

The Py3DViewer project:

a Python library for fast prototyping in geometry processing

G. Cherchi, L. Pitzalis, **G. L. Frongia** & R. Scateni

Py3DViewer

Python library for fast prototyping in Geometry Processing

Manipulation and Visualization of:

- Surface meshes
- Volume meshes
- Skeletons
- Others (asap)

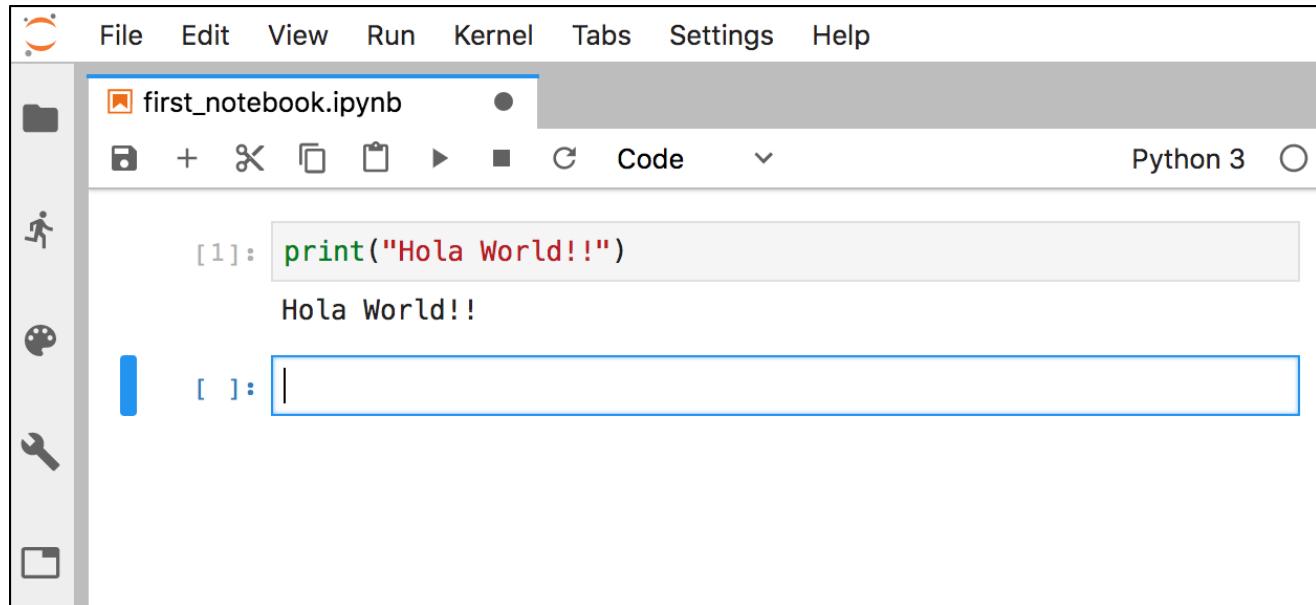
```
from Py3DViewer import Trimesh  
m = Trimesh('goyle.obj')  
m.show()
```



Jupyter environment

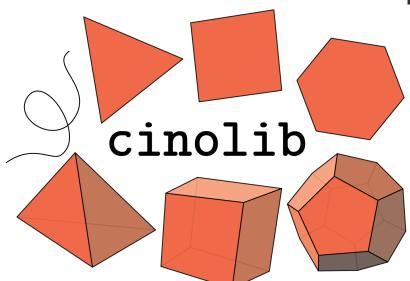
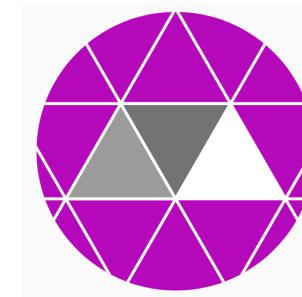
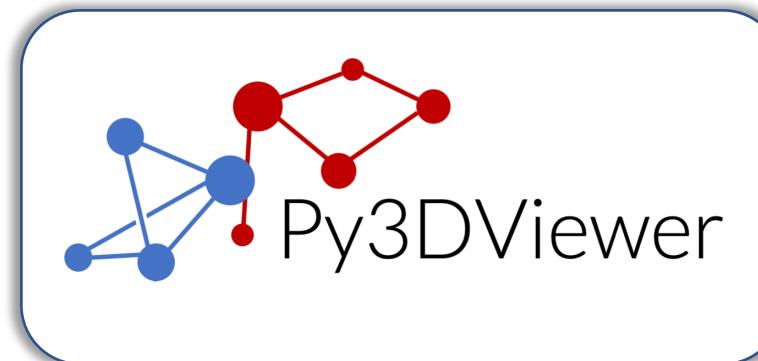
Jupyter Lab

- Interactive Python IDE in a browser



You can easily install jupyter: jupyter.org/install.html

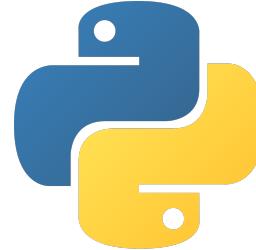
Why Py3DViewer?



