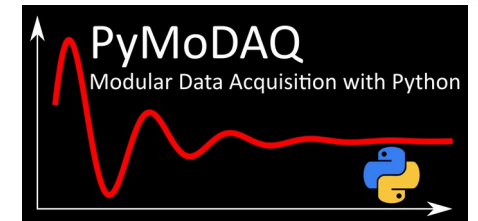




CEMES

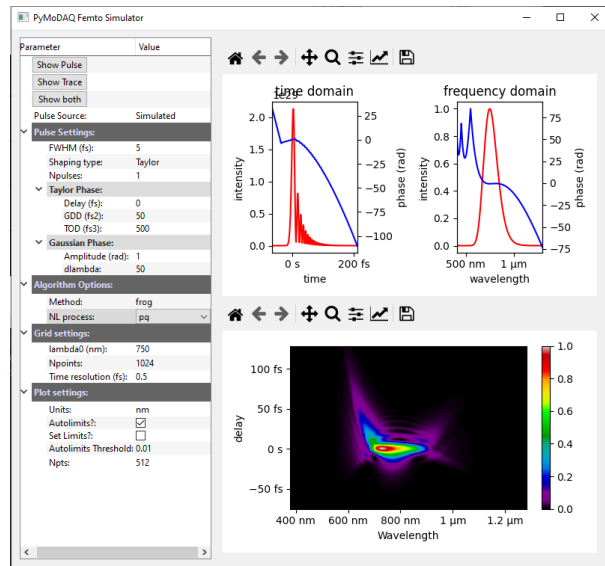
PyMoDAQ-Femto Practical Session Sébastien Weber

© Patri



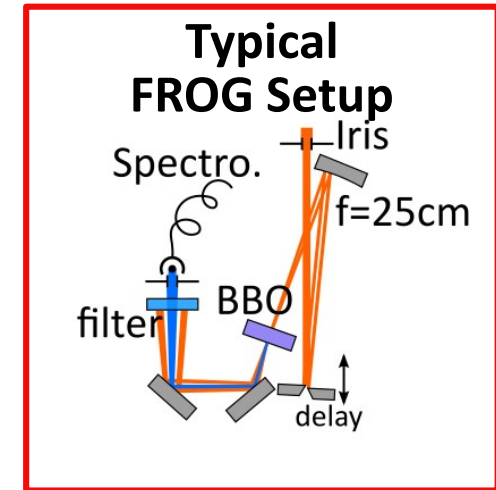
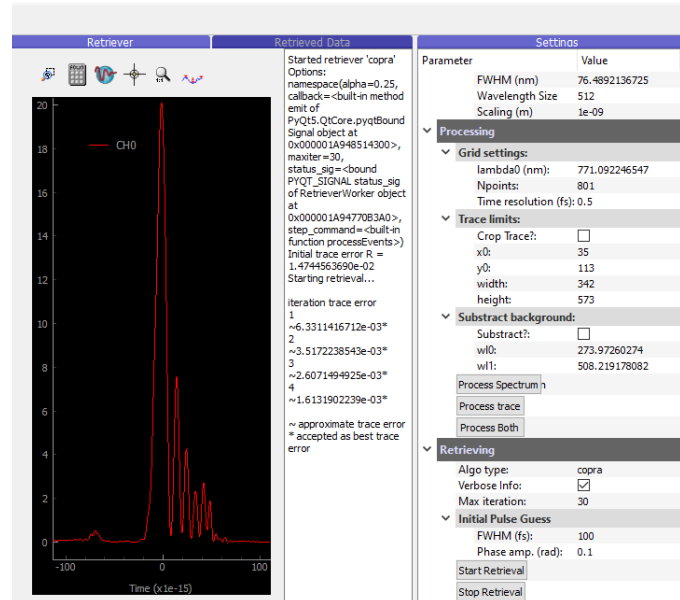
Goal of the practical:

- Learn how to use the Simulator
- Use the Retriever to process the data and export results
- Use PyMoDAQ to simulate real lab experiment (SHG FROG, DSCANs) and load data into the retriever

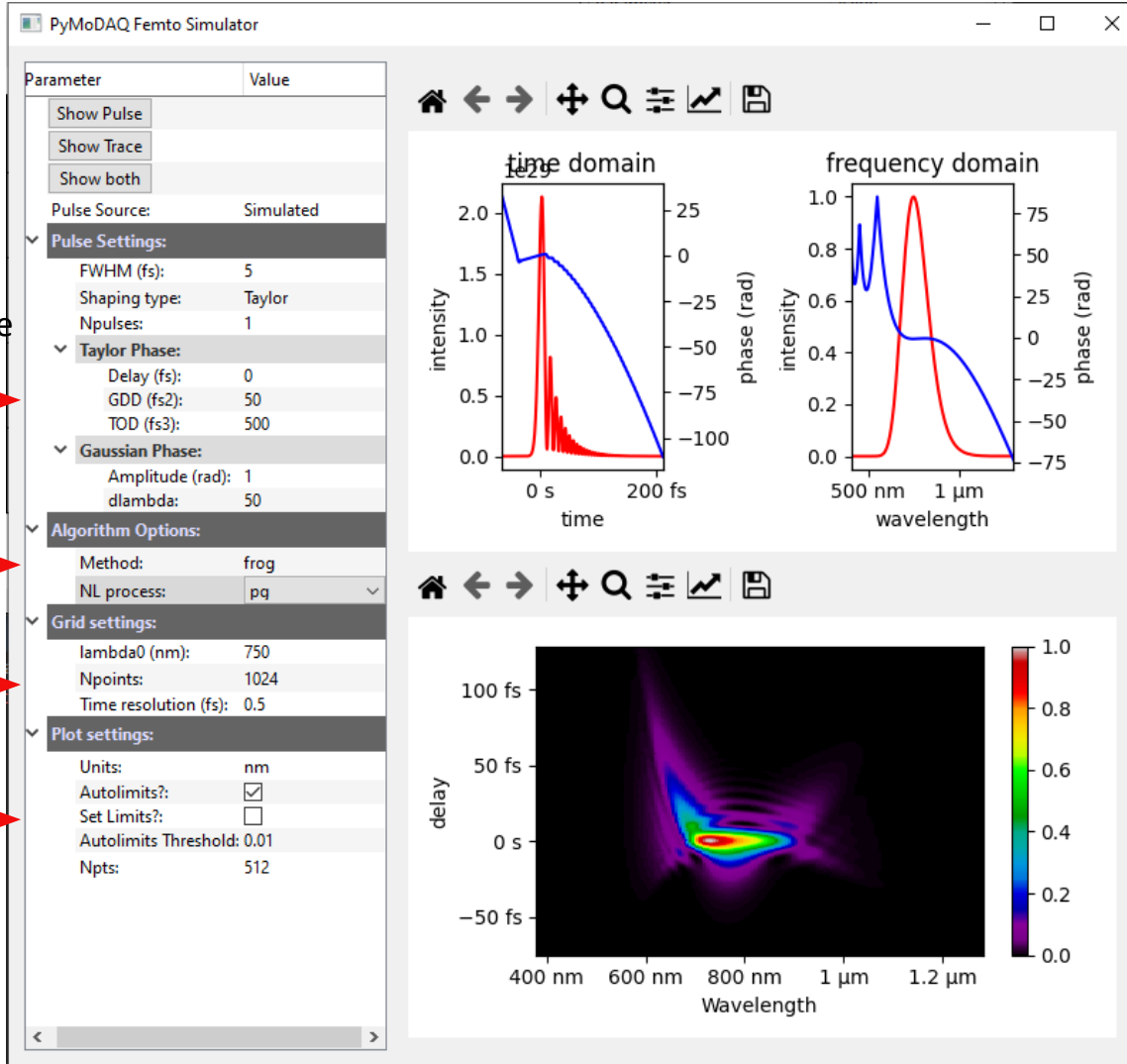


Simulator

Retriever



Acquisition



Practical (plot and export graphs):

