



# SARDANA – Scientific SCADA Suite

by Guifré CUNÍ, Grzegorz KOWALSKI, Zbigniew RESZELA  
on behalf of **Sardana and Taurus Communities**

Sardana Workshop @ ICALEPCS2019, New York, 6<sup>th</sup> October 2019

# Outline

- Sardana - What is? And what is not?
- Sardana Suite overview
- Software architecture & layers
- Sardana Community

# Sardana – What is? And what is not?

# Beamline Hardware Components

Motion



Vacuum



PLC



CCDs



AI, AO, DAQ

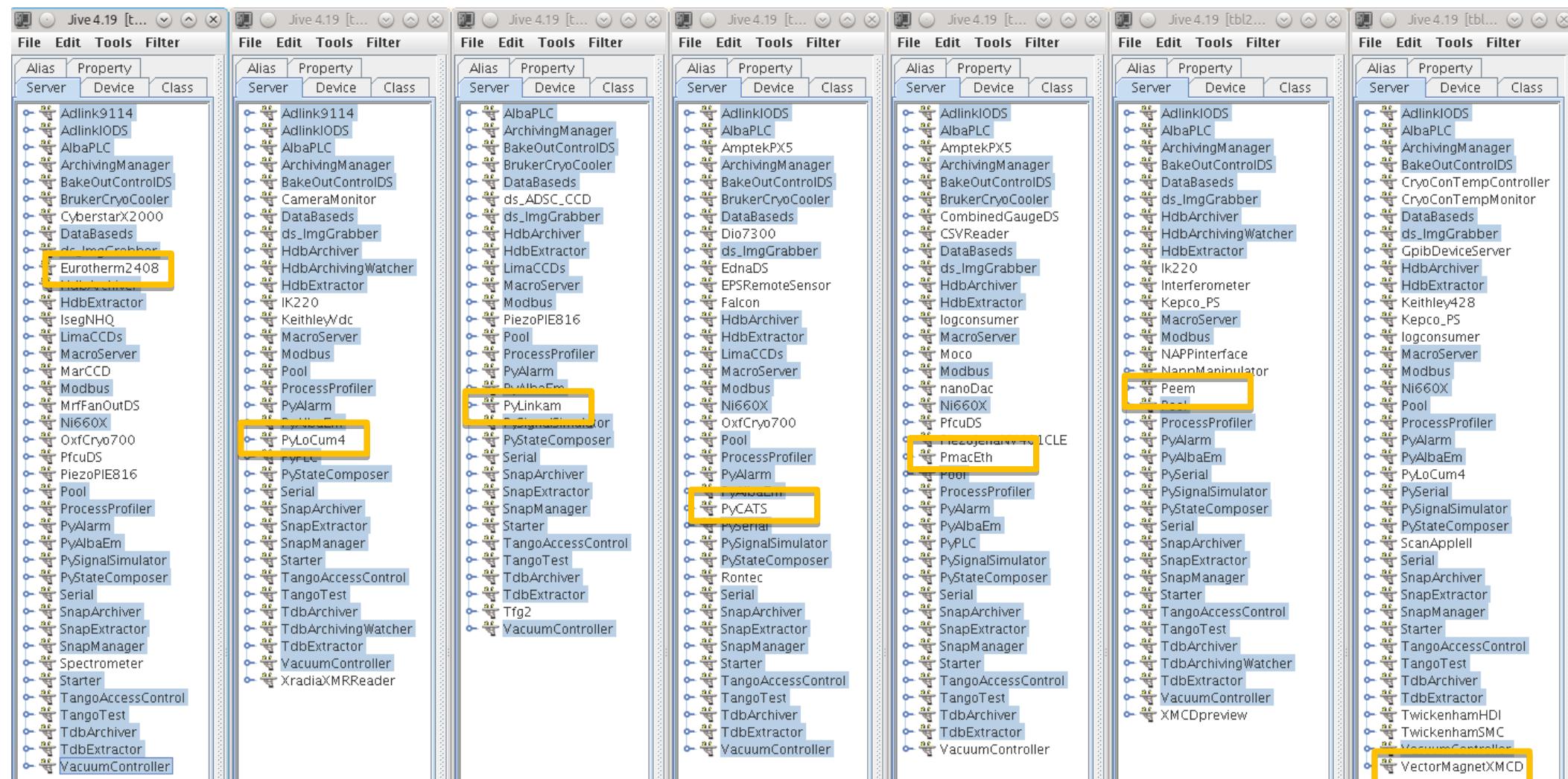


IPC



Workstations

# Beamline Control System (on example of Tango)



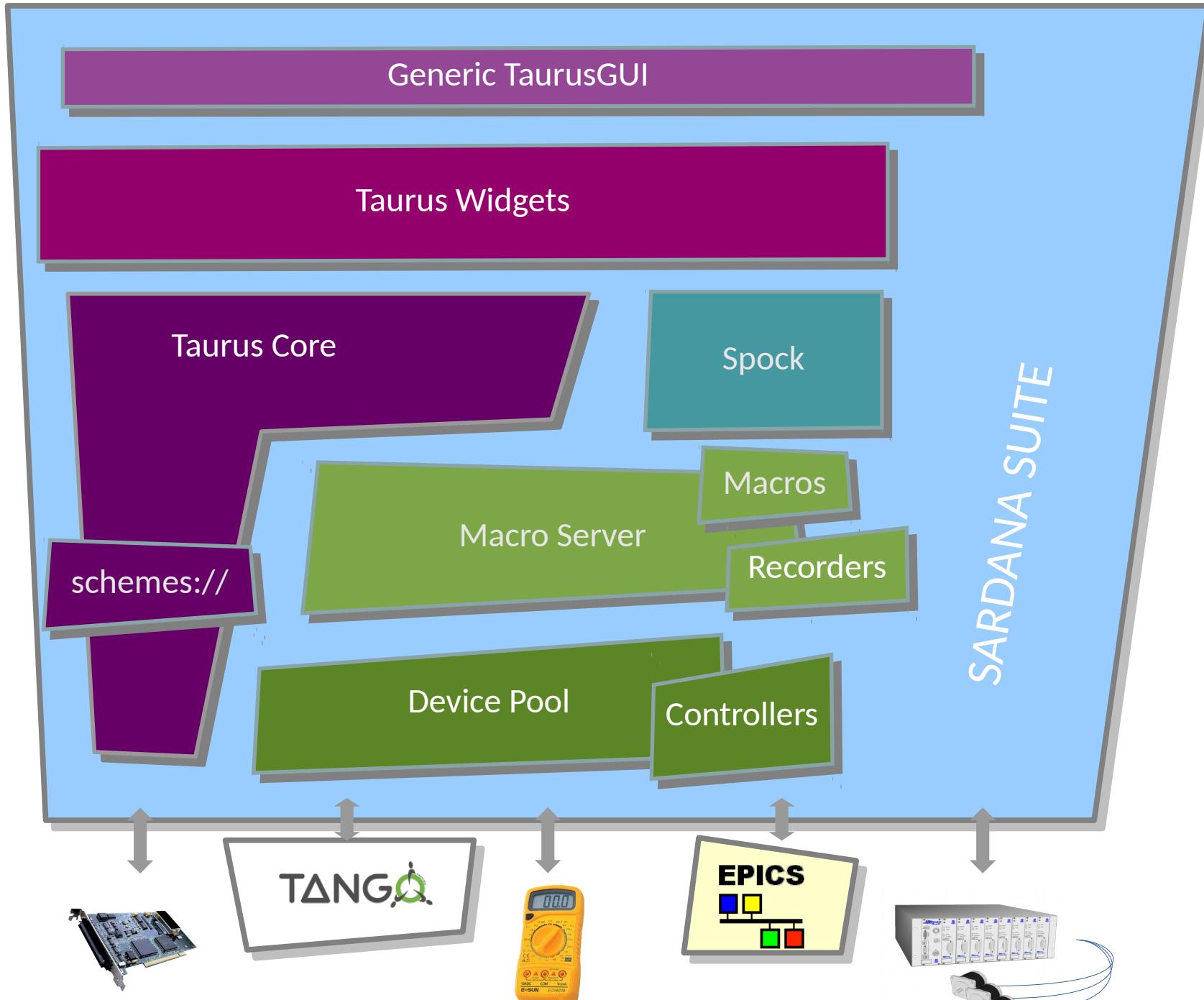
# What Sardana is? And what is not?

- Motion Control 
- Vacuum Pumps and Gauges 
- Equipment Protection System PLC 
- Diagnostic Cameras, Electrometers, ... 
- AI, AO, DAQ (NI6602, Adlink 2005, ...) 
- Archiving System 
- Software Alarm System 
- Experiment automation and synchronization 
- Program and execute user procedures 

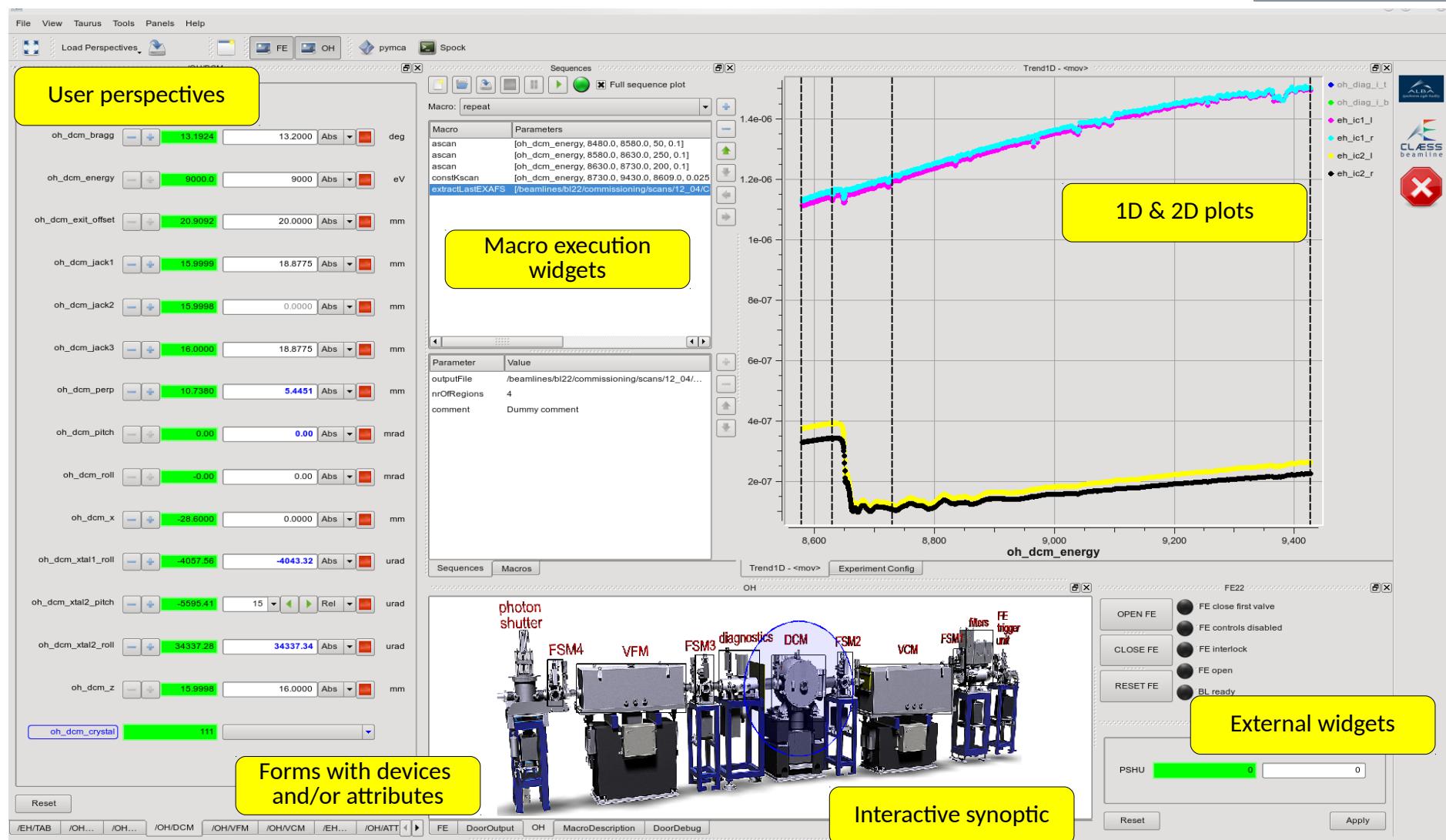
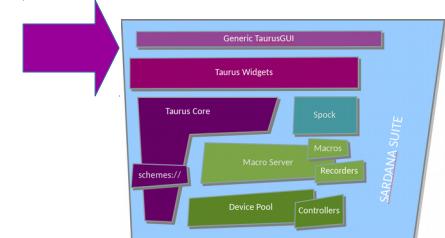
Sardana is a software suite for Supervision, Control and Data Acquisition in scientific installations. It aims to reduce cost and time of design, development and support of the control and data acquisition systems. Sardana development was started at the [ALBA](#) synchrotron and today is supported by a larger community which includes several other laboratories and individuals ([ALBA](#), [DESY](#), [MaxIV](#), [Solaris](#), [ESRF](#)).

