Jupyter & ROS

Wolf Vollprecht
QuantStack

Visualization
Scientific Computing
Open Source
C++ (xtensor, xsimd)
Interactive Computing
Robotics
Jupyter
What is Jupyter

- Interactive Computing environment
- Really successful in Data Science, Machine Learning ...
- Extensible
- Core: Notebook format
Open a CSV file using Pandas

```python
import pandas
df = pandas.read_csv('..//data/iris.csv')
df.head(20)
```

```
    sepal_length  sepal_width  petal_length  petal_width  species
     0       5.1          3.5         1.4          0.2     setosa
     1       4.9          3.0         1.4          0.2     setosa
     2       4.7          3.2         1.3          0.2     setosa
     3       4.6          3.1         1.5          0.2     setosa
     4       5.0          3.6         1.4          0.2     setosa
     5       5.4          3.9         1.7          0.4     setosa
     6       4.6          3.4         1.4          0.3     setosa
     7       5.0          3.4         1.5          0.2     setosa
     8       4.4          2.9         1.4          0.2     setosa
     9       4.9          3.1         1.5          0.1     setosa
    10       5.4          3.7         1.5          0.2     setosa
    11       4.8          3.4         1.6          0.2     setosa
    12       4.8          3.4         1.4          0.1     setosa
    13       4.3          3.0         1.1          0.1     setosa
    14       5.8          4.0         1.2          0.2     setosa
```
LIVE DEMO

Never do a live demo
— Maarten Breddels
ROS & Jupyter

- Interact with ROS through the notebook
- Problem #1: multi-threading of rospy
- Ad-hoc widgets for sending & receiving messages
- Widgets for 3D visualization of robot state
Laser Scan demo

Make sure that the websocket rosbag is running on your host computer, and that you publish a laser scan on the `/scan` topic.

```python
from sidecar import Sidecar
from jupyteros import ros3d

v = ros3d.Viewer()
rc = ros3d.ROSCconnection()
tf_client = ros3d.TFClient(ros=rc, fixed_frame='')

sc = Sidecar(title='Sidecar Output')
with sc:
    display(v)

v.height = '1000px'

laser_view = ros3d.LaserScan(topic=''/scan', ros=rc, tf_client=tf_client)

g = ros3d.GridModel()

v.objects = [g, laser_view]
g.color = '#CCC'
```
from jupyros import ros3d

v = ros3d.Viewer()
rc = ros3d.ROSConnection()
tf_client = ros3d.TFClient(ros=rc, fixed_frame='')

path_view = ros3d.Path(ros=rc, tf_client=tf_client, topic="/test_optim_node/local_plan")
rapyuta.io

- Cloud service to reliably connect robots to infinite compute & do fleet management
- Run JupyterLab as Robot ⇥ Cloud interface
Links

https://github.com/RoboStack/jupyter-ros

https://github.com/RoboStack/jupyterlab-zethus

https://github.com/voila-dashboards/voila

https://medium.com/@wolfv/robot-development-with-jupyter-ddae16d4e688

https://medium.com/@wolfv/ros-on-conda-forge-dca6827ac4b6
Thank You

GITHUB
@wolfv

TWITTER
@wuoulf

EMAIL
w.vollprecht@gmail.com

WEB
quantstack.net