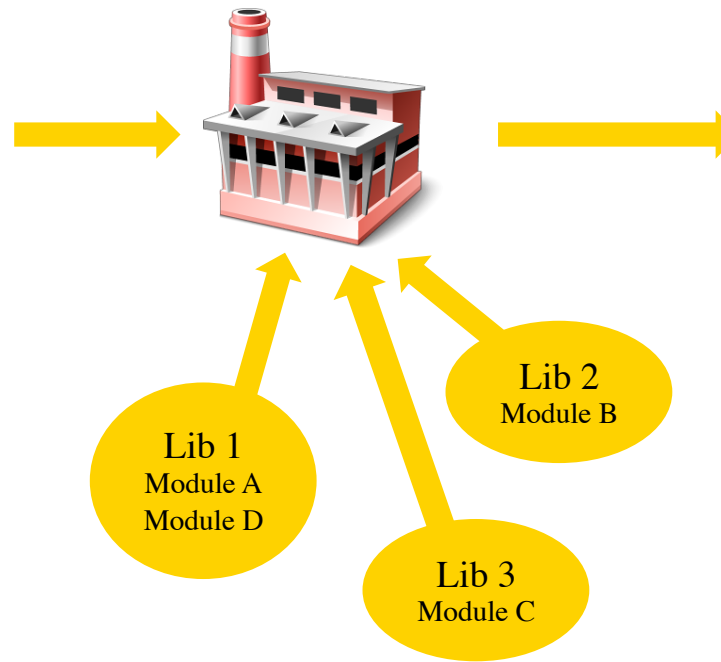
Configuration script

```

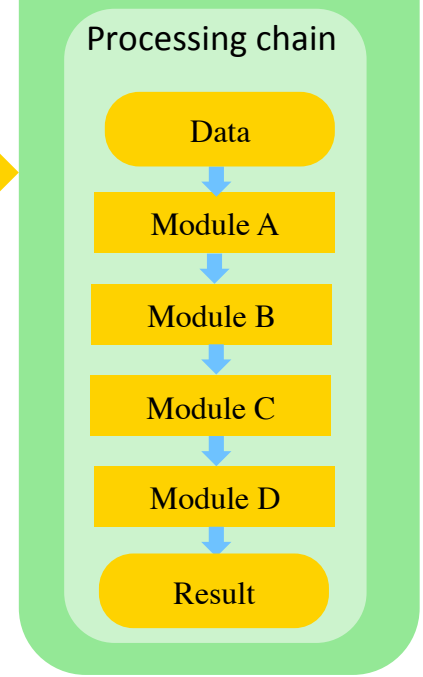
<!-- SpotClean2 -->
<module name="SpotClean2" />
<sharedobject>StdPreprocModules.dll</sharedobject>
<active>false</active>
<parameters>
  <parameter name="gamma" value="0.3" />
  <parameter name="iterations" value="1" />
  <parameter name="maxlevel" value="10" />
  <parameter name="minlevel" value="-0.2" />
  <parameter name="sigma" value="0.001" />
</parameters>
</module>
+ <module>
</pre>

```

Factory



Processing engine



Configuration structs are filled using

- GUI widgets
- XML-scripts

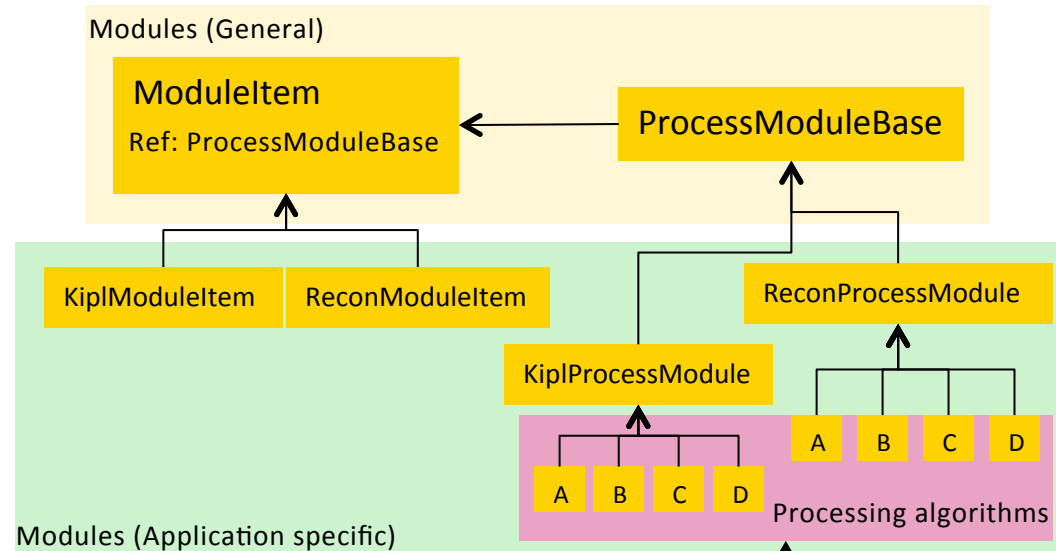
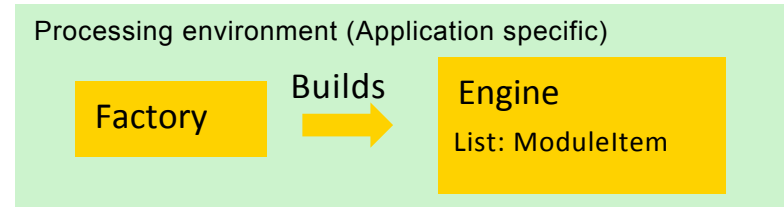
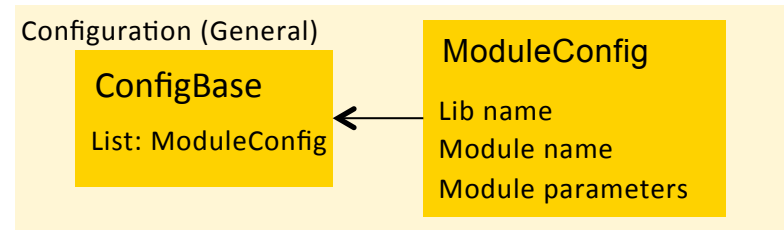
- Dynamic load requested modules
- Parameterize modules

