Using Python to Access NASA's Astrophysics Archives

Brought to you by the

NASA Astronomical Virtual Observatories

collaboration:

High Energy Astrophysics Science Archive Research Center — HEASARC
NASA/IPAC InfraRed Science Archive — IRSA
Mikulkski Archive for Space Telescopes — MAST
NASA/IPAC Extragalactic Database — NED

WIFI: AASsummer Pass: summer2023!

How does a user get the data?

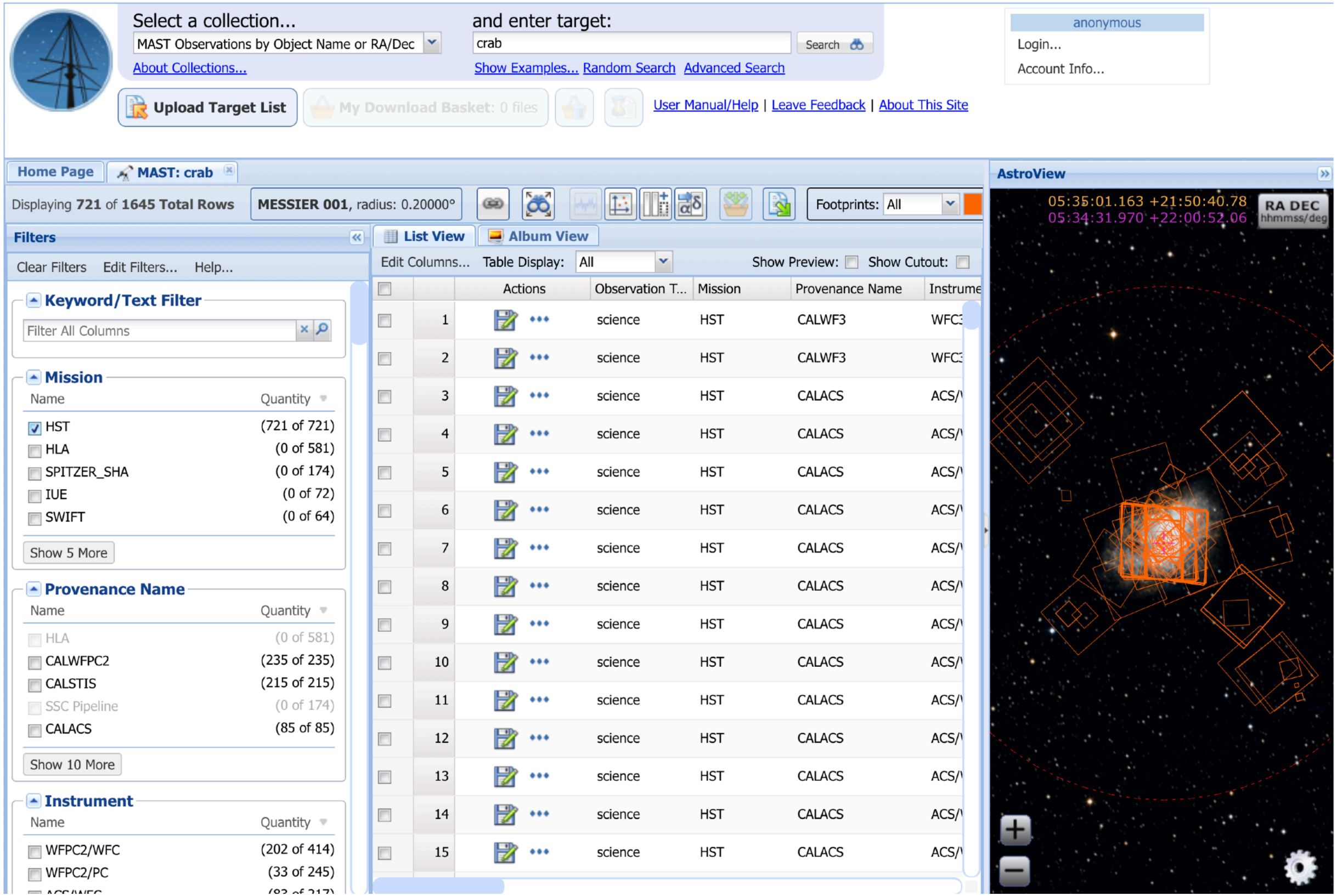
End User

?