- 5fs SHG Frog with :
 - 200 fs² GDD
 - -200 fs² GDD
 - Conclusion ?
- 5fs PG FROG
 - 200fs² GDD
 - -200fs² GDD
 - Conclusion
- 5fs PG FROG
 - 500fs² GDD
 - Conclusion
 - Increase temporal axis (Npts)
- 20fs SHG-FROG
 - Gaussian phase (amp=1rad, dt=10fs)
 - Same but 2 pulses Delay 25fs

Pulse Shaper

- Taylor spectral phase
- Gaussian temporal phase
- multipulse

Algo. options

Temporal Grid
(for fft)

Plot options



Retriever

```
(pymodaq_femto) C:\>retriever
```

Load data from file

Load Simulator and data from it

Method and nonlinearity

PyMoDAQ Retriever

Main Data In

Data In Processed Data Retriever Retrieved Data

Wavelength (μm)

Delay (fs.u.)

Spectrum

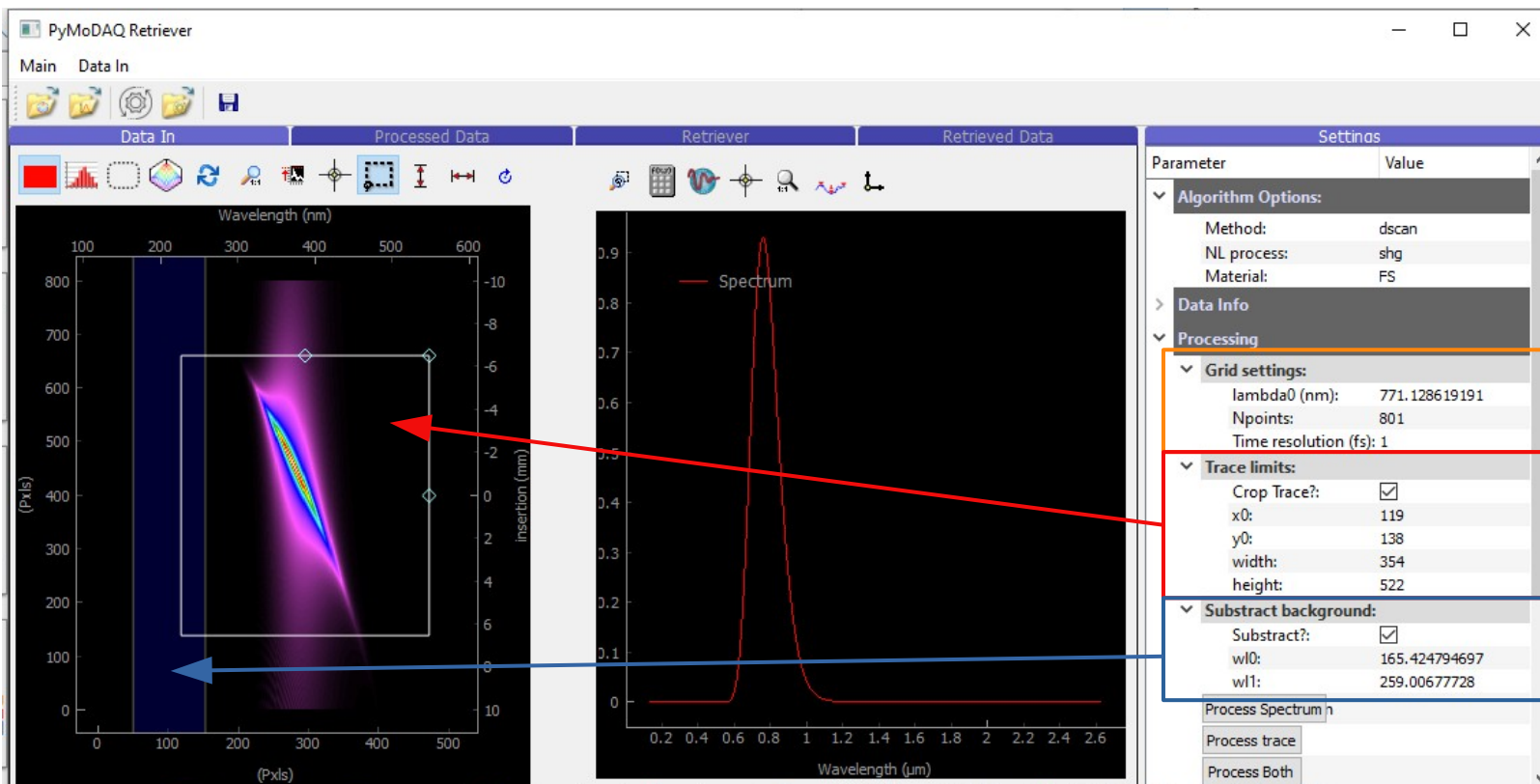
Wavelength (μm)

Settings

Parameter	Value
Algorithm Options:	
Method:	dscan
NL process:	shg
Material:	FS
Data Info	
Loaded file:	
C:\Data\2021\20210315\Dataset_20210315_001\Dataset_20210315_001.h5	
Loaded node:	/Raw_datas/Scan001/t
Trace Info	
W10 (nm)	780.566716128
FWHM (nm)	95.2889277432
Param Size	4
Wavelength Size	512
Scaling (m)	1e-09
Scaling Parameter	1e-15
Spectrum Info	
W10 (nm)	803.543283834
FWHM (nm)	85.2354945141
Wavelength Size	512
Scaling (m)	1e-09

