XAFSScanTool

Data Collection, XAFS/XRF Beamline and Control teams SESAME

Jul 09, 2023

CONTENTS:

This is a documentation for the scanning tool that collects experimental data from XAFS/XRF (X-ray Absorption Fine Structure/X-ray Fluorescence) spectroscopy beamline. The tool has been developed by data collection and analysis team of SESAME in collaboration with control team.

This documentation is targeting beamline end-users and scientists.

The scanning tool is GUI based that helps users to easily understand the scanning parameters and thus saving the time of explaning how to get XAFS-XRF experimental data.

CHAPTER

ONE

CURRENT FEATURES AND FUTURE PLANS

1.1 Current featues

At the moment, the scanning tool has the following features:

- User friendly GUI
- Collect data from detectors simultaneously (currently from three detectors ICs, KETEK and FICUS)
- Ability to calibrate the DCM energy based on the choosen metal foil
- Moves the DCM energy in eV or K units
- Runs as “unattended scanning mode”
- The tool is smart enough to **pause** in case of problems (e.g. current goes below certain limits, shutter is closed, problem in the vacuum, …etc)
- Runs as “unattended scanning mode”
- Enable data writing in xdi files
- Online data visualization
- Online / offline logging
- Automatic data transfer to data center (after each scan)
- Input data validation
- Public documentation targeting end-users (how-to)

1.2 Scanning tool | future plan

- Finalize automatic proposal and user metadata extraction from users DB
- Store the data in HDF5 dxFile format
- Develop a tool to convert HDF5 dxFile to xdi files
- Apply on-fly scanning mode (fast scan)

CHAPTER

TWO

INSTALLING XAFS/XRF SCANNING TOOL

This page includes information about the needed packages to run the scan tool.

2.1 Prerequisites

The following should be installed on the computer before running the scanning tool:

- Linux redhat based OS (This work has been done under CentOS 7.4, however, there should be no reason to not work on other distributions)
- EPICS XAFS/XRF IOCs (motion and scan IOCs)
- Python 3.9
- QT 4.1.0 based on 5.9.7.

2.2 Python virtual environment

venv module of Python is being used as a virtual environment for this setup.

The venv module of python provides support for creating **virtual environments** that is isolated from system site directories. Normally, each virtual environment has its own Python binary (which matches the version of the binary that was used to create this environment) and can have its own independent set of installed Python packages in its site directories.

to install and create venv:

```
$ pip3.9 install virtualenv
$ python3.9 -m venv /opt/DAQ/venv
```

to create alias of you environment:

```
$ vi ~/.bashrc
```

add the following line to the file:

```
alias p3='source /opt/DAQ/venv/bin/activate'
```

resource your bashrc:

```
source ~/.bashrc
```

2.3 Packages and libraries

The tool needs set of python packages and Qt libraries installed and configured.

2.3.1 Pyhon packages:

The list below contains the list of python packages needed for the scanning tool to run. After activating the python virtual environment (by typing **p3** in the terminal), you can use **pip** to install them in the virtual environment or you can copy this list in a text file (requirements.txt) and install them at once using this command (pip install -r requirements.txt)

```
backports.entry-points-selectable==1.1.0
bcrypt==3.2.0
cffi==1.15.0
colorama==0.4.4
cryptography==35.0.0
cycler==0.10.0
distlib==0.3.2
filelock==3.0.12
h5py==2.10.0
importlib-metadata==4.8.1
importlib-resources==5.2.2
kiwisolver==1.2.0
matplotlib==3.2.1
numpy==1.18.5
pandas==1.0.4
paramiko==2.8.0
Pillow==7.1.2
platformdirs==2.3.0
progressbar==2.5
pycparser==2.20
pyepics==3.4.1
PyNaCl==1.4.0
pyparsing==2.4.7
PyQt5==5.12
PyQt5_sip==4.19.19
pyqtgraph==0.10.0
python-dateutil==2.8.1
pytz==2020.1
scipy==1.4.1
six==1.15.0
typing-extensions==3.10.0.2
virtualenv==20.7.2
zipp==3.5.0
```

2.3.2 Qt and its libraries:

1. Install epics from SESAME's local repo.
2. Download Qt creator: https://drive.sesame.org.jo/owncloud/index.php/s/LO3GLyDkPMWZKU9.
3. Install qt-creator-opensource-linux-x86_64-4.13.3.run.
4. Install epics-qt, qt5, qwt, or anything related to *qt* packages by `yum` command.
5. Go to `.bashrc` and copy the following:

```
export EPICS_BASE='/opt/epics/base'
export EPICS_HOST_ARCH=linux-x86_64
export PATH=${PATH}:/opt/qtcreator-4.13.3/bin/
export QWT_ROOT=/usr/local/qwt-6.1.3
export QWT_INCLUDE_PATH=${QWT_ROOT}/include
export QE_TARGET_DIR=/usr/local/epics-qt
export PATH=${EPICS_BASE}/bin/$EPICS_HOST_ARCH:${QE_TARGET_DIR}/bin/${EPICS_
↪HOST_ARCH}:/usr/lib64/qt5/bin:${PATH}
export LD_LIBRARY_PATH=${EPICS_BASE}/lib/${EPICS_HOST_ARCH}:/usr/local/qwt-6.1.
↪3/lib:${QE_TARGET_DIR}/lib/${EPICS_HOST_ARCH}:${QE_TARGET_DIR}/lib/${EPICS_
↪HOST_ARCH}/designer
export QT_PLUGIN_PATH=${QT_PLUGIN_PATH}:${QWT_ROOT}/plugins:$QE_TARGET_DIR/lib/
↪$EPICS_HOST_ARCH
```

6. `source .bashrc`
7. To validate your setup, create a new project and open the designer, you should get qwt and epics qt widgets shown.

2.4 Clone and run the scanning tool

Note: Make sure that the python environment is activated before proceeding with this section

The scanning tool (XAFSXRFScanTool) is available on github. The most recent version can be found on this link: https://github.com/SESAME-Synchrotron/XAFSScanTool.git. To clone and run, launch your terminal then do the follwoing:

```
$ cd ~
$ git git@github.com:SESAME-Synchrotron/XAFSScanTool.git
$ cd XAFSScanTool
$ python main.py --testingMode yes
```

Warning: If all is fine, you should see the GUI pops up, otherwise, error messages and alerts should be shown in the terminal.

CHAPTER

THREE

RUN EXPERIMENT AND COLLECT DATA

Note: All beamline EPICS IOCs should be up and running before using the scanning tool.

Note: In order to run the scanning tool, you need to activate python environment that you have already setup.

The scanning tool home directory is located in the home directory of the control user. To access it:

```
$ cd ~
$ cd XAFSScanTool
```

to run the scanning tool:

```
$ python main.py
```

the main function will validate and execute some procedures and functions, if all is fine GUI will appear:

Warning: if a PV is disconnected, the scanning tool will show such PV in "red" color (instead of green as shown ubove), this will cause the tool to not run!!.