Data

HST
TESS
K2
JWST
Kepler
FUSE
Fermi
NICER
NuStar
Swift
ROSAT
Suzaku
IRAS
Spitzer
WISE
SOFIA
Herschel
Akari
GALEX
SDSS
2MASS
AllWISE
…

Example: MAST portal



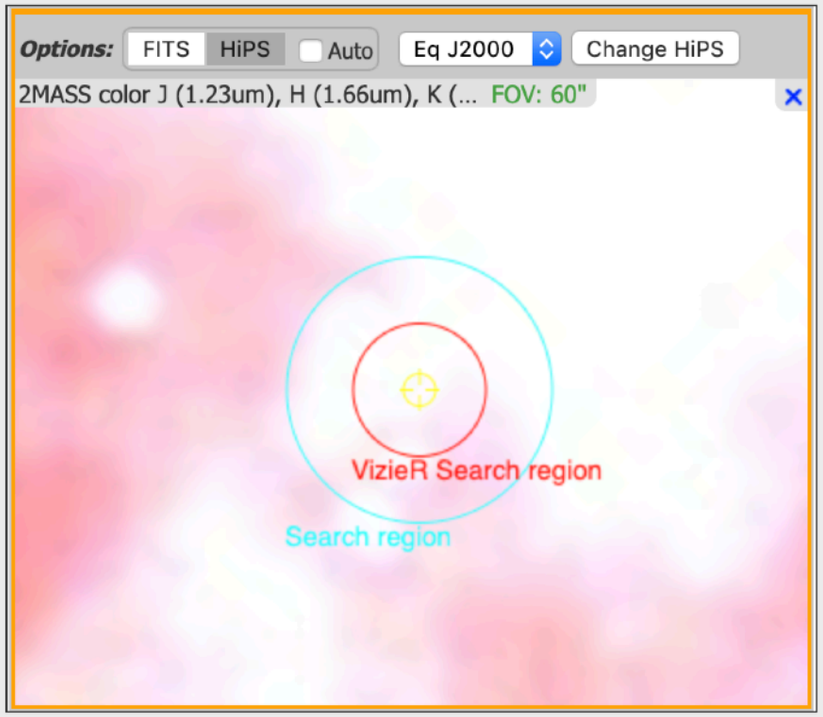
Example: IRSA portal

Data Discovery Search Result

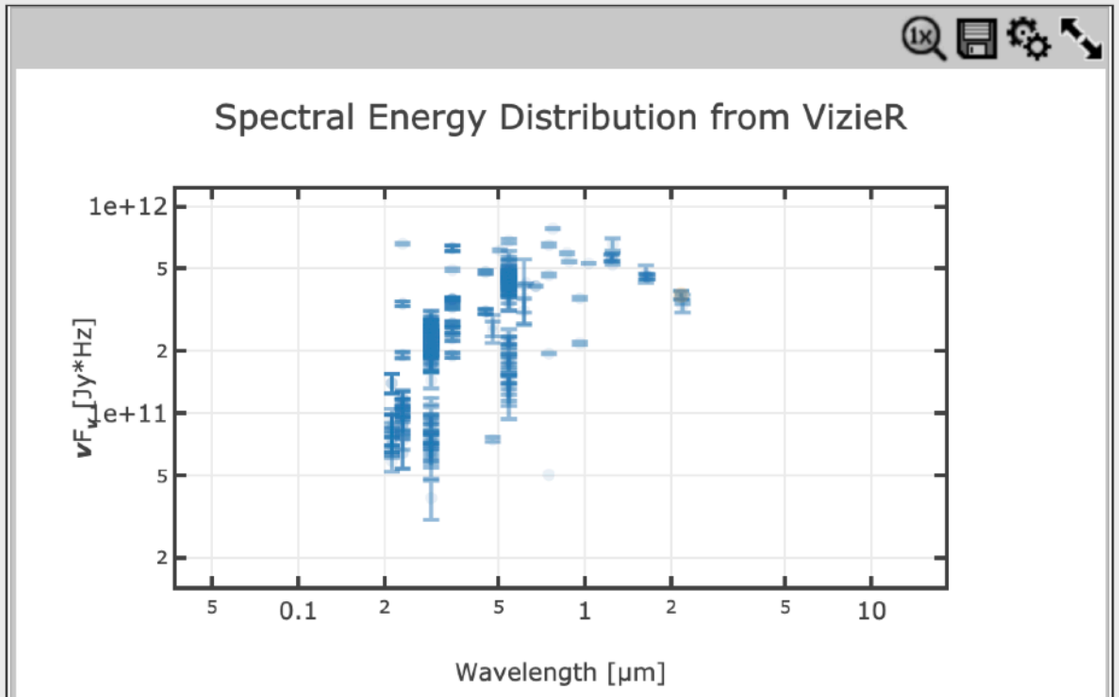
Object/Coordinate	Source	Type	Glon	Glat	Equatorial J2000
M 1	SIMBAD	SNR	184.5575	-5.7843	05h 34m 31.94s +22d 00m 52.2s