infos

Retriever: preprocessing

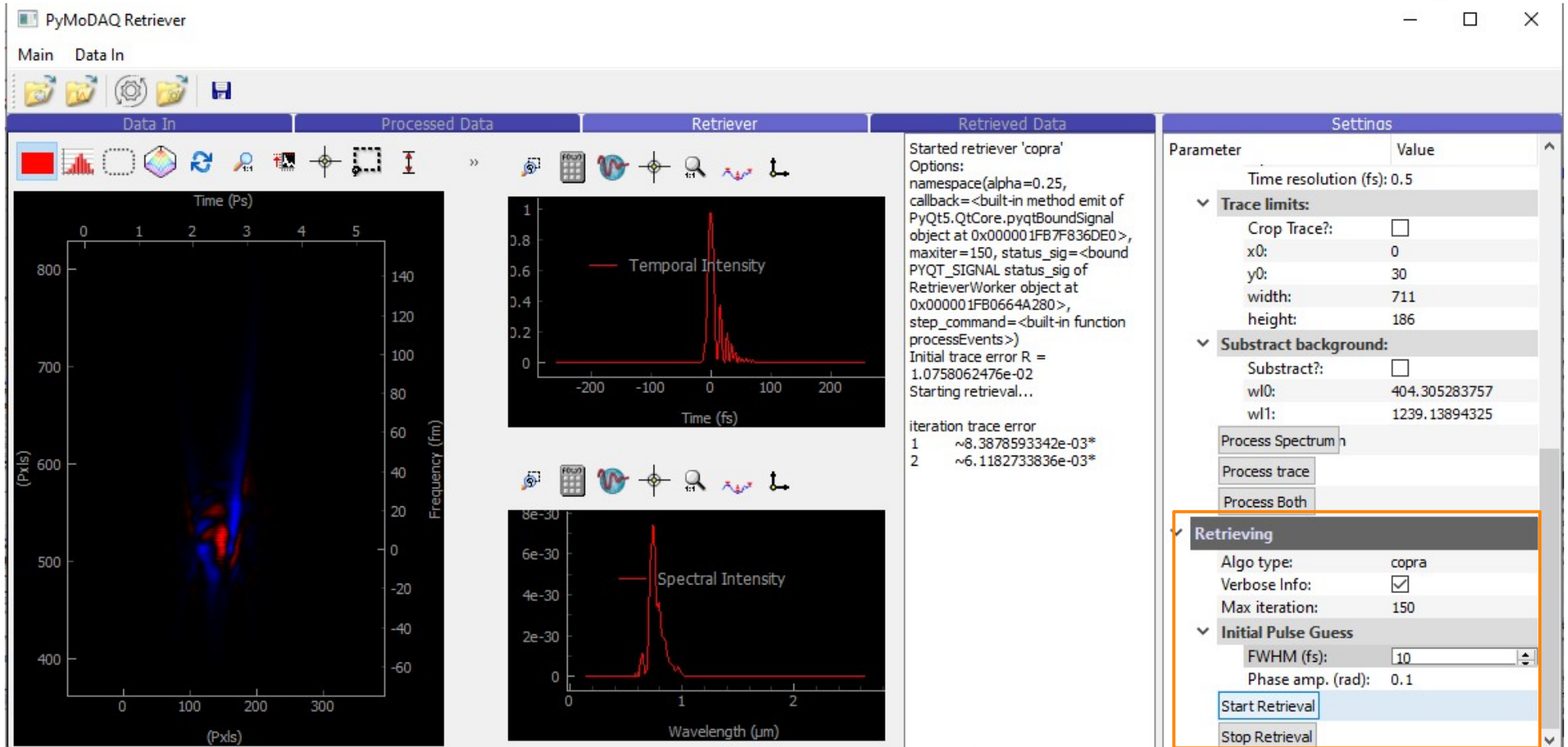


FFT Grid definition

Limiting the trace

Subtracting some background

Retriever: retrieval algo



The screenshot displays the PyMoDAQ Retriever software interface, which is divided into several sections:

- Data In:** Shows a 2D spectrogram plot with Time (Ps) on the x-axis (0 to 5) and Frequency (fm) on the y-axis (400 to 800).
- Processed Data:** Contains two plots:
 - Temporal Intensity:** A line plot showing intensity versus Time (fs) from -200 to 200, featuring a sharp peak at 0 fs.
 - Spectral Intensity:** A line plot showing intensity versus Wavelength (μm) from 0 to 2, with a peak around 0.8 μm .
- Retriever:** A central text area providing real-time feedback on the retrieval process:

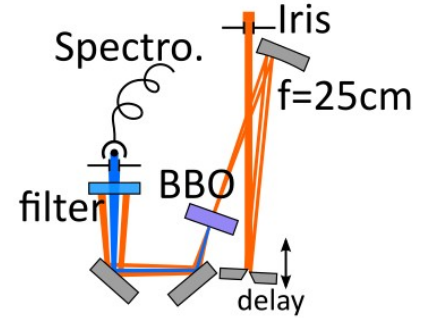

```
Started retriever 'copra'
Options:
namespace(alpha=0.25,
callback=<built-in method emit of
PyQt5.QtCore.pyqtBoundSignal
object at 0x000001FB7F836DE0>,
maxiter=150, status_sig=<bound
PYQT_SIGNAL status_sig of
RetrieverWorker object at
0x000001FB0664A280>,
step_command=<built-in function
processEvents>)
Initial trace error R =
1.0758062476e-02
Starting retrieval...

iteration trace error
1 ~8.3878593342e-03*
2 ~6.1182733836e-03*
```
- Retrieved Data:** A section for displaying the results of the retrieval process.
- Settings:** A configuration panel on the right with various parameters:

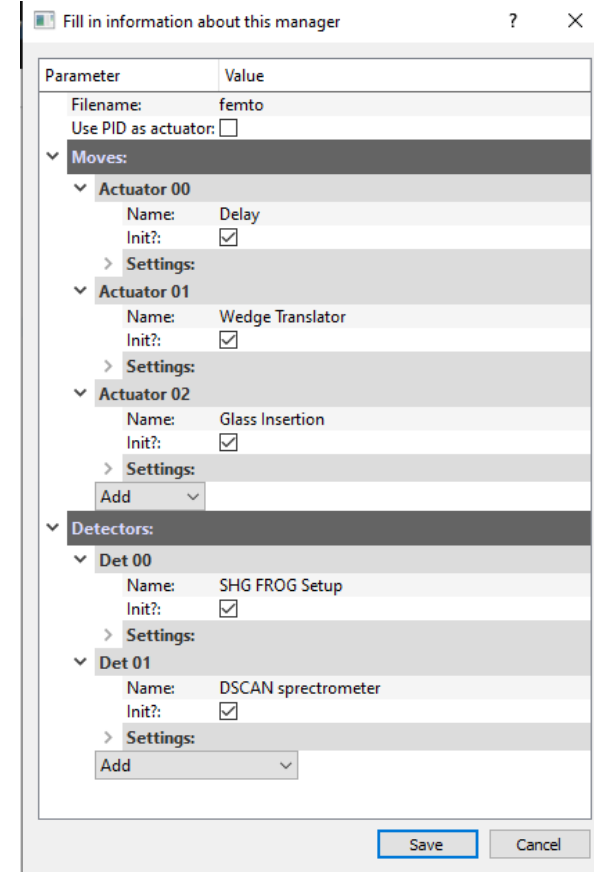
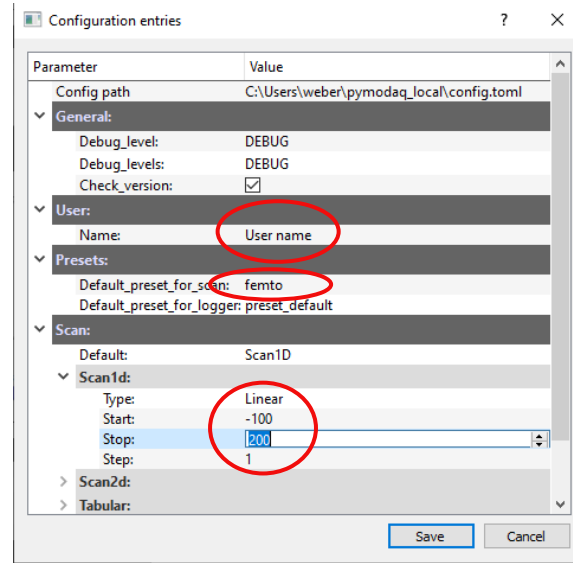
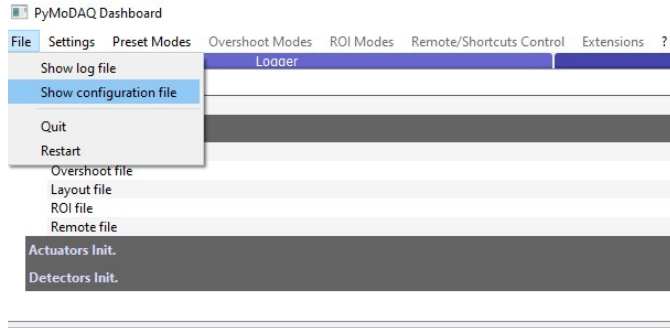
Parameter	Value
Time resolution (fs):	0.5
Trace limits:	
Crop Trace?:	<input type="checkbox"/>
x0:	0
y0:	30
width:	711
height:	186
Subtract background:	
Subtract?:	<input type="checkbox"/>
wl0:	404.305283757
wl1:	1239.13894325
Process Spectrum	<input type="checkbox"/>
Process trace	<input type="checkbox"/>
Process Both	<input type="checkbox"/>
Retrieving	
Algo type:	copra
Verbose Info:	<input checked="" type="checkbox"/>
Max iteration:	150
Initial Pulse Guess	
FWHM (fs):	10
Phase amp. (rad):	0.1
Start Retrieval	<input type="button"/>
Stop Retrieval	<input type="button"/>

