

Generic Data Acquisition Interfaces and Processes in Sardana

Z. Reszela, J. Andreu, G. Cuní, T. Coutinho, C. Falcon-Torres, D. Fernandez-Carreiras (on leave), R. Homs-Puron, C. Pascual-Izarra, D. Roldan (on leave), M. Rosanes-Siscart (ALBA-CELLS Synchrotron, Barcelona, Spain), M. T. Nunez Pardo de Vera (DESY, Hamburg, Germany), A. Milan-Otero (MAXIV Laboratory, Lund, Sweden), G. W. Kowalski (Solaris, Krakow, Poland)

GENERAL PROBLEM

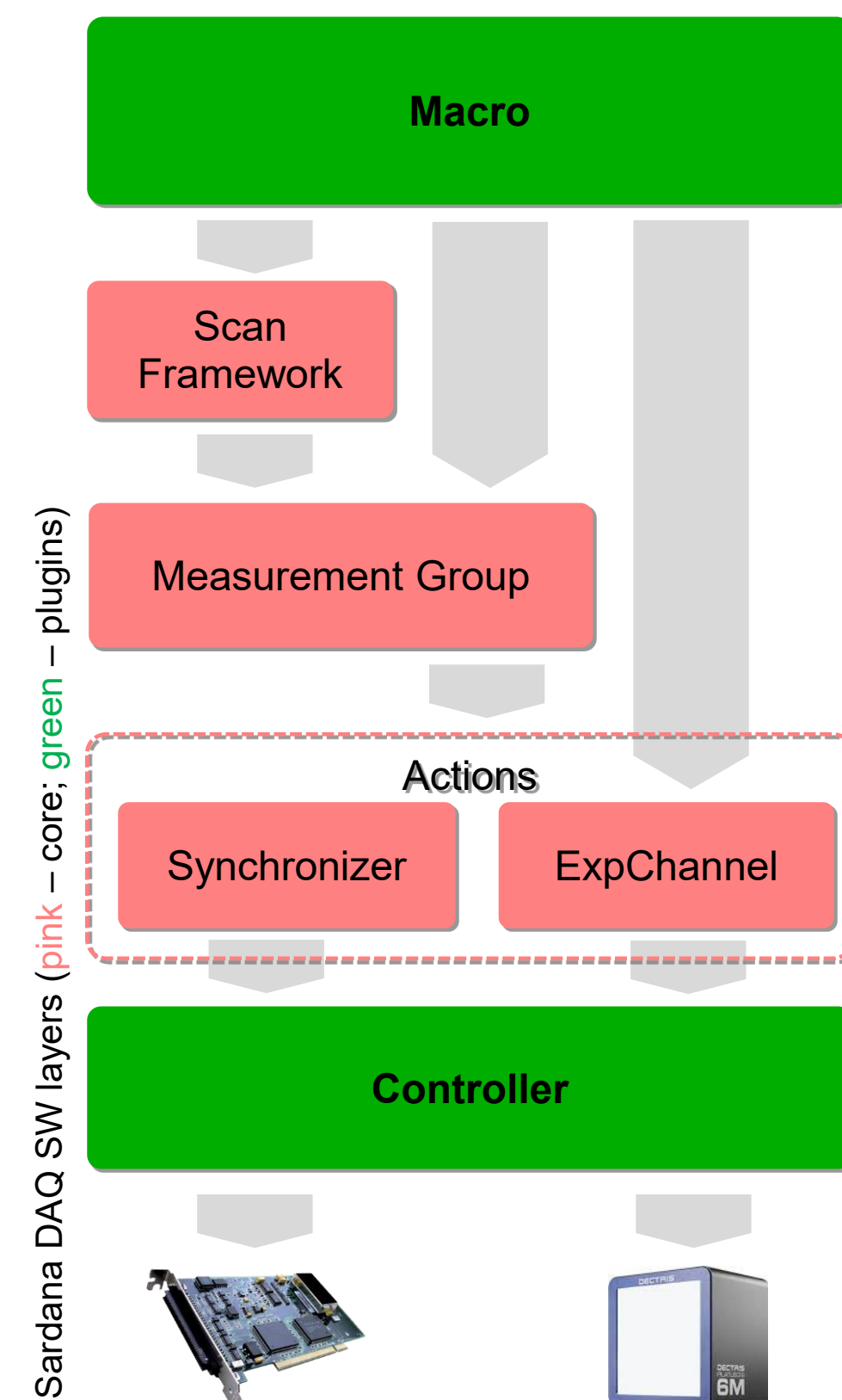
- Diverse experiment control apps across different facilities
- Multiple tools for different DAQ modes within the same lab
- Users lack of time to understand DAQ architecture