Cone Search with center of 83.6331 22.0145 Equatorial J2000 and radius of 10.00 arcsec. **51 datasets found.**

Search Region



SED Plot

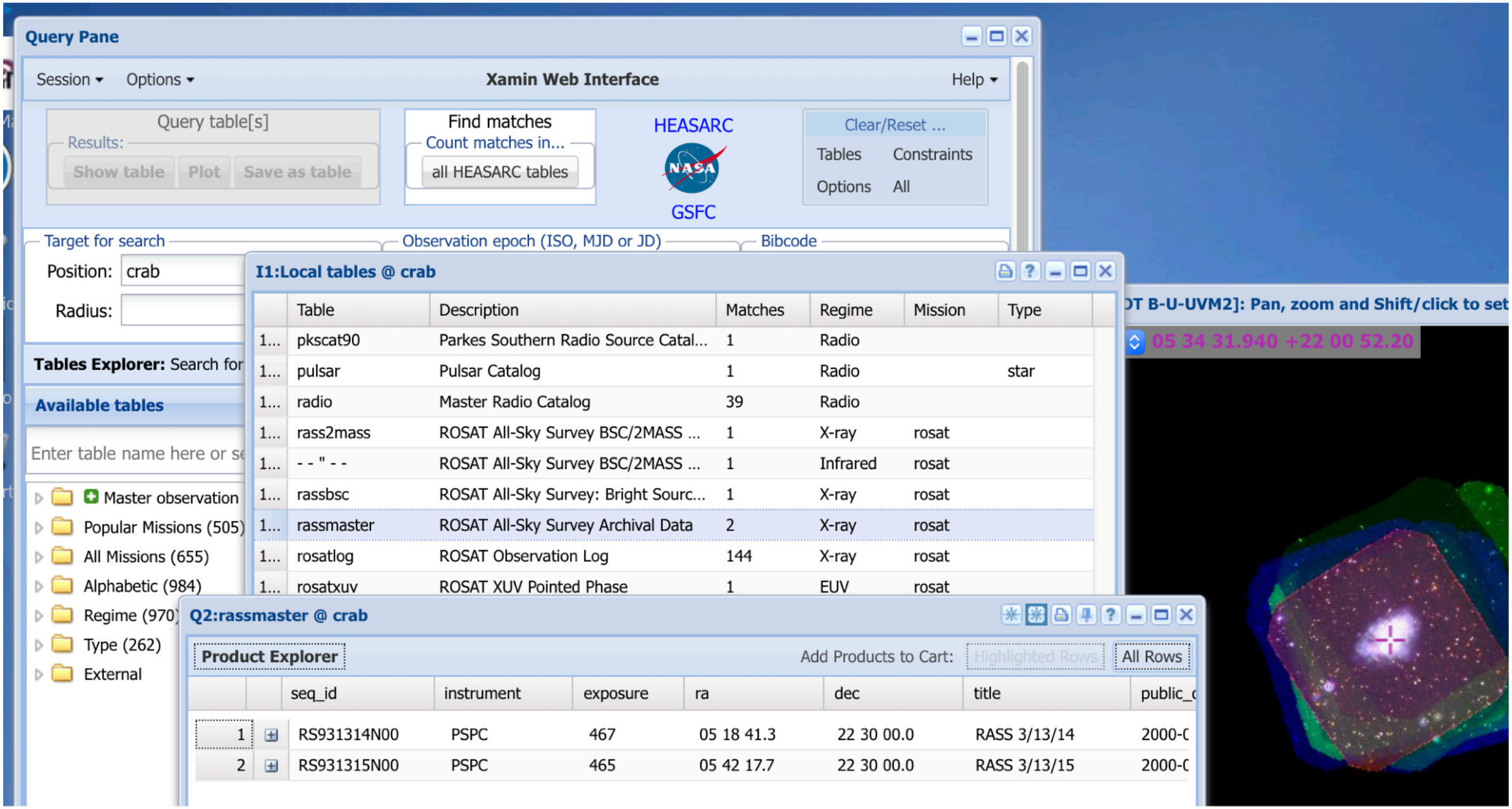


Matching Data Sets | VizieR SED

1 of 1 (1 - 51 of 51)

Mission	Data Set	Data Type	Link to Data
WISE	WISE 3-Band Cryo Single Exposure (L1b) Source Table	Catalog	To IRSAViewer
WISE	WISE Preliminary Release Single Exposure (L1b) Source Table (Superseded)	Catalog	To IRSAViewer
2MASS	2MASS All-Sky Point Source Catalog (PSC)	Catalog	To IRSAViewer
2MASS	The 2MASS Large Galaxy Atlas	Catalog	To IRSAViewer
2MASS	2MASS Second Incremental Release Point Source Catalog (PSC)	Catalog	To IRSAViewer
2MASS	2MASS First Incremental Release Point Source Catalog (PSC)	Catalog	To IRSAViewer
SPITZER	SEIP Source List	Catalog	To IRSAViewer