# Sardana Suite overview

# Sardana Suite overview

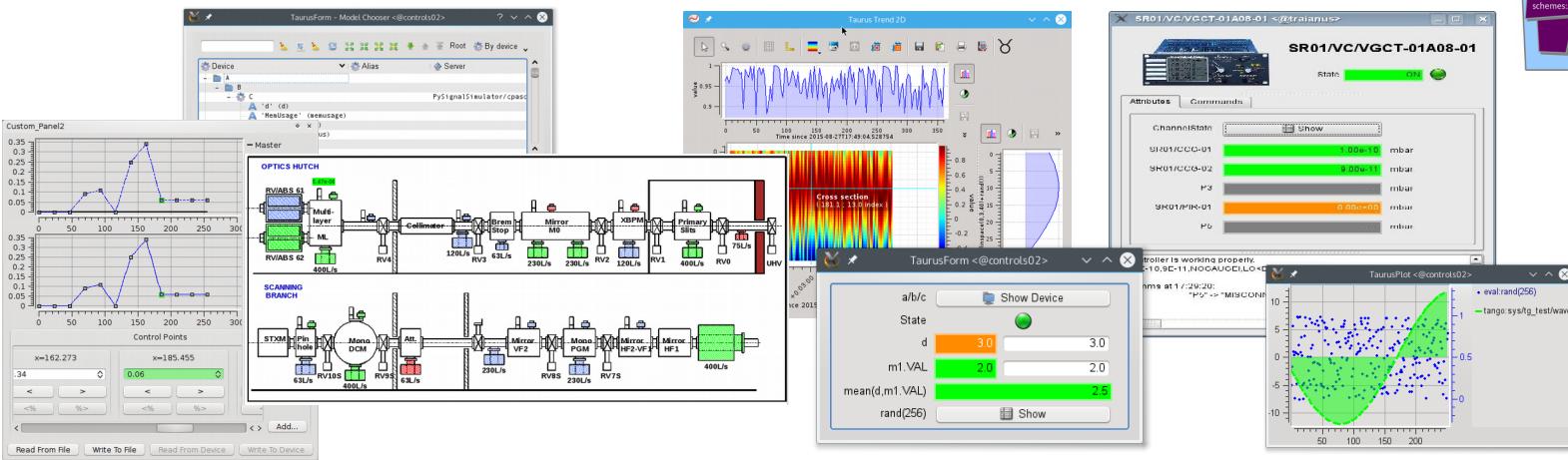
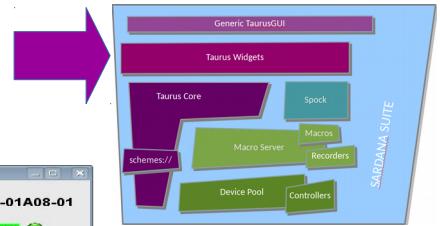


# Generic Taurus GUI



TaurusGUI framework for creating complete GUIs **without programming a single line of code!**

# Taurus Widgets



- Qt based graphical widgets: generic forms, plots, ...
- Sardana specific widgets: macro executor, motor, experiment configuration, scan plots, ...

**Left Window (TaurusForm):**

Parameter	Value
motor	m_cpl_1
start_pos	0.0
final_pos	100.0
nr_interv	10
integ_time	0.1

**Center Window (TaurusForm):**

Active Measurement Group: cp\_mg1

Channel	output	Shape	Plot Type	Plot Axes
ct_cp1_1	true	[]	Spectrum	<mov>
ct_cp1_2	true	[]	Spectrum	<mov>
sys/tg_te...	true	[251, 251]	Image	<idx>:<idx>

**Right Window (Code Editor):**

```

ms/v3/1
Macro libraries
  training
    where_moveable_2
    where_moveable
    snapshot_ccd
    set_scand
    scan_info
    set_scand
    scan
    move_movable
    move_energy
    move2
    mandelbrot
    hello_world_2
    fixed_ascan
    fft_freq
    cosine
    test01
    submacros
    standard
    sequence
    scans
    scan
    parameters
    aca
    lists
    ioregister
    hooks
    funcs
    expert
    env
    demo
    communication

```

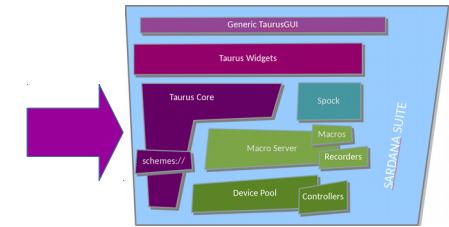
```

71 """This macro changes the current energy"""
72 E = self.getMoveable("Energy")
73 E.move(energy)
74 self.output("Energy is now at %s", E.getPosition())
75
76
77 @macro([ ["ccd", Type.String, None, "ccd name"],
78         [ "fname", Type.String, None, "filename to save" ] ])
79 def snapshot_ccd(self, ccd, fname):
80     """Saves the current ccd image to the specified file"""
81     ccd_dev = self.getDevice(ccd)
82     image = PIL.Image.fromarray(ccd_dev.image)
83     image.save(fname)
84
85 @macro([ ["p1", Type.Float, None, "absolute position of mot
86           1"],
87         [ "p2", Type.Float, None, "absolute position of mot
88           2" ] ])
89 def move2(self, p1, p2):
90     motion = self.getMotion(["mot01", "mot02"])
91     motion.move([p1, p2])
92
93 @macro([ ["moveable", Type.Moveable, None, "moveable to get
94           position" ],
95         [ "position", Type.Float, None, "absolute position" ] ])
96 def where_moveable_2(self, moveable):
97     """This macro prints the current moveable position"""
98     self.wm(moveable)
99
100 @macro([ ["moveable", Type.Moveable, None, "moveable to mov
101           e" ],
102         [ "position", Type.Float, None, "absolute position" ] ])
103 def move_moveable(self, moveable, position):
104     """This macro moves a moveable to the specified position
105
106     self.mv(moveable, position)
107     self.output("%s is now at %s", moveable.getName(), move
108               able.getPosition())

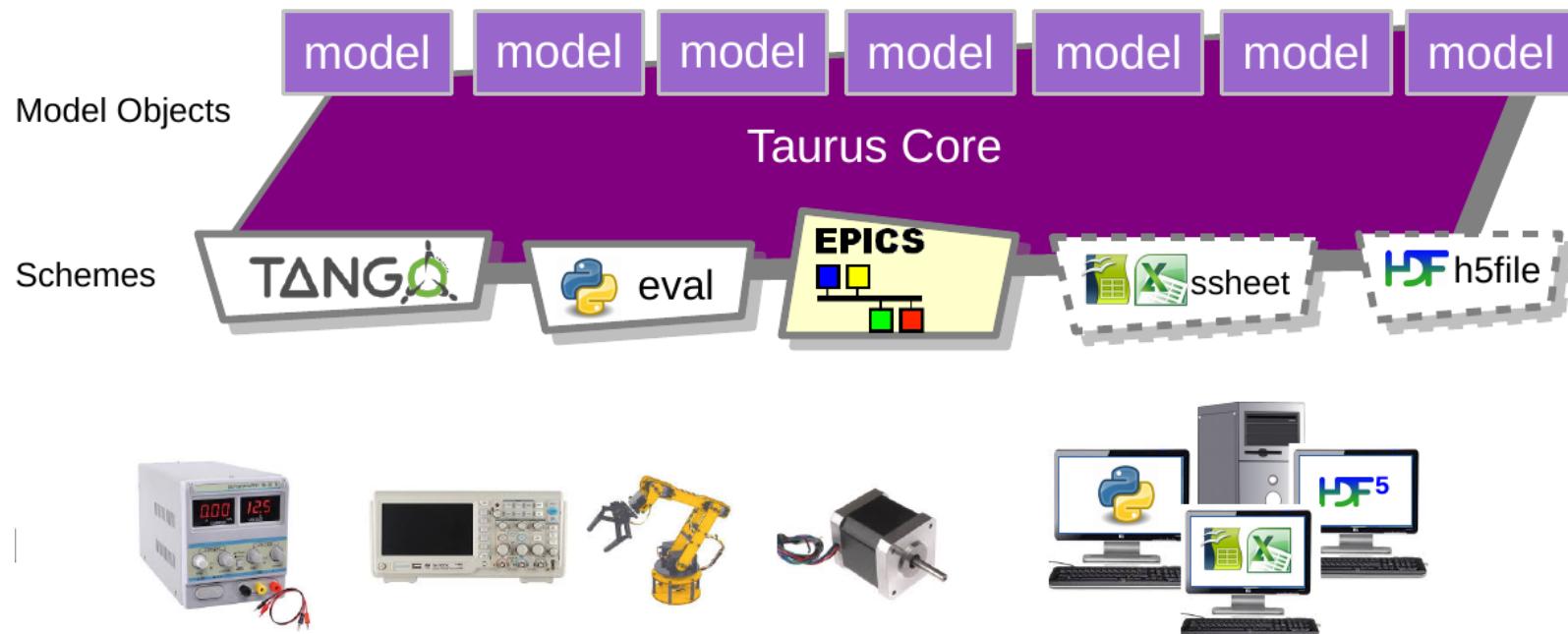
```

# Taurus Core

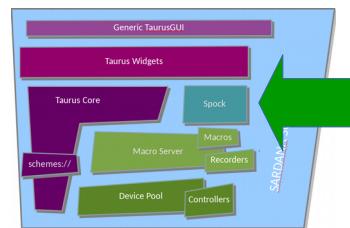
11



- “Container of unique models”.
- Schemes provide access to different type of data sources.
- Polling and event mechanism.