C++ vs Python

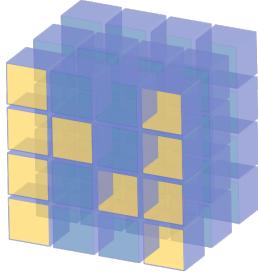


- Verbose but fast
- Efficient loops
- Not ideal for prototyping
- Troublesome library installation



- Compact but slow,
“interpreted pseudo code”
- Efficient tensorial operations
- Ideal for prototyping
- Easy package manager (pip)

Python... but fast?

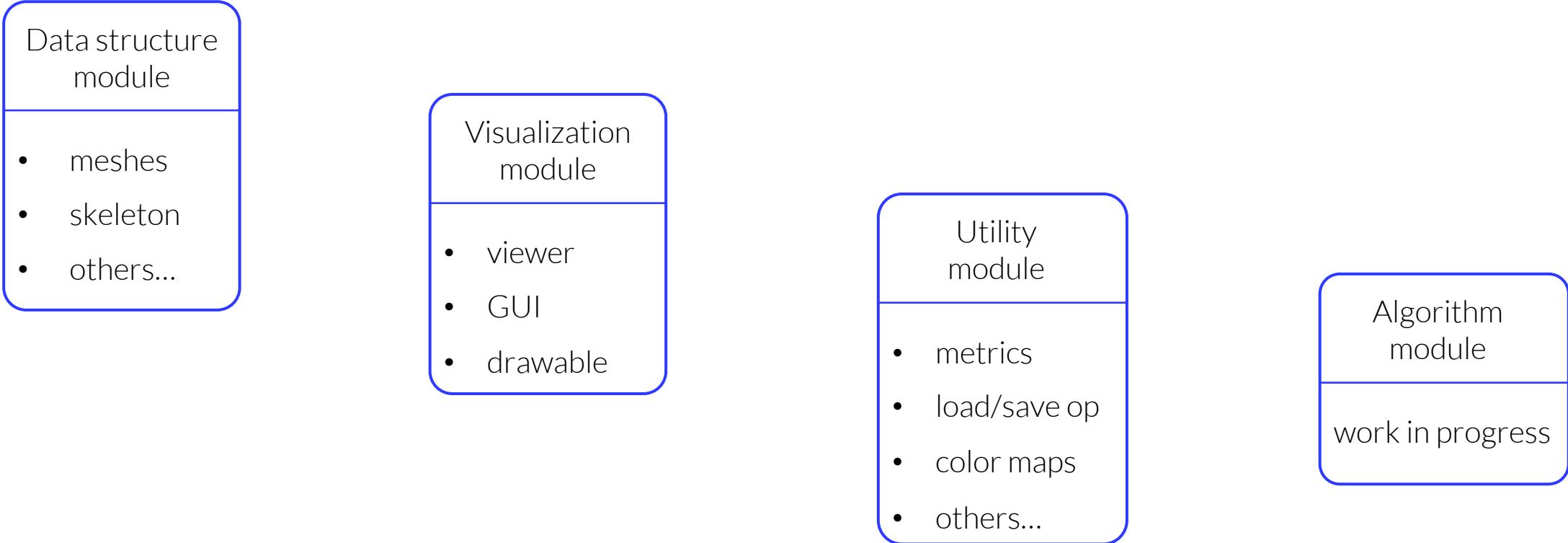


NumPy

 Numba

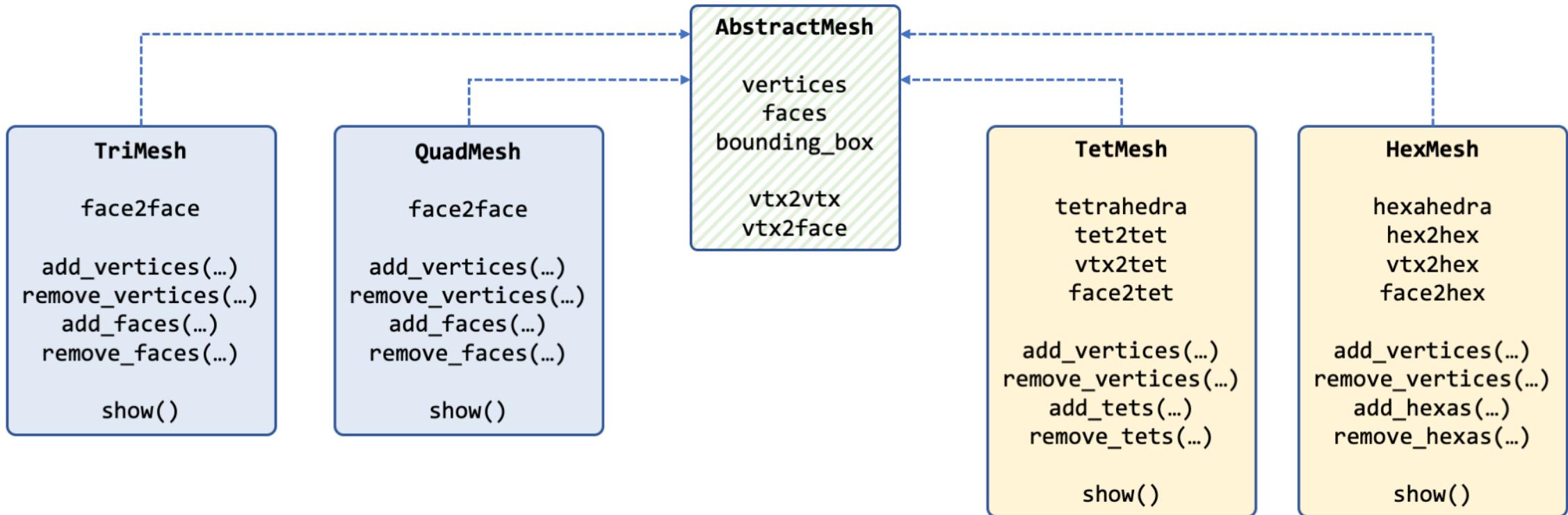
 SciPy

The library structure