Example: HEASARC portal



Example: NED portal

Results for object MESSIER 051 (m51)

Overview | Cross-IDs (14) | Coordinates (5) | Redshifts (1) | Distances (50) | Classifications (3) | Galactic Extinctions | Notes (11) | Diameters (0)

Photometry & SED (37) | Spectra (1) | Images (74) | References (634) | External Links | Survey Coverage

Photometry & SED for MESSIER 051

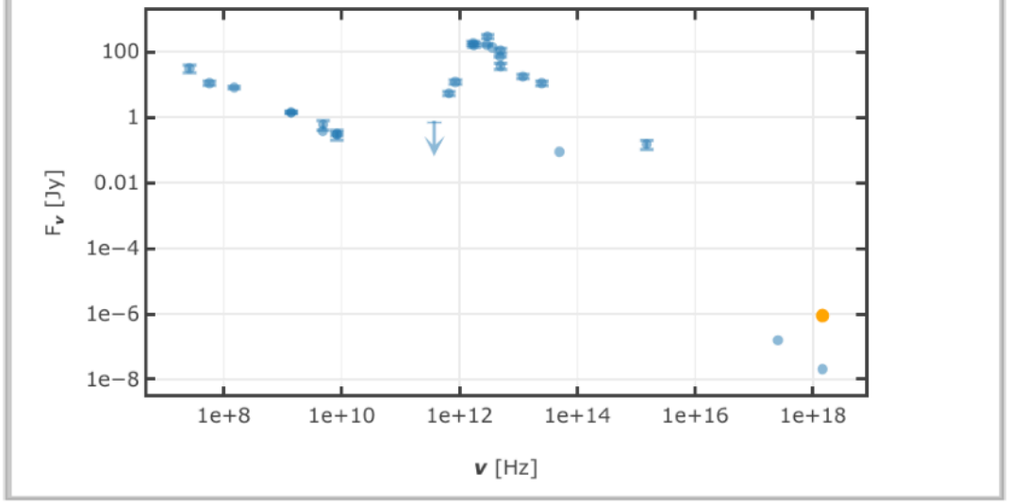
Quick-look Photometry & SED plot for MESSIER 051

Quick Look Photometry & Luminosities (brightest flux in each spectral region)

View References in ADS (1) — 1 of 1 — (1 - 6 of 6)

	Spectral Region	Band	Apparent Mag or ...	Reference co...	Absolute Mag or vLv [W]
	char	char	char	char	char
☑	X-Ray	2-10 keV	1.28E-14 W/m^2^	2004A&A...4...	8.13E+33 [W]
☐	Ultraviolet	m(2000)	8.7 +/- 0.3 mag	1987A&A...1...	-20.47 +/- 0.37 [mag]
☐	Mid-Infrared	25 microns (IRAS)	17.47 +/- 15 % Jy	1988ApJS.....	1.33E+36 +/- 2.30E+35 [W]
☐	Far-Infrared	FIR (IRAS)	4.91E-12 W m^-2^	1988ApJS.....	3.11E+36 [W]
☐	Sub-Millimeter	350 microns	11.9 +/- 1.7 Jy	1989ApJ...33...	6.47E+34 +/- 1.08E+34 [W]
☐	Radio	26.3 MHz	31. +/- 8. Jy	1975AJ.....80...	5.17E+30 +/- 1.41E+30 [W]

SED Plot



Photometry for MESSIER 051

View References in ADS (14) — 1 of 1 — (1 - 27 of 27)