# Spock



- **Spock** – IPython based Sardana CLI.
- Spock syntax mimics **SPEC** commands.
- Provides total control over the system: executes procedures, interacts with the elements, ...

**Terminal Session:**

```
/bin/bash 90x39
tcoutinho@pc151:~/workspace/Spock$ ./spock -p BL98

Spock 7.2.1 -- An interactive Tango client.

Running on top of Python 2.6.6, IPython 0.10 and PyTango 7.2.1dev

help      -> Spock's help system.
object?   -> Details about 'object'. ?object also works, ?? prints more.

Spock's sardana extension 0.5.0 loaded with profile: BL98 (linked to door 'Door_BL98')

Door_BL98 [1]: %ascan bl98_m1 0 100 10 0.1
ExtraColumns is not defined
ScanDir is not defined. This operation will not be stored persistently
SharedMemory is not defined.
SharedMemory is not defined.
Scan started at Tue Jun 28 18:06:16 2011. It will take at least 0:00:01.100000
 #Pt No    BL98_M1    BL98_Timer    BL98_C1    BL98_C2    BL98_C3
  0        0.0       0.1       0.103096   0.206192   0.309288
  1       10.0       0.1       0.10095    0.2019     0.30285
  2       20.0       0.1       0.102416   0.204832   0.307248
  3       30.0       0.1       0.105096   0.210192   0.315288
  4       40.0       0.1       0.111601   0.223202   0.334803
  5       50.0       0.1       0.113532   0.227064   0.340596
  6       60.0       0.1       0.115527   0.231054   0.346581
  7       70.0       0.1       0.101574   0.203148   0.304723
  8       80.0       0.1       0.117536   0.235072   0.352608
  9       90.0       0.1       0.101459   0.202918   0.3043
 10      100.0      0.1       0.113926   0.227852   0.3417
Scan ended at Tue Jun 28 18:06:33 2011, taking 0:00:16.645132 (dead

Door_BL98 [2]: wa
Current Positions (user, dial)

BL98_M1    BL98_M2    BL98_MP1
100.0000   43.0000   100.0000
100.0000   43.0000   100.0000

Door_BL98 [3]:
```

**File Browser:**

A screenshot of a file browser window titled "File/Group/Dataset". It shows a hierarchical tree structure of datasets under "scans.h5". The "measurement" group is expanded, showing datasets for PT\_No, ct01, ct02, ct03, ct04, and gap01, along with their respective shapes and data types. Below the tree is a plot window showing a linear trend over time.

**Plot Viewer:**

A screenshot of a plot window titled "TreeID = <root>". The plot shows multiple data series (blue, magenta, cyan, yellow, black) against "oh\_dcm.energy" on the x-axis (ranging from 8.600 to 9.400) and time on the y-axis (ranging from 1e-07 to 10). Vertical dashed lines are present at approximately 8.65, 8.85, and 9.05.

**Configuration GUI:**

A screenshot of a configuration interface for a measurement group. It shows a table with columns for Channel, enabled, output, shape, data type, plot type, plot axes, timer, monitor, synchronizer, synchronization, conditioning, normalization, and Nexus Path. Several rows are listed, each defining a specific monitoring or control channel with its properties.

# MacroServer

**Hooks**

```
Door_1 [8]: loop 0 10 3
Starting loop
At step 0
running hook with hints=['pre-acq']
In hook 1
At step 3
running hook with hints=['pre-acq']
In hook 1
At step 6
running hook with hints=['pre-acq']
In hook 1
At step 9
running hook with hints=['pre-acq']
In hook 1
Finished loop
```

**Input parameters & results & data**

**SPEC like commands**

**Plotting**

**pcapac2014.py**

```
13
14 @macro([["moveable", Type.Moveable, None, "moveable to move"], ["position", Type.Float, None, "absolute position"]])
15 def move(self, moveable, position):
16     """This macro moves a motor to the specified position"""
17     moveable.move(position)
18     self.output("Motor ended at ", moveable.getPosition())
19
20 class loop(Macro, Hookable):
21     """A macro that executes a for loop. It accepts hooks.
22     hints = {'allowsHooks':('pre-move', 'post-move', 'pre-acq', 'post-acq')}
23
24     param_def = [[['start', Type.Integer, None, 'start point'], ['stop', Type.Integer, None, 'end point'], ['step', Type.Integer, 1, 'step']]]
25
26     result_def = [['result', Type.Integer, None, 'result']]
27
28     def hook1(self):
29         self.output("In hook 1")
30
31     def run(self, start, stop, step):
32         self.info("Starting loop")
33         self.hooks = [(self.hook1, ["pre-acq"])]
34         for i in xrange(start, stop, step):
35             self.output("At step %d" % i)
36             self.flushOutput()
37
38     def hook2(self, hints):
39         for hook in self.hooks:
40             hook[0]()
41         self.info("finished loop")
42         return i
43
44 class hooked_scan(Macro):
45     """An example on how to attach hooks to the various hook points of a scan.
46     """
47     param_def = [
48         ['motor', Type.Moveable, None, 'Motor to move'],
49         ['start_pos', Type.Float, None, 'Scan start position'],
50         ['end_pos', Type.Float, None, 'Scan final position'],
51         ['nr_interv', Type.Integer, None, 'Number of scan intervals'],
52         ['integ_time', Type.Float, None, 'Integration time']]
53
54     def hook1(self):
55         self.info("\thook1 execution")
56
57     def run(self, mot, start, end, nr, intt):
58         self.ascan, pars = self.createMacro("ascan", mot, start, end, nr, intt)
59         self.ascan.hooks = [(self.hook1, ["pre-acq"])]
60         self.runMacro(ascan)
61
62     @property
63     def data(self):
64         return self.ascan.data
65
66     @macro()
67     def ask_number_of_points(self):
68         """Asks user for the number of points"""
69         nb_points = self.input("How many points?", data_type=Type.Integer)
70
71     @macro()
72     def J0_plot(self):
73         """Sample J0 at linspace(0, 20, 200)"""
74         x = linspace(0, 20, 200)
75         y = J0(x)
76         x1 = x[::10]
77
78         y1 = map(j0, x1)
79         self.pyplot.plot(x, y, label=r'$J_0(x)$')
80         self.pyplot.plot(x1, y1, 'ro', label=r'$J_0(x)\backslash\integ(x)$')
81         self.pyplot.title(r'Verify $J_0(x)=\frac{1}{\pi}\int_0^{\pi} \cos(x \sin\phi)$')
82         self.pyplot.xlabel('x$')
83         self.pyplot.legend()
```

**Adding, editing macros at runtime**

**Interactive macros**

**Macro Server**

**ShmRec**

**StdoutRec**

**JsonRec**

**NXScanRec**

**NXasRec**

**FioRec**

**System**

**ESRF .spec**

**DESY .fio**

**Nexus NXscan**

**Nexus NXas**

**Generic TaurusGUI**

**Taurus Widgets**

**Taurus Core**

**Macros**

**Recorders**

**schemes//**

**Device Pool**

**Controllers**

**Macro Server**

**Device Pool**

**Controllers**

**Generic TaurusGUI**

**Taurus Widgets**

**Taurus Core**

**Macros**

**Recorders**

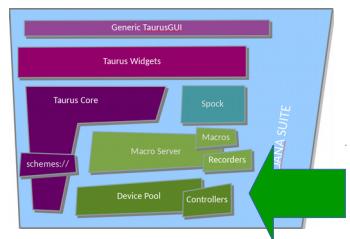
**schemes//**

**Device Pool**

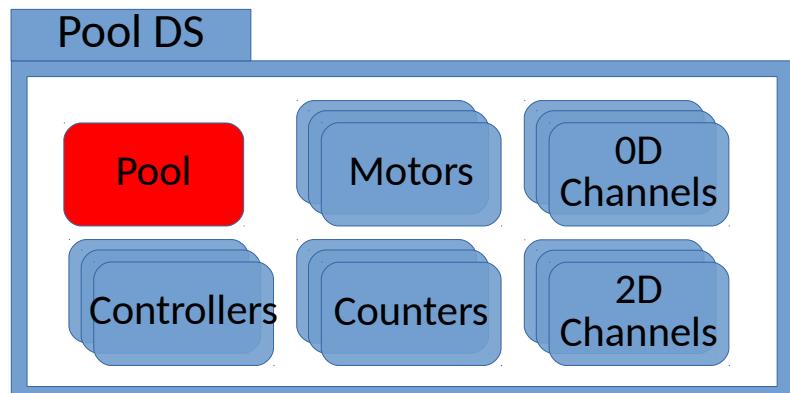
**Controllers**

# Device Pool

14



- All the equipments are interfaced via Device Pool and its **plug-in controller classes (Python)**
- Generic elements' interfaces allow building high level layers on top of them e.g. MeasurementGroup, pseudo elements, generic widgets, etc.