- Load data
- Execute chain
- Store result

The module system classes (non-GUI)

Core module system

- Module description (xml)
- Dynamic loading of modules
- OS specific details supported
- Abstract interface
- Supports
 - Single module loading
 - Chain loading using factory

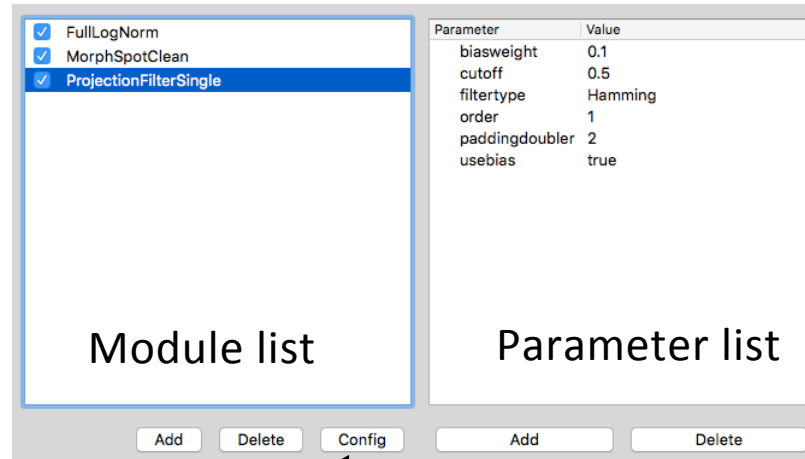


These add new processing and analysis functionality

Module configurator widget



Based on QWidget



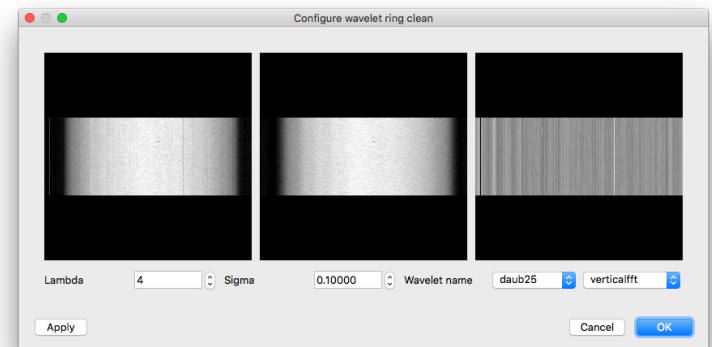
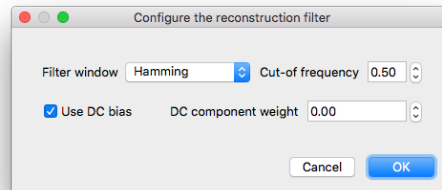
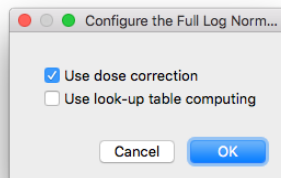
Features module list

- Add/delete modules
- Enable/disable modules
- Rearrange order
- Supporting configuration dialogs

Features parameter list

- Parameters appear for the selected module
- Add/delete parameters
- Edit parameter name
- Edit values (currently only as strings)