Live infos

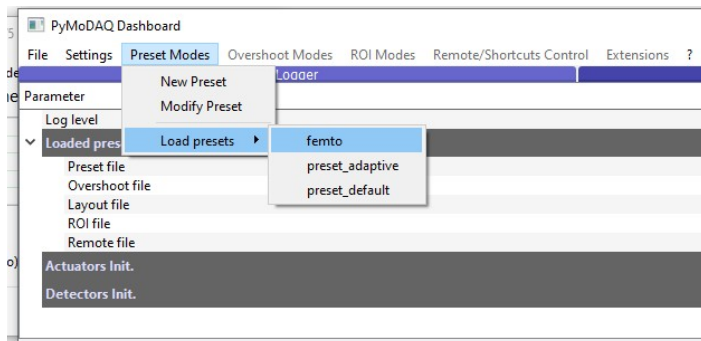
Retrieval algo options



1) Base Configuration file

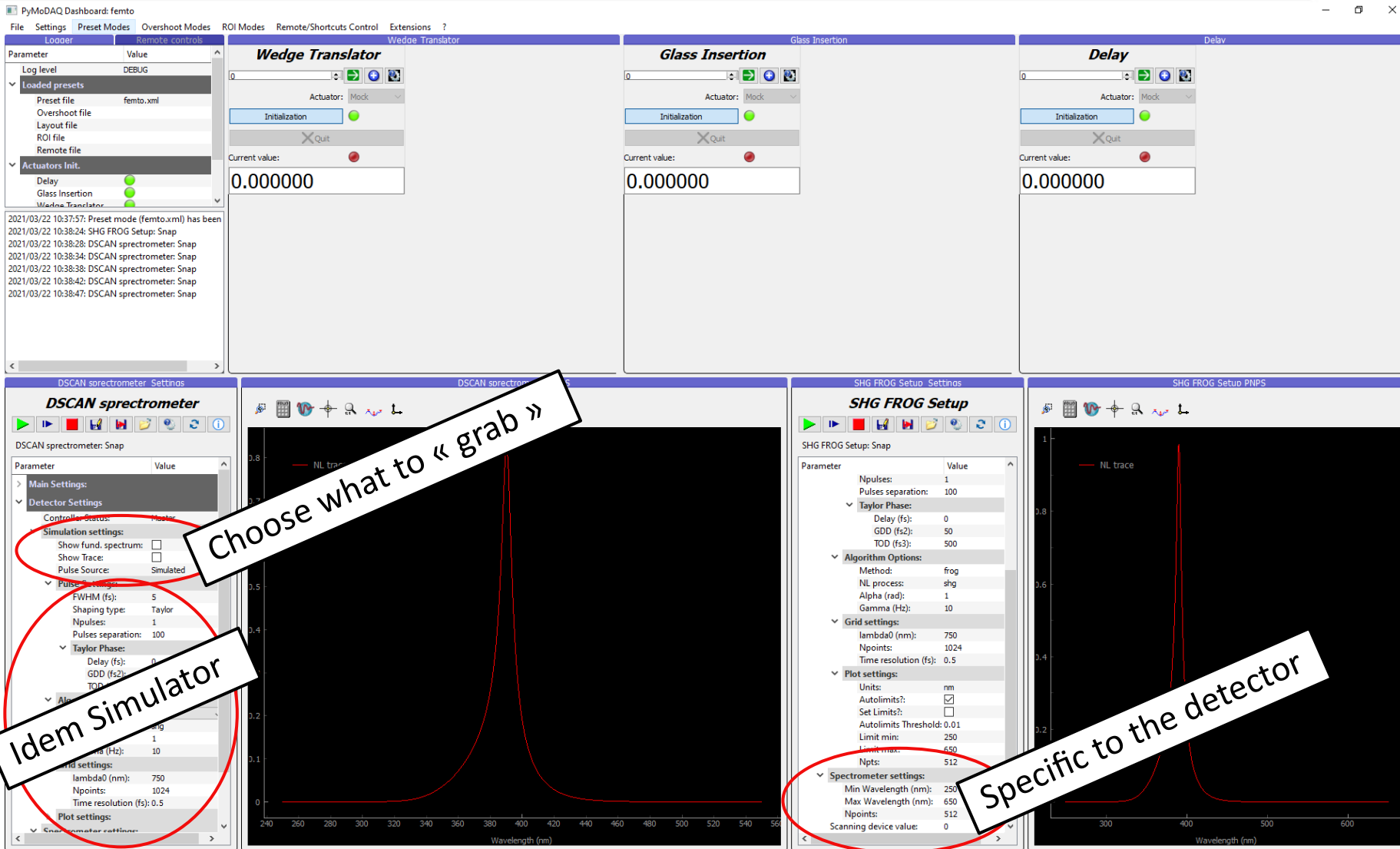


2) Experiment Configuration file (preset)



To edit the content
Use « Modify Preset »

Settings up the control Modules



The screenshot displays the PyMoDAQ Dashboard interface for the femto system. The top row contains control panels for **Wedge Translator**, **Glass Insertion**, and **Delay**, each with a slider, actuator dropdown (set to 'Mock'), and 'Initialization' button. The bottom row shows simulation results for **DSCAN spectrometer** and **SHG FROG Setup**.

DSCAN spectrometer Settings:

- Controller Status: Master
- Simulation settings: Show fund. spectrum, Show Trace, Pulse Source: Simulated
- Pulse settings: FWHM (fs): 5, Shaping type: Taylor, Npulses: 1, Pulses separation: 100
- Taylor Phase: Delay (fs): 0, GDD (fs²): 50, TOD (fs³): 500
- Algorithm Options: Method: frog, NL process: shg, Alpha (rad): 1, Gamma (Hz): 10
- Grid settings: lambda₀ (nm): 750, Npoints: 1024, Time resolution (fs): 0.5
- Plot settings: Units: nm, Autolimits?: , Set Limits?: , Autolimits Threshold: 0.01, Limit min: 250, Limit max: 650, Npts: 512
- Spectrometer settings: Min Wavelength (nm): 250, Max Wavelength (nm): 650, Npoints: 512, Scanning device value: 0