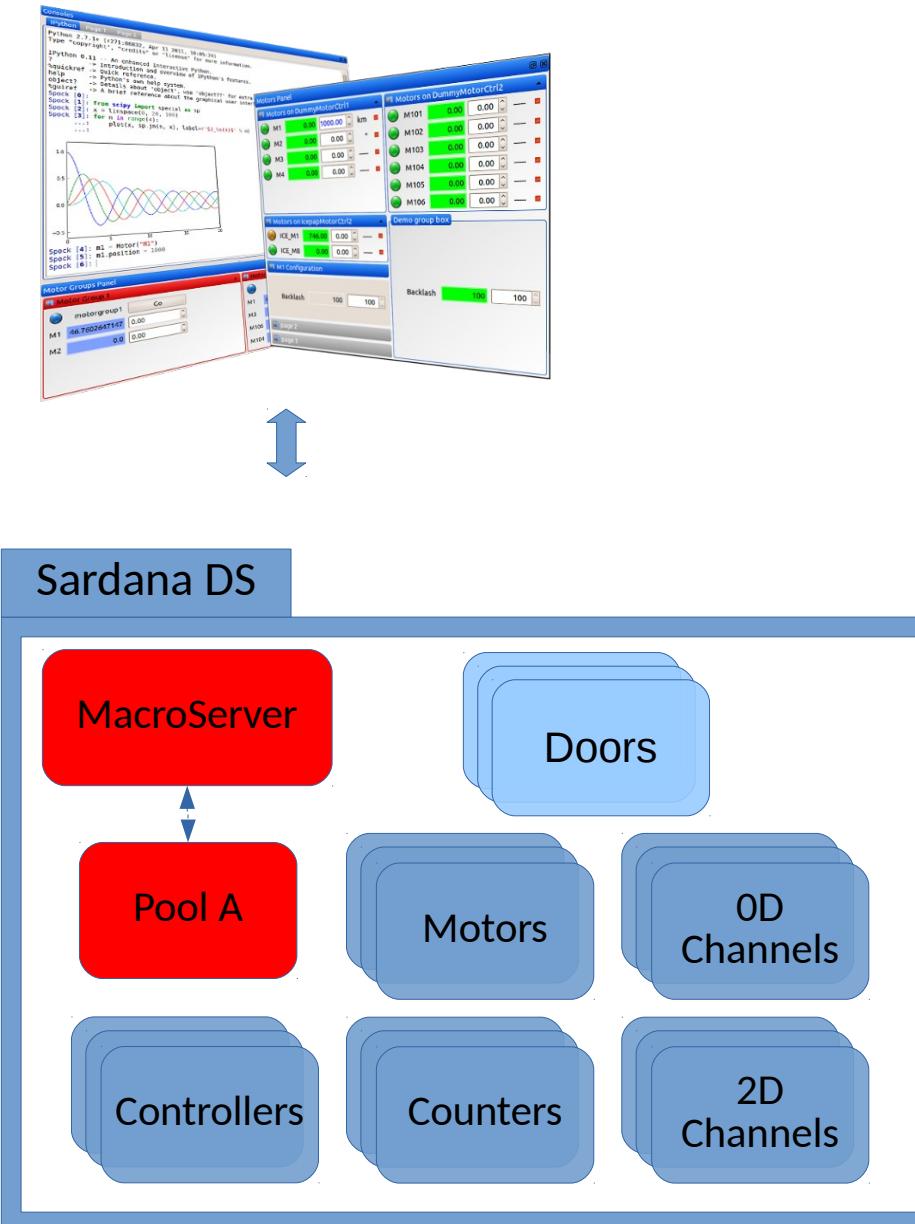


Pool Device Server and its elements

Element Type	Example of application
<b>Motor</b>	stepper, servo or piezo actuator
<b>PseudoMotor</b>	energy, HKL of a diffractometer, slit's gap or offset
<b>CounterTimer</b>	event counter, position measurement, but also... analog to digital converter (ADC), low current electrometer
<b>PseudoCounter</b>	vertical beam position in the X-ray beam position monitor (XBPM)
<b>0DExpChannel</b>	no controlled acq. (software sampling)
<b>1DExpChannel</b>	position sensitive detector (PSD), multichannel analyzer (MCA)
<b>2DExpChannel</b>	CCD camera, 2D X-ray detector
<b>TriggerGate</b>	generators of synchronization signals/events
<b>IOResister</b>	input/output registers of PLC

