

Python for User Interfaces at Sirius (THAPL04)

Douglas B. Beniz

(Brazilian Synchrotron)

douglas.beniz@lnls.br



ICALEPCS2017

Authors



Gabriel S. Fedel

gabriel.fedel@lnls.br



James R. Piton

james@lnls.br



Laís P. do Carmo

lais.carmo@lnls.br

→ EPICS

- ✓ Devices control via RS232/Ethernet;
- ✓ PVs (Process Variables) available on network, providing monitoring and “instant” operation;

→ Python

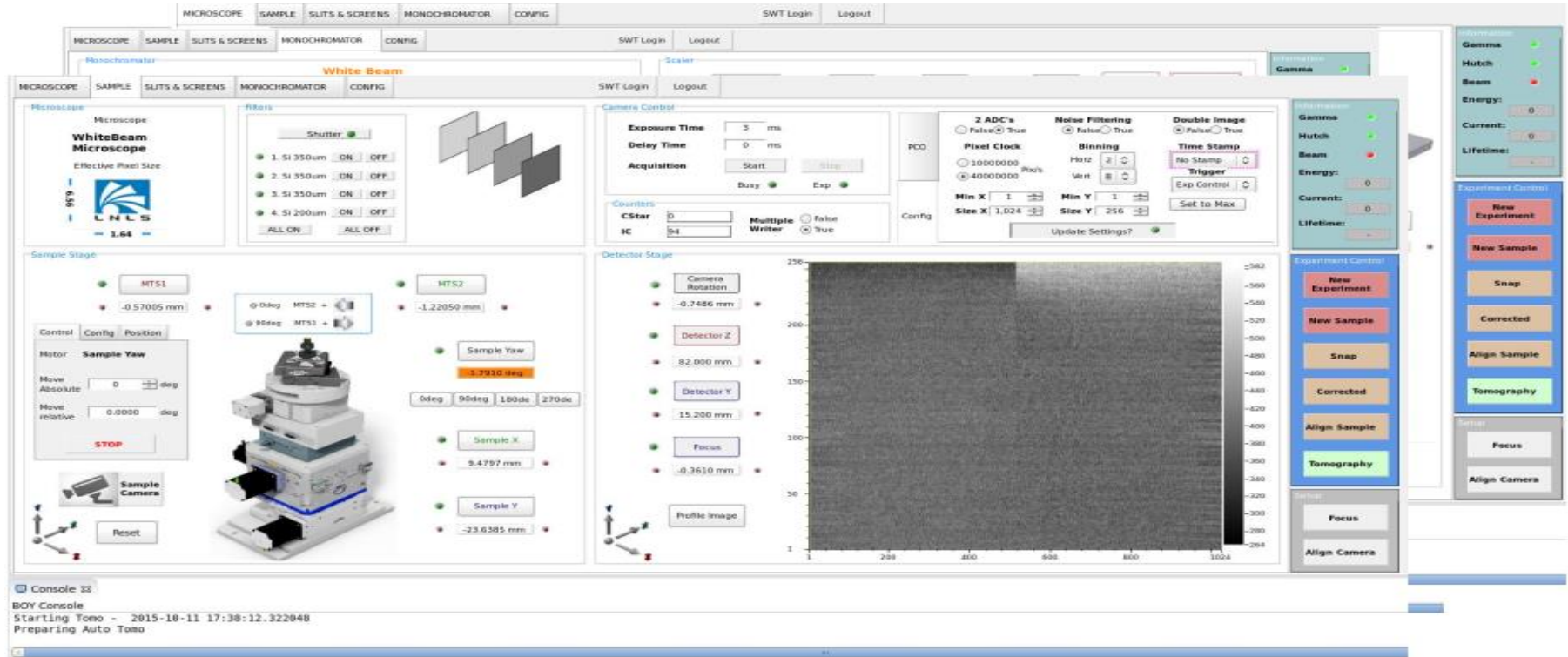
- ✓ Scripts to coordinate actions on beamlines with a set of devices synchronously;
- ✓ PyEPICS, Py4Syn*;

