HiPS2MOC

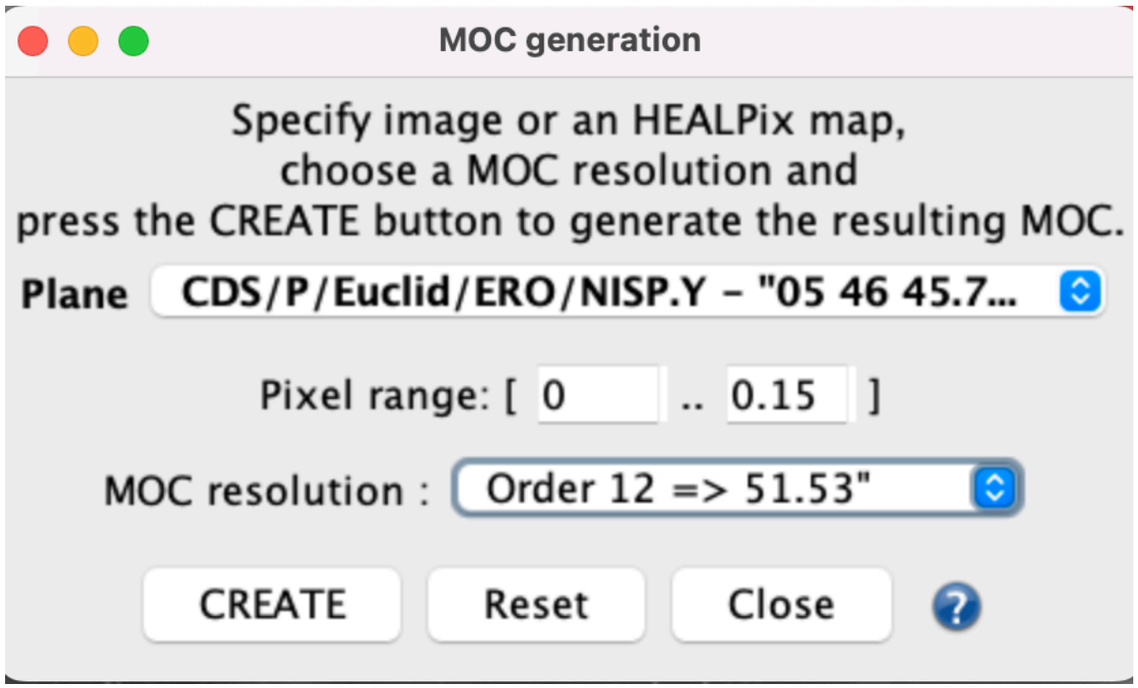
IVOA Nov 2024 - Malta

CENTRE DE DONNÉES
ASTRONOMIQUES DE STRASBOURG

Thomas Boch
& the Aladin team

Whats is this about?