Configuration dialog examples



Flow chart Module dialogs

Select module in config widget

Modules are in libModules.xyz
Dialogs in libModulesGUI.xyz

Check if GUI lib is available

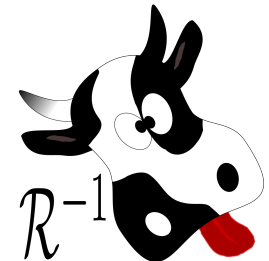
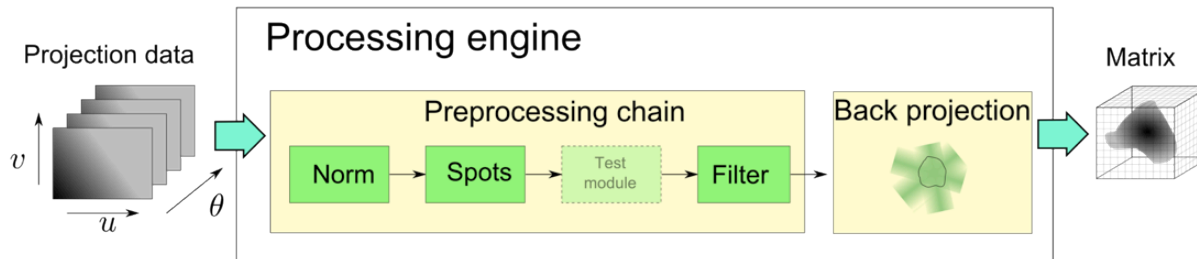
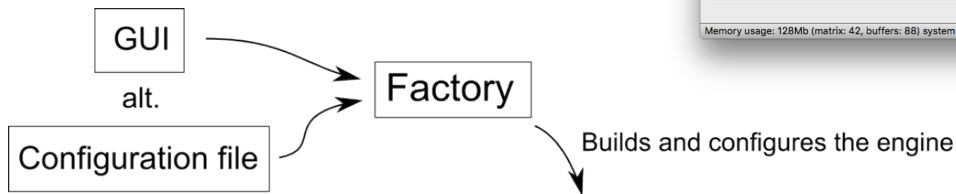
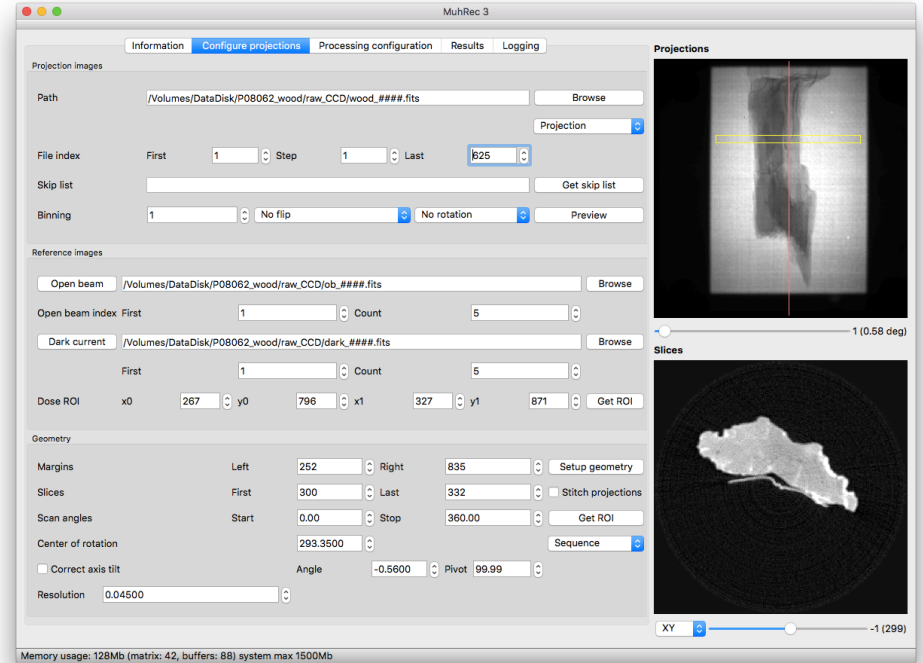
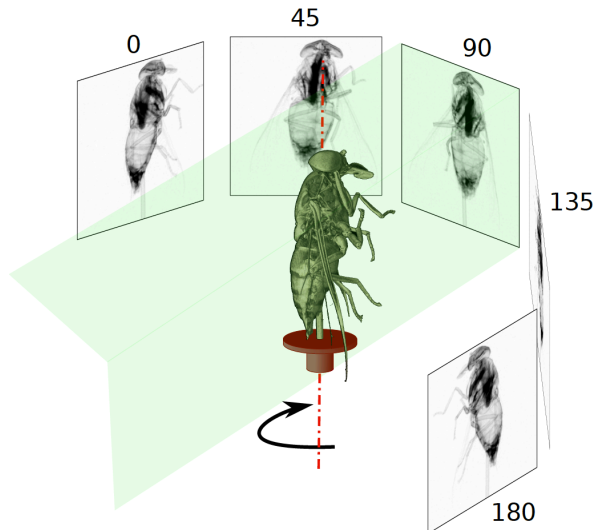
Throw exception

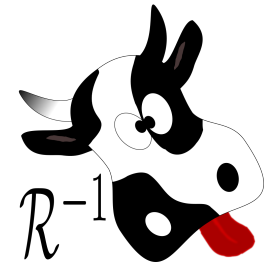
Try to get ModuleDialog from lib

Throw exception

Open dialog with current parameters

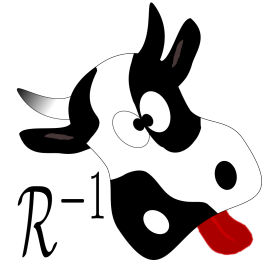
- Module dialogs are based on QDialog
- Dialogs can operate
 - Without images, faster opening
 - With images, the previous steps in the processing chain will be executed





- GUI based CT reconstruction tool
- Based on filtered back projection with parallel beam geometry
- Freely configurable preprocessing chain
 - Normalization
 - Artifact reduction (Spots and Rings)
 - Data correction
 - Projection filter
 - Inspectors
- New functionality can easily be added as modules for pre-proc and recon
- Supported file formats: fits and tiff
- Multi OS support (Windows, Linux, MacOS)
- Can be run using scripts or CLI

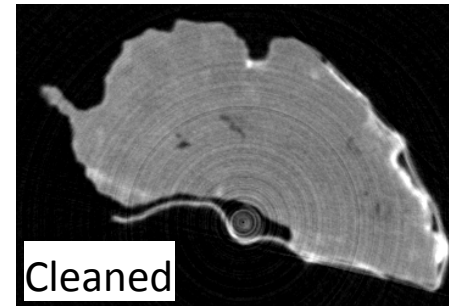
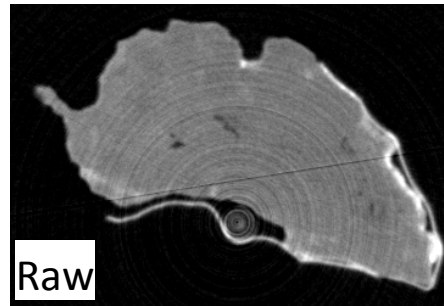
Some common preprocessing modules