→ GUIs

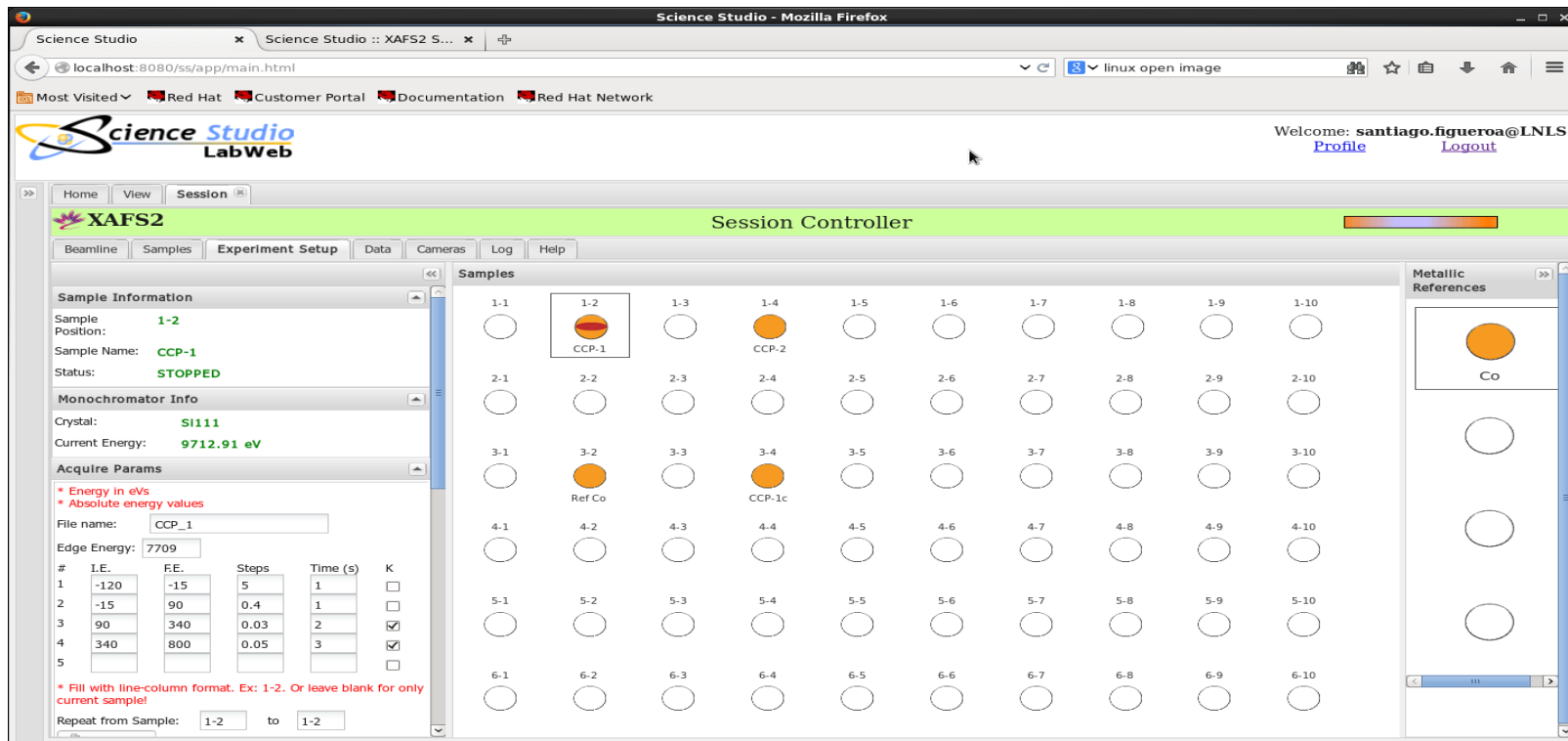
- ✓ Easy to operate actions (scripts);
- ✓ Current tools:
 - ◆ **Java based:**
 - CS-Studio (*Control System Studio*);
 - LabWeb (*Science Studio*);
 - ◆ **Python based:**
 - In-house developed TkInter solution;
 - MXCuBE and in-house solution, both based on PyQt;

*H. H. Slepicka et. al., 2015, “**Py4Syn: Python for synchrotrons**”. J. of Synchrotron Rad., V. 22, pp. 1182-1189.

CS-Studio (Control System Studio*)



*J.Hatje, M.Clausen et.al., "Control System Studio (CSS)", ICALEPCS 2007, Knoxville, MOPB03.