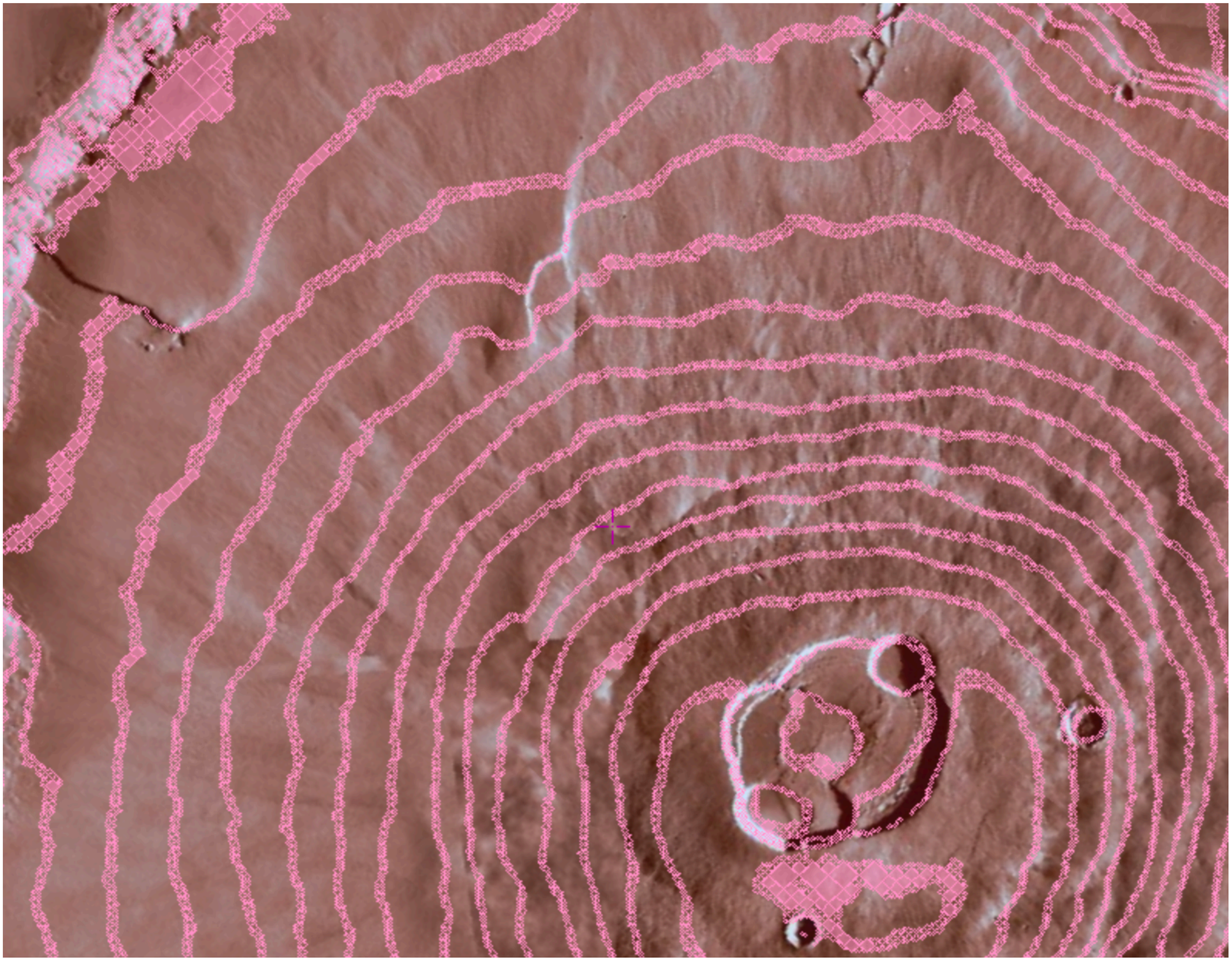
- Prototype service creating **MOCs** from **constraints** on **HiPS** tiles **pixels values**
- Computation done server-side, near the data
- Function already in Aladin Desktop
  - HiPS tiles have to be downloaded first



Motivations

- Coverage (MOC) of GALEX NUV pixels with a flux greater than 2MJy/sr
- Coverage of SIMBAD most dense regions
- MOC of Mars locations with an elevation between 6000 and 9000 meters

Demo



Implementation

- Loop over all HiPS FITS tiles at requested order
  - For each tile:
    - check which (HEAL)pixels satisfy the constraints (*numpy*)
    - Build a MOC (*MOCPy*)
- Build union of all MOCs

Performances

- IO bound
- We need to read:
  - 12 x 4**(tile_order) x *HiPS coverage fraction* FITS tiles
  - tile_order = 2 ==> 192 FITS tiles (all-sky HiPS)
  - tile_order = 3 ==> 768 FITS tiles

Caveats

- Lower-order HiPS tiles are computed averaging pixels on higher-order tiles
  - ==> slight differences in results when changing the requested order
  - Extreme values are averaged out
- HiPS in galactic frame not supported
- Works only for HiPS hosted at CDS

Possible improvements

- Use DATAMIN/DATAMAX keywords to speed up MOC computation
  - Useful for simple min/max queries

- Document units of all CDS HiPS
  - Action started and ongoing

- Allow for expressions on multiple HiPS?
  - Eg: pixel_GALEX_FUV / pixel_GALEX_NUV > 2

Going further(?)

- TAP service for HiPS pixels?

```
SELECT fuv.pixel_value / nuv.pixel_value as
ratio, fuv.ipix as ipix
  FROM "CDS/P/GALEX/FUV"  as nuv, « CDS/
P/.GALEX/NUV" as fuv
    WHERE ratio>10 and fuv.order=14 AND
nuv.order=14 AND
      1=CONTAINS(POINT('ICRS', fuv.RA_pixel,
fuv.DEC_pixel), CIRCLE('ICRS', 80.89417,
-69.75611, 1.)) AND
      fuv.ipix = nuv.ipix
```