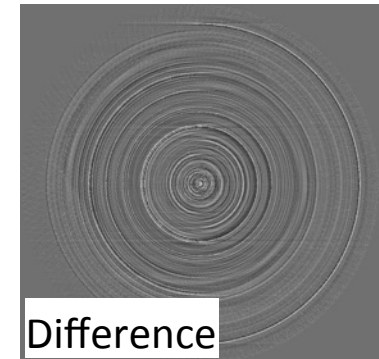
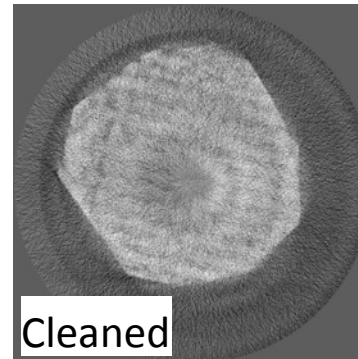
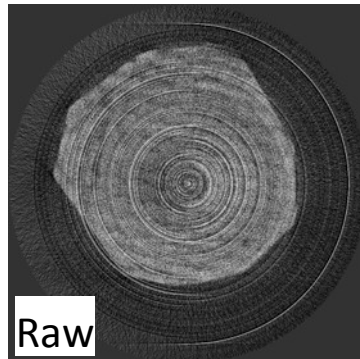
- Normalization

$$-\log \left(\begin{array}{c|c} \text{Raw} & \text{Mask} \\ \hline \text{Raw} & \text{Mask} \end{array} \right) = \text{Normalized}$$

- Spot cleaning



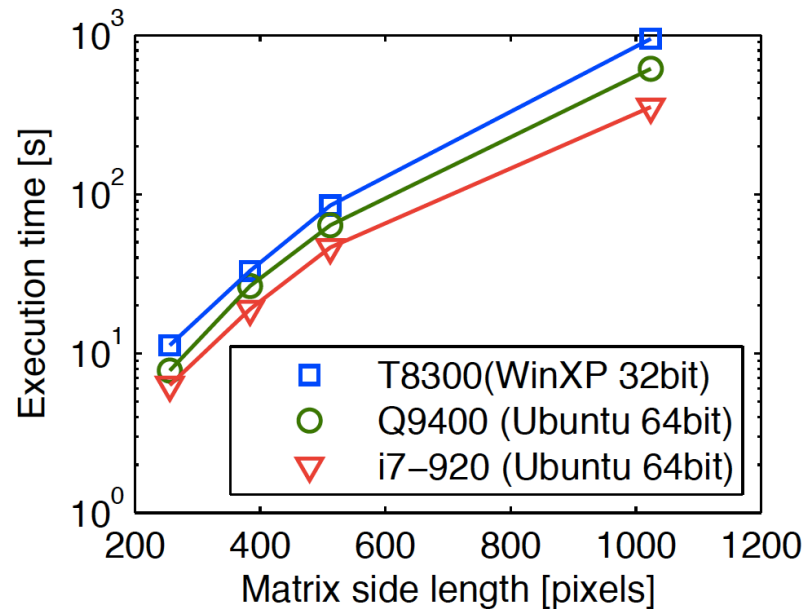
- Ring cleaning



- Reconstruction filter - Ramp + apodization filters

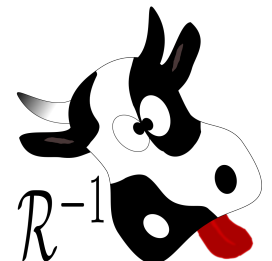
Muhrec: Some word about performance

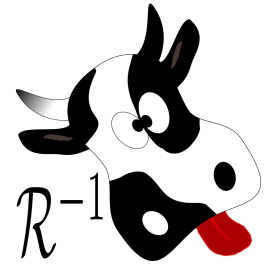
Execution time on a single thread (2011)



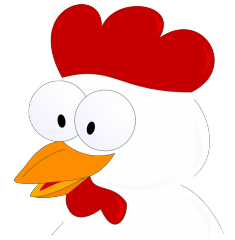
Today:

- The back projector is multi threaded
- Most time is spent on artifact removal
 - There is room for improvement





- Add cone beam geometry
- Improve preprocessing time
 - artifact reduction is time consuming
- Add iterative reconstruction methods
 - Predefined SIRT, DART, SART for single dat
 - Add support for a priori information
- Add support for 4D CT (data flow + regularization)
- Improve GUI stability (inkl. improve user friendliness)
- Improve user documentation
- Add HDF support (e.g. NeXus, Matlab, others?)
- Support for 6D data?
- Port to cluster based recon
- Python scripting (for rapid prototyping modules)



KIPTool was developed to provide a user interface to image processing algorithms

- Works mainly with 3D data
- Load a stack of images
- 2D slice inspection
- Based on a chain processor
- Results can be compared with original
 - Histogram
 - Profiles
 - Difference
- Several module libraries

Base

- Scale data
- Data clamping
- Dose correction
- Volume projection

Advanced filters

- ROF filter (ISS)
- Non-linear diffusion
- Non-local means

Classification

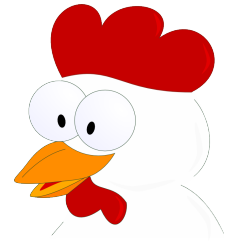
- Double threshold
- Fuzzy C-Means
- Remove background

PCA

- PCA filter (experimental)

Pore space

- Pore size distribution (experimental)



The ROF filter model is an edge preserving denoising filter

Parameter	Value
alpha	0.01
autoscale	true
initialim...	InitialImageOriginal
intercept	0
iteration...	./issiteration_###.tif
iterations	100
lambda	1
regulariz...	RegularizationTV2
saveitera...	false
slope	1
tau	0.01