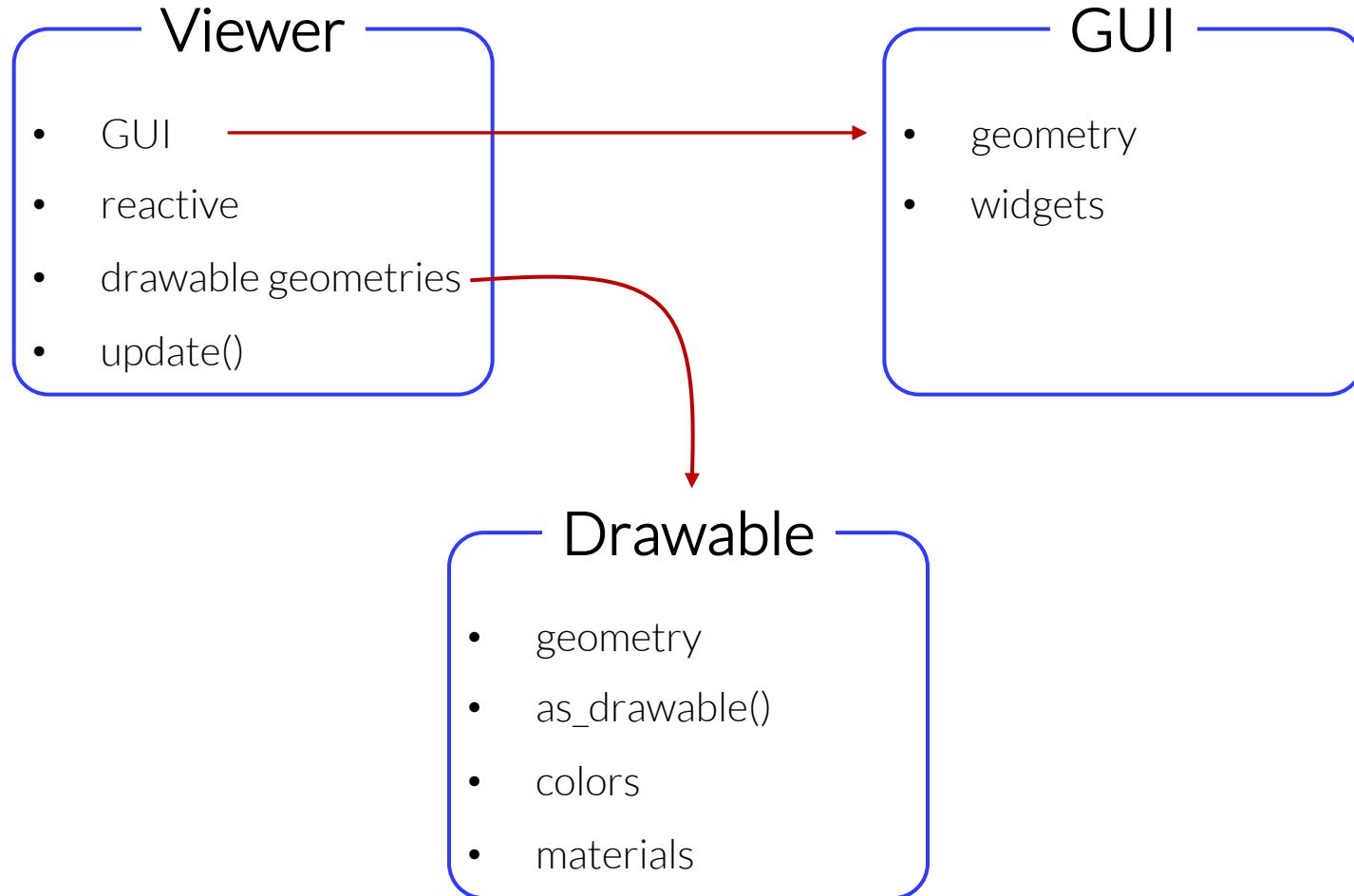


Data structures module

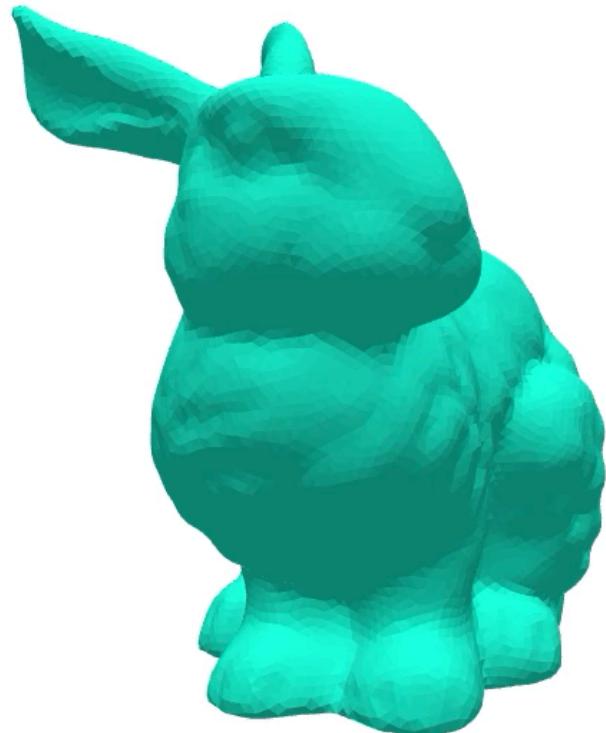
Index-based storage



Visualization module

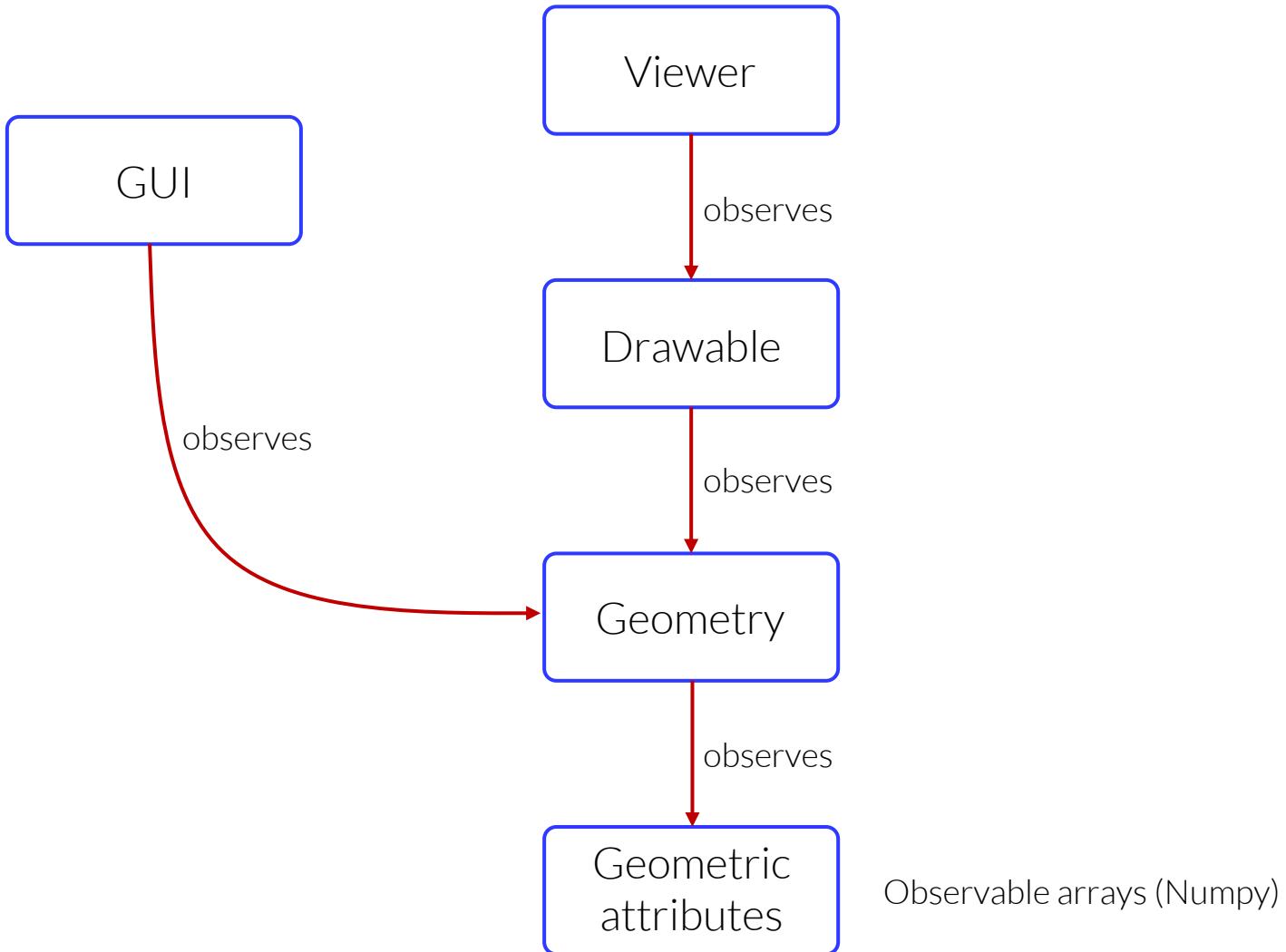


Responsive viewer

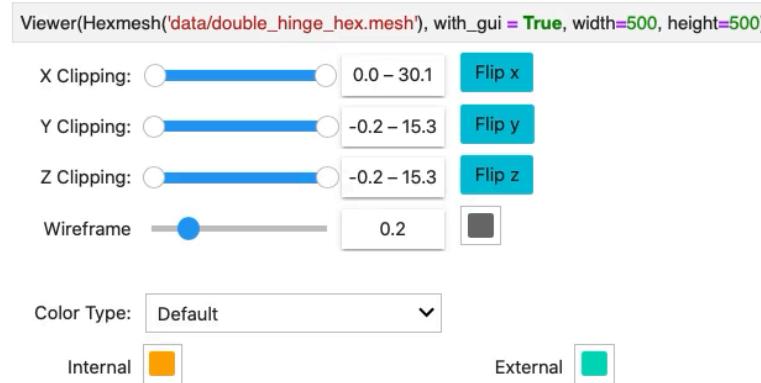


```
while True:  
    rot = Rotation.from_euler('y', .5, degrees=True)  
    mesh.vertices = rot.apply(mesh.vertices)  
    sleep(1/20)
```

Reactive dataflow



The GUI



Algorithms module

Work in progress module

At least one for the main GP problems like:

- Smoothing
- Hole filling
- Triangulation
- Decimation
- and so on...



Active development

We developed and improved a lot from the paper submission...

Model	#Vertices	#Simplices	Time [Liv19]	Time ours (paper)	Time ours (now)
tri-mesh	14.2k	25.5k	0.25s	1.05s	0.35s
quad-mesh	9.9k	9.6k	0.11s	0.49s	0.15s
tet-mesh	4.6k	15.4k	0.52s	0.58s	0.32s
hex-mesh	18.9k	15.2k	0.61s	0.61s	0.58s

Loading times (with adjacencies)

Let's code!

tinyurl.com/py3dviewer

How to install and use Py3DViewer

- GitHub repo:
`github.com/cg3hci/py3DViewer`
- You can easily install Py3DViewer by using pip:
`pip install git+https://github.com/cg3hci/py3DViewer`
- Documentation and examples available on GitHub or:
`py3dviewer.readthedocs.io/en/latest`
- Interactive example on binder:
`tinyurl.com/py3dviewer`

Thanks
for your attention

github.com/cg3hci/py3DViewer