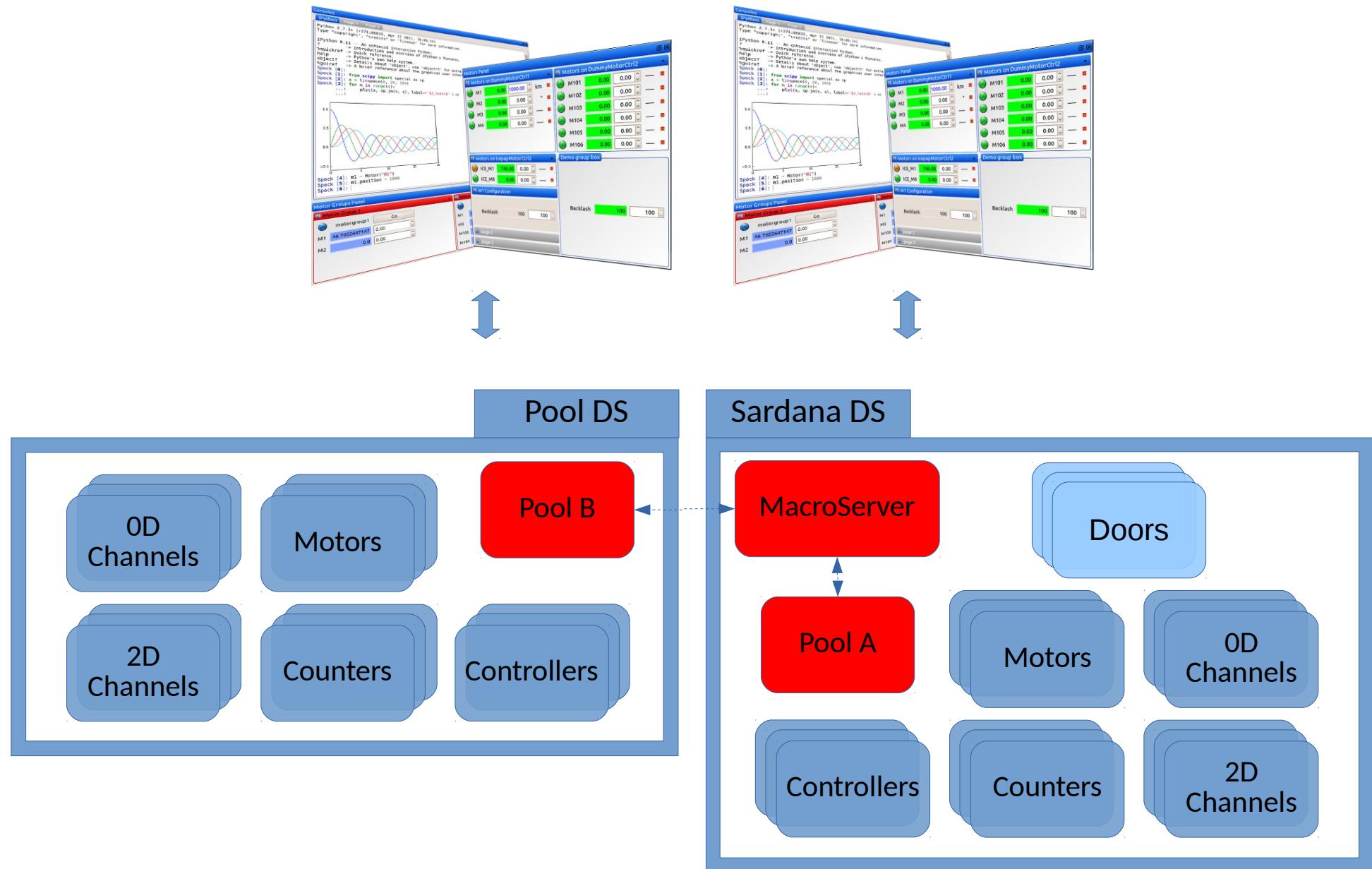
# Software architecture & layers

# Client – Server model



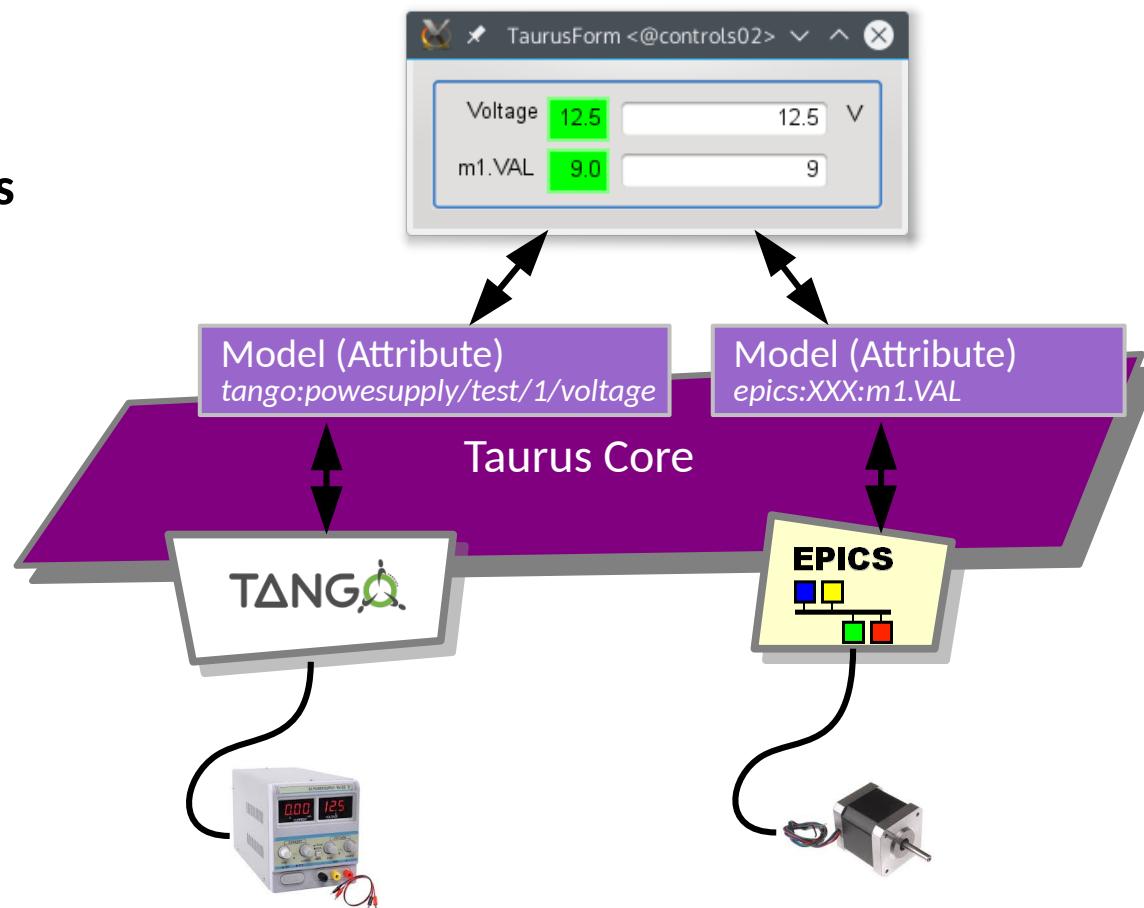
# Client – Server model

17

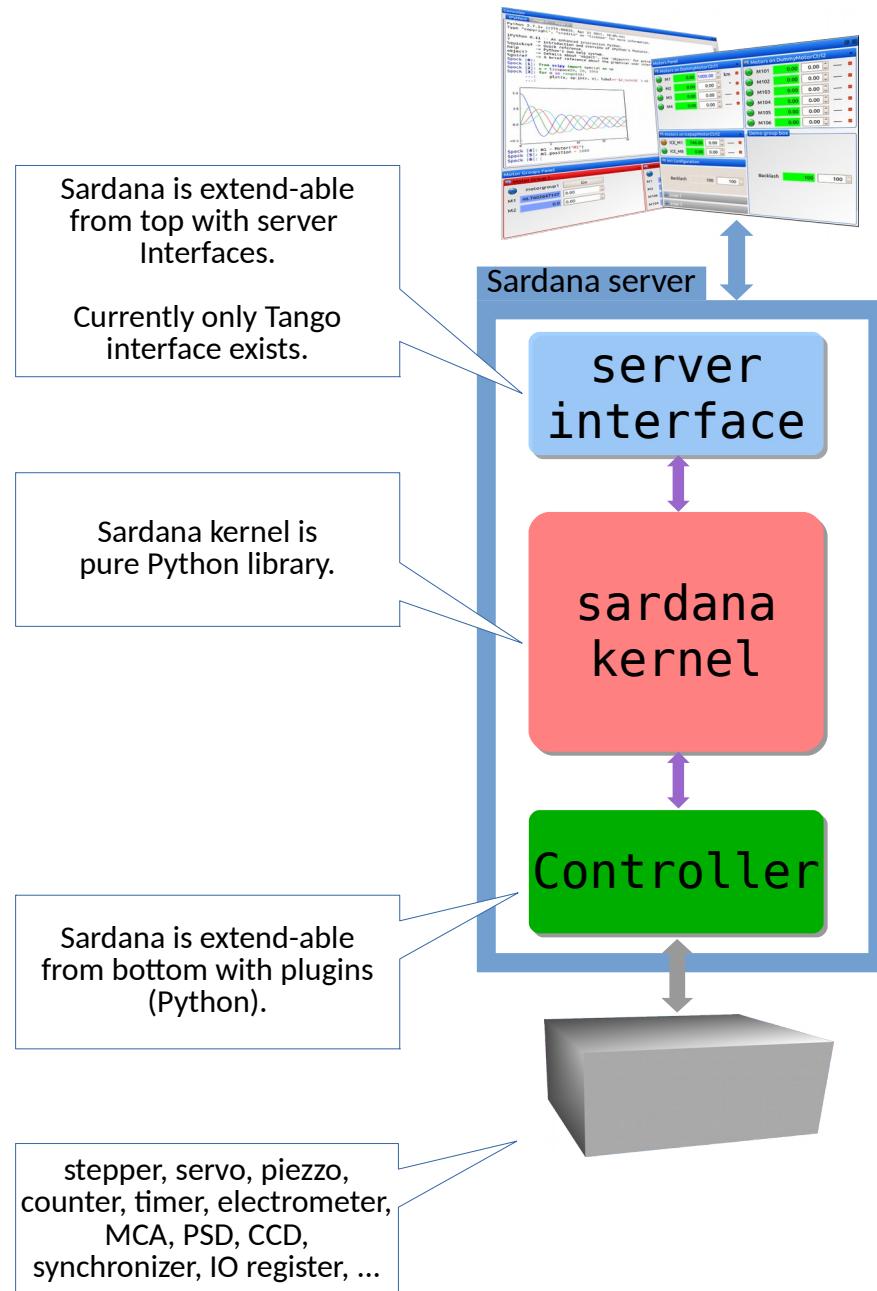


# Software layers - Taurus models & schemes

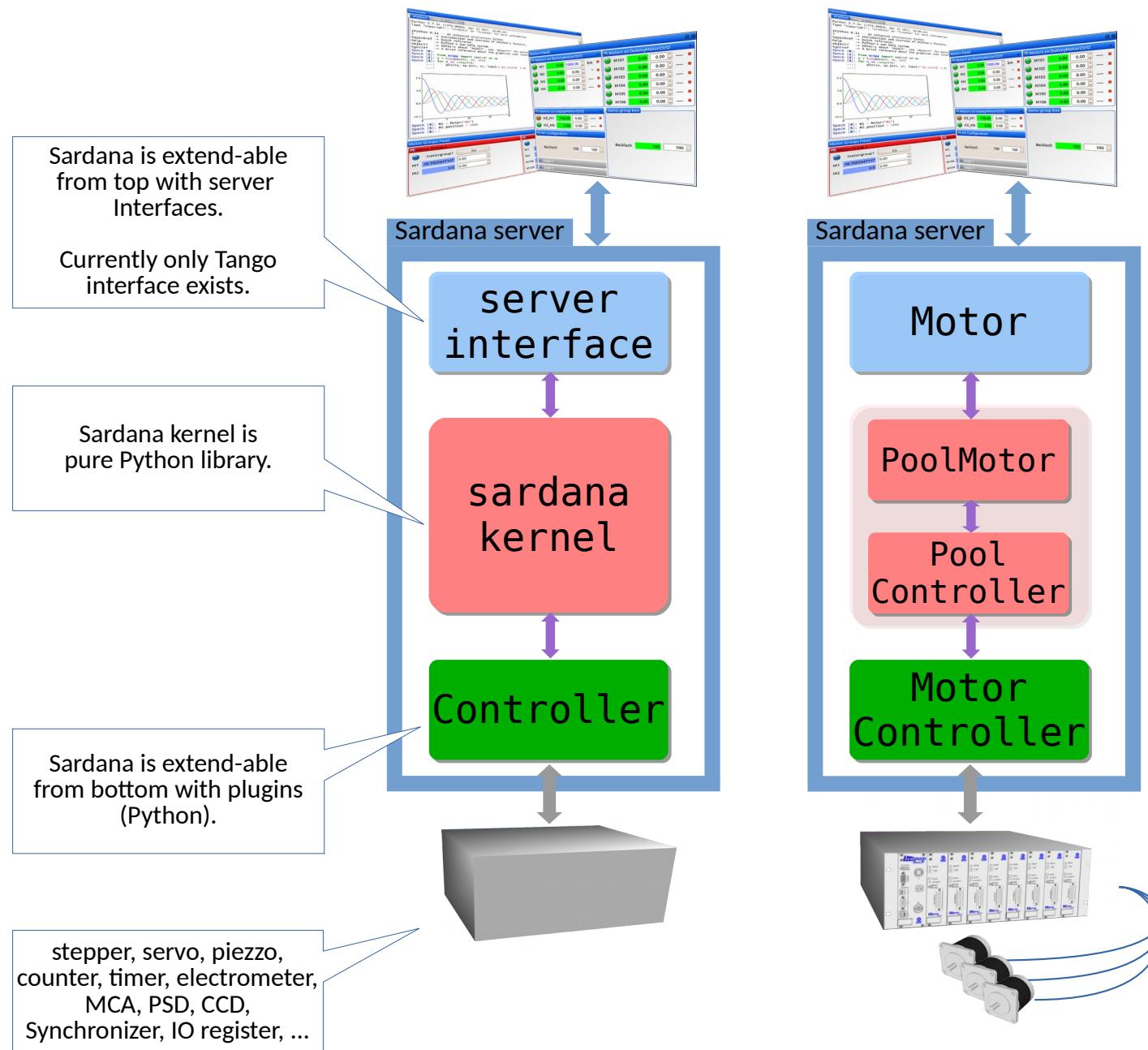
- The model objects are **singletons**
- Model names are **URIs**
- Each scheme provides:
  - A **model factory** for:
    - Authority
    - Device
    - Attribute
  - **Model name validators**



# Software layers - Controllers



# Software layers - Controllers



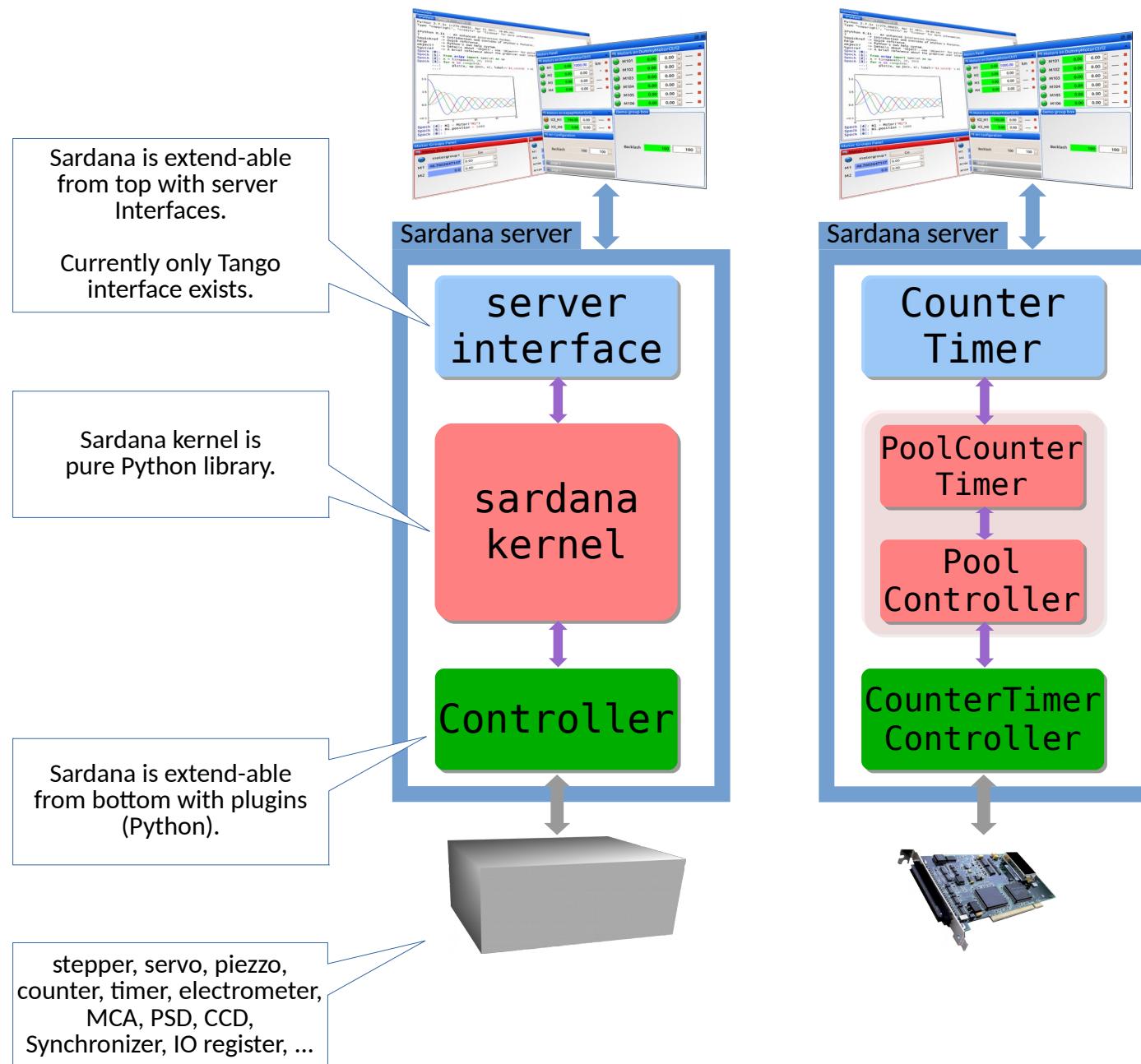
## Motor

```
position: float
state: enum
offset: float
sign: int
steps_per_unit
...
```