Science Studio - Mozilla Firefox

Science Studio :: XAFS2 S...

localhost:8080/ss/app/main.html

Most Visited Red Hat Customer Portal Documentation Red Hat Network

Welcome: [santiago.figueroa@LNLS](#)
[Profile](#) [Logout](#)

Home View Session

XAFS2 Session Controller

Beamline Samples Experiment Setup Data Cameras Log Help

Sample Information

Sample: 1-2
Position: CCP-1
Sample Name: CCP-1
Status: STOPPED

Monochromator Info

Crystal: Si111
Current Energy: 9712.91 eV

Acquire Params

* Energy in eV
* Absolute energy values

File name: CCP_1

Edge Energy: 7709

#	I.E.	F.E.	Steps	Time (s)	K
1	-120	-15	5	1	☐
2	-15	90	0.4	1	☐
3	90	340	0.03	2	☒
4	340	800	0.05	3	☒
5					☐

* Fill with line-column format. Ex: 1-2. Or leave blank for only current sample!

Repeat from Sample: 1-2 to 1-2

Samples

1-1 1-2 1-3 1-4 1-5 1-6 1-7 1-8 1-9 1-10

2-1 2-2 2-3 2-4 2-5 2-6 2-7 2-8 2-9 2-10

3-1 3-2 3-3 3-4 3-5 3-6 3-7 3-8 3-9 3-10

4-1 4-2 4-3 4-4 4-5 4-6 4-7 4-8 4-9 4-10

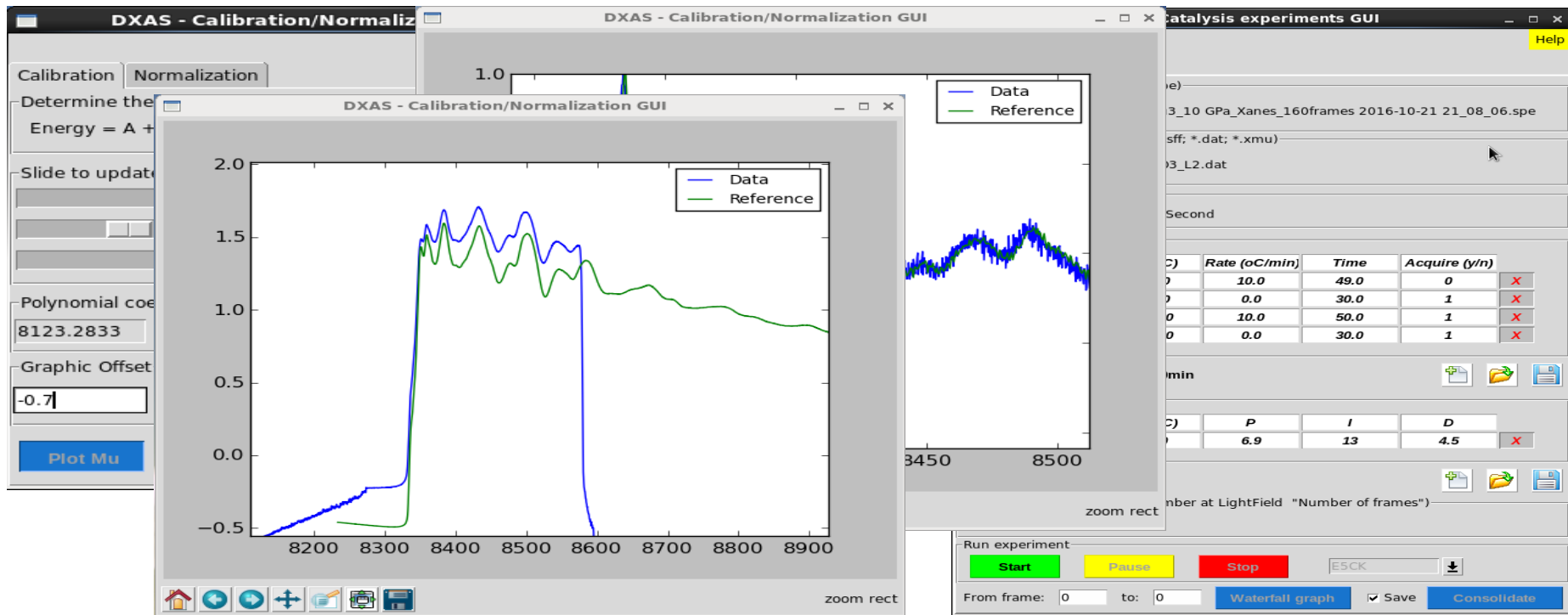
5-1 5-2 5-3 5-4 5-5 5-6 5-7 5-8 5-9 5-10