Plots and information

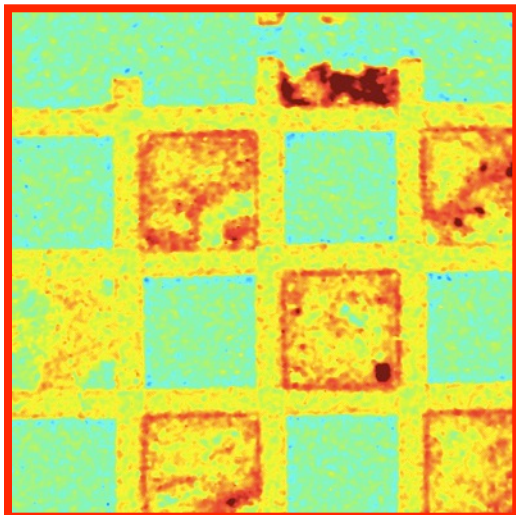
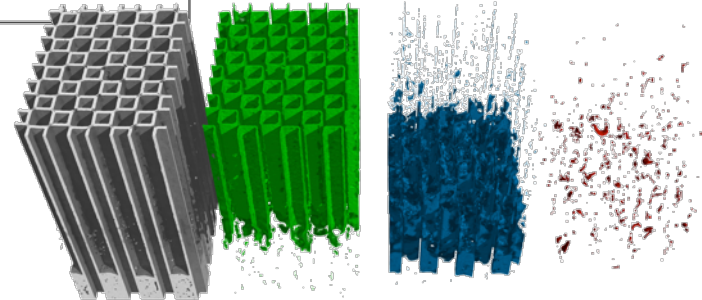
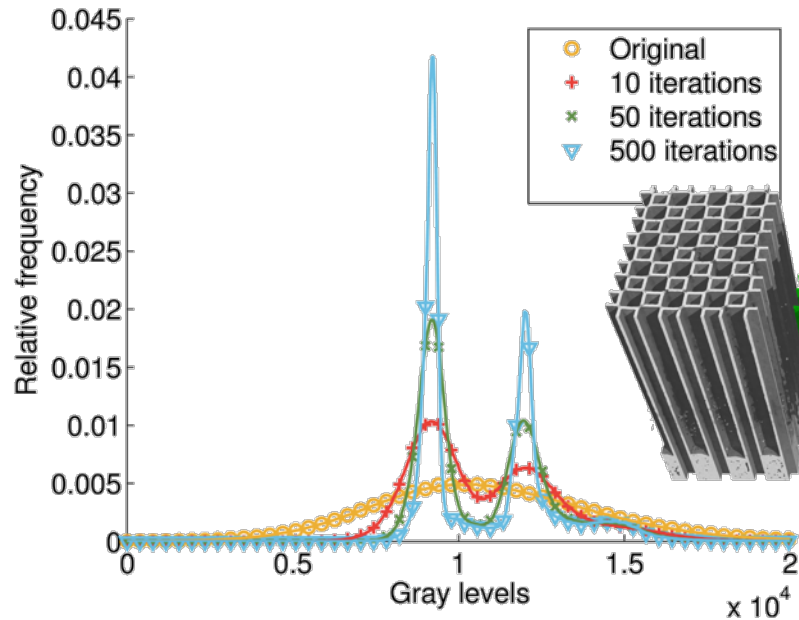
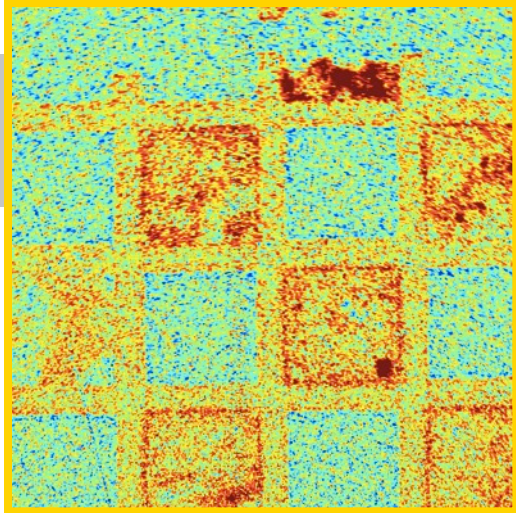
Horizontal profile

Vertical profile

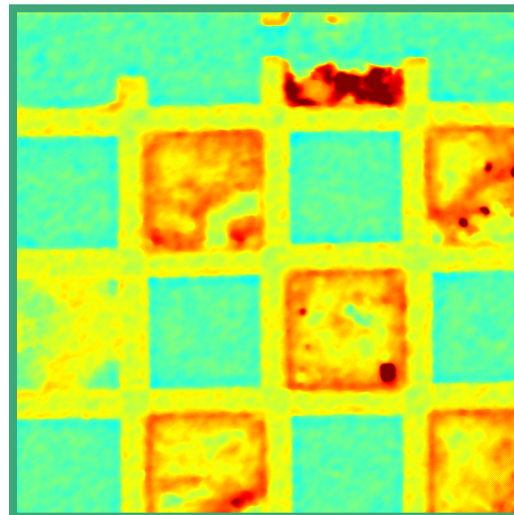
[message] IS
[message] IS
[message] IS
[message] IS
[message] KiplEngine: Execution times :
Module ISSFilterModule: 4.74628s
[message] KipToolMainWindow: The process
ended successfully
[message] ImagePainter: Erase

