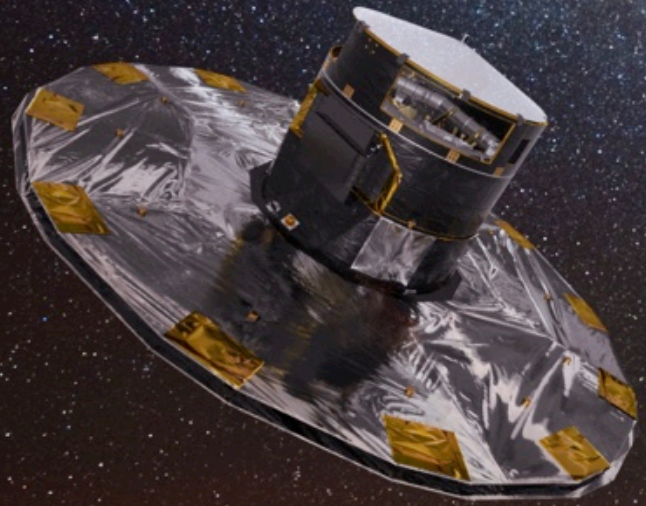
Driving Gaia Science from the ESA Archive: DR2 to DR3

Juan González-Núñez, J. Salgado, R. Gutiérrez-Sánchez, JC. Segovia, J. Duran, E. Racero, J. Osinde, P. De Teodoro, F. Giordano, D. Baines, A. Mora, J. Bakker, B. Merín, C. Arviset, F. Aguado-Agelet

ESAC Science Data Centre (ESDC)

The main Gaia Archive Challenges



Data Release 1

- How to provide high throughput access and server side analysis to a 1.1e10 sources catalogue?

Data Release 2

- How to link catalogue data with associate epoch photometry dataset with billion product level scalability?

Data Release 3

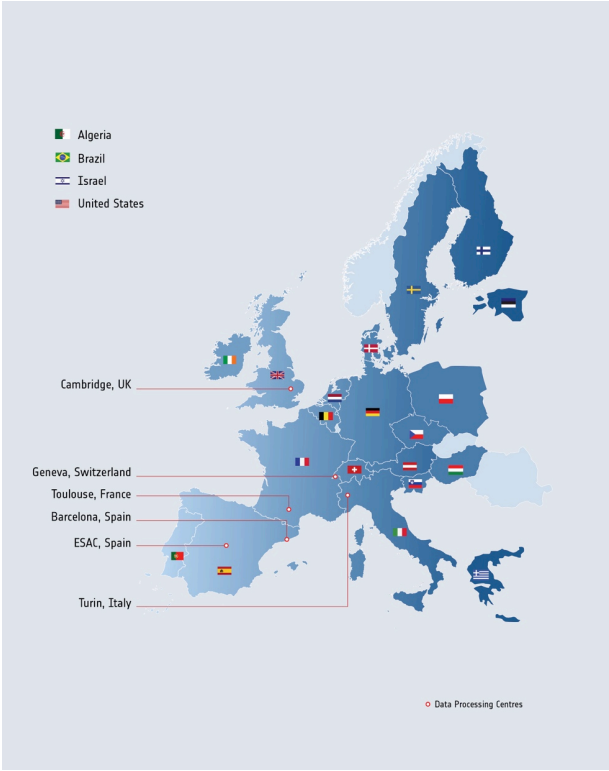
- How to provide effectively access to Spectra and further Epoch data products in the scale of tenths/hundreds of TB?

How?

If you want to go far, go together

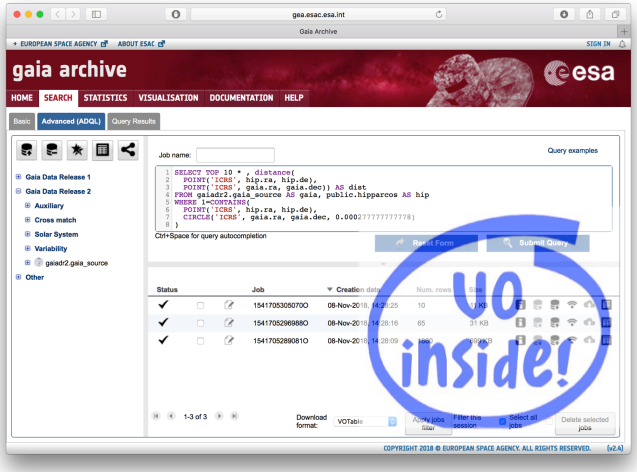
- Gaia DPAC Coordination Unit 9

  - Work packages covering main activities (Visualization, Validation, Operations, etc.) contributing to the Archive
  - Hundreds of experts throughout Europe providing feedback to the Archive, including VO experts
  - Associate and Partner Data Centres serving replicas of Gaia Data