From the GUI above you can choose the experiment type:

A. Choose **Users Experiment** if there is a scheduled beamtime for an accepted proposal.

Upon choosing Users Experiment, you will be asked to provide scheduled proposal ID as shown in Figure 2 below:

By choosing "Users Experiment", the scan tool will:

- Ask you to provide the proposal number.
- Validate whether the provided proposal number is correct and valid for this beam time.
- If the validation result is okay, the tool will import the proposal metadata and include them in the experimental file. If not, user will be alerted and the tool will not be able to continue!!

Note: The scanning tool is already integrated with the users database. All validation and metadata importing processes are done through such integration. metadata of a validated proposal includes but not limited to proposal title, principal investigator information, number of allocated shifts, proposal review committee.

B. Choose **Local Experiment** to run in-house experiment that is not associated with a proposal.

This scan mode is intended to run "not proposal based" experiments, example of such experiments:

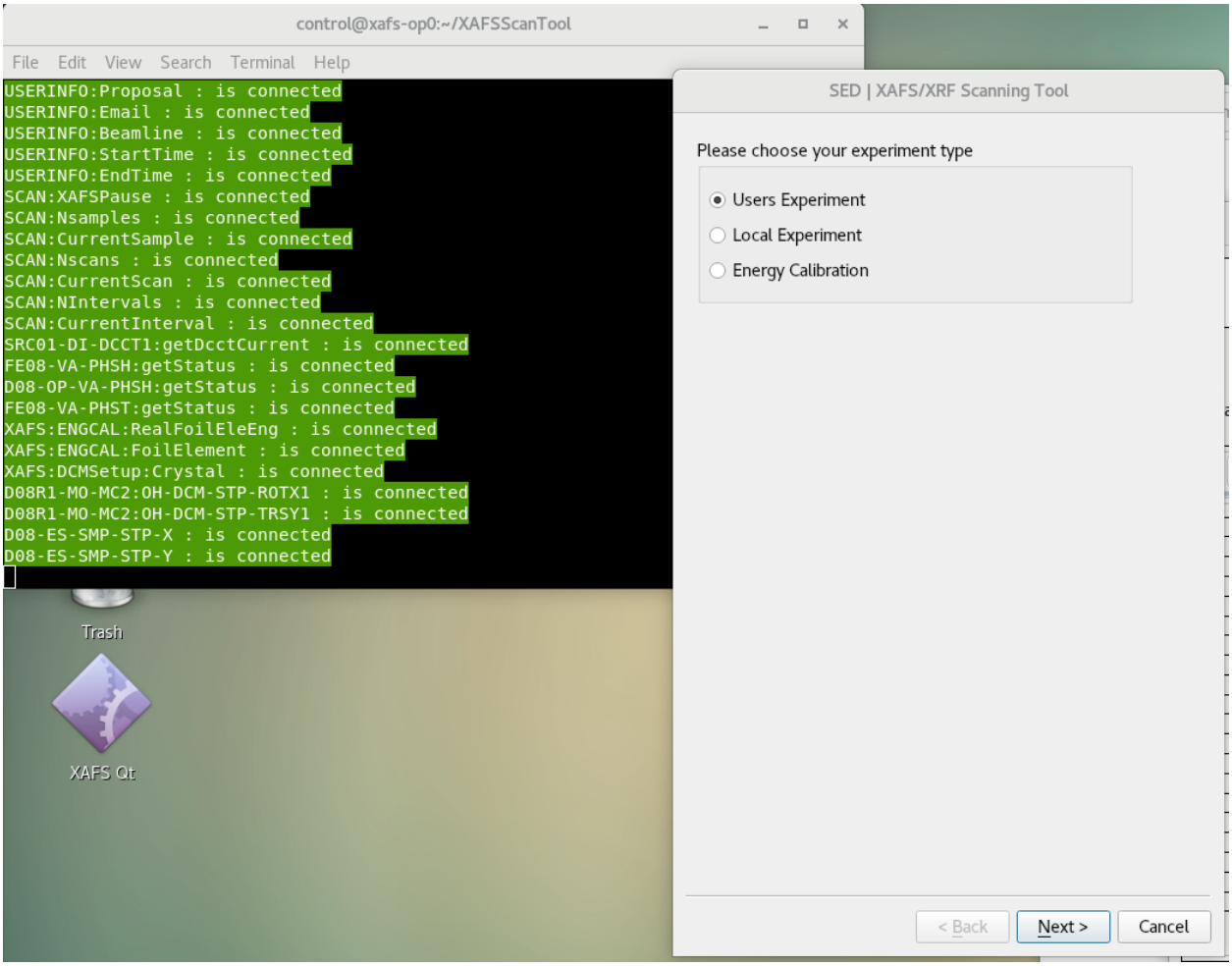

Fig. 1: *Figure 1: First popup GUI that allows you to choose experiment type*

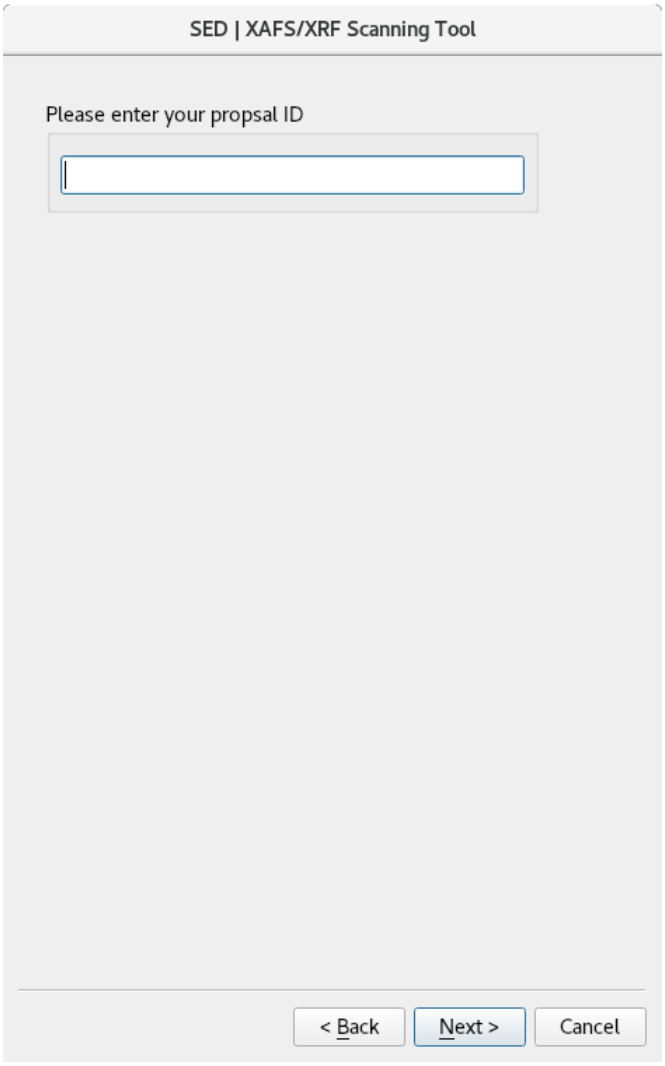

Fig. 2: *Figure 2: proposal ID*

- Director general beamtime
- In-house research experiment
- Testing / commissioning new components at the Beamline

Warning: Normally, this option is restricted to beamline scientists.

