

## Online tools for planetary sciences



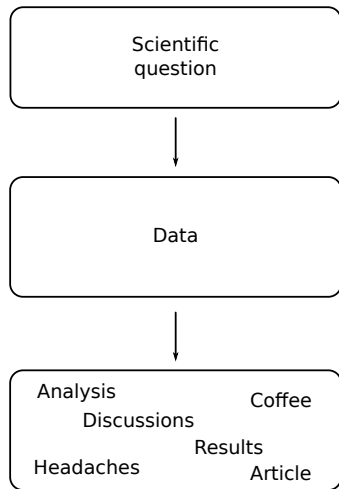
**rocks**

**B. Carry<sup>1</sup> & M. Mahlke<sup>2</sup>**

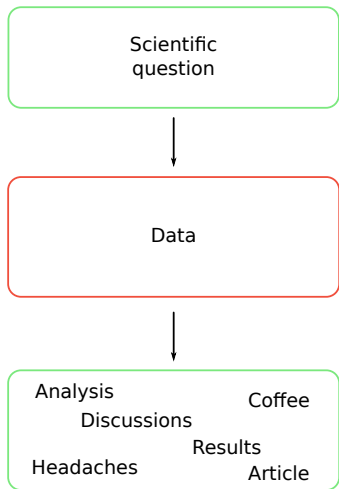
<sup>1</sup>Lagrange, Observatoire de la Côte d'Azur, Nice

<sup>1</sup>Institut d'Astrophysique Spatiale, Orsay

# A typical research project



# A typical research project



## Repetitive (and tedious) tasks!

- **Planning and conduction of observations**
  - Observations already exist?
  - Target/sample available? visible?
- **Gathering ancillary data for the analysis**
  - Complementary information diameter, fall/find, ...
  - Context for research another population
- **Repetitive low-level analysis**
  - Spectral classification
  - Cross-matches & merges

# Shared resources save community time

- **Tedious tasks? Share the load!**

- Many agencies have the mission to support the community  
ESO/ESA/NASA, JPL/MPC/IMCCE, ...
- The expertise is in the community → individual initiatives  
SSHADE, Meteoretical Bulletin, SMASS
- ▶ More time for your research

- **Tedious tasks? Automatize them!**

- Click, click, click... copy-paste, click...
- Or code some processes to work for you
- ▶ Virtual Observatory & Community librairies

- **Community services are less prone to errors!**

- One user → one  $\alpha$ -,  $\beta$ -tester, user...
- Many users → bug reports! and community solutions & patches!
- ▶ Robustness of analysis → results

# Pointing a telescope

## Example

Where do I point the telescope from the name of a target?

# Pointing a telescope

## Example

Where do I point the telescope from the name of a target?

Answer: CDS, IMCCE Miriade, JPL SSD, MPC, Lowell AstEph

The screenshot displays the homepage of the Centre de Données astronomiques de Strasbourg (CDS). The header includes the CDS logo and the text "Centre de Données astronomiques de Strasbourg" and "Strasbourg astronomical Data Center". Below the header, there are four service icons: "Entry point to all services", "Object database", "Catalogue database", and "Interactive sky atlas". Each icon has a corresponding search input field labeled "Object position".

Below the service icons, there is a section titled "Formulaires de calcul d'éphémérides" (Ephemeris calculation forms). This section contains three cards, each with a title, a description, and a "Calculer" (Calculate) button:

- Visibilité des astres**: Calcul des instants de lever, de passage au méridien et de coucher du Soleil, de la Lune et des planètes.
- Observation des planètes**: Calcul des éphémérides utiles à l'observation du Soleil et des corps du Système solaire depuis la Terre.
- Éphémérides de position**: Calcul des éphémérides de position du Soleil et des corps du Système solaire.