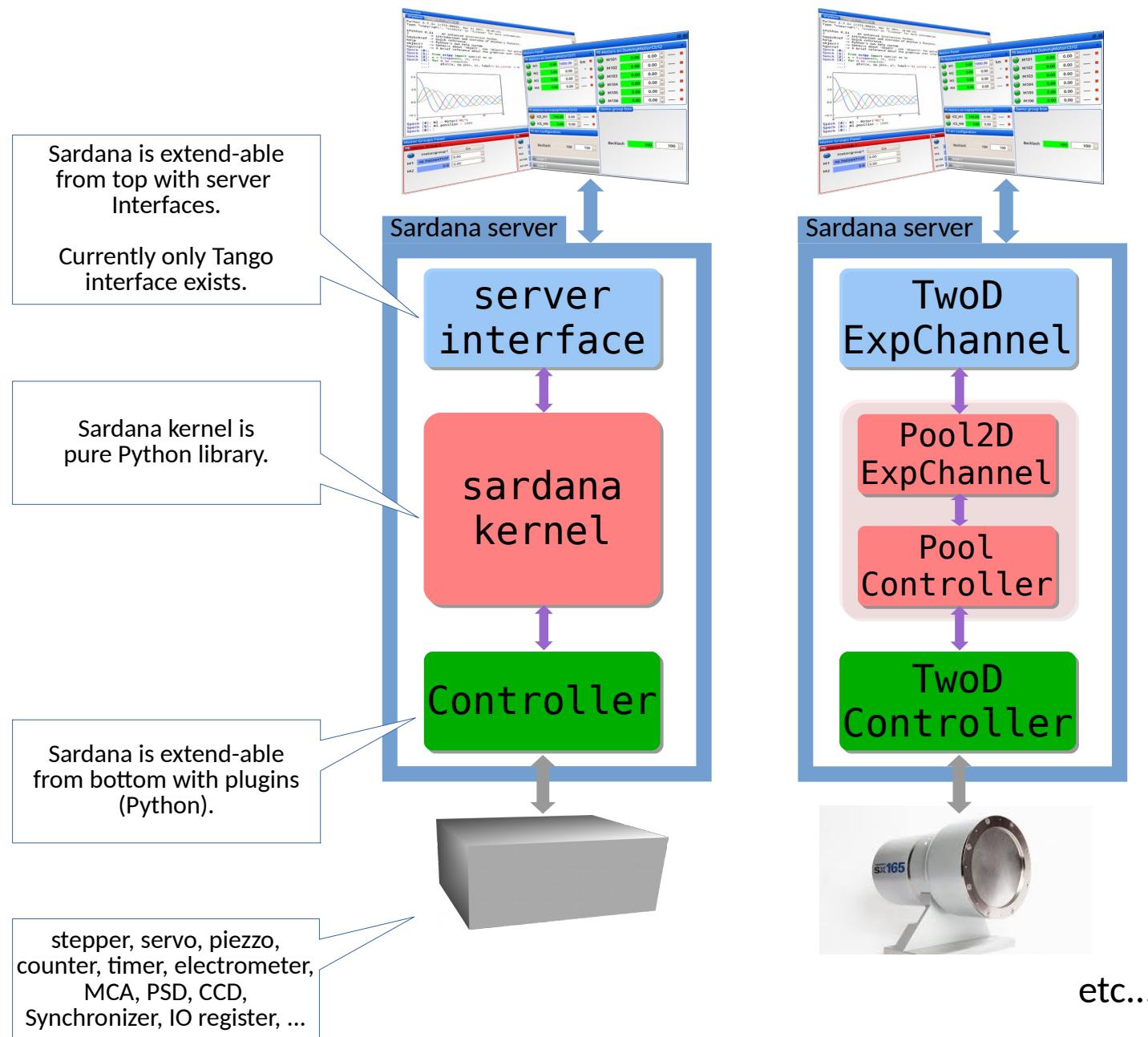
```
Stop()
Abort()
...
```

```
class MyMotorCtrl(MotorController):
    def StateOne(self, axis):
        [...]
    def ReadOne(self, axis):
        [...]
    def StartOne(self, axis, pos):
        [...]
    def AbortOne(self, axis):
        [...]
```