Warning: Access to experimental data generated out of this kind of experiment is restricted to beamline scientists and authorized SESAME staff only. On the other hand, the generated data will not be **mapped/linked** with any proposal or PI work.

C. Choose **Energy Calibration** to calibrate the beam energy in reference to the DCM crystal and metal foil you are using.

The beam energy is linked to the monochromator via the Bragg formula, calibrating the energy means adjusting the Bragg angle (theta) of the monochromator in reference to the crystal and the metal foil you are using.

Currently, the following crystals are available:

- Si(111)
- Si(311)

Also, the following metal foils:

1. Ti (4966)
2. V (5465))
3. Cr (5989)
4. Mn (6539)
5. Fe (7112)
6. Co (7709)
7. Ni (8333)
8. Cu (8979)
9. Zn (9659)
10. Se (12658)
11. Zr (17998)
12. Nb (18986)
13. Mo (20000)
14. Pd (24350)
15. Ag (25514)
16. Sn (29200)
17. Sb (30491)
18. Ta (9881)
19. Pt (11564)
20. Au (11919)

21. Pb (13035)

The scanning tool allows you to either enter new configuration and thus generate a new configuration file or load an already existed configuration file. These two options can be chosen from this GUI:

Next GUI is meant to enter new experiment configurations or see/edit a loaded one. This GUI allows you to move the energy over a range by driving the theta motor of the Double Crystal Monochromator (DCM).

The user can enter many intervals, each interval has start energy(eV), end energy(eV), energy move step size, Ionization Chamber (IC) integration time, fluorescence detector integration time, external trigger and step unit.

The step unit can be either in eV or K. When eV is chosen, the "step" is used as energy incerment value across the interval starting from "start" until reaching the "end" energies. By choosing K as step unit, the energy increase size (step size) increases as the scan moves further above the edge.

Note: "The XAFS region is most naturally thought of as a function of k. Because E is proportional to the square of K, features will tend to broaden and reduce in amplitude as getting further above the edge. In addition, the signal falls off with increasing energy, further reducing the amplitude of features high above the edge."" reference: XAFS for every one, page 161, point# 3

The equations of calulating DCM energy with K step unit are shown below:

Where K is energy step size in K, E_a is the current DCM energy in K, E_c is the calibrated energy in K and E_n is the next energy value that the DCM is going to.

You can define many samples and align them with respect to the beam (depending on the number of holders installed on the sample stage). Through this GUI you can change the sample position horizontally and vertically in order to target the right position of the sample. Also, for each sample you must assign name where it will be used as part of the experimental file name.

Note: sample name is added as part of the experimental file name

---

Detectors GUI allows you to choose among the available transmission and florescence detectors. ICs detectors are already chosen by default, you just need to enter the gas mixture that you use in each IC. For the fluorescence detectors, either FICUS or KETEK. For more information about the detectors, please see this page: https://www.sesame.org.jo/beamlines/xafs-xrf#tabs-7

Other scan parameters in the main confirmation GUI like "Experiment metadata", "Mirror coating" and "Comments" sub-boxes are used to provide some experimental meta data.

Note: Some experiment metadata fields are mandatory because they are needed to comply with xdi file format.

Fields that are highlighted in green (refer to Figure 4) are write protected when you run Users Experiment or Local Experiment (refer to Figure 1). This means that the DCM has been already calibrated and has got these values in which can't be changed for this kind of experiments.

However, to re-calibrate the DCM with different metal foil element and crystal you can choose Energy Calibration (refer to Figure 1), then, such fields are not "write protected" and you will see them highlighted in orange:

By clicking "Next", if all is fine, the last GUI will pop up as shown below:

Once scan is started, interactive logs will be printed on the terminal showing exactly what is being processed. Also, an interactive data visualization tool will start plotting the experimental data.

In case of energy calibration, two main GUIs will be popped up as soon as the calibration scan is finshed as below:

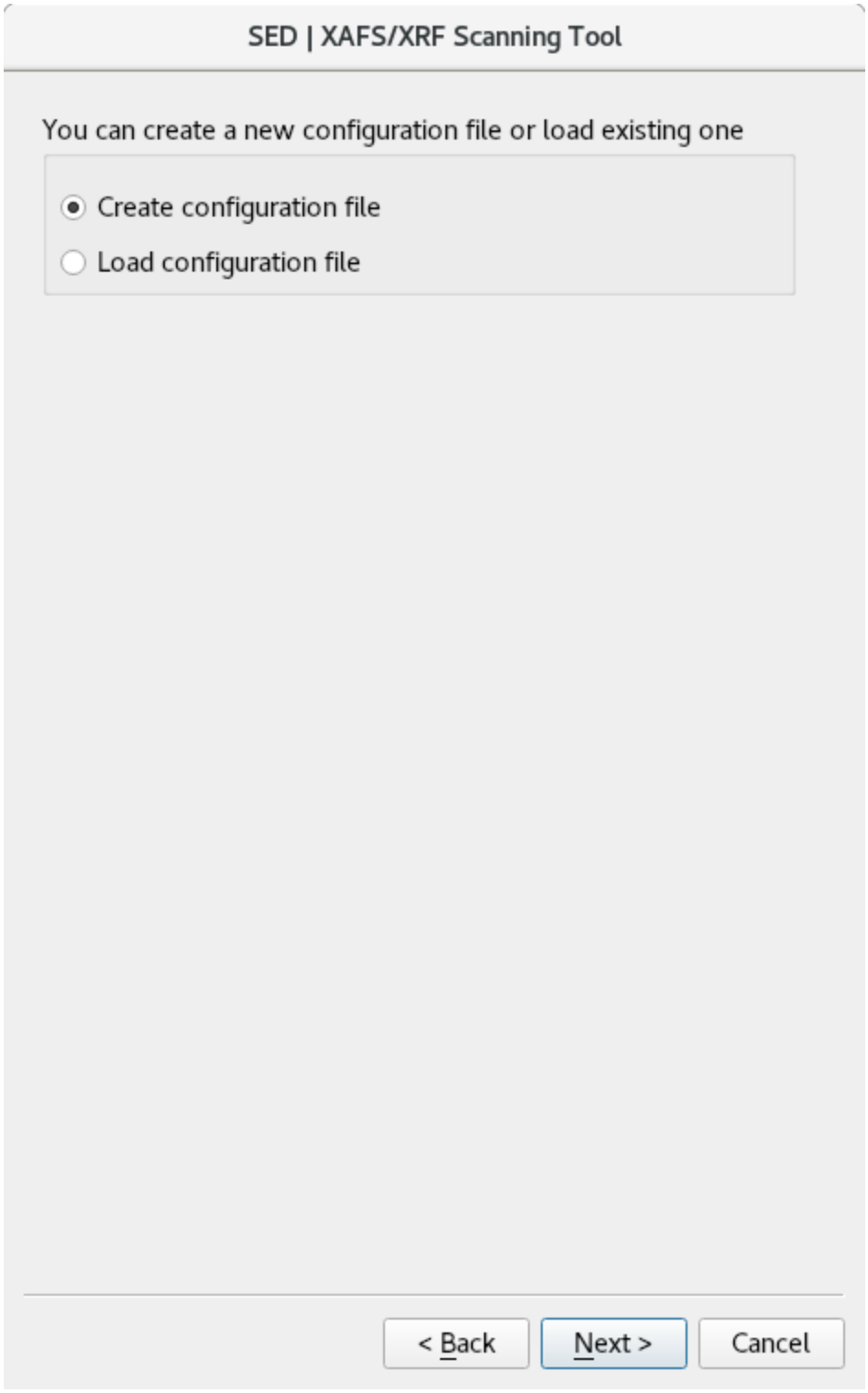

Fig. 3: *Figure 3: configration mode choosing GUI, either to create new config file or load already existed one*