	No.	Observed Passband	Photometry Measurement	Uncertainty	Units	Frequency (Hz)	Flux Density (Jy)	Upper limit...	Lower limit...	Upper limit...	Lower limit...	NED Uncertainty	NED Units	
	int	char	double	char	char	double	double	double	double	double	double	char	char	
													= 'Jy'	
☐	1	2-10 keV	1.28e-14		W/m^...	1.45e+18	8.83e-7						Jy	2004A
☐	2	2-10 keV	3e-16		W/m^...	1.45e+18	2.07e-8						Jy	2004A
☐	3	0.1-2 keV (Chandra)	4e-13		ergs/...	2.54e+17	1.57e-7						Jy	2004A
☐	4	m(2000)	8.7	+/-0.3	mag	1.5e+15	0.15	0.0477	0.0477			+/-4.77E-02	Jy	1987A
☐	5	6 microns (ISO)	0.088		Jy	5e+13	0.088						Jy	2004A
☐	6	12 microns (IRAS)	11.02	+/-15 %	Jy	2.5e+13	11	1.65	1.65			+/-1.65E+00	Jy	1988A
☐	7	25 microns (IRAS)	17.47	+/-15 %	Jy	1.2e+13	17.5	2.62	2.62			+/-2.62E+00	Jy	1988A
☐	9	60 microns (ISO)	36.4	+/-20 %	Jy	5e+12	36.4	7.28	7.28			+/-7.28E+00	Jy	2001A
☐	10	60 microns (IRAS)	108.68	+/-15 %	Jy	5e+12	109	16.3	16.3			+/-1.63E+01	Jy	1988A
☐	11	ISO 60 microns	70.3	+/-3.0	Jy	4.93e+12	70.3	3	3			+/-3.00E+00	Jy	2003A
☐	14	FIR (IRAS)	4.91e-12		W m^...	3.63e+12	135						Jy	1988A
☐	16	100 microns (IRAS)	292.08	+/-15 %	Jy	3e+12	292	43.8	43.8			+/-4.38E+01	Jy	1988A

Searching Several Archives

- Suppose you are interested in a specific source and would like to compare *all* of the available multi-wavelength data for it.

- You could:

1. collect a list of multi-wavelength observations in the literature, although NED has done most of the work for extra galactic objects but it does not have all objects and data from all archives;
2. for each observation, search on-line for the relevant public archives;
3. following each archive's instructions for how to navigate their system and/or use their API to find and download the products you are interested in (e.g., images);
4. do it again for your next source.

- Wouldn't it be nice if there were an easier way?

- And a way that's scriptable in Python?!

Example: astroquery.mast

- See
  - https://astroquery.readthedocs.io/en/latest/mast/mast.html

```
>>> from astroquery.mast import Observations

>>> obs_table = Observations.query_region("322.49324 12.16683")
>>> print(obs_table[:10])
```

```
dataproduct_type obs_collection instrument_name ... distance
---------------- -------------- --------------- ... --------
            cube          SWIFT            UVOT ...      0.0
            cube          SWIFT            UVOT ...      0.0
            cube          SWIFT            UVOT ...      0.0
            cube          SWIFT            UVOT ...      0.0
            cube          SWIFT            UVOT ...      0.0
            cube          SWIFT            UVOT ...      0.0
            cube          SWIFT            UVOT ...      0.0
            cube          SWIFT            UVOT ...      0.0
            cube          SWIFT            UVOT ...      0.0
            cube          SWIFT            UVOT ...      0.0
            cube          SWIFT            UVOT ...      0.0
```