6-1 6-2 6-3 6-4 6-5 6-6 6-7 6-8 6-9 6-10

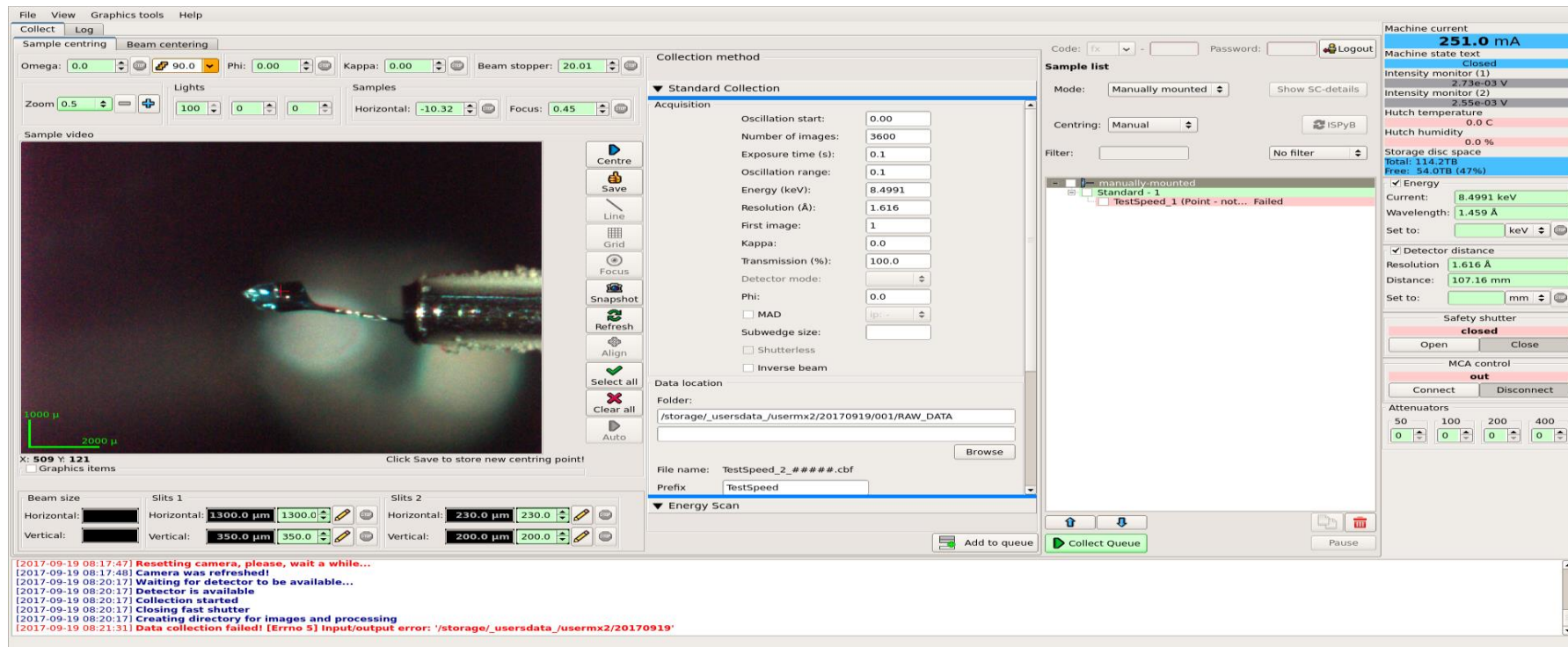
Metallic References

Co

*H. H. Slepicka et.al., "LabWeb – LNLS Beamlines Remote Operation System", ICALEPCS 2013, S. Francisco, TUPPC037.



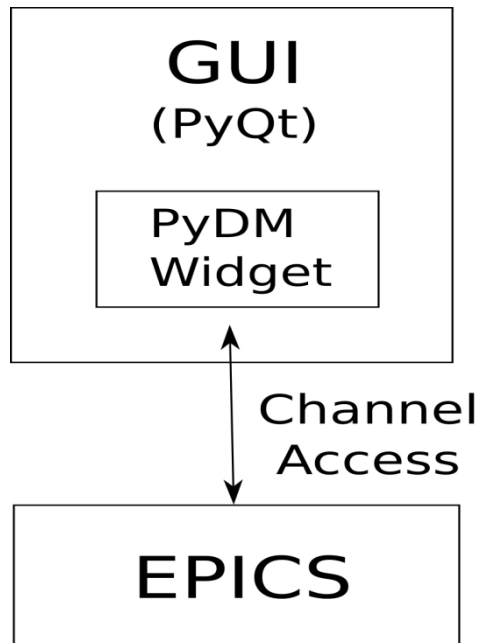
*D. B. Beniz et.al., "Using TkInter of Python to Create Graphical User Interface (GUI) for Scripts in LNLS", PCaPAC 2016, Campinas, WEPOPRPO25.



