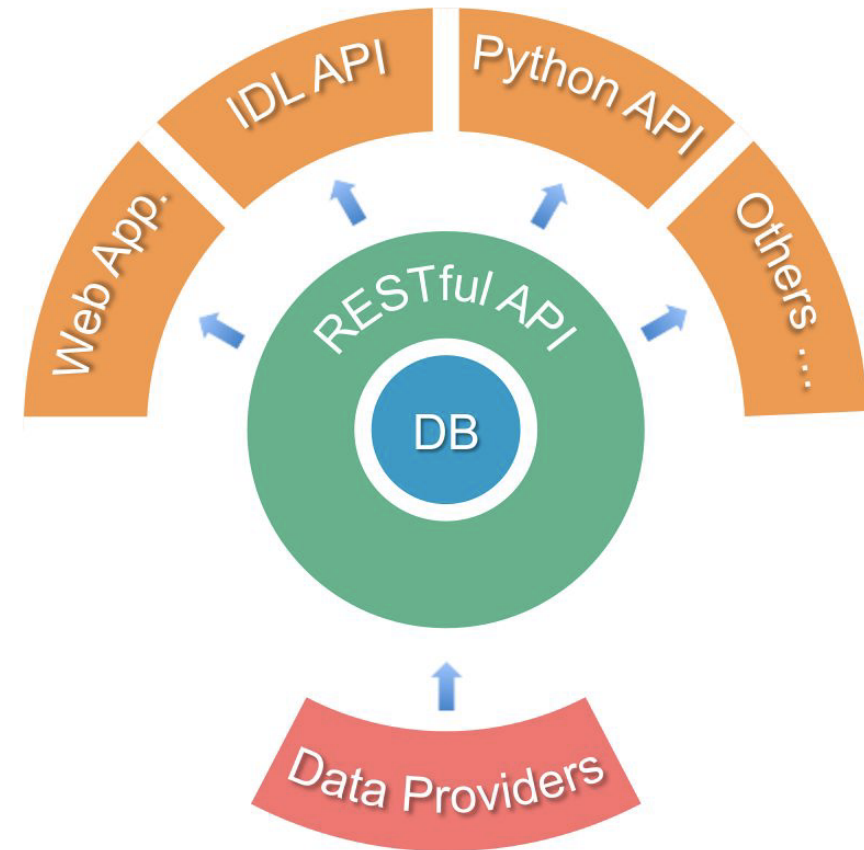


# The SOLARNET Virtual Observatory (SVO)

- Developed in the H2020 SOLARNET project
- At it's heart lies a database populated with meta-data from datasets taken by **space-** and **ground based** instruments.
- It lets you find and access data through a **Web App**, **IDL** and **Python API** and is easily interoperable with external tools (RESTful API)
- <https://solarnet.oma.be/>



Royal Observatory  
of Belgium

**SOLARNET**



This research has received financial support from the European Union's Horizon 2020 research and innovation program under grant agreement No. 824135 (SOLARNET)

# The SOLARNET Virtual Observatory (SVO)

The screenshot shows the SOLARNET interface with search filters for Telescopes (SDO, PROBA2), Characteristics (AIA level 1, HMI magnetogram, SWAP level 1), Tags (venus transit), and Observation date/wavelength. A 'Save selection' button is visible.

Cross dataset searches

The screenshot shows a 'Flare' event with a quick look image of the Sun. The event details include: Start time: 2017-09-01T00:54:20, End time: 2017-09-01T01:10:44, Recognition method: Flare Detective - Trigger Module, and Identifier name: Feature Finding Team.

Dataset specific searches and quick look images

The screenshot shows event-based search results for Flare and CME events. The results table is as follows:

Type	Start date	End date
Flare	2017-09-01T00:06:44	2017-09-01T00:09:44
CME	2017-09-01T00:48:08	2017-09-01T05:48:07
Flare	2017-09-01T00:54:20	2017-09-01T01:10:44
CME	2017-09-01T01:25:47	2017-09-01T05:48:07
CME	2017-09-01T01:25:47	2017-09-01T05:48:07
Flare	2017-09-01T01:26:00	2017-09-01T01:32:00
Flare	2017-09-01T01:26:00	2017-09-01T01:32:00
Flare	2017-09-01T01:28:32	2017-09-01T01:31:44
CME	2017-09-01T01:36:13	2017-09-01T04:36:13
CME	2017-09-01T01:36:15	2017-09-01T05:12:07

Event based searches.

```

from __future__ import print_function
from SOLARNET import datasets

# See all available datasets
for dataset in datasets:
    print(dataset)

# Get a specific dataset
aia_level1 = datasets["aia_level1"]

# Filter the record in that dataset for June 2012 the 6th with a wavelength of 171A
filtered_aia_level1 = aia_level1.filter("DATE-OBS", "2012 June 6", WAVELENGTH = 171)

# Display the date of observation and the wavelength in that filtered dataset
for record in filtered_aia_level1:
    print(record.meta_data["DATE-OBS"], record.meta_data["WAVELENGTH"])

# Download the data from a record
record = filtered_aia_level1[0]
record.download("/tmp")

# Get the data as a BytesIO [1] without saving to disk
data = record.data()

# Open the data as a fits file (see astropy.io.fits [2])
hdus = record.HDUs()
    
```

Python and IDL client



Royal Observatory  
of Belgium



This research has received financial support from the European Union's Horizon 2020 research and innovation program under grant agreement No. 824135 (SOLARNET)