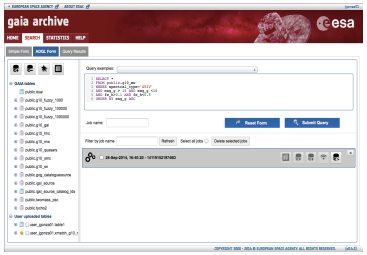
The ESA Gaia Archive: VO Inside

- TAP, UWS, DataLink, VOSpace are the **core** backbone of the Gaia Archive server side, not an on-top addition over tailored protocols
- All APIs used by the Archive **are public and documented**
- When a VO protocol does not fully fit the purpose, it is **extended**, keeping compatibility. Eg. TAP+



http://archives.esac.esa.int/gaia

A VO Inside Architecture



Command line tools

External Apps

Data Validation

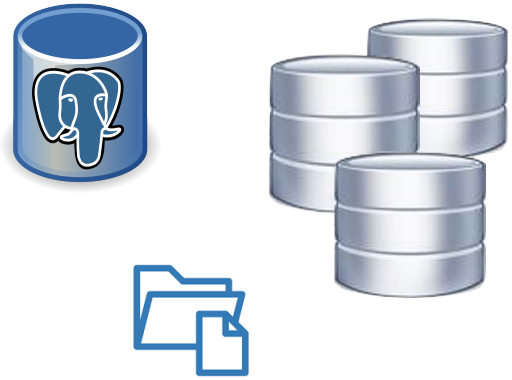
Open APIs

TAP+

DataLink

VOSpace

CSearch



Public area

- Publicly released data

Restricted area

- Dataduring validation

User Space

- User-uploaded data

Scaling Up in Data Release 2

Data Release 2

- How to link catalogue data with associate epoch photometry dataset with billion product level scalability?

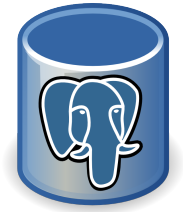
Combining VO protocols

- **TAP+**
  - Catalogues, source clasification, SSOs.
  - Efficiently “indexable” data
  - Benefits from storage in RDBMS

- **DataLink**
  - Associated data products (Spectra, Light Curves).
  - DataLink allows for efficient DataModel-agnostic search over large datasets based on product level metadata; perfect fit for Gaia Spectra or Light curves
  - Mechanisms for linking TAP searches with associated data products
  - Scales to DR3/4 data volumes

TAP+

- Load Balancing
- High availability



- **PostgreSQL + Q3C**
  - Master/slave setup
  - Streaming replication
  - Warm Standby
  - High performance HW
    - SSD, TB RAM

DataLink

- Stateless service
- High Scalability

- **Postgres-XL**
  - “Shared nothing” architecture
  - Virtualized Infrastructure
  - Read Only
  - Standard HW