Save Clear Messag

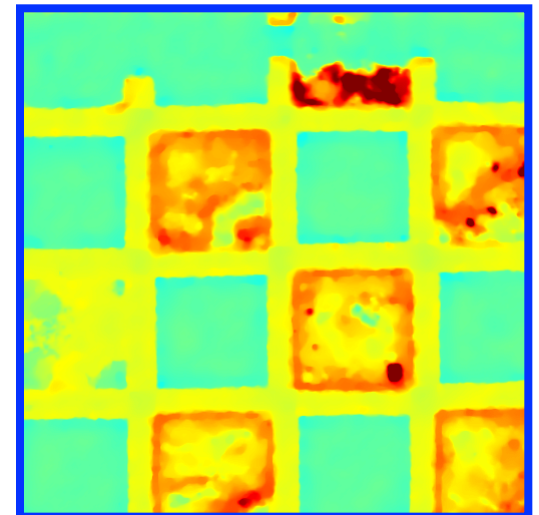
Demonstration ROF filter



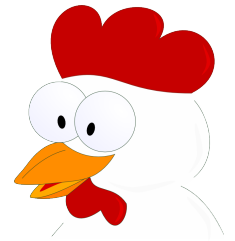
10 iterations



50 iterations



500 iterations



- Add more modules
 - Instrument related processing
 - Registration
 - Segmentation
 - Analysis
 - Quantification
- Change processing chain to network
 - Node editor
 - Multiple image support
- 2D and 4D support
- Scripting in Python
 - Using modules
 - Calling library functions

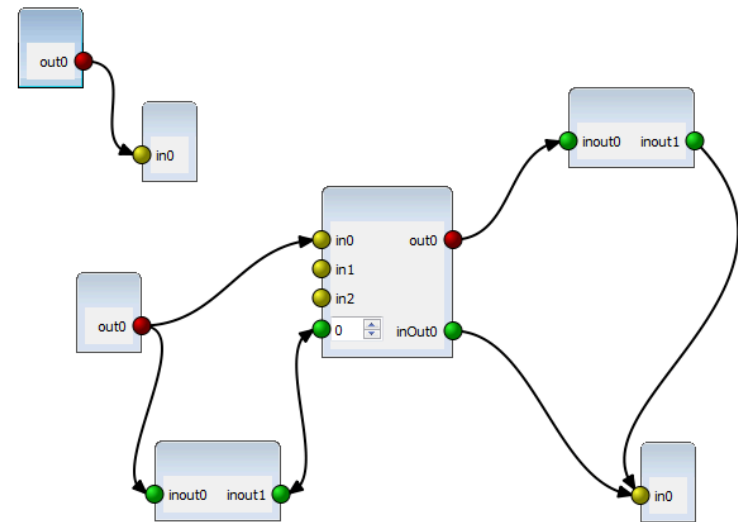
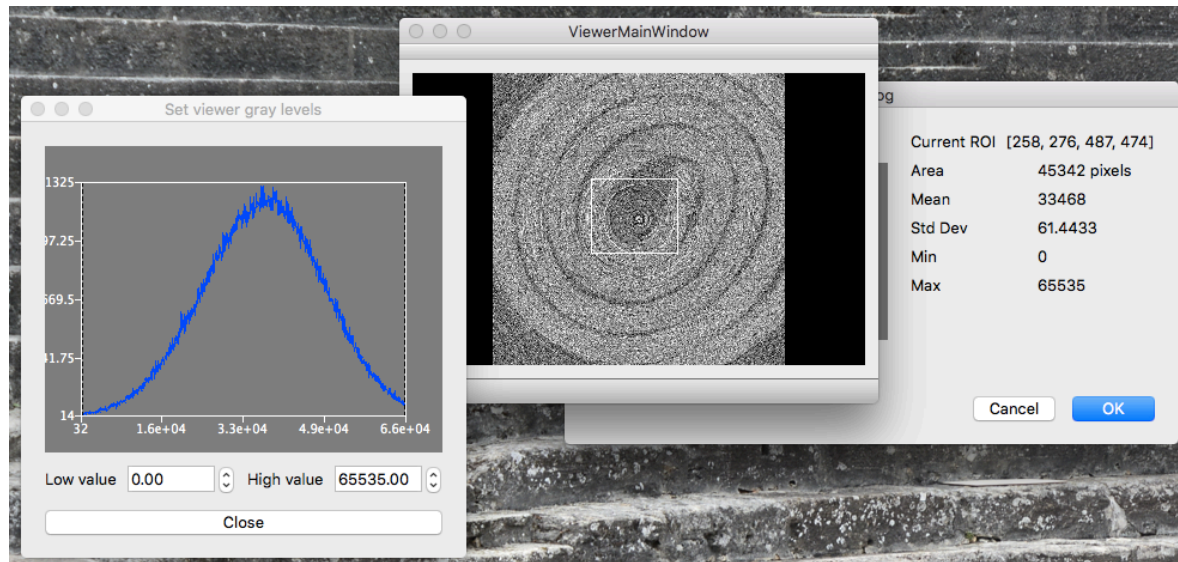
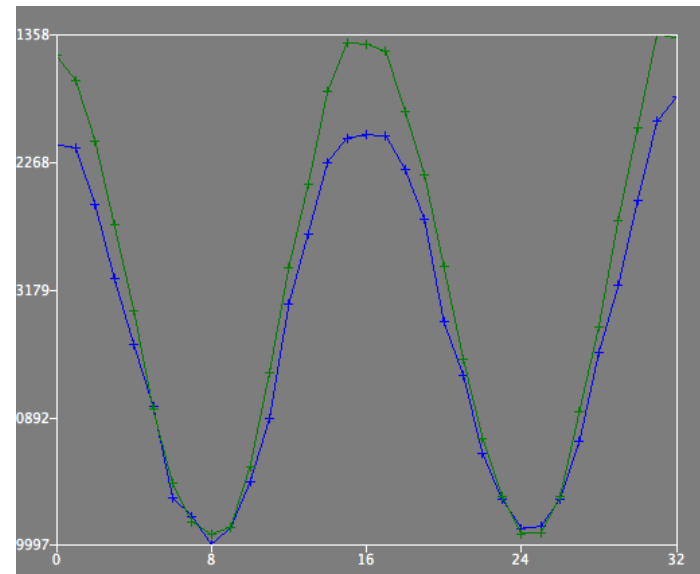