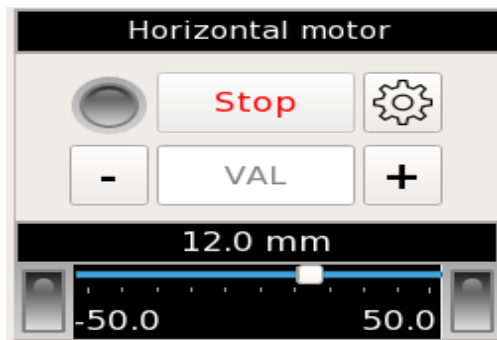
The screenshot displays the MXCuBE PyQt interface, a software environment for synchrotron beamline control. The interface is divided into several functional areas:

- Top Menu:** File, View, Graphics tools, Help.
- Sample centering / Beam centering:** Controls for Omega (0.0), Phi (0.00), Kappa (0.00), and Beam stopper (20.01). Includes a zoom control (0.5) and a sample video window showing a live feed of the sample.
- Lights / Samples:** Controls for lights (100, 0, 0) and sample parameters (Horizontal: -10.32, Focus: 0.45).
- Collection method:** A panel for configuring the acquisition method, including parameters like Oscillation start (0.00), Number of images (3600), Exposure time (0.1), Energy (8.4991), Resolution (1.616), and Transmission (100.0%).
- Sample list:** A list of samples, including 'Standard - 1' and 'TestSpeed_1 (Point - not... Failed)'. Includes a 'Collect Queue' button.
- Machine current:** A panel showing machine status, including current (251.0 mA), intensity monitors, and storage disc space.
- Detector distance:** Controls for resolution (1.616 Å), distance (107.16 mm), and safety shutter status (closed).
- MCA control:** Controls for MCA status (out) and attenuators (50, 100, 200, 400).
- Log window:** A bottom panel showing a log of system events, including camera resets, detector availability, and collection status.