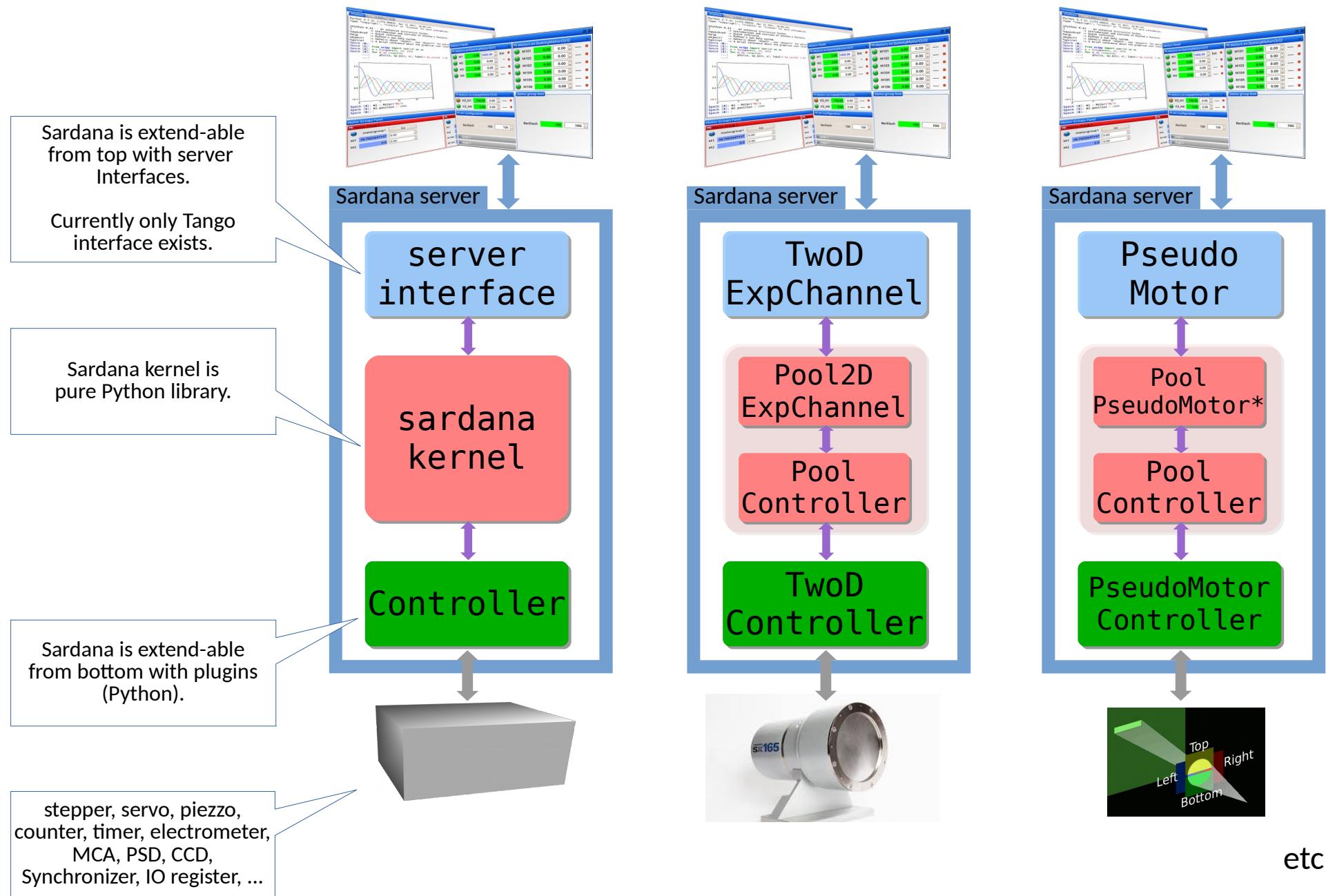
# Software layers



# Software layers - Controllers

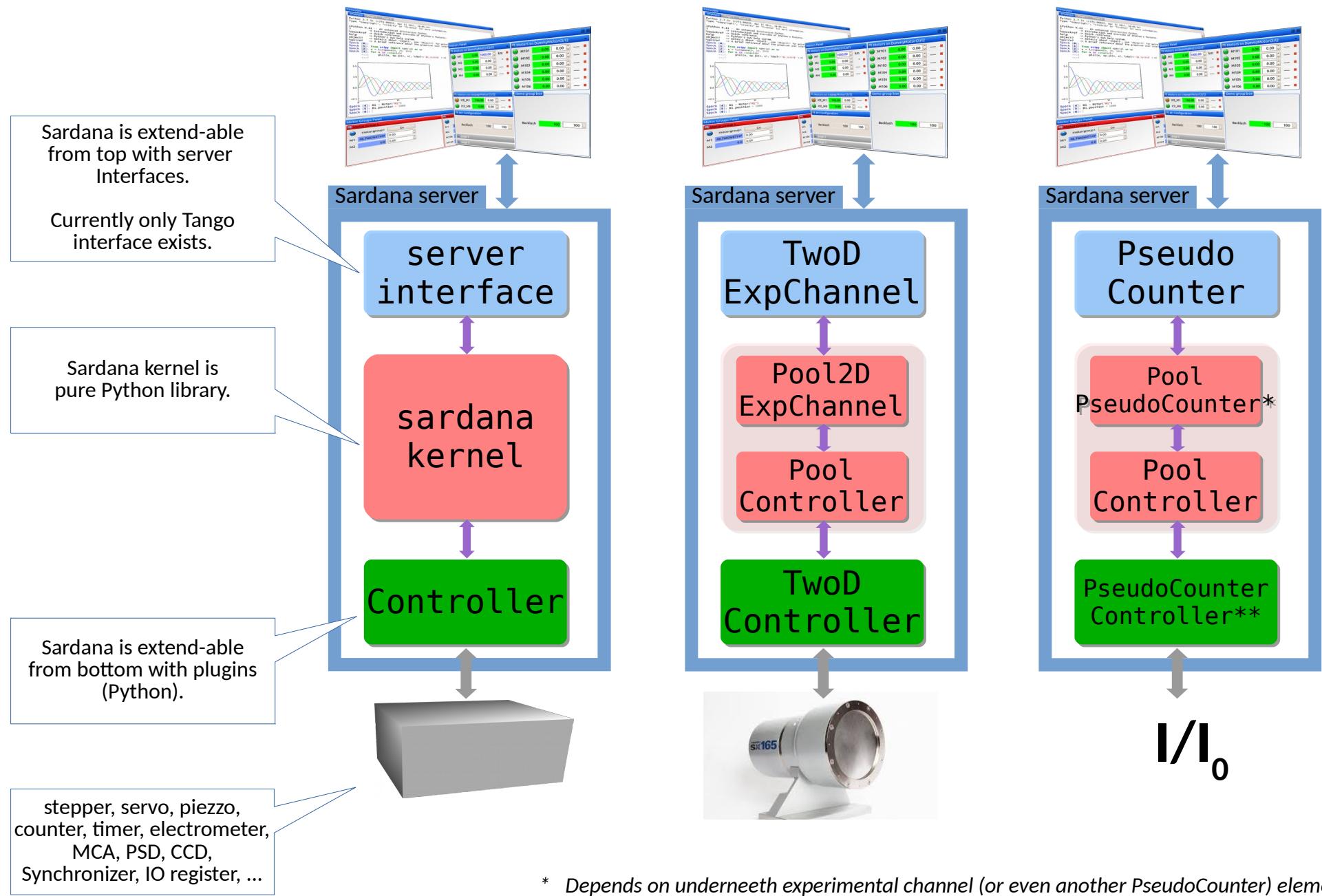


# Software layers - Controllers



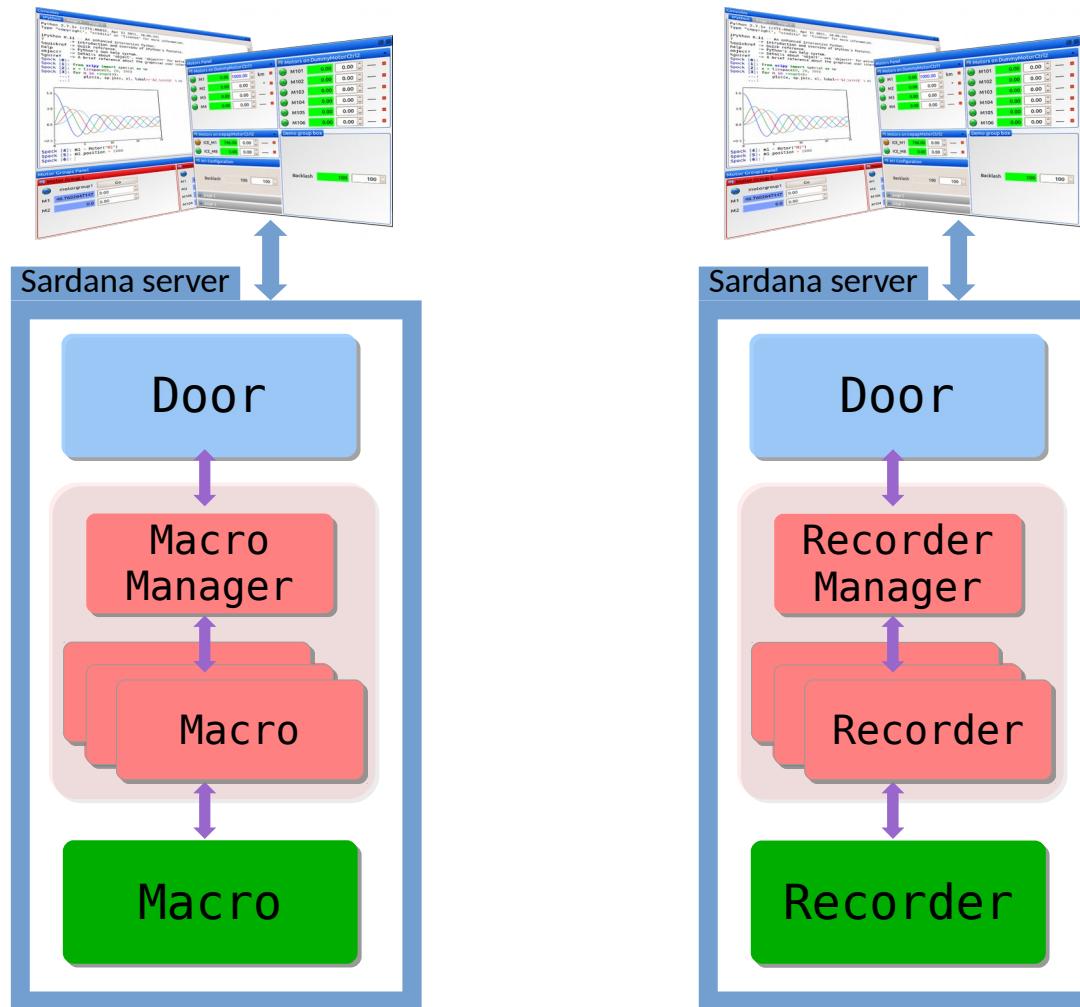
\* Depends on underneath PoolMotor (or even another PseudoMotor) elements

# Software layers - Controllers



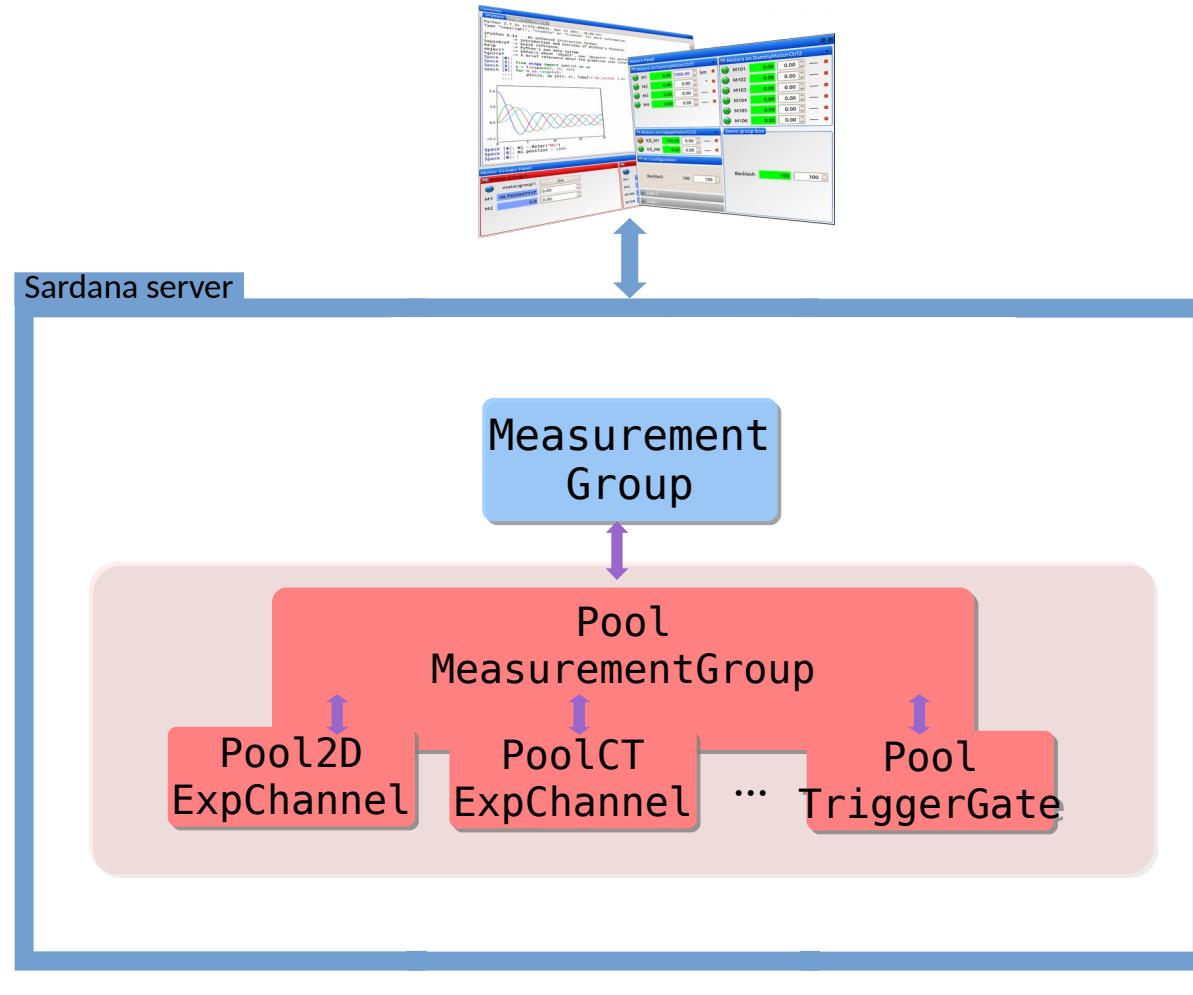
\* Depends on underneath experimental channel (or even another PseudoCounter) elements  
 \*\* Can be based on any type of experimental channel and provide any shape of data (0D, 1D or 2D)

# Software layers – Macros & Recorders



- Macros and Recorders are loaded in the Sardana server but **are not long-lived elements**.
- Managers are factories of Recorders or Macros and **create their instances on demand at runtime**.
- Macros may use Recorders to propagate the data to the destination.
- Door is an interface to execute Macros.

# Software layers - groups



- Hide the complexity of multi element actions e.g. acquisition or motion, behind a **common interface**.
- Ensure **synchronization** of multi element actions e.g. start of movement and **optimization** of multi element queries e.g. read motor positions.

## Other software related aspects...



Travis CI



AppVeyor

GitHub Pages



debian



All software layers have **automatic tests** (unit, integration or acceptance tests) run in CI (Travis CI & AppVeyor). We practice as much as possible TDD.

**Static code analysis** (flake8) and **documentation builds** are also part of the CI.

Sardana and Taurus form part of the **Debian releases**.

Sardana and Taurus **Docker images** are provided for newcomers.

# Sardana and Taurus Communities

# Sardana applications



## DESY:

- 14 beamlines (all in production)

## MAXIV:

- 3 beamlines + 8 beamlines in construction
- used in the Accelerator for motion control

## SOLARIS:

- 2 beamlines (both in production)
- planned to be used in the Accelerator

## Max Born Institute:

- 2 laboratory setups for time-resolved experiments studying magnetism using femtosecond light pulses (MOKE and XMCD)
- Sardana within Tango and EPICS environment!

## ESRF:

- used in the Accelerator - MacroServer only (control the automatic topup system; start/stop the equipments in order to send new electrons in the storage ring)

## LPS Orsay (Paris)

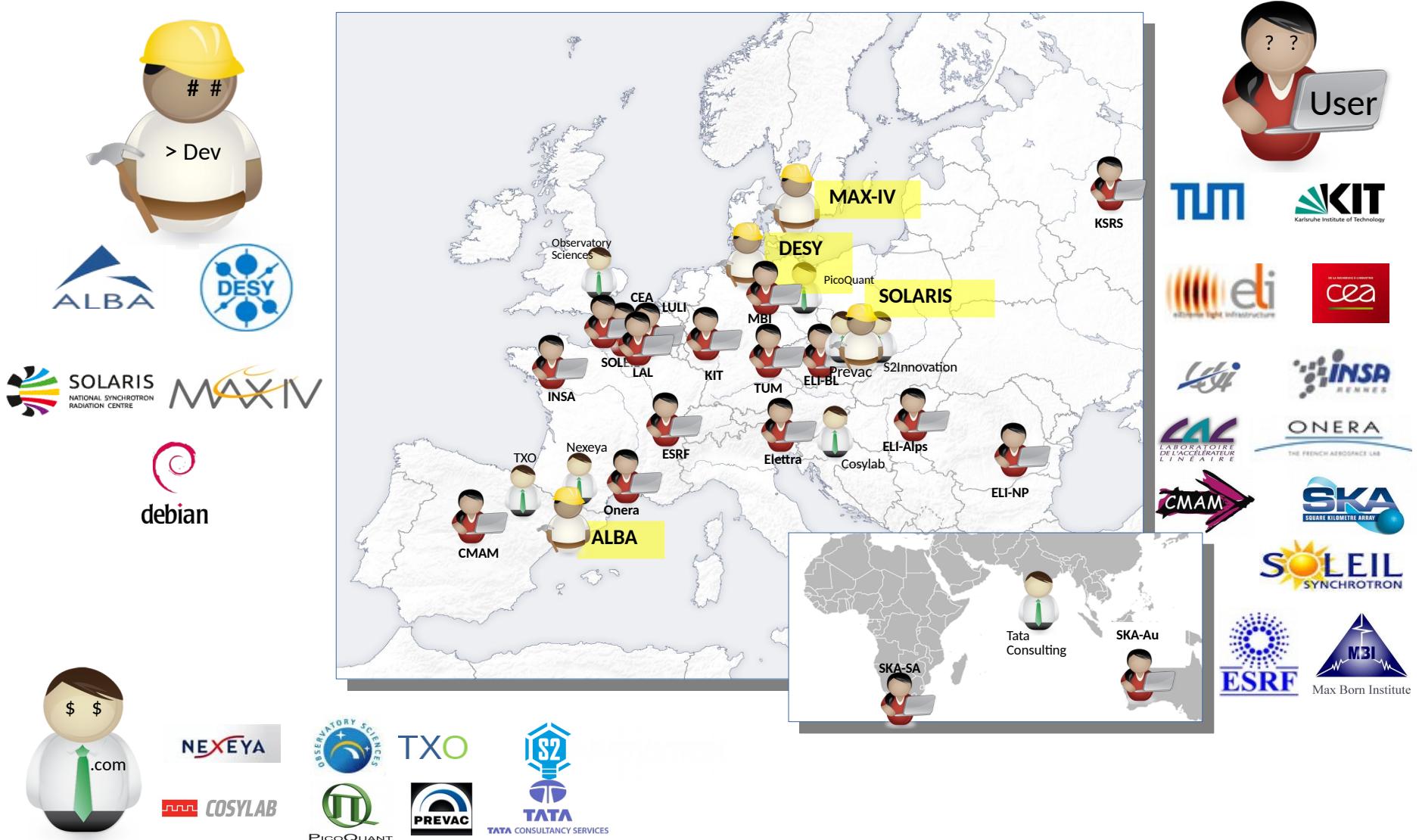
- control an Eulerian-4 circle diffractometer to perform scans