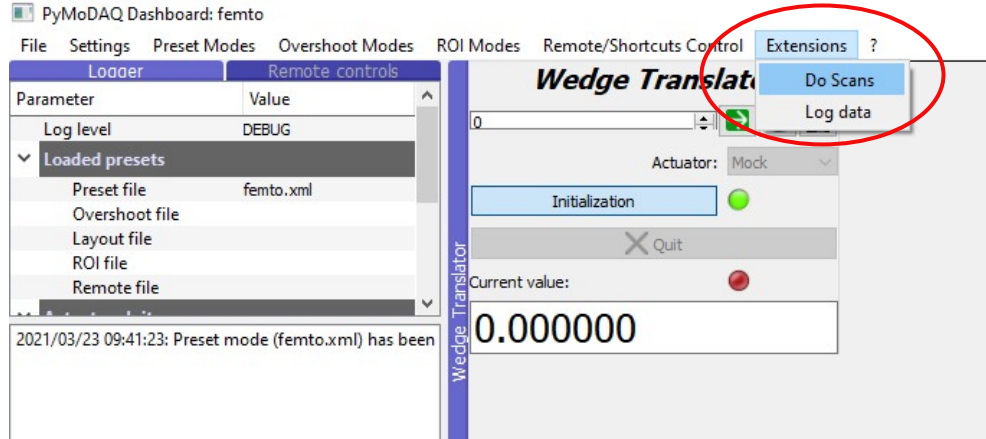
SHG FROG Setup Settings:

- Npulses: 1, Pulses separation: 100
- Taylor Phase: Delay (fs): 0, GDD (fs²): 50, TOD (fs³): 500
- Algorithm Options: Method: frog, NL process: shg, Alpha (rad): 1, Gamma (Hz): 10
- Grid settings: lambda₀ (nm): 750, Npoints: 1024, Time resolution (fs): 0.5
- Plot settings: Units: nm, Autolimits?: , Set Limits?: , Autolimits Threshold: 0.01, Limit min: 250, Limit max: 650, Npts: 512
- Spectrometer settings: Min Wavelength (nm): 250, Max Wavelength (nm): 650, Npoints: 512, Scanning device value: 0

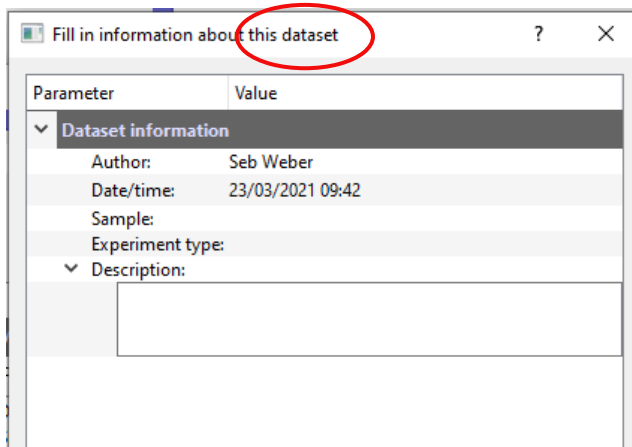
Two callout boxes provide instructions:

- Choose what to « grab »**: Points to the 'Show Trace' checkbox in the DSCAN spectrometer settings.
- Idem Simulator**: Points to the 'Simulation settings' section in the DSCAN spectrometer settings.
- Specific to the detector**: Points to the 'Spectrometer settings' section in the SHG FROG Setup settings.

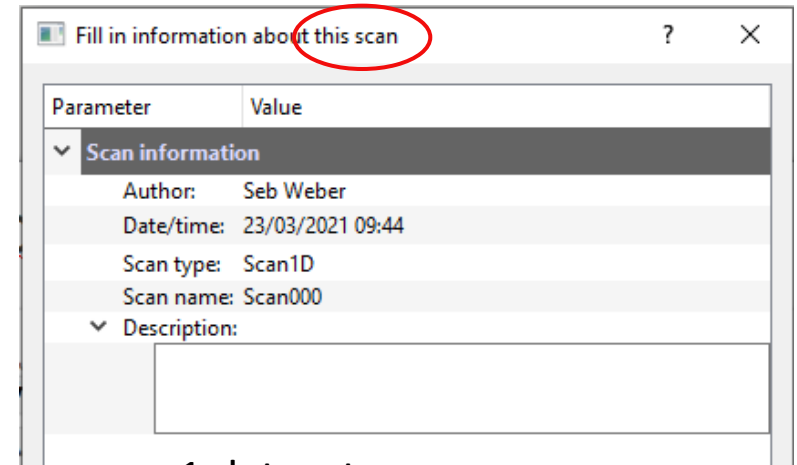
DAQScan Extension: loading and metadata



Metadata



1 hdf5 file == 1 dataset
General information about
the file content



1 dataset == many scans
Specific information about each scan

python Scan

File Settings

Quit Set Scan Init Positions

Parameter	Value
Actuators/Detectors Selection	
detectors	
Det 1D	
Det 0D	
Det 2D	
Actuators	
Theta Axis	
Yaxis	
Xaxis	
Moves done?	<input checked="" type="checkbox"/>
Detections done?	<input checked="" type="checkbox"/>
Data dimensions	
Actuators positions	

General Settings

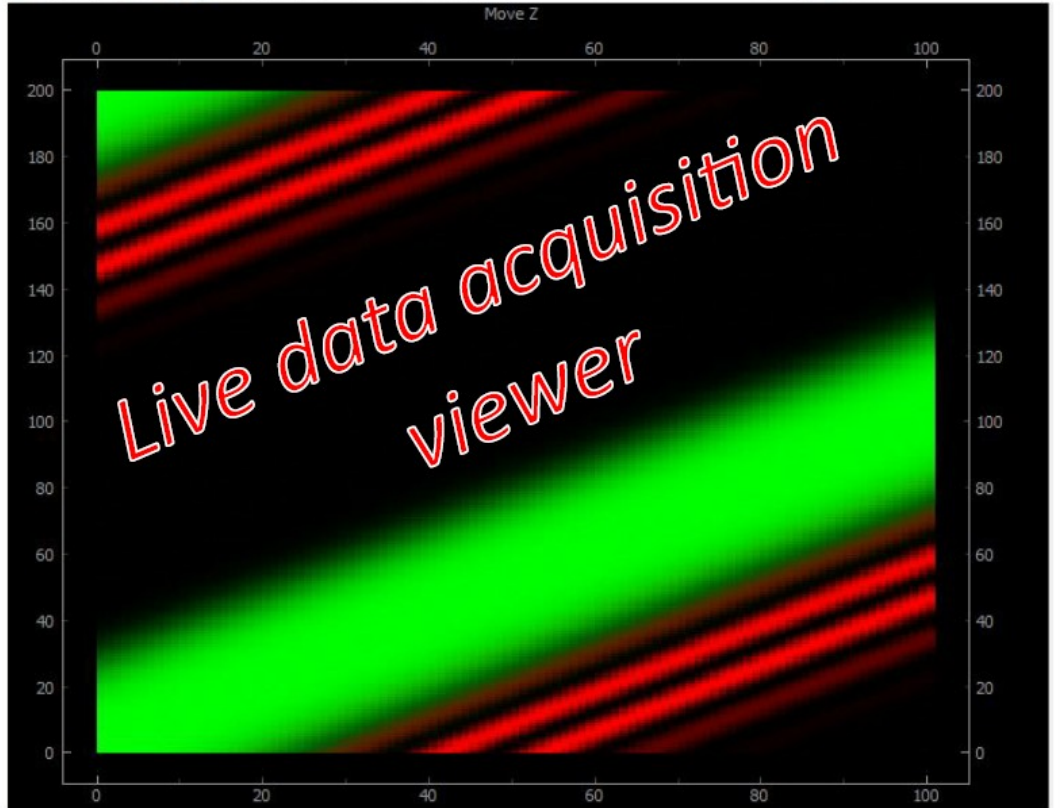
Save Settings

Scanner Settings

Parameter	Value
Scanner settings	
calculate_positions	
N steps:	11
Scan type:	Scan1D
Scan1D settings	
Scan type:	Linear
Start:	-2
stop:	3
Step:	0.5
load_xml	
save_xml	

