

# QtSpock

## Spock in a Qt widget

Tim Schoof

Hamburg, 03.06.2019

- > Spock in a Jupyter QtConsole widget
- > Most of Spock functionality supported
- > Executing commands does not block the Qt application
- > QtConsole features include
  - Proper multi-line editing
  - Inline figures
  - Graphical calltips

```
qtspock.py

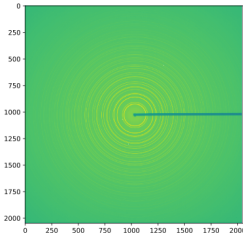
Jupyter QtConsole 4.2.1
Spock 2.7.2.2 -- An interactive laboratory application.

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

IPython profile: spockdoor

p02/door/haso232s.01 [1]: ascan ehla_mot01 0.0 10.0 10 1.0
ScanDir is not defined. This operation will not be stored persistently. Use Use
"expcnf" (or "senv ScanDir <abs directory>") to enable it
Scan #27 started at Wed May 29 17:42:44 2019. It will take at least
0:00:18.189000
#Pt No    ehla_mot01    ct05    ct06    ct07    ct08    dt
0      0            1        2        3        4      2.59801
1      1            1        2        3        4      3.90757
2      2            1        2        3        4      5.2213
3      3            1        2        3        4      6.53252
4      4            1        2        3        4      7.83733
5      5            1        2        3        4      9.15842
6      6            1        2        3        4     10.4592
7      7            1        2        3        4     11.7827
8      8            1        2        3        4     13.0834
9      9            1        2        3        4     14.3962
10     10           1        2        3        4     15.7101
Scan #27 ended at Wed May 29 17:43:01 2019, taking 0:00:16.756201. Dead time
34.4% (motion dead time 31.0%)

p02/door/haso232s.01 [2]: showcam perkin_elmer
Result [2]: <matplotlib.image.AxesImage at 0x7f76ae141650>
```



```
Signature: ehla_mot01.state(*args, **kwargs)
Docstring:
state(self) -> DevState

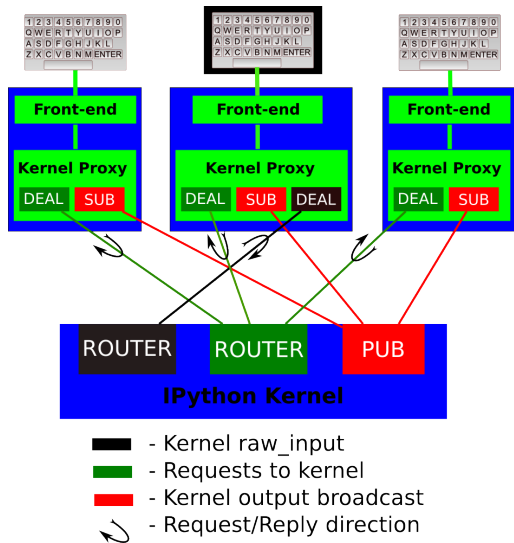
    A method which returns the state of the device.

Parameters : None
Return      : (DevState) constant
Example :
    dev_st = dev.state()
    if dev_st == DevState.ON : ...
File:       /usr/lib/python2.7/dist-packages/tango/green.
Type:       instancemethod
```

```
p02/door/haso232s.01 [3]: ehla_mot01.state(
```

# Implementation Details

- > Jupyter server-client architecture
- > Kernel subprocess started on frontend initialization
- > IPython kernel loads Spock profile and extensions
- > Communication via ZeroMQ
- > No shared namespace
- > Kernel can be (auto)restarted
- > Door name obtained from kernel on every restart
- > `shutdown_kernel` must be called on exit



<https://jupyter-client.readthedocs.io/en/latest/messaging.html#messaging>

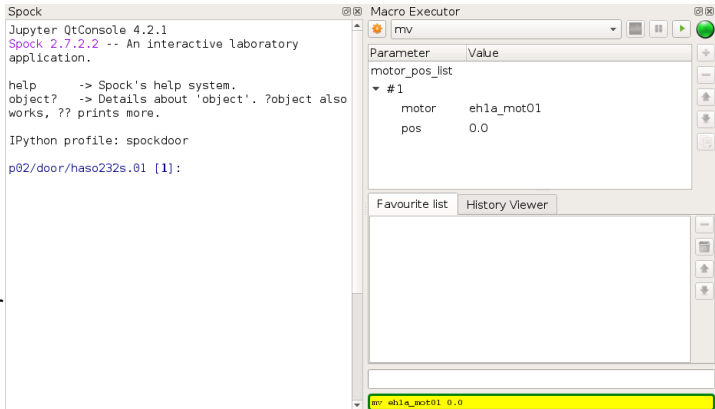
[//jupyter-client.readthedocs.io/en/latest/messaging.html#messaging](https://jupyter-client.readthedocs.io/en/latest/messaging.html#messaging)

# Open Questions and Missing Features

- > Creating/updating Spock profile
- > edmac not working, other macros need testing
- > Input dialogs (`SPOCK_INPUT_HANDLER == "Qt"`) crash application
- > Integration with Taurus/Sardana
- > Setting macroserver/door name on frontend side
- > ?

# Integration with Taurus Macro Executor

- > GUI support for
  - Users unfamiliar with CLI
  - Complex macros with repeat parameters
  - Favorite macros
- > TODO:
  - Show output in QtSpock
  - Enable Pause/Resume/Stop for macros in QtSpock
  - Synchronize history



# Thank you!

## Contact

**DESY.** Deutsches  
Elektronen-Synchrotron

[www.desy.de](http://www.desy.de)

Tim Schoof

FS-EC  
[tim.schoof@desy.de](mailto:tim.schoof@desy.de)