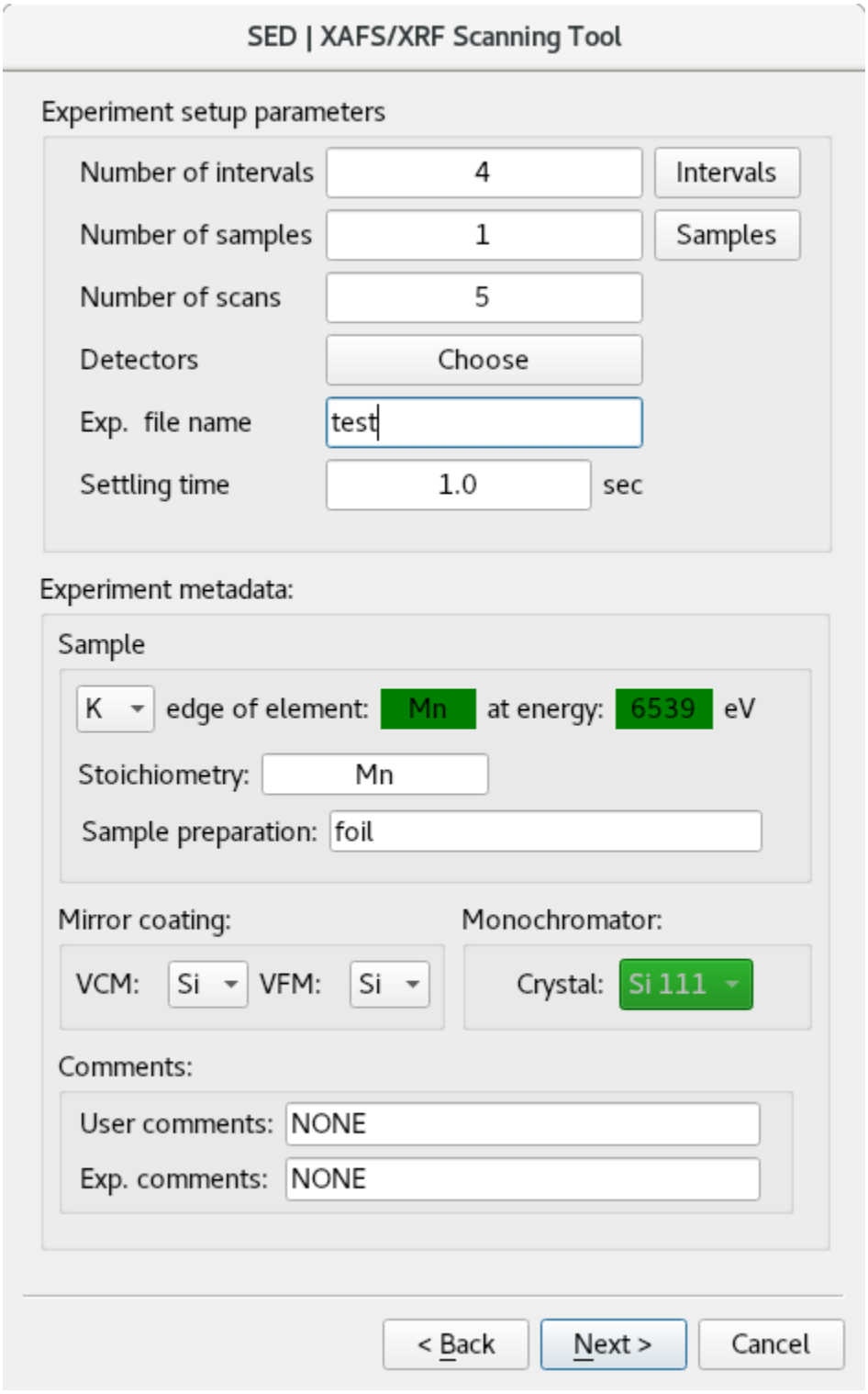

Fig. 4: *Figure 4: Main experiment configration GUI*

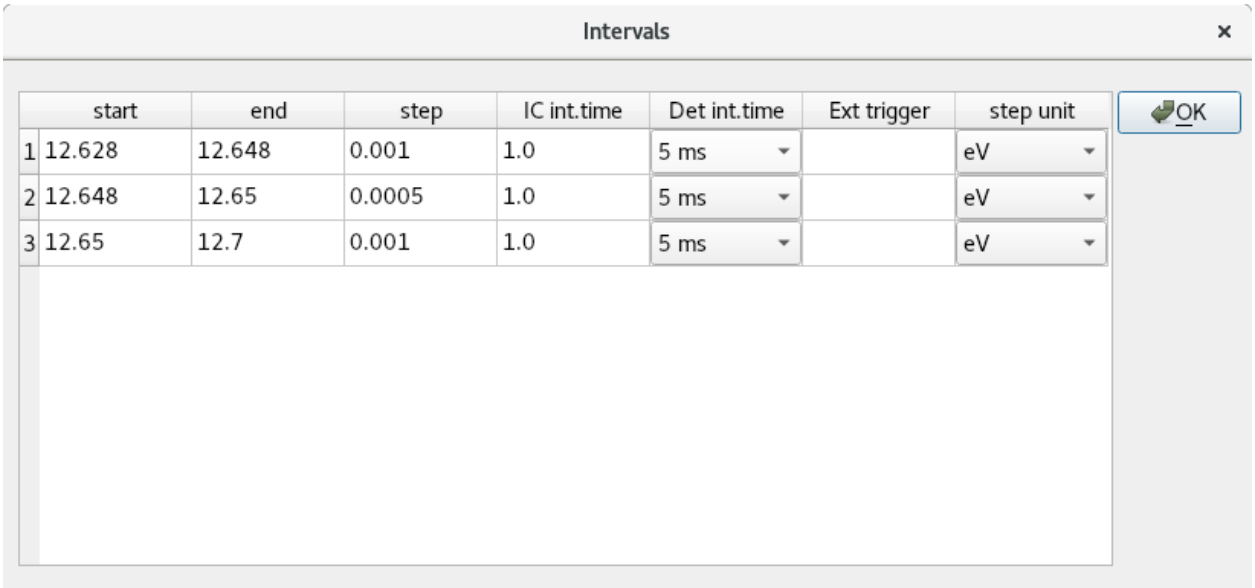

Fig. 5: *Figure 5: DCM energy equations with K step unit*

$$\Delta E = 4\Delta K\sqrt{E_a - E_c}$$
$$\mathrm{E}_n = E_a + \Delta E$$

Fig. 6: *Figure 6: DCM energy equations with K step unit*

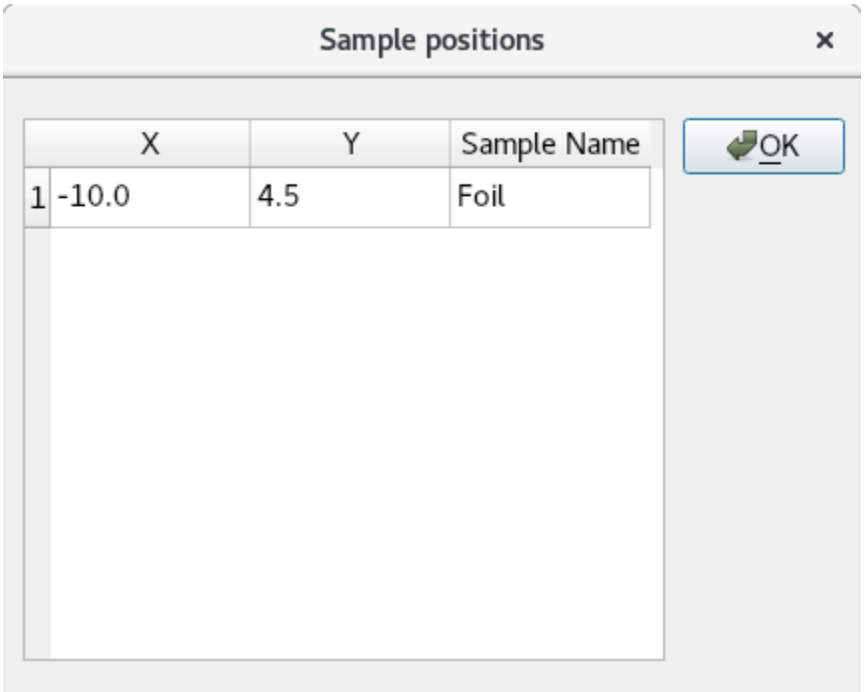

Fig. 7: *Figure 7: Sample position & name GUI*

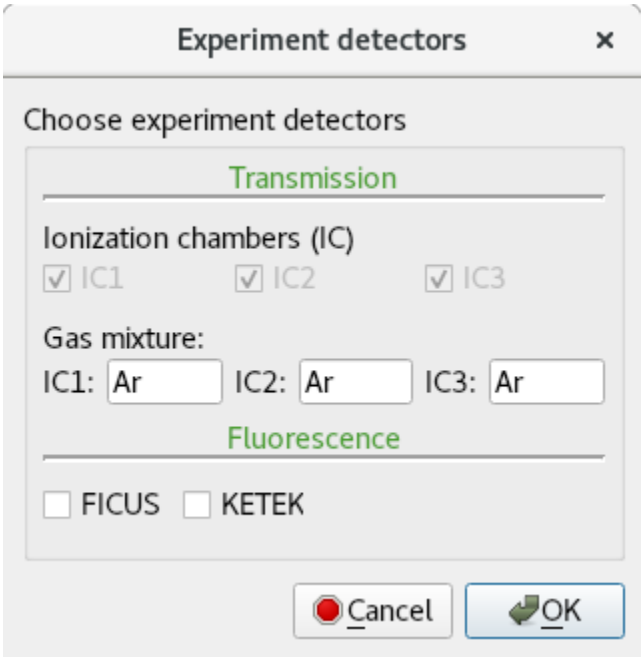

Fig. 8: *Figure 8: Detectors choosing GUI*

- **Main plots GUI as shown in Figure 12:**
  It contains the main analysis plots of the data imported from xdi data file:
  - Upper left corner: Normalization (Linear Scalling).
  - Upper right corner: Smoothing (Savitzky-Golay filter with default parameters (W,P):(5,3)).
  - Lower left corner: 1st derivative of normalized data.
  - Lower right corner: All of the uppove including 2nd derivative.
- **1st derivative plot GUI as shown in Figure 13:**
  This GUI allows user to select the best peak energy value either by selecting the blue dot, or by choosing any other value on the curve.

The main functions of this GUI are:

- Shows the 1st derivative of the normalized data (refer to Figure 13).
- Allows entering smoothing parameters, window length and polynomial order of Savitzky-Golay filter.
- Allows choosing the best peak energy value Interactively.
- Includes confirm button in order to confirm the chosen value and close the plots.

Note: According to smoothing parameters, please make sure that window length must be greater than ploynomial order, otherwise, an popup alert will be shown.

Once the peak value is chosen (either the blue dot, or any value on the curve), it will be printed on the terminal as shown in figure 14.

After clicking the confirm button, the results (the old offset, and the new calculated offset) will be printed on the terminal as shown in figure 15.

