

MOCPy: a Python library to handle MOCs



MOCPy in a nutshell

MOC (Multi Order Coverage maps) is an Virtual Observatory (IVOA) standard, based on the HEALPix tessellation, allowing to **describe** and compare **arbitrary sky regions**.

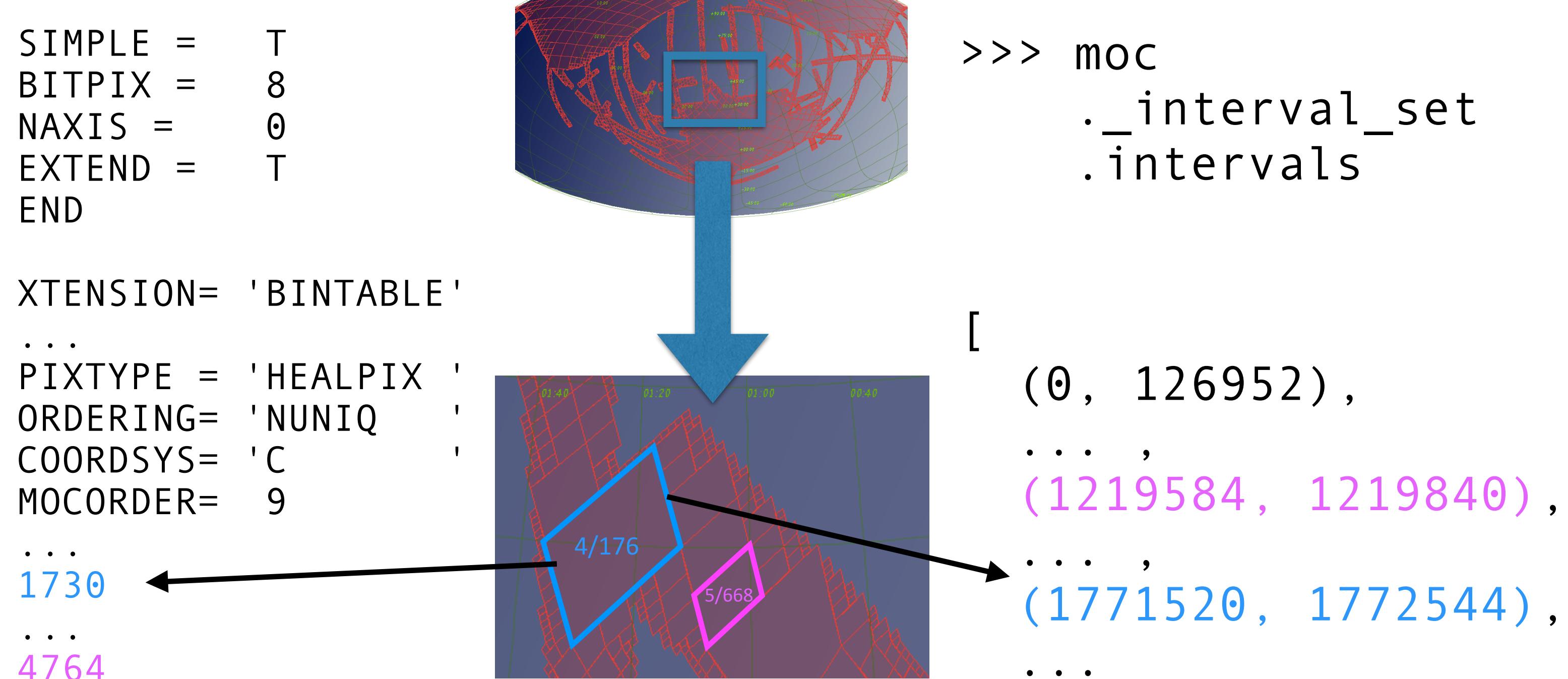
MOCPy is a GPL3-licensed **Python library** to handle and manipulate MOC **spatial coverage maps**.

MOCPy API allows for:

- » easy creation of MOC objects from a list of sources or for a given VizieR table
- » fast computation of coverages intersection
- » filtering of a list of positions to keep rows inside the MOC
- » fast query by MOC of any VizieR table

Several iPython notebooks are provided as API examples made interactive thanks to the Binder service.

Storing a MOC



MOCs are serialized as FITS files enumerating the list of HEALPix cells, using the NUNIQ encoding scheme (eg: 4764 = 4 x 4**5 + 668). In MOCPy, **MOC objects are stored as a list of npix integer intervals** at the deepest *norder* (resolution of the MOC). This allows for fast computation of set operations (intersection, union, etc).

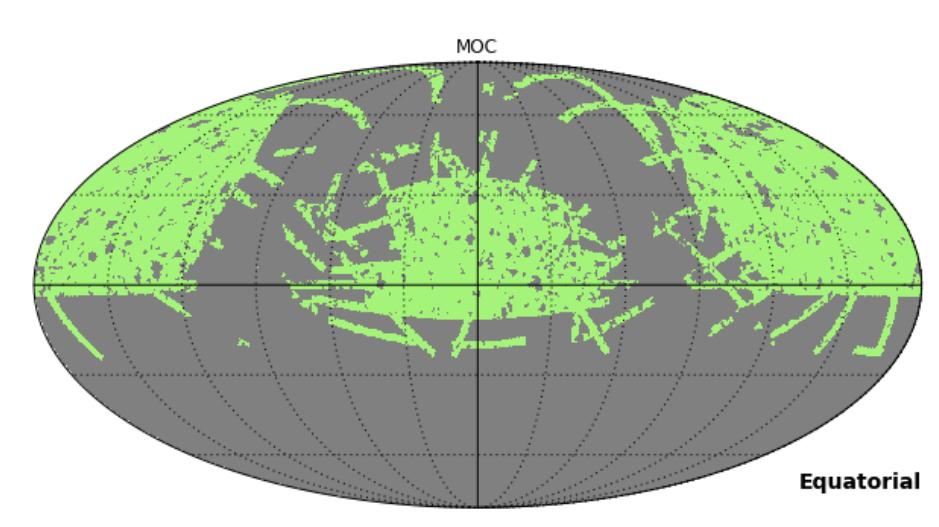
API

Below are typical usages of MOCPy:

Intersection of 2 MOCs

```

from mocpy import MOC
moc_sdss = MOC.from_vizier_table(
    'V/139/sdss9', nside=64)
moc_galex = MOC.from_file(
    'data/GALEX-moc.fits')
intersection =
moc_sdss.intersection(moc_galex)
intersection.plot()
    
```



Create a MOC from a list of positions

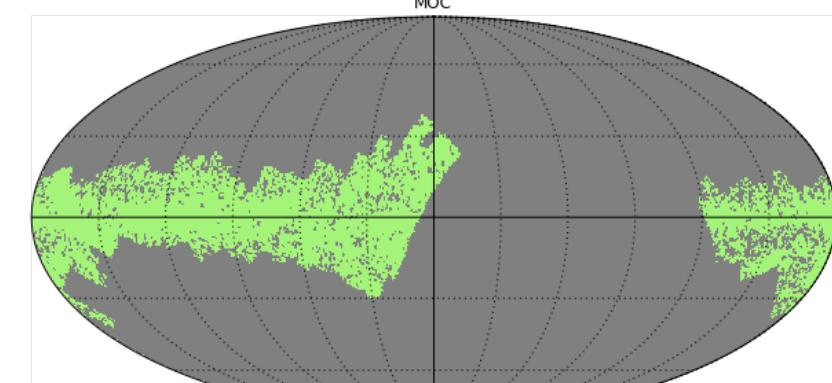
```

from astroquery.vizier import Vizier
table = Vizier.get_catalogs(
    'I/293/npm2cros')[0]
print(table)
    
```

_RAJ2000	_DEJ2000	NPM2	Star	KLA	...	col	n_mag	n_col	tyc	hip	
deg	deg				mag	...			
339.6383	83.1667	+83.0016	LP	002-38	HPM	...	6	0			
271.3571	-23.0056	-23.0168	LSS	4637	OB	...	4	0	T		
Length = 46887 rows											

```

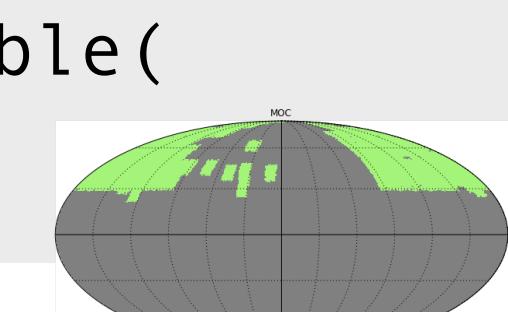
moc = MOC.from_table(table,
    '_RAJ2000', '_DEJ2000', 6)
moc.plot(coord='G')
    
```



Query VizieR by MOC

```

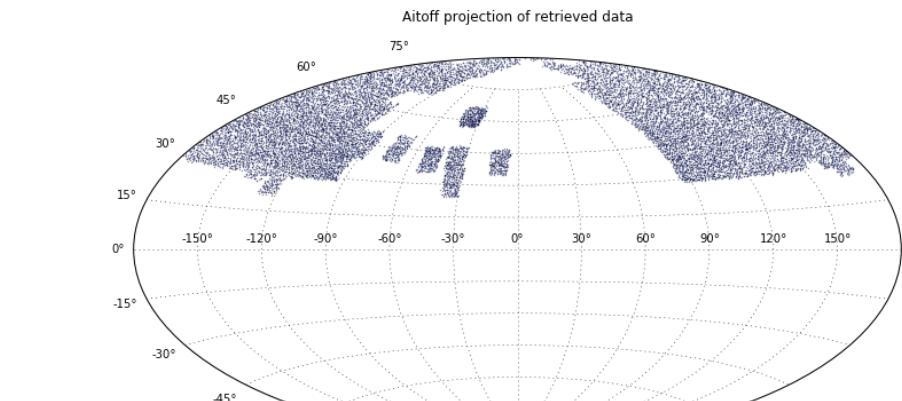
moc = MOC.from_vizier_table(
    'VIII/84/7c', nside=128)
moc.plot(coord='C')
    
```



```

# retrieve Hipparcos sources inside
# MOC coverage
table = moc.query_vizier_table(
    'I/239/hip_main', max_rows=100000)
print(table)
    
```

_RAJ2000	_DEJ2000	HIP	RAhms	...	pmDE	e_Plx	B-V	Notes
deg	deg			...	mas / yr	mas	mag	
5.137508	45.508927	1642	00 20 32.94	...	-21.95	1.02	0.563	
282.127586	72.433073	92284	18 48 30.63	...	22.47	1.49	1.400	D
Length = 16379 rows								



Installation

```
pip install mocpy
```

Dependencies

numpy
healpy
astropy
astroquery

Documentation

The user guide is provided through a set of Jupyter notebooks.



Binder (mybinder.org) is a free service which builds Docker containers for Python notebooks hosted on github and run them in the cloud.



We use *binder* to provide interactive versions of the notebooks.

Roadmap

- » integration in *astropy Regions* package
- » new feature: building a MOC from a set of WCS headers
- » performance improvement for HiPS creation and intersection/union operations
- » in the long term: replace *healpy* with another HEALPix library and switch license from GPL3 to BSD

github.com/tboch/mocpy

Thomas Boch

