Sardana

Software for building SCADAS in scientific environments

Tiago Coutinho - ALBA's controls team
Do you want to know what it is?

Morpheus: Do you believe in fate, Neo? Why not?

Neo: Because I'm not in control of my laboratory

Morpheus: [...] do you know what I am talking about?

Neo: The Sardana

Morpheus: Do you want to know what it is?

Neo: Yes!

Morpheus: Sardana is everywhere. It is all around us (in the lab). You can see it when you move a motor, when you monitor a temperature, when you do an energy scan.

A software package for instrument control and data acquisition
Generic Lab GUI

- Fully customizable
- Auto discovery
- Extensive widget catalog
- Drag-n-drop between applications
1. At runtime
2. Designer

3. Write your own code

```python
from taurus.qt.QtGui.panel import TaurusForm

panel = TaurusForm()
props = ['state', 'status', 'position', 'velocity', 'acceleration']
model = ['sys/taurustest/1/%s' % p for p in props]
panel.setModel(model)
```
SPEC like CLI

- IPython based
- Context sensitive command completion
- Command history
- Help system
- Error handling
Free, open source (LGPL)

- Python oriented
- Distributed
- Extensible
- Scalable
- Tango based
- Event based
- Set of tools

Morpheus: You take the red pill, you stay in Wonderland, and I show you how deep the rabbit hole goes
Where do you intervene

Controller Plug-in

```python
from sardana import State
from sardana.pool import MotorController

class NSC200Controller(MotorController):
    """This class is the Tango Sardana motor controller for the Newport NewStep handheld motion controller""

ctrl_properties = {
    'SerialCh': {'Type': str, 'Desc': '[...]'},
    'SwitchBox': {'Type': bool, 'Desc': '[...]'},
}

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

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

def StateOne(self, axis):
    if nsc200.is_moving(axis):
        return State.Moving, 'Motor in motion'
    return State.On, 'Motor is stopped'

def ReadOne(self, axis):
    return nsc200.motor_position(axis)

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

def StartAll(self):
    nsc200.send_move_command(self.pos)

def AbortOne(self, axis):
    nsc200.send_abort_command(axis)
```

Macro Plug-in

```python
from sardana.macroserver import Macro, Type

class move(Macro):
    """moves a motor to the specified absolute position.""

param_def = [
    ['motor', Type.Motor, None, 'motor to move'],
    ['pos', Type.Float, None, 'absolute position']
]

self.debug('Moving %s to %s', motor, pos)
motor.move(pos)
self.info('Ended moving %s to %s', motor, pos)
```

Device Pool

```python
from sardana import State
from sardana.pool import MotorController

class NSC200Controller(MotorController):
    """This class is the Tango Sardana motor controller for the Newport NewStep handheld motion controller""

ctrl_properties = {
    'SerialCh': {'Type': str, 'Desc': '[...]'},
    'SwitchBox': {'Type': bool, 'Desc': '[...]'},
}

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

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

def StateOne(self, axis):
    if nsc200.is_moving(axis):
        return
    return State.On, 'Motor is stopped'

def ReadOne(self, axis):
    return nsc200.motor_position(axis)

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

def StartAll(self):
    nsc200.send_move_command(self.pos)

def AbortOne(self, axis):
    nsc200.send_abort_command(axis)
```

GUI / CLI

```python
from taurus.qt.qtgui.application import TaurusApplication
from taurus.qt.qtgui.panel import TaurusForm

gui, panel = TaurusApplication(), TaurusForm()
attrs = 'state', 'status', 'position', 'velocity'
model = [ 'BL00/ENERGY/1/%s' % a for a in attrs ]
panel.model = model
panel.show()
gui.exec_()
```

```
Neo: [...] I can dodge bullets?
Morpheus: [...] when you're ready, you won't have to.
```
Morpheus: We have only bits and pieces of information but [...] at some point in the early twenty-first century all of mankind was united in celebration. We marveled at our own magnificence as we gave birth to... *Sardana!*
Why?

- Fully customizable/programmable GUI framework
- Familiar looking CLI
- Python oriented
- Free, open source

Distributed C-S arch
- Scalable
- Multi-threaded

- Tango, Qt, Python
- Fairly documented

Icalepcs 2011
Morpheus: Unfortunately, no one can be told what the Sardana is. You have to see it for yourself:

Acknowledgment

- ALBA's computing division
- ELETTRA computing group
- ESRF Tango & BLISS groups
- FRM II computing group
- HASYLAB computing group
- MAX-lab computing group
- SOLEIL computing group