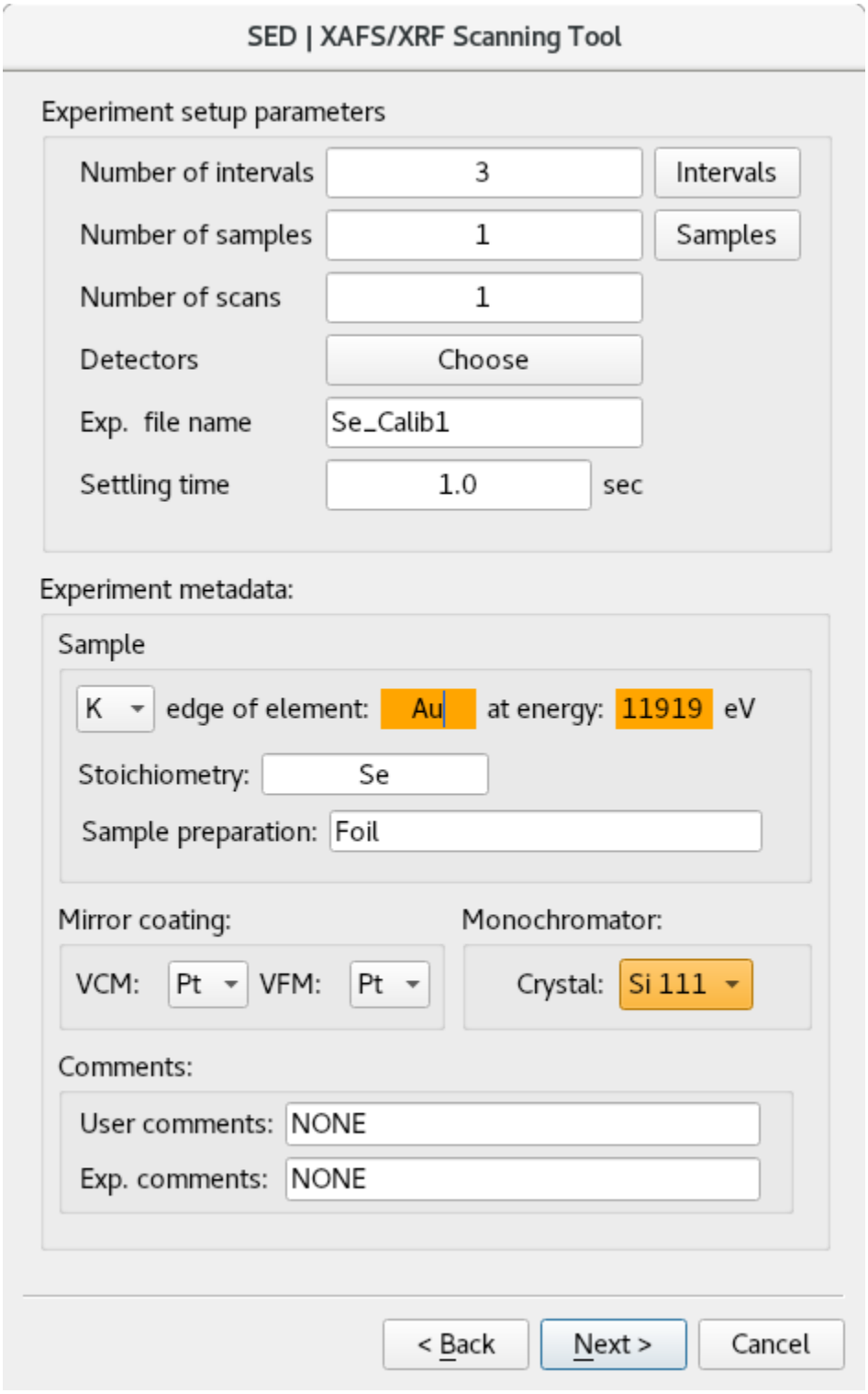

Fig. 9: *Figure 9: Main configration GUI that belongs to DCM energy calibration*

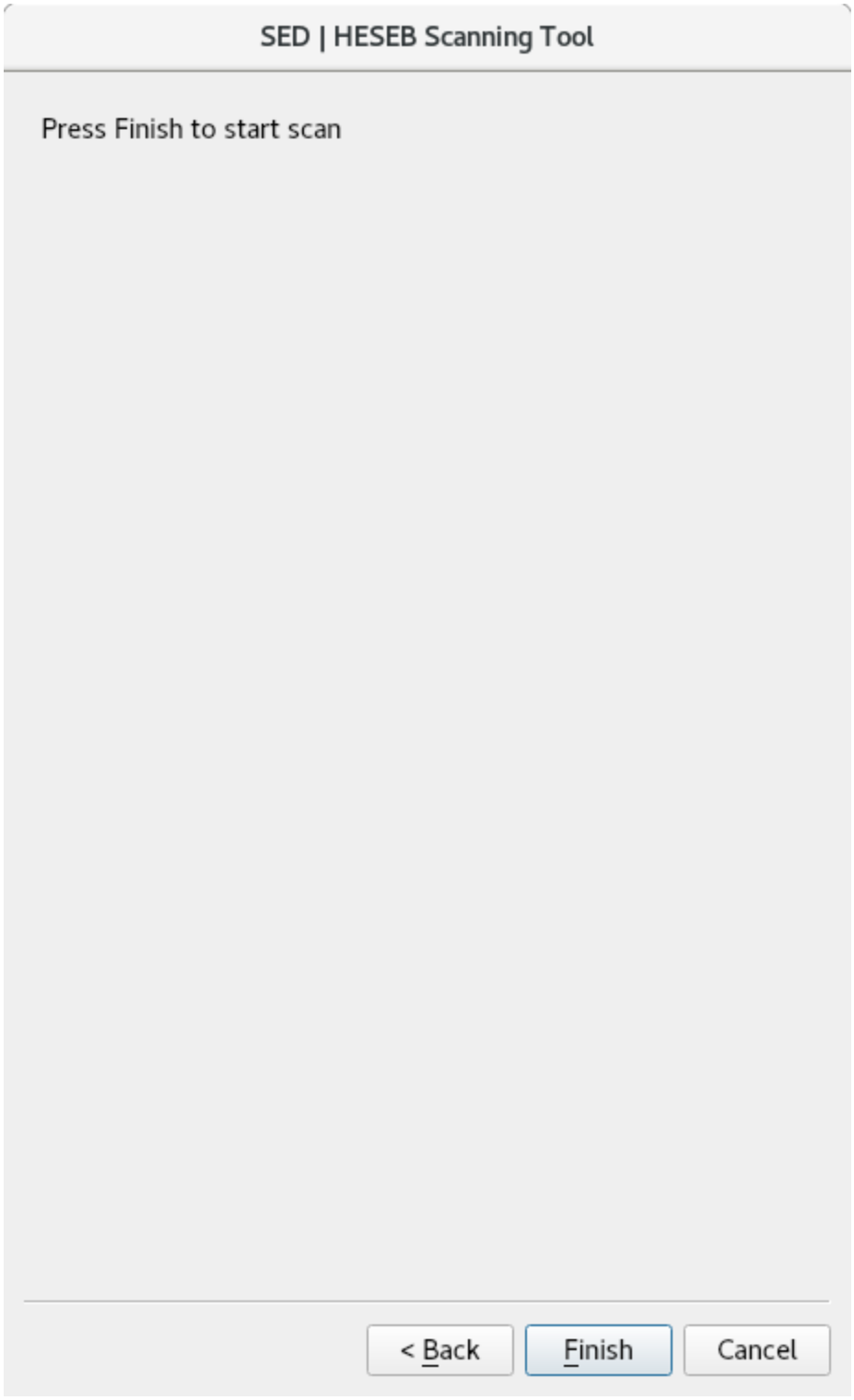

Fig. 10: *Figure 10: Last GUI before triggering the scan to start*

Fig. 11: *Figure 11: Interactive data visualization GUI*

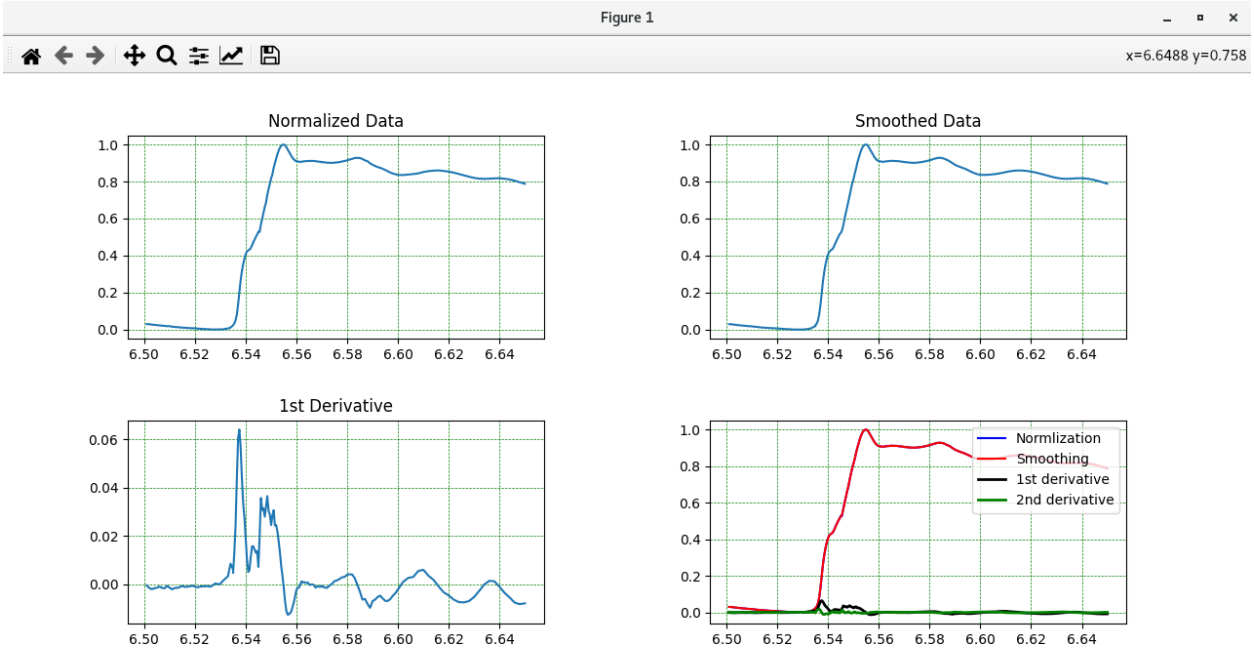

Fig. 12: *Figure 12: Main plots of xdi file*

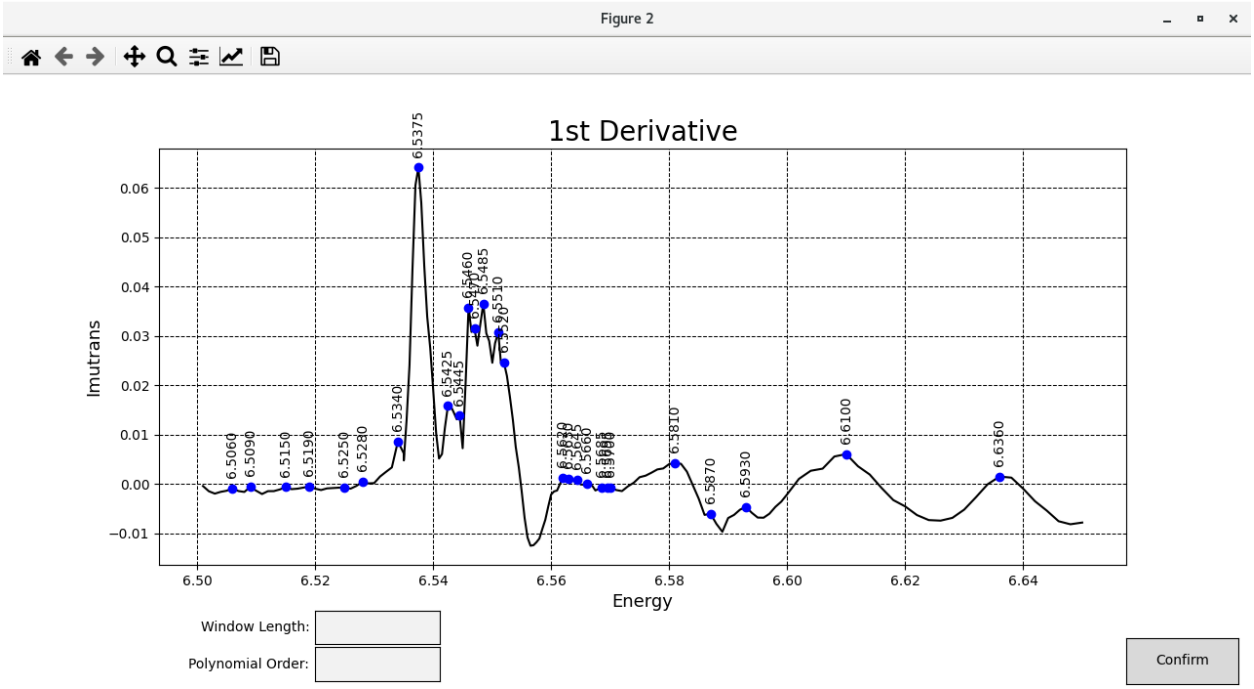

Fig. 13: *Figure 13: 1st derivative GUI of energy calibration*

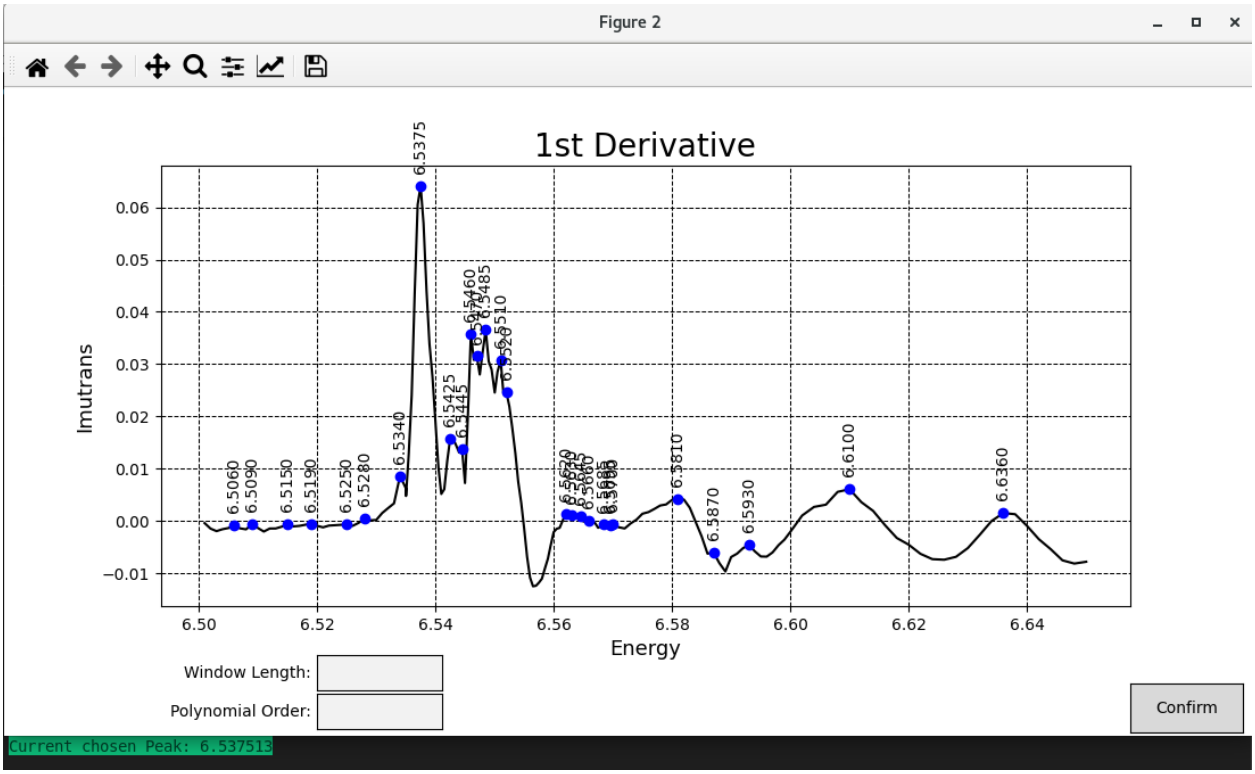

Fig. 14: *Figure 14: The chosen value for energy calibration*

```
Current chosen Peak: 6.537513
Crystal setup: Si 111
The old offset is: 0.025
The new offset is: -17.563543565972623
```

Fig. 15: *Figure 15: Energy calibration results*

Warning: If the *Confirm* button is clicked without choosing a value, an error messages will be shown in the terminal.

Note: To ignore the smoothing filter, both smoothing parameters in the GUI should be zeros.

Note: To repeat the energy calibration process without running a scan (or when you already have a calibration xdi file), type the following command in terminal:

```
python main.py --engCalib (xdi path)
e.g. python main.py --engCalib /home/XAFSScanTool/DATA/CalibTest_Foil_Scan1_
↪20220725T111439.xdi
```

CHAPTER

FOUR

EXPERIMENTAL DATA FILE LAYOUT AND FORMAT

The scanning tool of XAFS-XRF beamline generates experimental data in compliance with "SESAME Experimental Data Management Policy (SEDMP)" from the data acquisition point of view. The policy can be found here: https://www.sesame.org.jo/for-users/user-guide/sesame-experimental-data-management-policy.

One of the pillars of the SEDMP is to generate expiremntal data in stander and well defined data formats, thus at SESAME the stander experimental data files for XAS beamlines is XAS Data Interchange Format (XDI) version 1.0. XDI format is an open-source data format aims to standardize the XAS data format. All information about this format can be found on this page: https://github.com/XraySpectroscopy/XAS-Data-Interchange/blob/master/specification/spec.md