TECHNICAL PROBLEM

- Rich in features DAQ HW are hard to abstract in SW
- Detectors HW produce larger volumes of data at higher rate

SOLUTION

- Flexible GUI and CLI interfaces to the CS
- Active synoptic as navigation tool to instrument panels
- Python based macro execution environment
- Turn-key step and continuous scan macros
- Scan framework for development of custom scans
- HDF5, SPEC, FIO or any custom data format
- 1D and 2D data references accessible via HDF5 VDS
- Generic DAQ synchronization (SW or HW)
- Single, grouped and autonomous DAQ modes
- Generic interface of experimental channels



```
h5demo_1 [1]: ct 1 adc
Sat Sep 28 18:56:32 2019
adc = 105.5

h5demo_1 [2]: defmeas mntgrp adc mca det
Created mntgrp

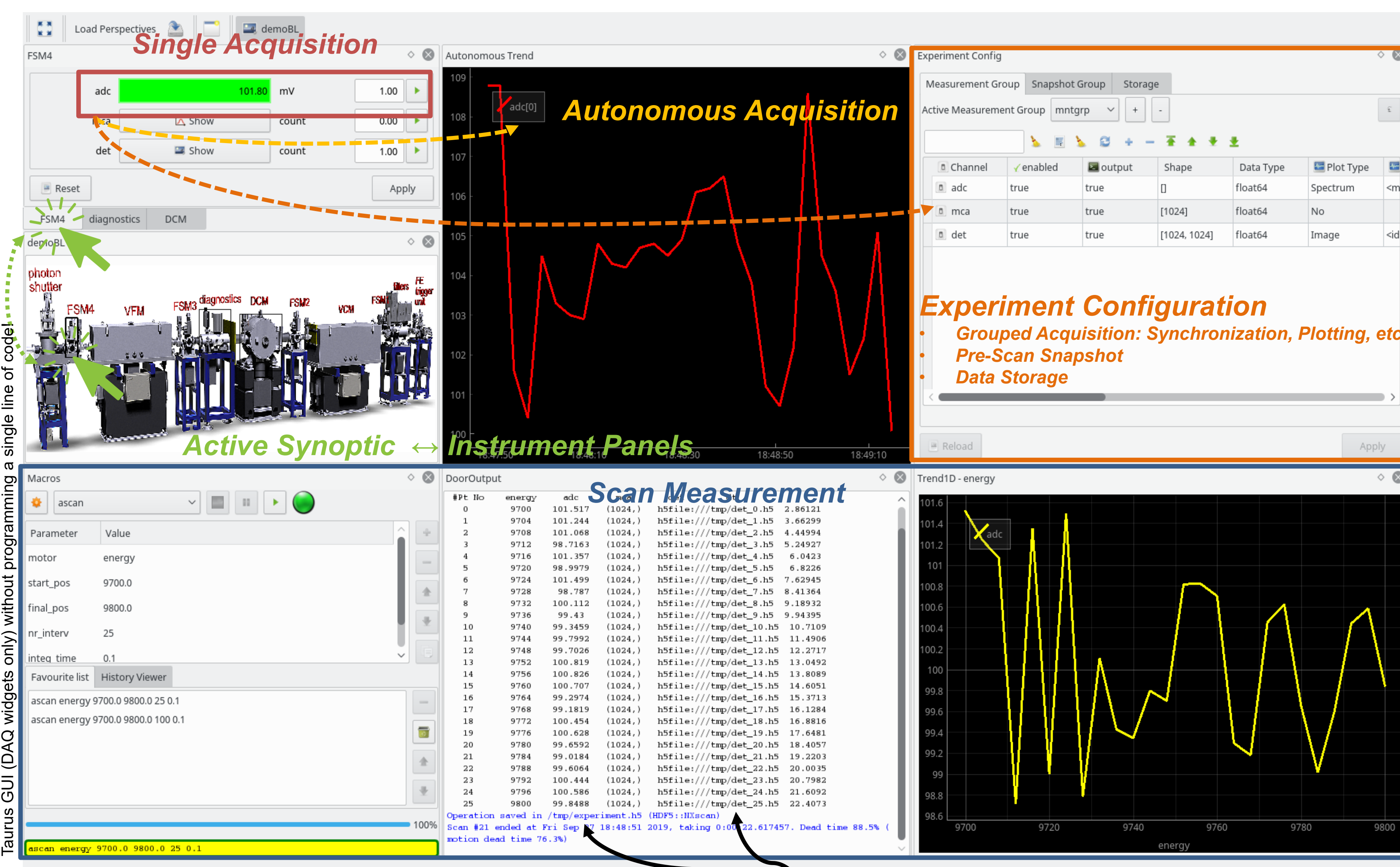
h5demo_1 [3]: ct 1
Sat Sep 28 18:56:53 2019
adc = 100.7
mca = [1024]
det = [1024, 1024]

h5demo_1 [4]: defsnap table_z

h5demo_1 [5]: newfile /tmp/experiment.h5
ScanDir is : /tmp
ScanFile set to : experiment.h5
Next scan is : #1