Towards Data Release 3

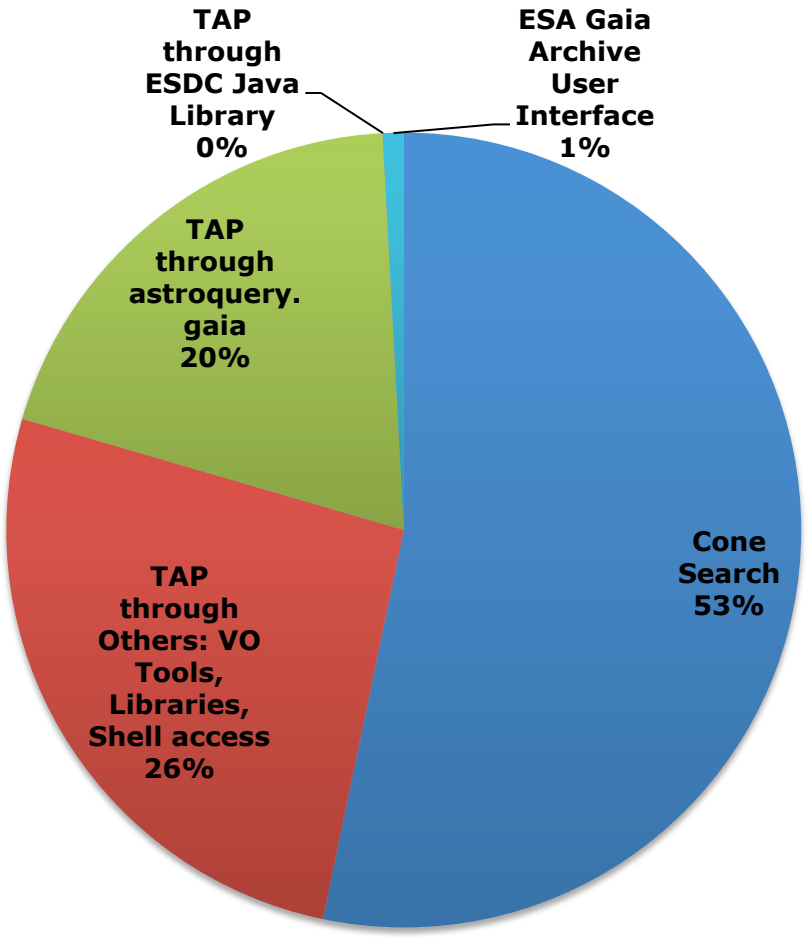
Data Release 3

- How to provide effectively access to Spectra and further Epoch data in the range of tenths/hundreds of TB?

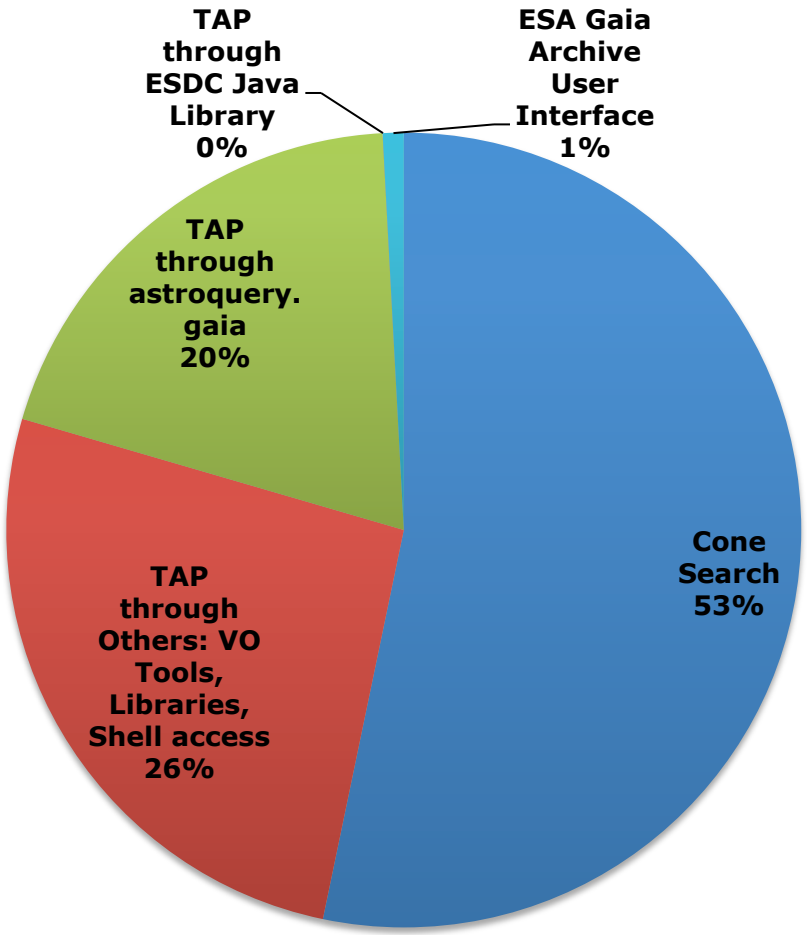
User feedback through User Stats?

- A plethora of information about archive usage has been compiled through traditional methods: surveys, user groups, beta testers, science usage scenarios, over 400 support tickets, etc.

- Do usage statistics enforce these trends, eg. interest in the Python language and associated ecosystem, or even introduce newer feedback?