XDI is an ASCII file that can be opened using any text editor, also the file itself is self-descriptive. The figure below shows an example of XDI file layout:

Calib_Cr_foil_Scan1_20220811T091948.xdi

```
# XDI/1.0 SED_XAFS/0.9
# Column.1: energy eV
# Column.2: I0
# Column.3: Itrans
# Column.4: Irefer
# Column.5: mutrans (log(I0/Itrans))
# Column.6: murefer (log(Itrans/Irefer))
# Experiment.Type: Local
# Base.file_name: Calib_Cr
# Element.edge: K
# Mono.name: Si 111
# Mono.d_spacing: 3.1356
# Mono.settling_time: 1.0
# Beamline.name: XAFS/XRF Beamline (08-BM)
# Beamline.collimation: slits
# Beamline.focusing: no
# Beamline.harmonic_rejection: mirror coating VCM: Si, VFM: Si
# Facility.name: SESAME Synchrotron-light
# Facility.energy: 2.50 GeV
# Facility.current: 239.62618600000002
# Facility.xray_source: SESAME Bending Magnet
# Exp.start_time: 2022-08-11T09:19:46
# Scan.start_time: 2022-08-11T09:19:55
# Scan.end_time: 2022-08-11T09:29:12
# Scan.edge_energy: 5989
# Scan.number: 1/1 -- intervals: 3, samples: 1
# Detector.IC1: 15cm  HeN
# Detector.IC2: 30cm  ArN
# Detector.IC3: 15cm  Ar
# Element.symbol: Cr
# Sample.stoichiometry: Cr
# Sample.prep: foil
# ///
# Experiment comments and remarks: NONE
# User comments and remarks: NONE
#----
#(1)energy    (2)I0    (3)Itrans    (4)Irefer    (5)mutrans    (6)murefer
5.962001e+00  2.772691e+00  4.282301e+00  3.149920e-04  -4.346721e-01  9.517454e+00
5.964004e+00  2.718606e+00  4.210298e+00  1.000000e-16  -4.374139e-01  3.827889e+01
5.966013e+00  2.790605e+00  4.330777e+00  4.020080e-04  -4.394886e-01  9.284786e+00
5.968011e+00  2.790417e+00  4.352545e+00  2.394192e-04  -4.445697e-01  9.808055e+00