Image viewer widget

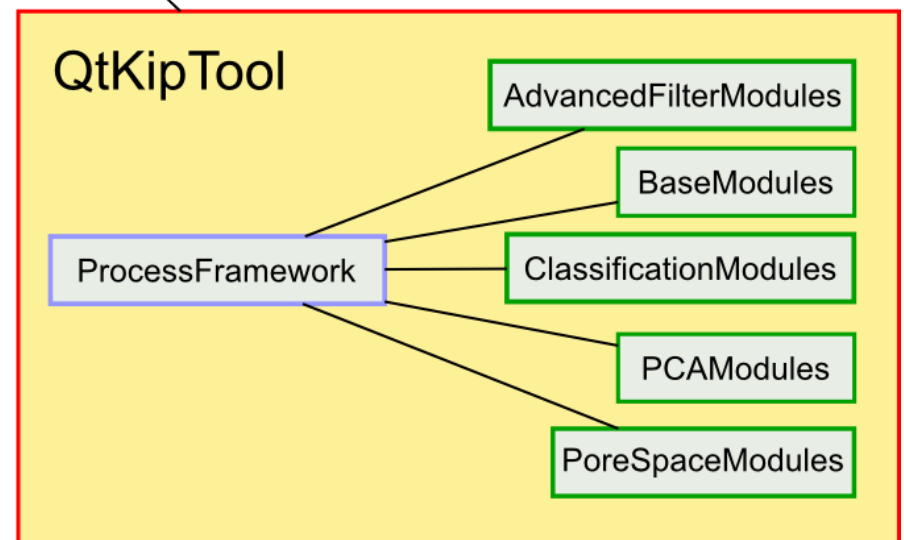
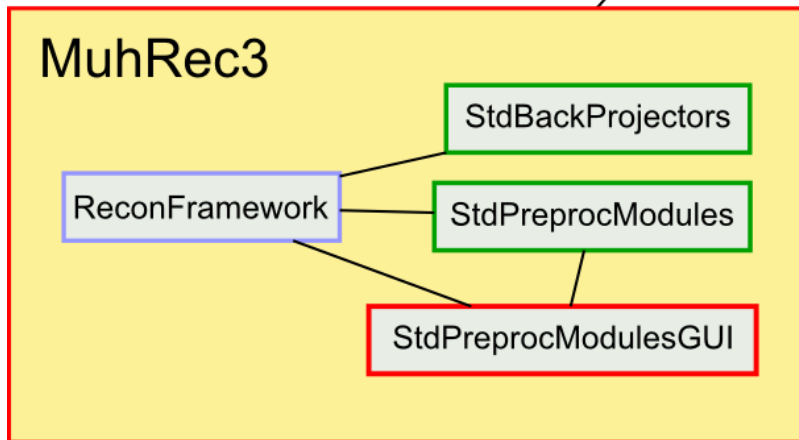
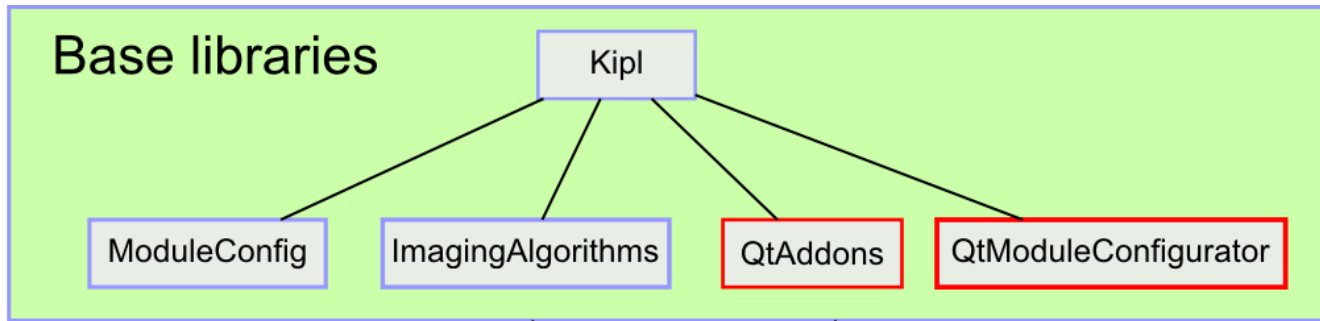


- Display of floating point data
- Handles rectangular ROIs
- Intensity measurements in ROI
- Setting displayed intensity interval using dialog or right mouse
- Shows intensity and mouse position as tool tip
- Can link settings with other image viewers
- Supports plots and markers
- Based on Qwidget
- Can be used in PDF reports

- A basic plotting widget
- Multiple plots and colors
- Horizontal and vertical markers lines
- Zoom function
- Glyphs on data points
- Based on Qwidget
- Can be used in PDF reports