1D plot 2D plot

Move Z



Initializing 11 0

Module manager

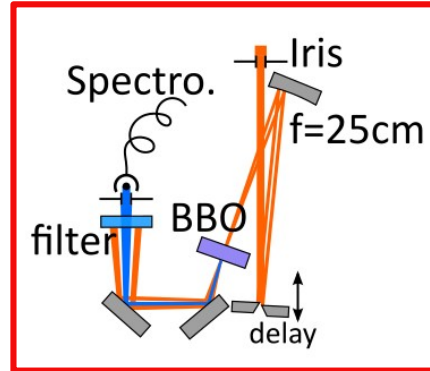
Scanner UI

Live data acquisition viewer

DAQScan Extension: Acquisition

Selection of 1 detector and 1 actuator

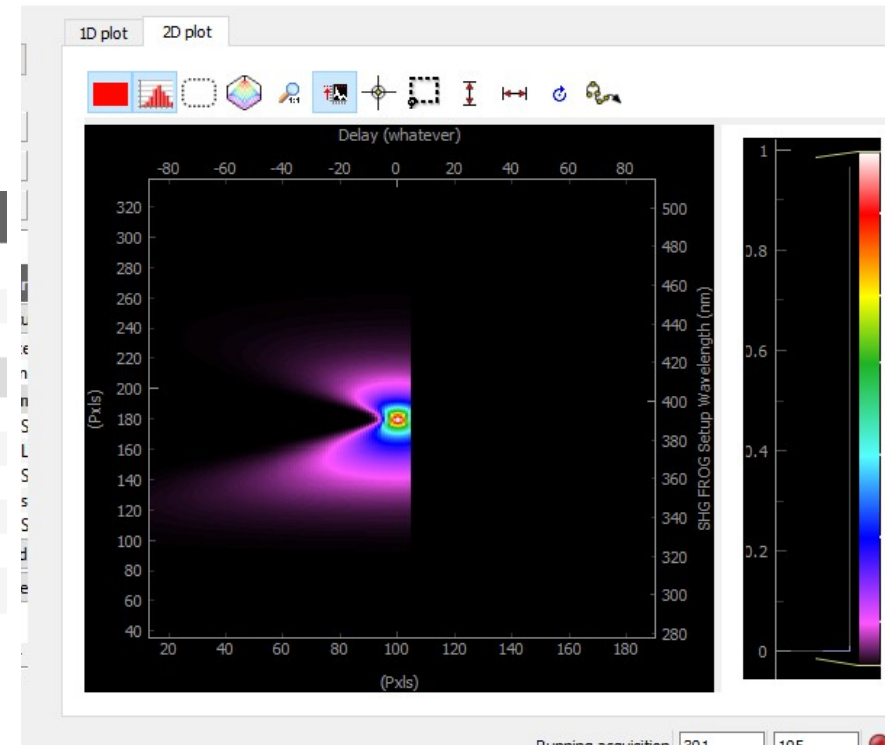
Parameter	Value
Actuators/Detectors Selection	
detectors	
<input type="checkbox"/> DSCAN spectrometer <input checked="" type="checkbox"/> SHG FROG Setup	
Actuators	
<input checked="" type="checkbox"/> Delay <input type="checkbox"/> Glass Insertion <input type="checkbox"/> Wedge Translator	
Moves done?	<input type="checkbox"/>
Detections done?	<input type="checkbox"/>
> Data dimensions	
> Actuators positions	



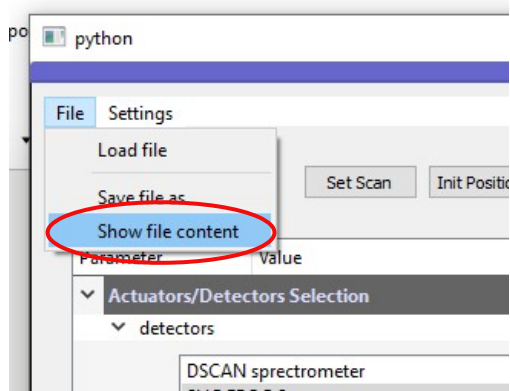
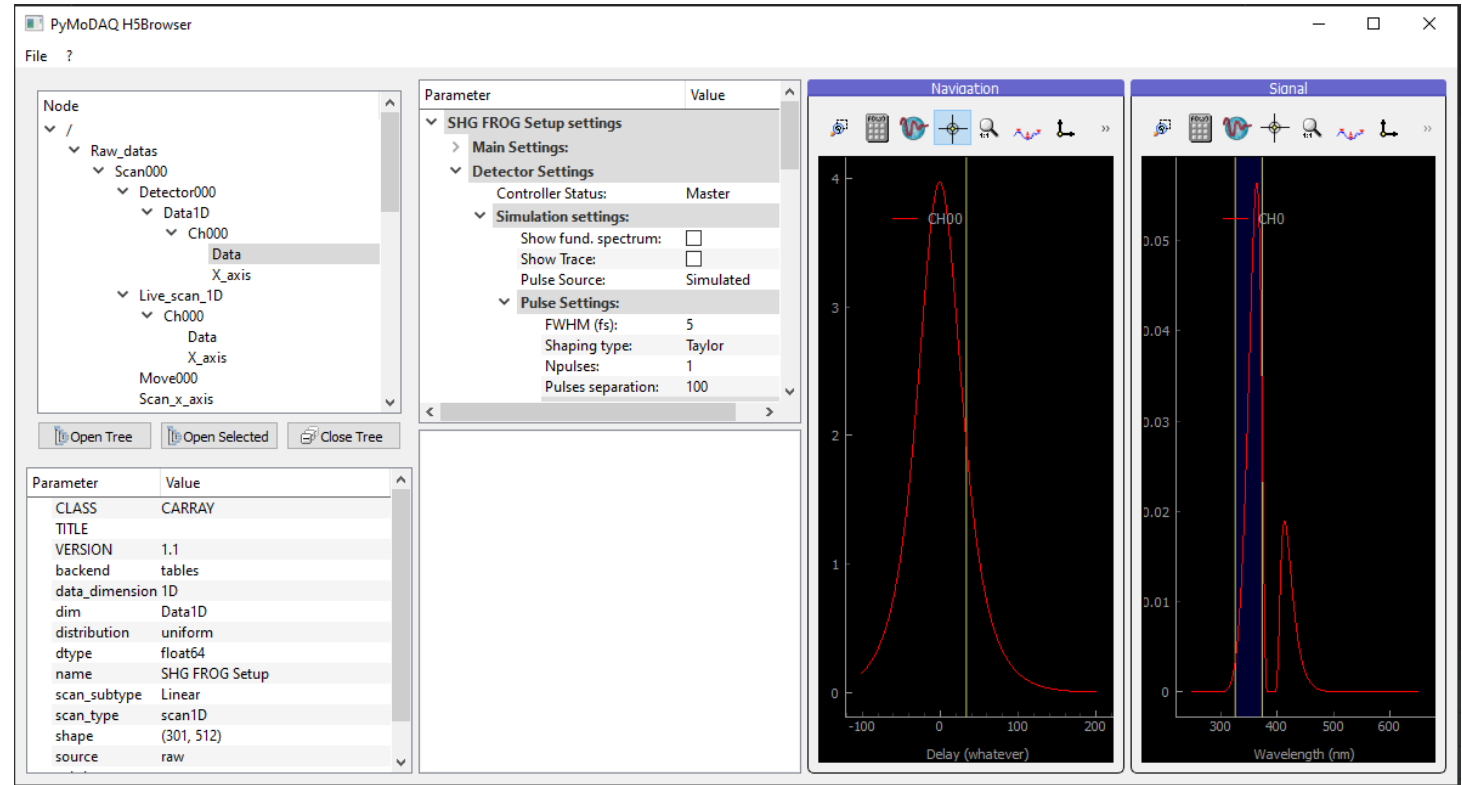
Scanner settings	
Calculate positions:	s:
N steps:	301
Scan type:	Scan1D
Scan1D settings	
Scan subtype:	Linear
Loss type	
Start:	-100
stop:	200
Step:	1
Load settings	
Save settings	

Setting up the scan

Checking and starting



DAQScan Extension: Browsing data

The PyMoDAQ H5Browser application window is shown. It features a tree view on the left, a parameter settings panel in the middle, and two plots on the right.