Example: astroquery.ipac.irsa

- See
  - https://astroquery.readthedocs.io/en/latest/ipac/irsa/irsa.html

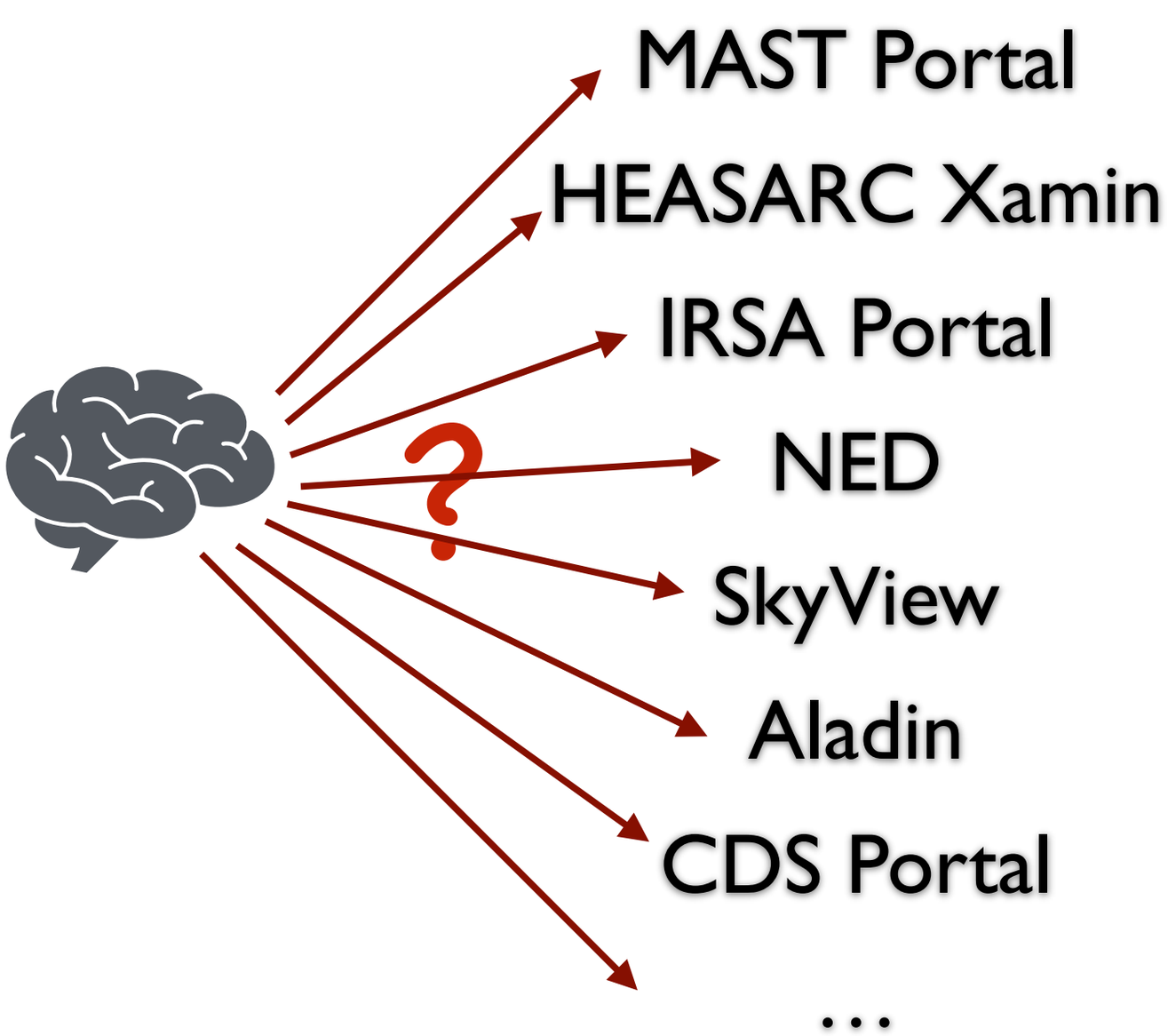
```
>>> from astroquery.irsa import Irsa
>>> import astropy.units as u
>>> table = Irsa.query_region("m31", catalog="fp_psc", spatial="Cone",
...                           radius=2 * u.arcmin)
>>> print(table)

      ra      dec        clon          clat      err_maj ...  j_h   h_k    j_k    id
------- ------- ------------ ------------ ------- ... ----- ------ ------ ---
10.685  41.248 00h42m44.45s 41d14m52.56s    0.14 ... 1.703  0.921  0.971   0
...     ...          ...          ...     ... ...   ...    ...    ...  ...
10.686  41.271 00h42m44.60s 41d16m14.16s    0.13 ...    --     --     -- 768
10.694  41.277 00h42m46.55s 41d16m36.13s    0.27 ...    --     --     -- 769
10.690  41.277 00h42m45.71s 41d16m36.54s    0.15 ...    --     --     -- 770
10.679  41.281 00h42m42.88s 41d16m51.62s    0.43 ...    --     --     -- 771
10.689  41.237 00h42m45.26s 41d14m13.32s    0.22 ...    --     --     -- 772
10.661  41.274 00h42m38.53s 41d16m24.76s    0.18 ...    --     --     -- 773
10.653  41.281 00h42m36.78s 41d16m52.98s    0.17 ...    --  0.795     -- 774
```