Current libraries – an overview



Visualization is still an open question

Currently commercial (very expensive) full featured tools are used

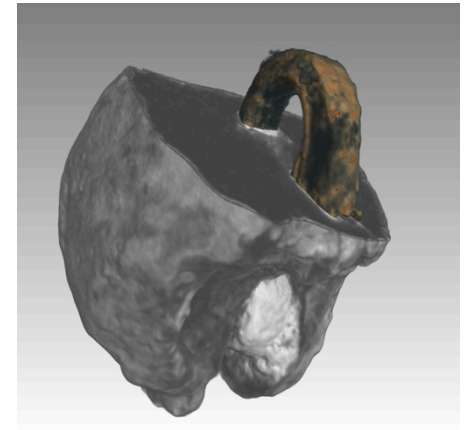
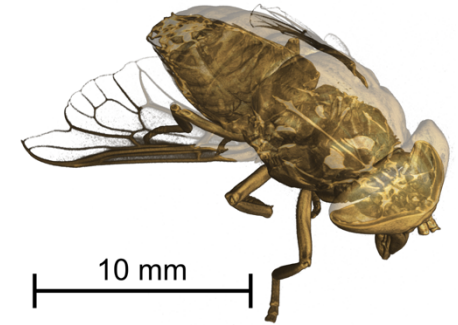
- VG Studio
- Aviso

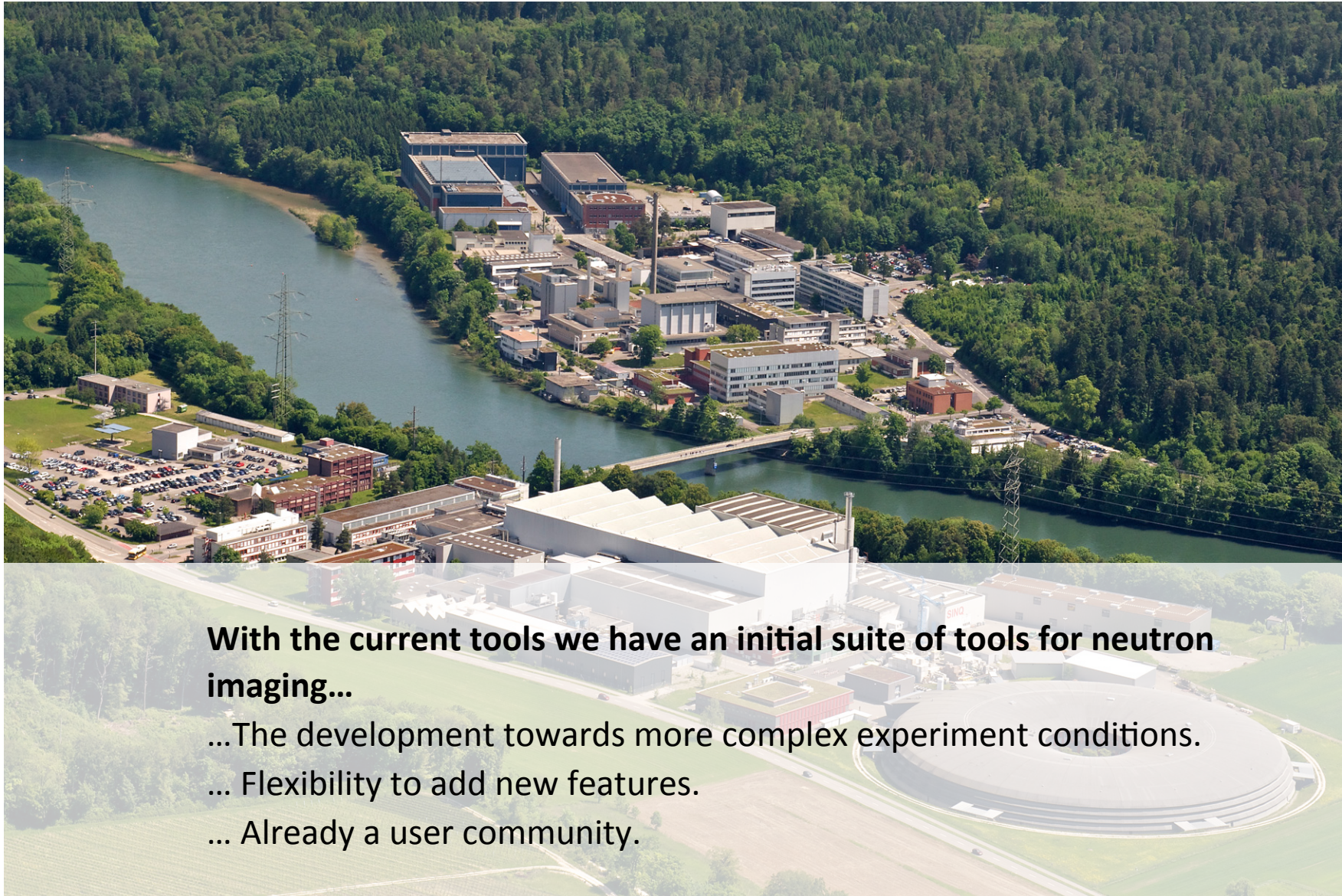
They solve many tasks but fail with more specific tasks

- 4D data (time series of volumes)
- 6D (vectors in 3D)
- Multivariate visualization

These topics will be targeted during a planned workshop organized with COST action MP1207: **Visual analysis of dynamic processes**

Place: Rigi Kulm, CH, January 2017





With the current tools we have an initial suite of tools for neutron imaging...

...The development towards more complex experiment conditions.

... Flexibility to add new features.

... Already a user community.