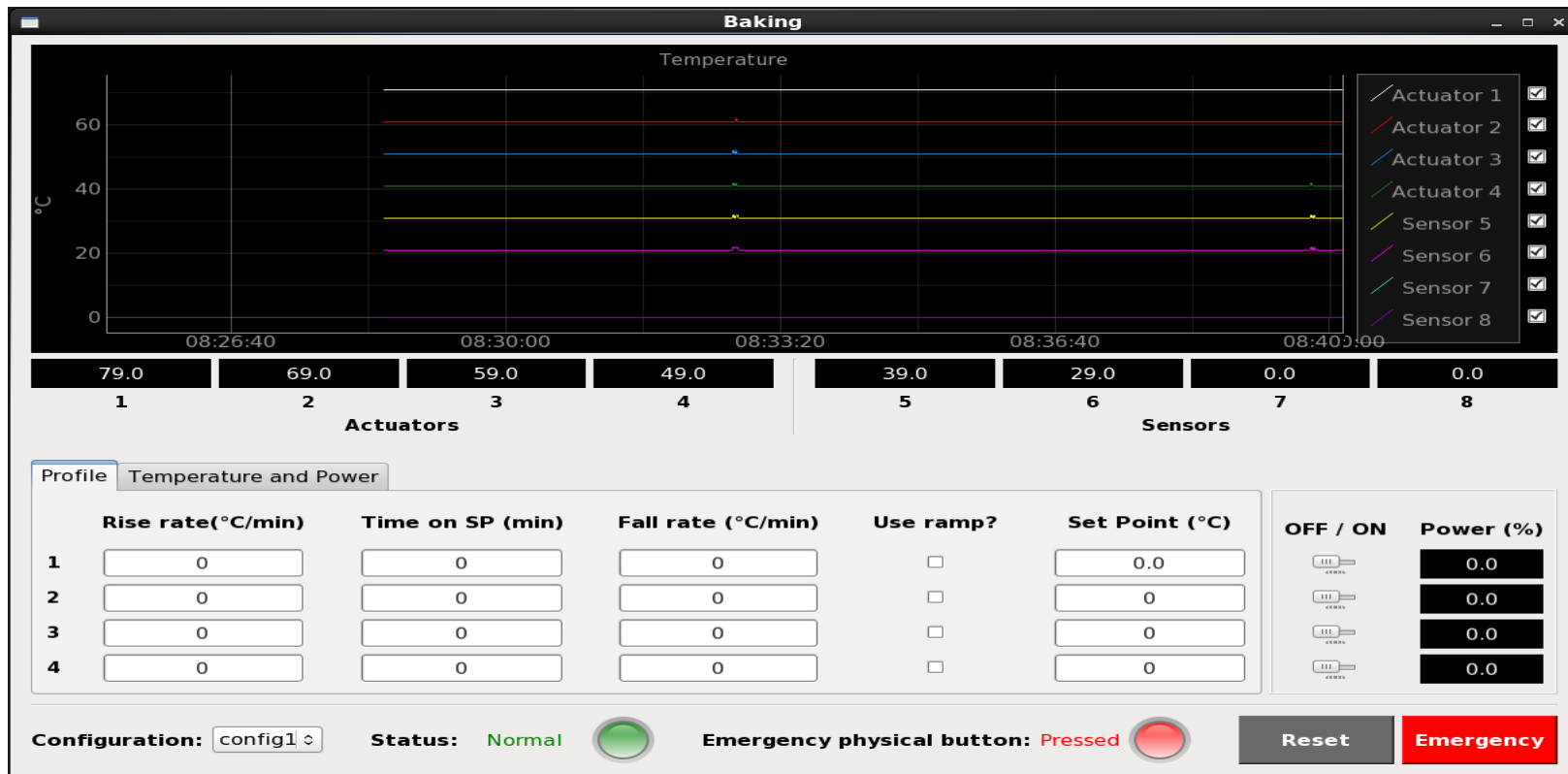
*J. Gabadinho et.al., 2010, "MxCuBE: a synchrotron beamline control environment customized for macromolecular crystallography experiments", J. of S. Radiation, V. 17, pp. 700-707.



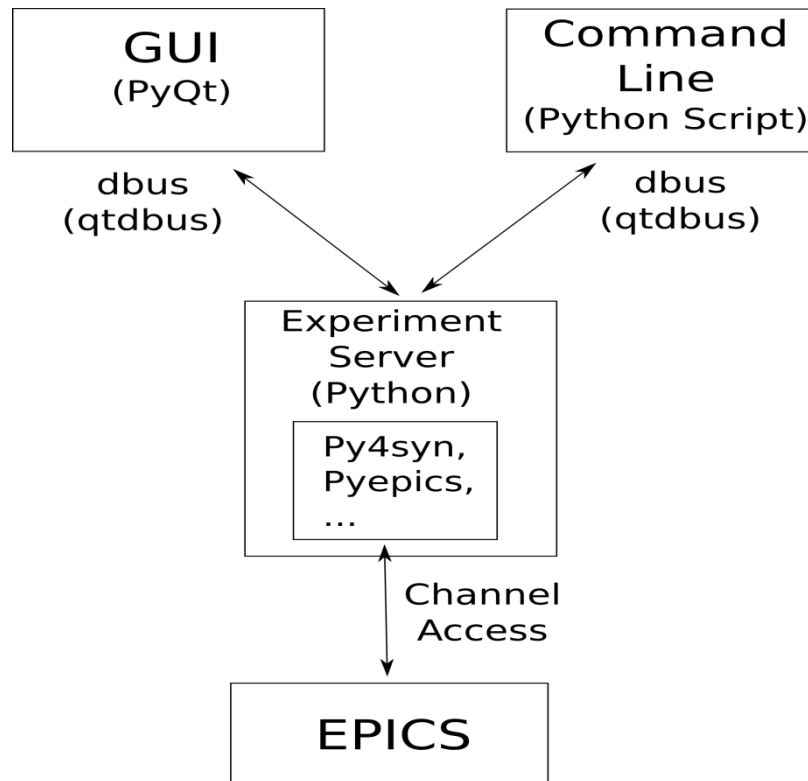
New PyQt + PyDM based UIs



New PyQt + PyDM based UIs



Framework



New PyQt + PyDM based UIs

Scan Motors - [Preview] — Qt Designer

Parameters

Output Path: ...