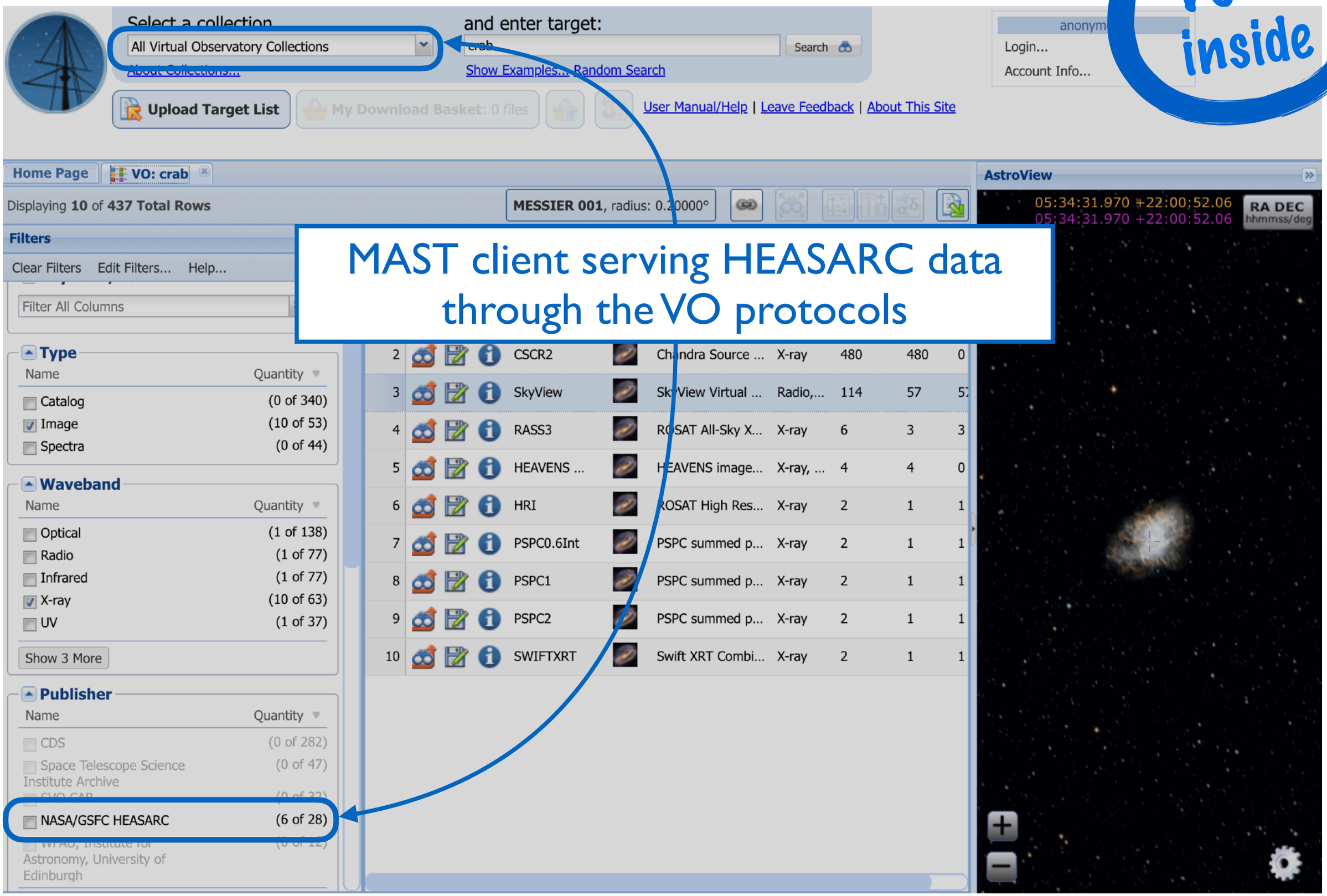
This is standardized and easy to use, but it still depends on knowing which archive to query and doing them all individually.

Note: astroquery.ipac.irsa and .heasarc are *not* maintained by us.

How does a user get the data?



MAST Portal with VO inside



MAST client serving HEASARC data through the VO protocols

Introducing the Virtual Observatory

- The VO was conceived to standardize all of the archives' interfaces.
- Any client able to access one VO-compliant archive can access *all*.

End User

Python?

MAST Portal

HEASARC Xamin

IRSA Portal

NED

SkyView

Aladin

CDS Portal

…

Archive

Barbara A. MIKULSKI ARCHIVE FOR SPACE TELESCOPES

HEASARC

irsa

NED

…

Data

HST
TESS
K2
JWST
Kepler
FUSE
Fermi
NICER
NuStar
Swift
ROSAT
Suzaku
IRAS
Spitzer
WISE
SOFIA
Herschel
Akari
GALEX
SDSS
2MASS
AllWISE
…

…

PyVO allows searches across NASA archives

- The client we are interested in today is anything using Python, whether interactively in a Jupyter notebook or programmatically in a batch queue.
- PyVO is an Astropy-affiliated package that:
  - is a collaboration among a number of archives;
  - is an open development project on GitHub;
  - includes
    - data discovery,
    - catalog searches,
    - cross-correlations, and
    - image and spectra searches;
  - is continuing to expand its suite of user-friendly functions;
  - may become your favorite new Pythonic way to get data!
  - https://pyvo.readthedocs.io/en/latest/

PyVO and Astroquery

- Astroquery is the most popular package in Astropy ecosystem.
- PyVo is *complementary* to Astroquery.
- Astroquery often uses PyVO underneath.
- Astroquery is archive-specific and includes specialized APIs specific to the archive.
- PyVO is general and works the same everywhere the VO is implemented.
- If you don't already know where your data are (i.e., which archive), then you can use PyVO to find out.