Tree View:

- Node
 - Raw_data
 - Scan000
 - Detector000
 - Data1D
 - Ch000
 - Data
 - X_axis
 - Live_scan_1D
 - Ch000
 - Data
 - X_axis
 - Move000
 - Scan_x_axis

Parameter Settings:

- SHG FROG Setup settings
 - Main Settings:
 - Detector Settings
 - Controller Status: Master
 - Simulation settings:
 - Show fund. spectrum:
 - Show Trace:
 - Pulse Source: Simulated
 - Pulse Settings:
 - FWHM (fs): 5
 - Shaping type: Taylor
 - Npulses: 1
 - Pulses separation: 100

Navigation Plot:

Y-axis: 0 to 4. X-axis: -100 to 200. A red curve shows a peak at 0. A vertical green line is at 0. A label 'CH00' is above the peak.

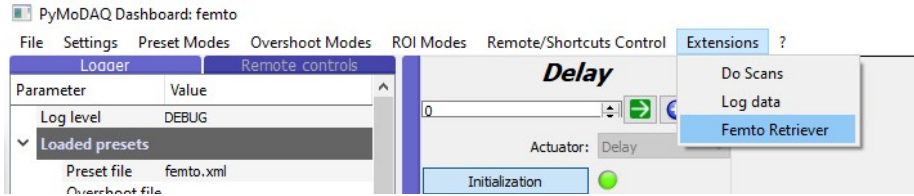
Signal Plot:

Y-axis: 0 to 0.05. X-axis: 300 to 600. A red curve shows a peak at approximately 400 nm. A vertical blue line is at approximately 380 nm. A label 'CH0' is above the peak.

Parameter Table:

Parameter	Value
CLASS	CARRAY
TITLE	
VERSION	1.1
backend	tables
data_dimension	1D
dim	Data1D
distribution	uniform
dtype	float64
name	SHG FROG Setup
scan_subtype	Linear
scan_type	scan1D
shape	(301, 512)
source	raw

Retriever Extension: analysing data



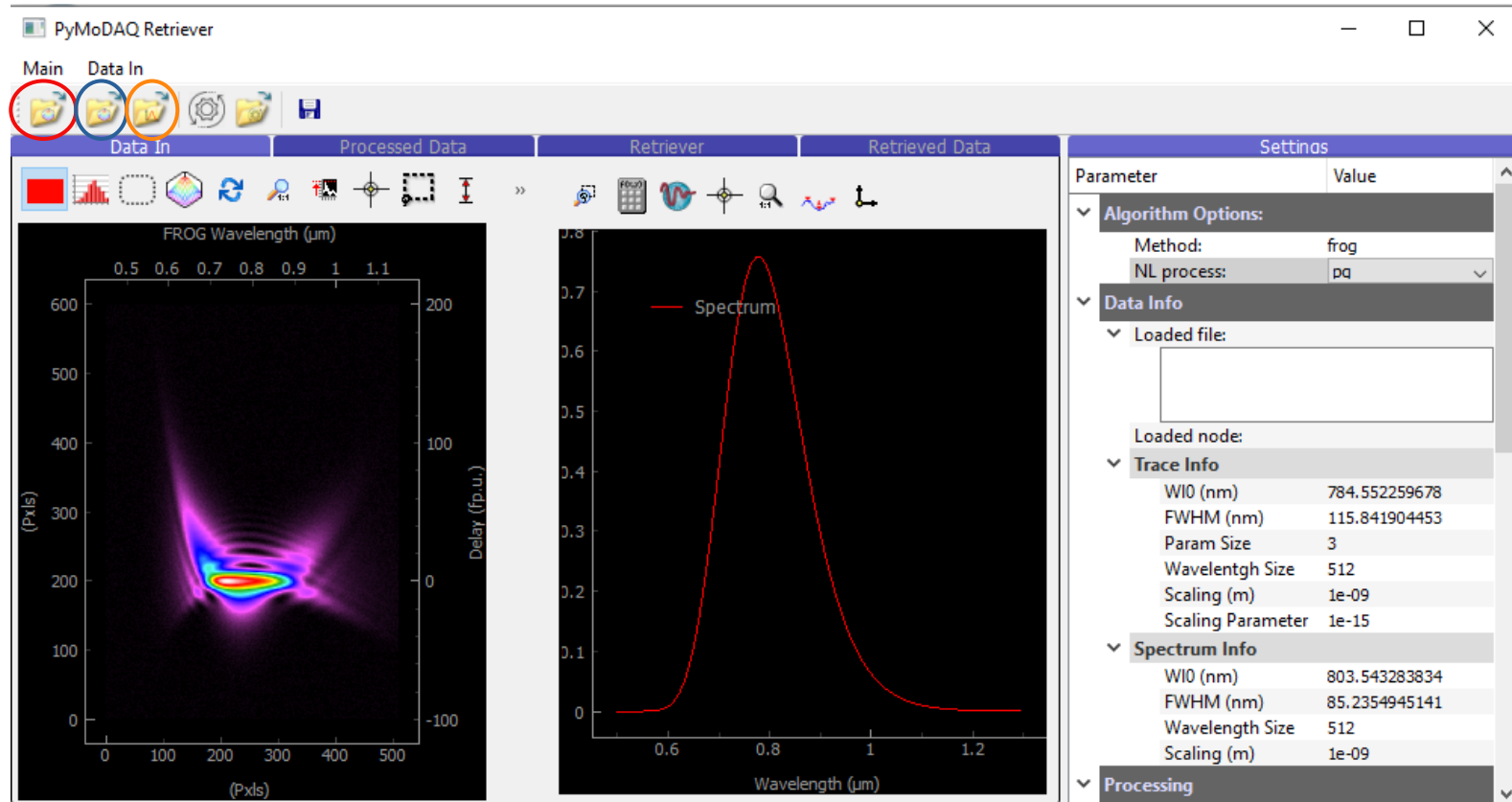
Load Trace from last Scan

Load Trace from file

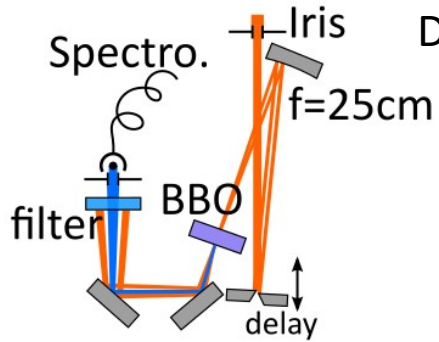
Load Spectrum from file

Practical (plot and export graphs):

- 1) 5fs SHG Frog with :
 - 50 fs² GDD
 - 500 fs³ TOD
 - Conclusion ?
- 2) 5fs PG FROG
 - 50fs² GDD
 - 500fs³ GDD
 - Conclusion
- 3) 5fs SHG Dscan with :
 - 50 fs² GDD
 - 500 fs³ TOD
 - Conclusion ?
 - Take care of scaling, algo,...



