

hips(py) and ipyaladin, 2 new Python tools for the HiPS ecosystem

Thomas Boch, Christopher Deil, Adeel Ahmad, Pierre Fernique, Jérôme Desroziers

IVOA Interop, Santiago Chile, Apps1



hips(py)



- **A HiPS client for Python**
- Developed during GSoC 2017
(under the umbrella of OpenAstronomy organization)



Google
Summer of Code

- Student



● Adeel Ahmad

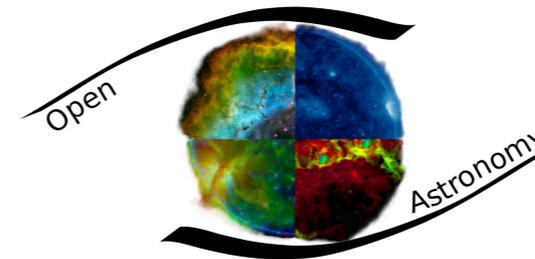
- Mentors



● Christoph Deil (MPIK, Heidelberg)



● Thomas Boch (CDS)



- ***Design and create a Python client for Hierarchical Progressive Surveys (HiPS). The package will enable users to view astronomical figures in an interactive environment. Currently, similar clients exist, such as Aladin and Aladin Lite, but they are written in Java and JavaScript, respectively. The goal of this project is to provide similar functionality using Python.***

□ hips(py): v0.1



- Released on July 28, 2017
- Announced on *astropy-dev* mailing-list
- Feature
 - *"At the moment, the Python hips package supports **fetching** and **drawing HiPS image tiles** into a **sky image** of a **geometry** (a **WCS projection** and **shape**) of your choosing. »*
- Demo

□ Using hips(py)



- Installation
 - Requires Python ≥ 3.6
 - `pip install hip`
- Documentation
hips.readthedocs.org
- Notebooks
github.com/hips/py/hips-extra/tree/master/notebooks

□ Improving hips(py)



- Roadmap for v0.2
 - Implement HiPS **tile caching** mechanism
 - Improve tiles **drawing speed**
 - Test other drawing algorithms
(current one uses projective transformation)
 - Switch from healpy to **astropy-healpix** BSD-licensed **HEALPix library**
- Eventual goal
 - Become an **astropy-affiliated** package
- Contributing
 - Source code on github: github.com/hips/py

- **A Jupyter widget for Aladin Lite**

- **Features**

- Easy integration of Aladin Lite in Python notebooks
- Control of field of view (target, zoom level, HiPS to display)
- Linked views
- Overlay VOTable, Astropy Tables, MOCs
- Register callbacks triggered by action in widget view

- **Demonstration**

```
In [4]: import ipyaladin.aladin_widget as ipyal

In [5]: aladin = ipyal.Aladin(target='orion', fov=4)
        aladin
```



□ ipyaladin development



- Structure based on *widget-cookiecutter* template
github.com/jupyter-widgets/widget-cookiecutter
- Traits attributes
 - synchronization between Python and Javascript models through JSON objects
 - Allows communication back and forth between Python and JS
- Community very responsive
Gitter chatroom: gitter.im/jupyter-widgets/Lobby

□ ipyaladin installation



- pip package:

```
pip install ipyaladin
jupyter nbextension enable --py widgetsnbextension
jupyter nbextension enable --py --sys-prefix ipyaladin
```

- conda:

```
conda install -c tboch ipyaladin
```

- Source code on GitHub

github.com/cds-astro/ipyaladin

- Notebooks examples

github.com/cds-astro/ipyaladin/tree/master/examples

□ ipyaladin roadmap



- Offer same set of functions than in Javascript API
 - Investigate if this could be done (semi-)automatically
- Make it work in JupyterLab
- Your ideas/suggestions are welcome