# Visibility of targets

## Example

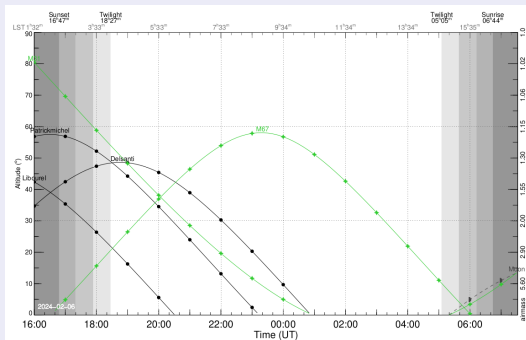
Can I observe asteroids Raymond, Delsanti, 7561 and 10281? And M31 and M67?

## Visibility of targets

### Example

Can I observe asteroids Raymond, Delsanti, 7561 and 10281? And M31 and M67?

Answer: IMCCE ViSiON, Lowell AstObs, airmass.org





## Accessing data

### Example

What is the taxonomy of Vernazza? the diameter of Groussin?

# Accessing data

## Example

What is the taxonomy of Vernazza? the diameter of Groussin?

Answer: IMCCE SsODNet, JPL sbdb, OCA MP3C, Lowell AstInfo, SiMDA

(20607) Vernazza

COPY LINK

EXPORT

Type: Asteroid

Class: MB>Outer

Parent body: Sun

Dynamical system: Sun

Dynamical parameters

Physical parameters

Absolute magnitude

$H = 13.03^{+0.2}_{-0.2}$

mag

Diameter

$D = 15.049^{+0.24}_{-0.24}$

km

Albedo

$p_V = 0.0479^{+0.0089}_{-0.0089}$

Taxonomy

Class = B

Complex = B

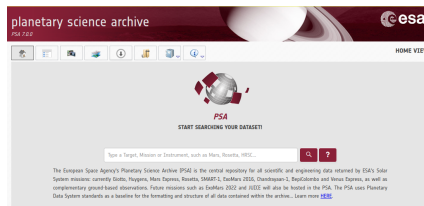
Wavelength range = VIS

Scheme = Bus-DeMeo

```
$ rocks diameters groussin
(16280) Groussin
+-----+-----+-----+-----+-----+
| | diameter | err_diameter_up | err_diameter_down | method | shortbib |
+-----+-----+-----+-----+-----+
| 1 | 3.081 | 0.105 | -0.105 | NEATM | Masiero+2011 |
| 2 | 3.19 | 0.84 | -0.84 | NEATM | Masiero+2012 |
+-----+-----+-----+-----+-----+
```

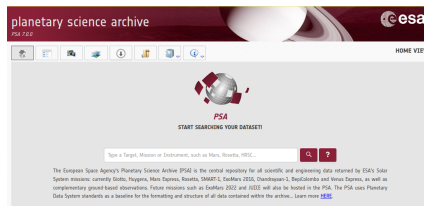
# Pimp my processing

- **Web forms** Access at human-scale
  - Reprocess archival observations
  - Need to contextualize and complement
  - Perform operations beyond our confort zone
- **Shared libraries** Automatize and rationalize
  - Local installation & calls
  - Part of codes, scripts → repeatability
- **Web services and APIs** Use remote resources
  - Send query & get answer
  - Maintenance on the provider side



# Pimp my processing

- **Web forms** Access at human-scale
  - Reprocess archival observations
  - Need to contextualize and complement
  - Perform operations beyond our confort zone
- **Shared libraries** Automatize and rationalize
  - Local installation & calls
  - Part of codes, scripts → repeatability
- **Web services and APIs** Use remote resources
  - Send query & get answer
  - Maintenance on the provider side

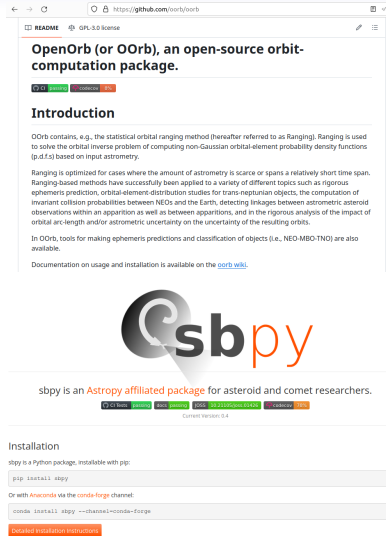


Broz (Astéroïde)

date	a	h	G	deg	R <sub>obs</sub>	R <sub>pl</sub>	Phase	Elong	Δ	Δ <sub>0</sub>	V <sub>0</sub>
					au	mag	mag	deg	deg	arcsec	km/s
2024-02-06T21:00:00.000	9:20:29.0387	20°50'05.305"	1.452625893759	17.85	2.095	174.8183	-0.6516374	0.2428899	4.9870915		