Retriever Extension: real actuators

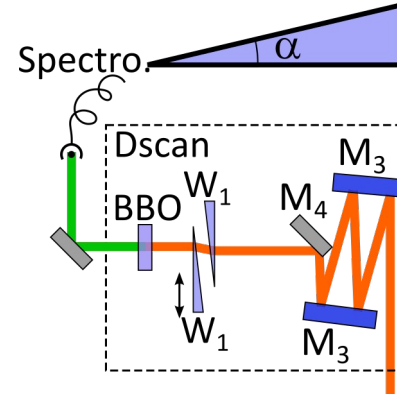


Delay from linear stage :
units in mm not fs ??

$$\Delta t = \frac{2 \Delta L}{c}$$

$$\Delta t_{fs} = \frac{210^4}{3} \Delta L_{mm}$$

$$\Delta t_{fs} \approx 6666.666 \Delta L_{mm}$$

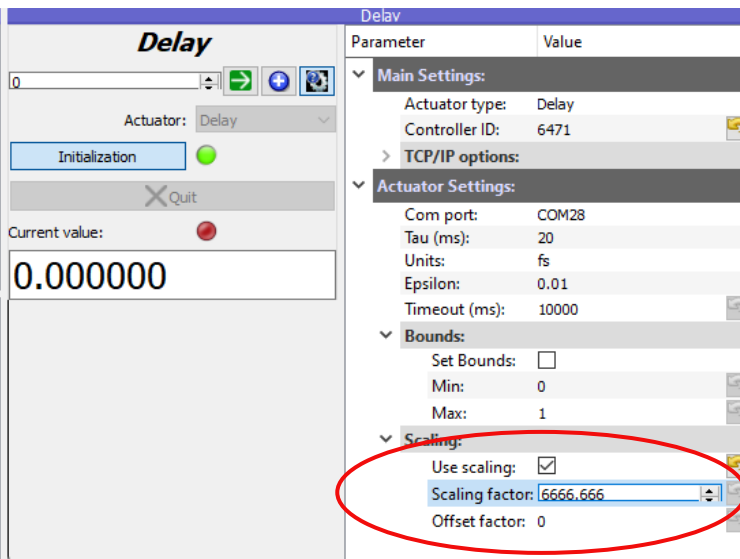


Insertion of wedge
not of glass ??

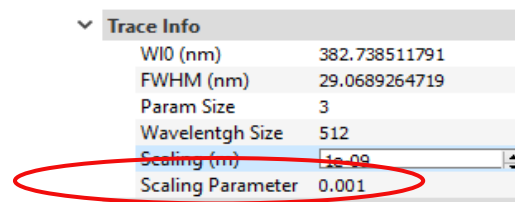
$$\tan(\alpha) = \frac{\Delta Ins}{\Delta L}$$

$$\alpha = 25^\circ$$

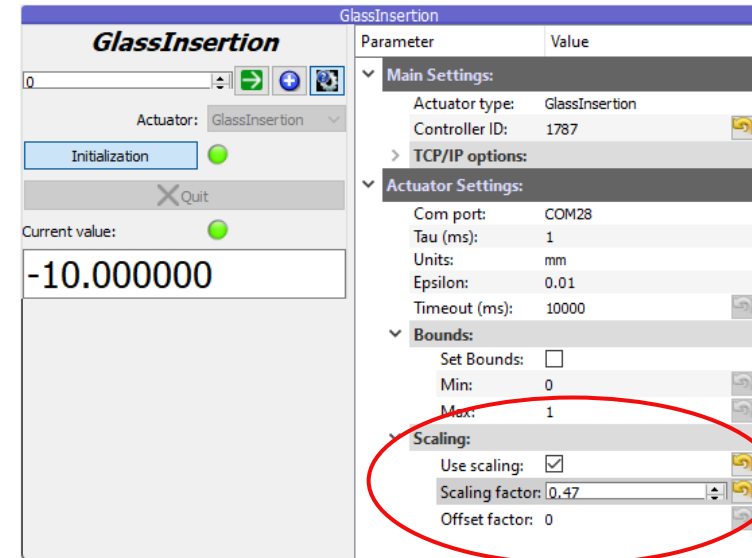
$$\Delta Ins = \Delta L \tan(\alpha) \approx 0.47 \Delta L$$



Parameter	Value
Actuator type:	Delay
Controller ID:	6471
Com port:	COM28
Tau (ms):	20
Units:	fs
Epsilon:	0.01
Timeout (ms):	10000
Min:	0
Max:	1
Scaling factor:	6666.666
Offset factor:	0



Parameter	Value
W10 (nm)	382.738511791
FWHM (nm)	29.0689264719
Param Size	3
Wavelength Size	512
Scaling (nm)	1e-09
Scaling Parameter	0.001



Parameter	Value
Actuator type:	GlassInsertion
Controller ID:	1787
Com port:	COM28
Tau (ms):	1
Units:	mm
Epsilon:	0.01
Timeout (ms):	10000
Min:	0
Max:	1
Scaling factor:	0.47
Offset factor:	0



Stay in touch and contribute

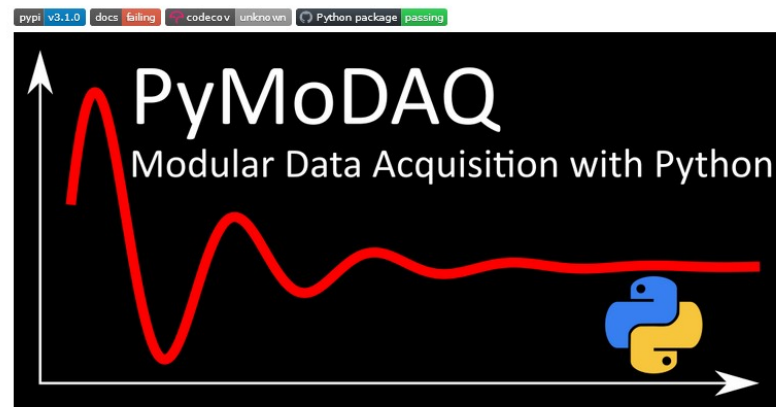
GitHub repository page for CEMES-CNRS / PyMoDAQ. The page shows the repository name, navigation tabs for Code, Issues (0), Pull requests (0), and Projects (0), and the repository description: Modular Data Acquisition with Python.

Modular Data Acquisition with Python

GitHub repository page for CEMES-CNRS / pymodaq_femto. The page shows the repository name, navigation tabs for Code, Issues, Pull requests, and Actions, and the repository description: PyMoDAQ Femto.

README.rst

PyMoDAQ



PyMoDAQ, Modular Data Acquisition with Python, is a set of `python` modules used to interface any kind of experiments. It simplifies the interaction with detector and actuator hardware to go straight to the data acquisition of interest.

PyMoDAQ Femto

PyMoDAQ extension for femtosecond laser pulse characterization

Published under the CeCILL-B FREE SOFTWARE LICENSE

GitHub repo: <https://github.com/CEMES-CNRS>

Documentation: <http://pymodaq.cnrs.fr/>