The PyVO workflow

Whether interactively or running an automated script — *and most likely, an iterative combination of the two* — the basic steps are:

1. Step I: search the Registry for data, e.g.,
   - UV images,
   - the latest Gaia catalog,
   - x-ray spectra,
   - etc.
2. Step II: ask each service about what it has, e.g.,
   - is there a Swift UVOT observation of Cen A?
   - what information (columns) does Gaia DR2 have?
   - is there a Chandra ACIS HETG spectrum for NGC 1365?
3. Step III: access the data, e.g.,
   - retrieve and view the images,
   - cross-correlate against your catalog, or
   - retrieve and analyze the spectra.

Concise example with PyVO

- Let's find all the available images of M51

(simplified example, no record-keeping, not storing filenames, etc.)

```
import pyvo as vo
from astropy.coordinates import SkyCoord
m51_pos = SkyCoord.from_name("m51")
from astropy.utils.data import download_file

services = vo.regsearch(servicetype='image')
for service in services:
    try:
        results=service.search(pos=m51_pos, radius=0.1)
        for result in results:
            try:
                print("Downloading {}".format(result.getdataurl()))
                download_file(result.getdataurl())
            except:
                print("Unable to download from {}".format(result.getdataurl()))
            break
    except:
        print("Service {} not working;  skipping".format(service.ivoid))
```

Step I: find services

Step II: ask what it has

Error trapping

Step III: get the data

```
Service ivo://3crsnapshots/sia not working;  skipping
Downloading http://galex.stsci.edu/data/GR6/pipe/01-vsn/05280-NGA_M51/d/01-main/0001-img/07-try/qa/NGA_M51-xd-int_2color.jpg
Downloading http://hla.stsci.edu/cgi-bin/getdata.cgi?config=ops&amp;dataset=hlsp_appp_hst_wfpc2_sfd-pu4k2f301_f606w_v2_sci
Service ivo://cadc.nrc.ca/sia not working;  skipping
Service ivo://cds.vizier/siap not working;  skipping
Downloading https://cdaftp.cfa.harvard.edu/cgi-bin/chaser_ftp_retrieve_file.cgi?filename=science/ao01/cat7/353/primary/acisf00353N005_e1_cntr_img2.jpg
Downloading https://cda.cfa.harvard.edu/csccli/retrieveFile?filename=acisf00354_000N020_b_img3.fits&filetype=ecorrimg&version=cur
```

Because these are living, changing services/archives.

Things to know

- Each archive is responsible for its own backends. They should obey the VO standard, but occasionally there are mistakes.
  - We invite you to contact the archive itself, or post to the PyVO channel on the Astropy Slack space: http://joinslack.astropy.org/
- Each archive has its own response and uptime issues. There *will* be servers that sometimes do not respond.
  - Ditto. Furthermore, for scripting loops over services, do NOT forget to enclose each in a **try:except** so that you can continue to the next. (And log what happens at each step so you can figure out after the fact what you got, or didn't get, and why.)
- Each archive is a living archive. Things change as a function of time, so what you did yesterday might not come out identically today.
- The VO is a collaboration consisting of **agreed standards** implemented by **growing services** and access by **evolving clients**, and it is an increasingly powerful one!

Tutorial Notebooks

- We have developed the following notebooks

  https://github.com/nasa-navo/navo-workshop

  - Download them to run locally and adapt, or
  - view them rendered on GitHub at

    https://nasa-navo.github.io/navo-workshop/
  - or run them in MyBinder (button on GitHub front page).

- Contents:
  - QuickReference.ipynb — example of each type of search;
  - EXERCISE / Use Case I — inspecting a candidate list;
  - EXERCISE / Use Case II — preparing a proposal;
  - EXERCISE / Use Case III — creating an HR diagram;
  - a set of more detailed cheat-sheets (CS*) for each type of search;
  - KNOWN_ISSUES.md — list of known oddities/errors/workarounds.

A peek at the tutorial notebooks

Have at it!

You can choose whether to

- fill in the code cells of the empty EXERCISE notebook yourself using the QuickReference etc. as a guide; or to
- go through the solutions to the Use Case to see how it works; modify, play, ask us questions…

Close out

- Thanks for coming, and we hope you learned a lot!
- We hope you continue using these tools on your own, and
- Give us feedback!
  - Look for a survey in your email and let us know what you think
    - https://bit.ly/aas242-pyvo-workshop
    - Link also posted in AAS Slack channel:
      - #workshop-accessing-nasa-astrophysics-archives-using-python
  - Report issues on the GitHub
    - https://github.com/NASA-NAVO/navo-workshop
- Ongoing support
  - Ask for help on the pyvo Astropy Slack channel
    - https://astropy.slack.com/
    - Join at http://joinslack.astropy.org/