# Pimp my processing

- **Web forms** Access at human-scale
  - Reprocess archival observations
  - Need to contextualize and complement
  - Perform operations beyond our confort zone
- **Shared libraries** Automatize and rationalize
  - Local installation & calls
  - Part of codes, scripts → repeatability
- **Web services and APIs** Use remote resources
  - Send query & get answer
  - Maintenance on the provider side



← → 🔍 <https://github.com/orb/orb>

README GPL-3.0 license

## OpenOrb (or OOrb), an open-source orbit-computation package.

📄 README 📄 LICENSE 📄 ISSUES

### Introduction

OOrb contains, e.g., the statistical orbital ranging method (hereafter referred to as Ranging). Ranging is used to solve the orbital inverse problem of computing non-Gaussian orbital-element probability density functions (p.d.f.s) based on input astrometry.

Ranging is optimized for cases where the amount of astrometry is scarce or spans a relatively short time span. Ranging-based methods have successfully been applied to a variety of different topics such as rigorous ephemeris prediction, orbital-elements-distribution studies for trans-neptunian objects, the computation of invariant collision probabilities between NEOs and the Earth, detecting linkages between astrometric asteroid observations within an apparition as well as between apparitions, and in the rigorous analysis of the impact of orbital arc-length and/or astrometric uncertainty on the uncertainty of the resulting orbits.

In OOrb, tools for making ephemeris predictions and classification of objects (i.e., NEO-MBO-TNO) are also available.

Documentation on usage and installation is available on the [orb wiki](#).

## sbpy

sbpy is an **Astropy affiliated package** for asteroid and comet researchers.

📄 README 📄 LICENSE 📄 ISSUES 📄 CONTRIBUTING 📄 CITATION

Current version: 0.4

### Installation

sbpy is a Python package, installable with pip:

```
pip install sbpy
```

Or with **Anaconda** via the **conda-forge** channel:

```
conda install sbpy --channel=conda-forge
```

[Detailed installation instructions](#)

# Pimp my processing

- **Web forms** Access at human-scale
  - Reprocess archival observations
  - Need to contextualize and complement
  - Perform operations beyond our confort zone
- **Shared libraries** Automatize and rationalize
  - Local installation & calls
  - Part of codes, scripts → repeatability
- **Web services and APIs** Use remote resources
  - Send query & get answer
  - Maintenance on the provider side

## Reddened solar spectrum

sbpy's `sbpy.spectroscopy.SpectralGradient` can be used to redden a solar spectrum, although this approach may be revised with future sbpy capabilities.

```
In [1]: import numpy as np
import matplotlib.pyplot as plt
import astropy.units as u
from sbpy.spectroscopy import SpectralGradient
from sbpy.units import hundred_nm
from sbpy.calib import Sun

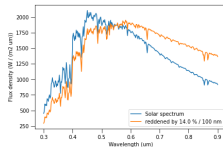
In [2]: sun = Sun.from_builtin('G490_2014LR')
wave = np.linspace(0.3, 0.9, 300) * u.um
fluxd = sun.observe(wave)
S = SpectralGradient(14 * u.percent / hundred_nm, wave@=0.55 * u.um)
print(S)
```

WARNING: Source spectrum is tapered. [synphot.observation]  
 WARNING:astropy:Source spectrum is tapered.  
 14.0 % / 100 nm