## ALBA:

- 8 beamlines (in production) + 4 more beamlines (planned/in construction)
- Optics and Material Science laboratories
- used in the Accelerator for: motion control, scans and operation



# Taurus Community



# Community – Tools & Events



**Docs:** <http://www.sardana-controls.org>  
<http://www.taurus-scada.org>

## GitHub

**Projects:** <https://github.com/sardana-org/sardana>  
<https://github.com/taurus-org/taurus>

## SEP      TEP

Sardana Enhancement    Taurus Enhancement  
Proposal                          Proposal

**SEP index:** <https://sardana-controls.org/sep/index.html>  
**TEP index:** <https://taurus-scada.org/tep/index.html>



**Videoconf:** <https://meet.jit.si/sardana>  
<https://github.com/sardana-org/sardana-followup>



**Plugins Catalogue:** <https://github.com/sardana-org/sardana-plugins>

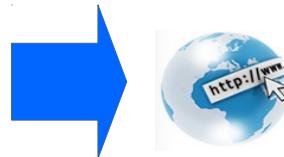


**Training and workshops:** <https://github.com/sardana-org/sardana-training>  
(satellite to Tango meetings or ICALEPCS conference)



**Announcements:** [sardana-users@lists.sourceforge.net](mailto:sardana-users@lists.sourceforge.net)  
[tauruslib-users@lists.sourceforge.net](mailto:tauruslib-users@lists.sourceforge.net)  
**Coordination:** [sardana-devel@lists.sourceforge.net](mailto:sardana-devel@lists.sourceforge.net)  
[tauruslib-devel@lists.sourceforge.net](mailto:tauruslib-devel@lists.sourceforge.net)

# Community – Tools & Events

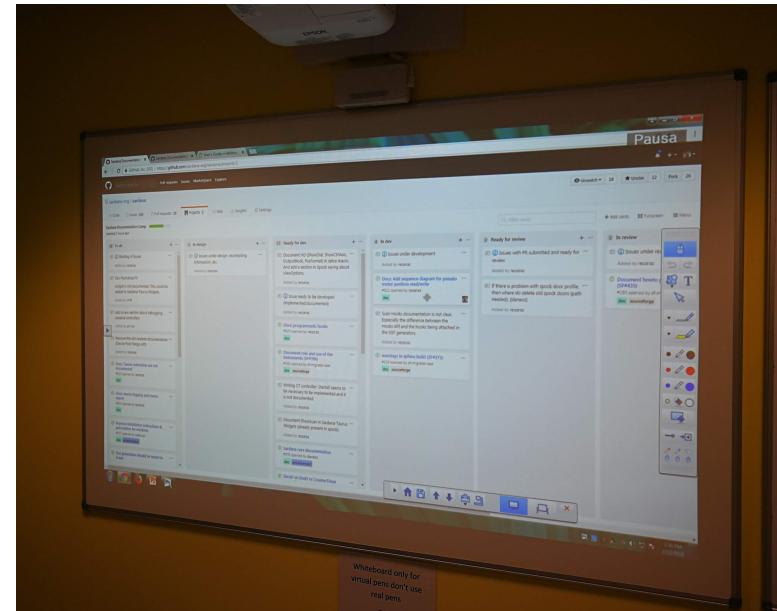


Docs: <http://www.sardana-controls.org>  
<http://www.taurus-scada.org>

## GitHub

## SEP      TEP

Sardana Enhancement    Taurus Enhancement  
Proposal                          Proposal



# Community – Tools & Events



**Projects:** <https://github.com/sardana-org/sardana>  
<https://github.com/taurus-org/taurus>

**SEP    TEP**  
 Sardana Enhancement    Taurus Enhancement  
 Proposal                      Proposal



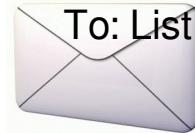


# Community – Tools & Events



**SEP    TEP**  
 Sardana Enhancement    Taurus Enhancement  
 Proposal                      Proposal

**SEP index:** <https://sardana-controls.org/sep/index.html>  
**TEP index:** <https://taurus-scada.org/tep/index.html>



## Sardana Enhancement Proposals

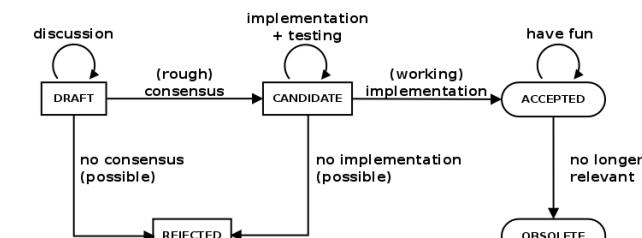
This is the main index of the Sardana Enhancement Proposals (SEP).

Each proposal should be in a separate file and be linked in the following table

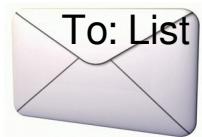
### Proposals list

Link	status	Title
<a href="#">SEP0</a>	OBSOLETE	Introducing Sardana Enhancement Proposal
<a href="#">SEP1</a>	OBSOLETE	Reorganization of code repos
<span style="color: blue;">i</span> <a href="#">SEP2</a>	CANDIDATE	Improve integration of 1D and 2D experimental channels
<span style="color: blue;">i</span> <a href="#">SEP3</a>	REJECTED (handled in #297)	Adapt to <a href="#">TEP3</a> (Tango-independent taurus.core)
<span style="color: blue;">i</span> <a href="#">SEP4</a>	ACCEPTED	HKL integration
<span style="color: blue;">i</span> <a href="#">SEP5</a>	ACCEPTED	Implementation of tests infrastructure
<span style="color: blue;">i</span> <a href="#">SEP6</a>	ACCEPTED	Continuous Scan Implementation
<a href="#">SEP7</a>	OBSOLETE	Code contribution workflow
<a href="#">SEP8</a>	CANDIDATE	Remove from Taurus objects t
<a href="#">SEP9</a>	ACCEPTED	Compact Read+Write widget:
<a href="#">SEP10</a>	OBSOLETE	Taurus separation
<a href="#">SEP11</a>	ACCEPTED	Direct load of .ui files
<a href="#">SEP12</a>	CANDIDATE	Use python Enum instead of t
<a href="#">SEP13</a>	REJECTED (moved to <a href="#">TEP13</a> )	Unified plugins support in Tau
<a href="#">SEP14</a>	DRAFT	MSENV taurus schema
<a href="#">SEP15</a>	ACCEPTED	Moving Sardana to Github
<span style="color: blue;">i</span> <a href="#">SEP16</a>	DRAFT	Plugins (controllers, macros, etc.) register
<a href="#">SEP17</a>	DRAFT	Ongoing acquisition formalization and implementation
<span style="color: blue;">i</span> <a href="#">SEP18</a>	ACCEPTED	Extend acquisition and synchronization concepts for SEP2 needs

### Proposal states



# Community – Tools & Events



**Videoconf:** <https://meet.jit.si/sardana>

<https://github.com/sardana-org/sardana-followup>

sardana-org / sardana-followup

Code Issues 9 Pull requests 1 Projects 0 Wiki Insights Settings

Unwatch 9 Unstar 2 Fork 6

Minutes and other stuff gathered during the Sardana project follow-up meetings Edit

Manage topics

162 commits 3 branches 0 releases 8 contributors

Branch: master New pull request Create new file Upload files Find file Clone or download

Commit	Description	Date
reszelaz Merge pull request #37 from reszelaz/20190110-minutes	Latest commit 4f3029a 5 days ago	
20150610-DESY	Upload Talks DESY15 WS	7 months ago
20171102	Formatting changes	a year ago
20171211	Adding tasks	a year ago
20180125	Merge pull request #7 from amilan/2018_01_25_meeting	11 months ago
20180405	corrected minor mistakes	10 months ago
20180517	Alba presentation	4 months ago
20180605-Prague	Removing moved files	7 months ago
20180619	Add sardana docs presentation	7 months ago
20180710-Barcelona	Update DocsStyleGuide.md	6 months ago
20180913	Fix typos	4 months ago
20181004	update agenda to reflect points covered during meeting	3 months ago
20181108	Error from ctrl-C at DESY	2 months ago
20181213	Applying Zibi suggestions	a month ago
20190110	Add minutes	6 days ago
README.md	Initial commit	a year ago

# Community – Tools & Events



**Plugins Catalogue:** <https://github.com/sardana-org/sardana-plugins>



**sardana-org / sardana-plugins**

[Code](#) [Issues 0](#) [Pull requests 0](#) [Projects 0](#) [Wiki](#) [Security](#) [Insights](#) [Settings](#)

Branch: master [sardana-plugins / hardware.md](#)

 reszelaz Fix typo in hardware category 66e5a73 on Aug 8

1 contributor

14 lines (11 sloc) | 928 Bytes

[Raw](#) [Blame](#) [History](#)

---

## Sardana plugins for specific hardware

Below you will find a table with Sardana plugins for specific hardware like for example motion controllers, detectors, etc.

Name	Description	Link(s) to project
ALBA Em Electrometer	Low current electrometer	<a href="#">sardana-albaem</a>
AdLink	AdLink DAQ cards e.g. 2005	<a href="#">sardana-adlink</a>
IcePAP	IcePAP motion controller	<a href="#">sardana-icepap</a>
Linkam	Temperature controller	<a href="#">sardana-linkam</a>
LoCuM	Low current monitor	<a href="#">sardana-locum</a>
OPUS	Bruker OPUS spectrometer	<a href="#">sardana-opus</a>

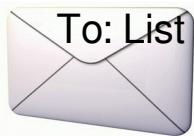
# Community – Tools & Events



Training and workshops: <https://github.com/sardana-org/sardana-training>  
(satellite to Tango meetings or ICALEPCS conference)



# Community – Tools & Events



**Announcements:** [sardana-users@lists.sourceforge.net](mailto:sardana-users@lists.sourceforge.net)  
[tauruslib-users@lists.sourceforge.net](mailto:tauruslib-users@lists.sourceforge.net)

**Coordination:** [sardana-devel@lists.sourceforge.net](mailto:sardana-devel@lists.sourceforge.net)  
[tauruslib-devel@lists.sourceforge.net](mailto:tauruslib-devel@lists.sourceforge.net)

## Questions?

... you can also reach us at any time!

### Acknowledgements:

- DESY: Teresa Nuñez, Jan Kotański, Tim Schoft, Thorsten Kracht
- SOLARIS: Grzegorz Kowalski, Stanisław Cabała, Ireneusz Zadworny, Michał Fałowski
- MAXIV: Antonio Milan, Juliano Murari, Paul Bell, Antoine Dupré, Miriam Lindberg, Vincent Hardion
- SOLEIL: Frédéric Picca
- Max Born Institute: Daniel Schick
- ALBA: all the Computing Division