5.970006e+00  2.799278e+00  4.382949e+00  1.807728e-04  -4.483603e-01  1.009599e+01
5.972010e+00  2.803093e+00  4.400111e+00  5.537440e-05  -4.509063e-01  1.128302e+01
5.974021e+00  2.804490e+00  4.411240e+00  1.814880e-04  -4.529341e-01  1.009848e+01
5.976012e+00  2.818224e+00  4.446074e+00  2.513392e-04  -4.559143e-01  9.780729e+00
5.978008e+00  2.821527e+00  4.464307e+00  2.177248e-04  -4.588359e-01  9.928393e+00
5.980010e+00  2.828461e+00  4.483540e+00  3.183296e-04  -4.606800e-01  9.552836e+00
5.980014e+00  2.829389e+00  4.482672e+00  3.390704e-04  -4.601585e-01  9.489522e+00
5.980312e+00  2.829300e+00  4.485996e+00  2.279760e-04  -4.609313e-01  9.887231e+00
5.980602e+00  2.828923e+00  4.483893e+00  2.286912e-04  -4.605958e-01  9.883630e+00
5.980917e+00  2.831460e+00  4.489586e+00  1.000000e-16  -4.609680e-01  3.834312e+01
5.981211e+00  2.831502e+00  4.491502e+00  1.388144e-04  -4.613797e-01  1.038456e+01
```

Fig. 1: *Figure 1: example of xdi file*

CHAPTER

FIVE

COLLECT YOUR EXPERIMENTAL DATA

Currently, to take a copy of your data you can ask the beamline scientist to do so. On the other hand, we are working on global solution for accessing and collecting the users experimental data. The solutions are:

- SESAME Experimental Data Dispenser (SEDD): which an onsite service that allow users to login into specific SEDD computers using their SESAME Users' Portal (SUP) account and then the user's data will be automatically mounted on the user home directory making it available for copying and analysis.
- Remote / onsite web service (ICAT Project): The ICAT project provides a metadata catalogue and related components to support experimental data management for large-scale facilities, linking all aspects of the research lifecycle from proposal through to data and articles publication. More details can be found here: https://icatproject.org