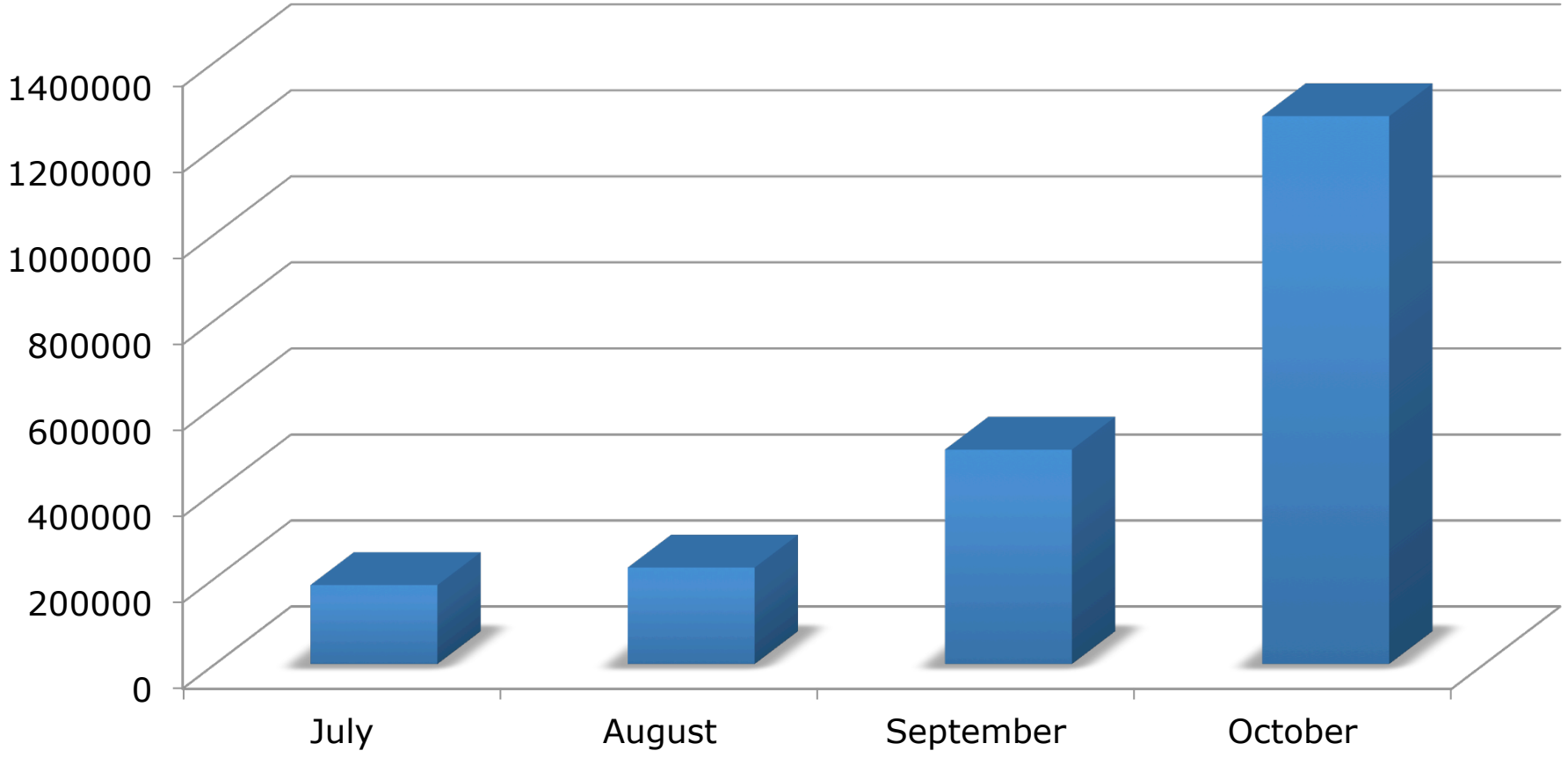
July-October 2018 query origin



- Over **17.5K** User interface users throughout the period "only" generate 1% of the traffic received – though they are the most complex
- The ESDC contributed library to Astroquery represents **almost half** of all the programmatic TAP queries received
- ConeSearch still generates significant traffic (with caveats)
- Well, astronomers definitely do not like Java

July-October 2018 query origin

Received TAP queries from astroquery.gaia



astropy libraries to rule them all

- **Gaia** modules:
  - **astroquery.gaia**: TAP client and Gaia TAP+ specific (public)
  - **astroquery.utils.tap**: generic TAP client (public)
  - **DataLink** access (in dev.)

- **ESASky** modules:
  - **astroquery.esasky** : ESASky data access module (public)
  - **pyESASky**: Widget to visualize data in eg. JupyterHub (in dev.)

- **Hubble** modules:
  - Data access module. Reusable to build access to any ESDC 2nd generation archive (in dev)

OK, what besides more Python

- New “types” of products: keep Data Models **in sync** with VO
- Data Volume grow
  - Provide **searchable access** to DR3 data
    - Covered by DR2 architecture, will require infrastructure extensions
  - Provide **deeper analysis capabilities** by moving code closer to the data

Moving code to the Data

- ESA efforts converging into a unified cloud computing platform: Science Exploitation and Preservation Plattform (**SEPP**)
  - Check P5-14 Poster from V. Navarro

- **JupyterHub** internal PoC for SEPP created by ESDC for JupyterLab awareness workshop
  - Authenticated through ESA CAS, User spaces
  - AstroPy and several COTS modules loaded
  - Fully scalable architecture