`SpectralGradient` works with two-band photometry expressed as magnitudes. To use it for a spectrum, the function `to_color()` must be repeatedly called:

```
In [3]: color_index = u.Quantity([S.to_color(0.55 * u.um, w) for w in wave])
# express in linear units (reflectance)
r = 10**(0.4 * color_index.value)
```

```
In [4]: ax = plt.gca()
ax.plot(wave, fluxd, label='Solar spectrum')
ax.plot(wave, r * fluxd, label='reddened by {}'.format(S))
plt.setp(ax, ylabel='Flux density ({}).format(fluxd.unit)',
xlabel='Wavelength ({}).format(wave.unit)')
plt.legend()
plt.show()
```



# Pimp my processing

## • Web forms Access at human-scale

- Reprocess archival observations
- Need to contextualize and complement
- Perform operations beyond our confort zone

## • Shared libraries Automatize and rationalize

- Local installation & calls
- Part of codes, scripts → repeatability

## • Web services and APIs Use remote resources

- Send query & get answer
- Maintenance on the provider side

```
>>> from astroquery.mpc import MPC
>>> result = MPC.query_object('asteroid', number=56788)
>>> result[0]['name'], result[0]['tisserand_jupiter']
('Guilbertepoutre', 3.2)

>>> result
[{'absolute_magnitude': '14.22',
 'aphelion_distance': '3.566',
 'arc_length': '9428',
 'argument_of_perihelion': '92.1275',
 'ascending_node': '317.2229282',
 'critical_list_numbered_object': False,
 'delta_v': '18.5',
 '...'}
 'updated_at': '2024-01-16T13:32:14Z',
 'uranus_mold': '15.1713',
 'venus_mold': '1.77253']
```

← → ↻ <https://api.imcce.fr/web/services/miriade/api/ephemcc.php?name=abonoi> 110%

```
# Flags: 1
# Ticket: 170015930207186699
# Solar system object ephemeris by IMCCE/OSPMV/CNRS
# Asteroid: 42927 Bonal
# Source: ASTORB
# Author: L.H. Wasserman
# Reference epoch: 2400300.5
# Number of observations: 1002
# Orbit arc covered: 9812 jours
# RMS or maximum residual: 0.280E-01 * 0.950E-04 */d
# Diameter: 0.08 km
# Absolute magnitude H: 15.100
# Slope parameter G: 0.150
# Orbital period: 1.9579452E+03 jours
# System mass: 0.0000E+00 kg
# Dynamical class: HbOuter
# Taxonomic class: ? . ? . ?
# Planetary theory: INPOP18a
# Frame: J2000 astrometric
# Coordinates: equatorial
# Frame center: geocenter
# Precession/nutation model: IAU2000
# Relativistic perturbations: yes
# PPM coordinate system: beta-gamma=1, alpha=0
```

#	Date UTC	RA	DEC	Obs	Mag	Phase	Elong.	dRacssDEC	dDEC	BV
	h m s	h m s	d °	au		deg	deg	arcsec/min	arcsec/min	km/s
2024-02-06T00:00:00.000	+09 54 20.85036	+14 07 30.4221	2.634580016	20.24	2.63	170.38	-0.4511E+00	0.1888E+00	-4.30531	
2024-02-07T00:00:00.000	+09 53 36.81275	+14 12 7.5592	2.612234127	20.22	2.31	171.57	-0.4547E+00	0.1885E+00	-3.82967	
2024-02-08T00:00:00.000	+09 52 50.82038	+14 16 39.3248	2.610157407	20.20	1.90	172.76	-0.4570E+00	0.1886E+00	-3.29170	
2024-02-09T00:00:00.000	+09 52 5.32512	+14 21 11.4027	2.608412551	20.17	1.66	173.95	-0.4605E+00	0.1891E+00	-2.75124	
2024-02-10T00:00:00.000	+09 51 19.57609	+14 25 43.7138	2.606979691	20.14	1.34	175.12	-0.4627E+00	0.1896E+00	-2.20901	

# Typical tasks and some solutions

TBD



# Let's get some hands-on experience

## 1. From GUI to scripts

- Find objects in an image
- **another example of API**

## 2. Easy access to data and parameters of objects

- Common resources for meteorites and Solar System objects
- Efficient data access with APIs

## 3. Getting and analyzing spectra

- How to search and obtain spectra?
- Tools for classification