h5demo_1 [6]: ascan energy 9700 9710 10 0.1
Operation will be saved in /tmp/experiment.h5 (HDF5::NXscan from NXscan5 FileRecorder)
Scan #1 started at Sat Sep 28 19:00:03 2019. It will take at least 0:00:03.928427
#Pt No energy adc mca det dt
0 9700 100.7 (1024,) h5file:///tmp/det_0.h5 0.0675027
1 9701 103.3 (1024,) h5file:///tmp/det_1.h5 0.563249
2 9702 108.8 (1024,) h5file:///tmp/det_2.h5 1.09719
3 9703 108.4 (1024,) h5file:///tmp/det_3.h5 1.62269
4 9704 106.9 (1024,) h5file:///tmp/det_4.h5 2.1518
5 9705 103.3 (1024,) h5file:///tmp/det_5.h5 2.67728
6 9706 109.6 (1024,) h5file:///tmp/det_6.h5 3.19647
7 9707 103 (1024,) h5file:///tmp/det_7.h5 3.73192
8 9708 102.5 (1024,) h5file:///tmp/det_8.h5 4.25441
9 9709 107.5 (1024,) h5file:///tmp/det_9.h5 4.77317
10 9710 104.1 (1024,) h5file:///tmp/det_10.h5 5.31879
Operation saved in /tmp/experiment.h5 (HDF5::NXscan)
Scan #1 ended at Sat Sep 28 19:00:09 2019, taking 0:00:05.559269. Dead time 80.2% (motion dead time 52.8%)

h5demo_1 [7]:
```