Counter Set: ...

☐ Go to optimum

Motor: + -

	Run 1	Run 2	Run 3
Couting Time (s)	500		
Initial - Motor sh2x (mm)	0		
Final - Motor sh2x (mm)	10		
Step - Motor sh2x (mm)	0.1		

Time Expected:

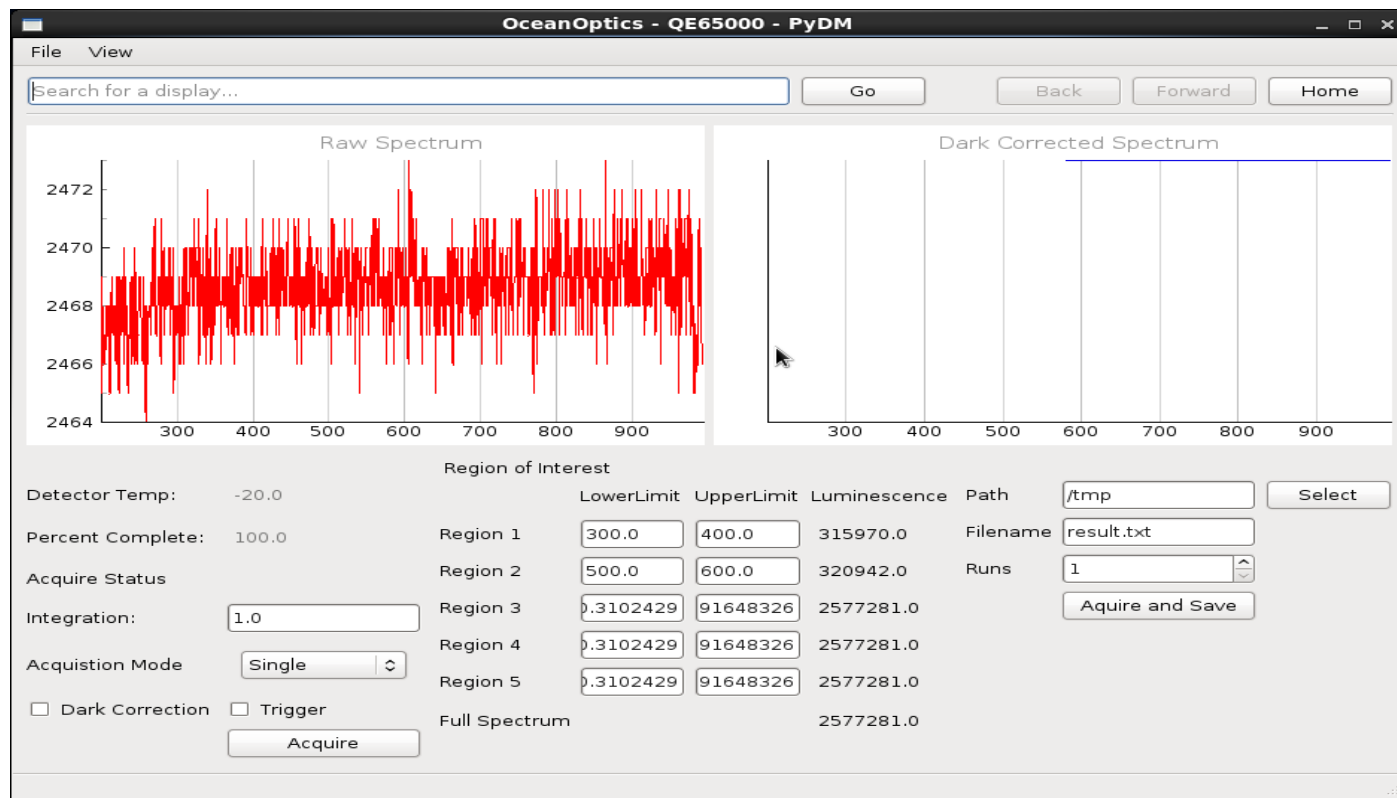
Save Load

Start Stop Pause Resume

Results

Output:

New PyQt + PyDM based UIs



Next steps...

- Standardize common windows
 - ✓ *To cover as many techniques as possible with single solution*
- Access control
 - ✓ *Different views for beamline technicians and researchers*
- Web access
 - ✓ *Proposed framework should facilitate migration to a web interface solution*

Questions...