- Several demo Notebooks made available in the workshop covering different **science cases** using our plattform and libraries

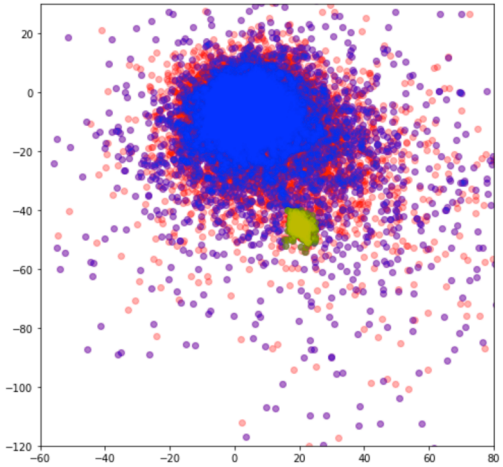
jupyter Gaia_astroquery_tutorial_clusters_DR2 (unsaved changes)

File Edit View Insert Cell Kernel Widgets Help

Run Markdown Publish your notebook into Anaconda.c

Visually inspect the results are the same by plotting the same as above.

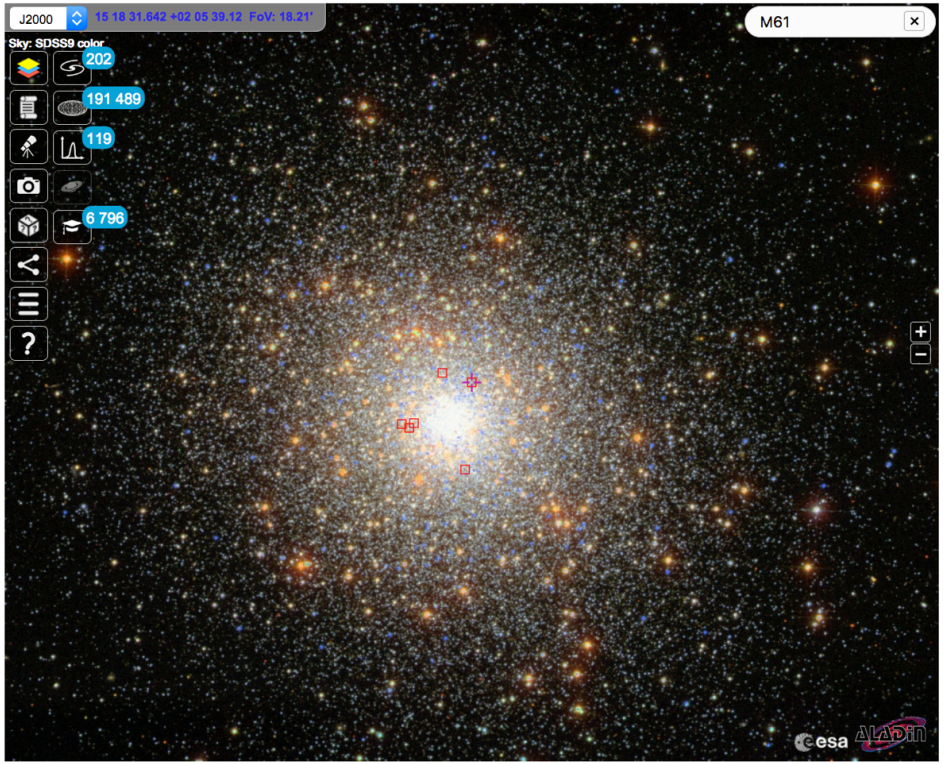
In [22]:
```
plt.figure(figsize=(8, 8))
plt.scatter(r['pmra'], r['pmdec'], color='r', alpha=0.3)
plt.scatter(j['pmra'], j['pmdec'], color='b', alpha=0.3)
plt.scatter(m45cluster['pmra'], m45cluster['pmdec'], color='g', alpha=0.3)
plt.scatter(test['pmra'], test['pmdec'], color='y', alpha=0.3)
plt.xlim(-60,80)
plt.ylim(-120,30)

plt.show()
```



Video:
https://goo.gl/YmLF3J

ESASky pyESASky and Hubble Source Catalogue (HSC) Use Case

In [1]:
```
# Import and instantiate the ESASky Widget
from pyesasky.pyesasky import ESASkyWidget
esasky = ESASkyWidget()
esasky
```



[2]:
```
# Go to the Globular cluster M5 (229.63842, +02.08103; coordinates in J2000)
esasky.setGoToRADec('229.63842', '+02.08103')
```