Virtual Data Sets

```
class ExpChannel1:
    value: float
    value_buffer: encoded
    state: enum
    value_ref*: str
    value_ref_buffer*: encoded
    integration_time*: float
    timer*: string
    current_value: float
    ...

    start()
    stop()
    abort()
```

Experimental channel interface

```
class MyDetCtrl(TwoDController):
    def LoadOne(self, axis, integ_time):
        [...]

    def StartOne(self, axis):
        [...]

    def StateOne(self, axis):
        [...]

    def ReadOne(self, axis):
        [...]

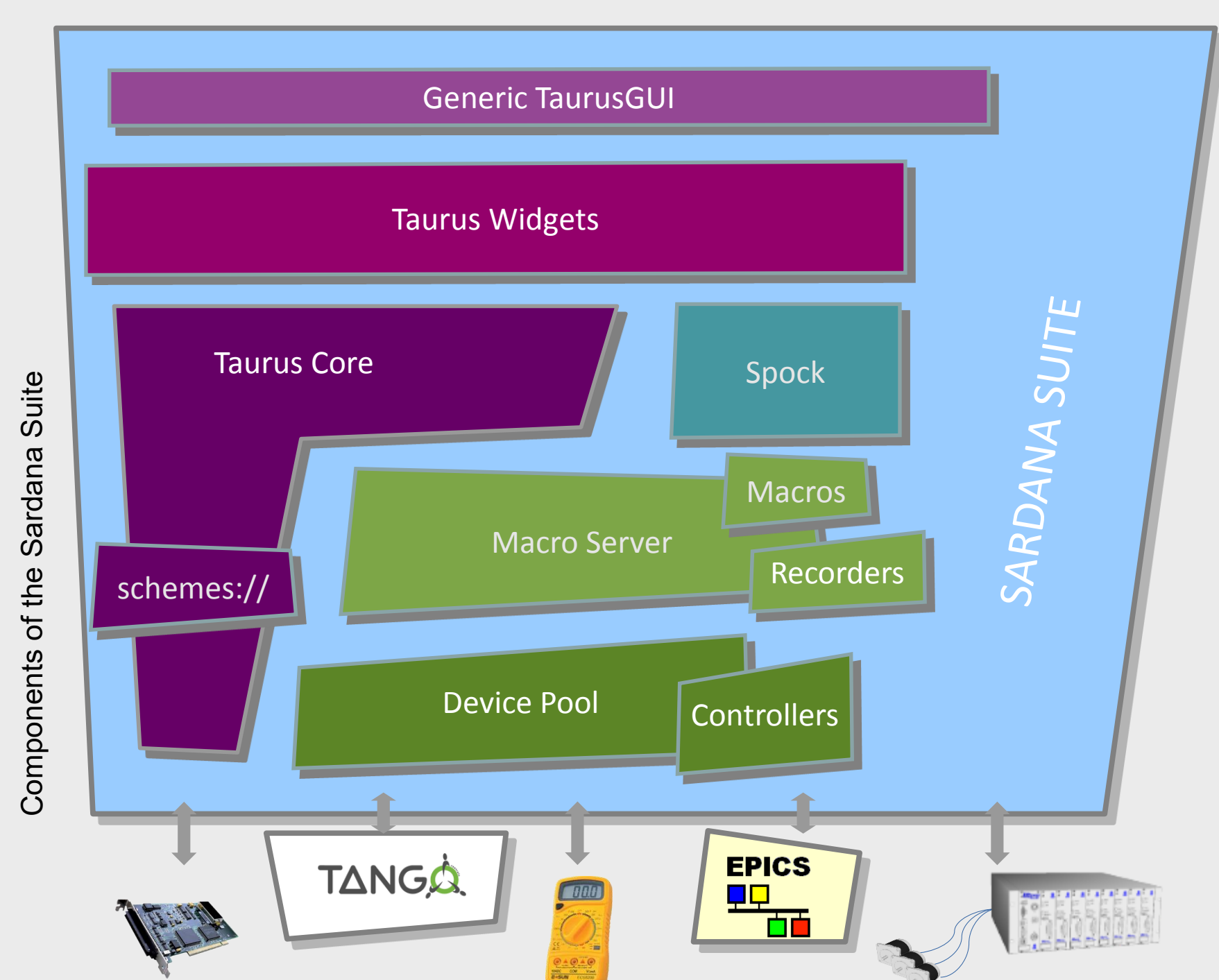
    def AbortOne(self, axis):
        [...]
```

Skeleton of the experimental channel controller

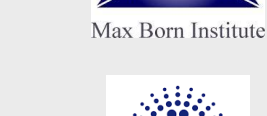
BENEFITS

- Improved UX at different sites within Sardana and Taurus collaboration
- Gentle learning curve to the CS:
 - Consistent look-and-feel
 - SPEC-like macro execution syntax
- Well isolated layers of DAQ SW
 - Easier to understand DAQ architecture
 - Promote code reusability
 - Facilitate new developments

SARDANA – Scientific SCADA Suite



- Widely used
- Production-ready
- Well supported
- Actively developed
- Free/Open Source
- Community-driven
- Modular
- Multi-platform
- Based on Python
- Easy to install



More than 30 labs using Sardana

NEXT

- Reduce dead times in scans
- Finish implementation of autonomous DAQ
- Refactor plugin system
- Generalize capabilities
- Support non-linear motion trajectories

MORE INFO

- <https://sardana-controls.org>
- <https://taurus-scada.org>

Acronyms and abbreviations

app – application, lab – laboratory, DAQ – data acquisition, HW – hardware, SW – software, GUI – Graphical User Interface, CLI – Command Line Interface, UX – user experience, CS – Control System