Gaia Data Queries with TAP/ADQL and TOPCAT

Markus Demleitner *(msdemlei@ari.uni-heidelberg.de)*
Hendrik Heinl *(heinl@ari.uni-heidelberg.de)*
Adapted by Enrique Solano (Nov 2019)

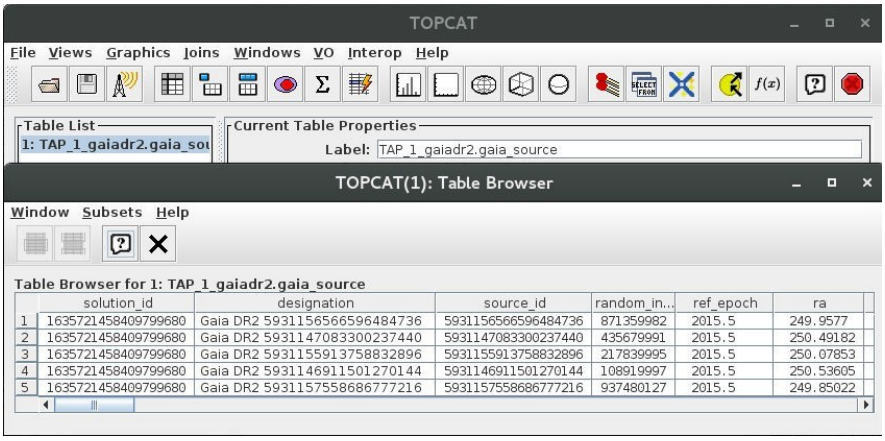
TAP: T(able) A(ccess) P(rotocol)
AQDL: A(stronomical) D(ata) Q(uery) L(anguage)

VO protocols like ConeSearch, SSAP or SIAP only allow simple queries, mainly by position and search radius. The table access protocol (TAP[1]) defines a service protocol to make more complex queries to astronomical catalogs as well as general database tables. TAP uses ADQL (Astronomical Query Data Languaje) to build expressions of relational algebra.

In this tutorial we will see a few introductory examples of queries with TAP/ADQL.

Note that all names in SQL (column names, table names, commands, etc) are case-insensitive.

- Launch TOPCAT
- IVO / Table Access Protocol (TAP) query. A new window (*"Table Access Protocol (TAP) Query"*) will pop up. If you wish, click the pin icon in the upper left corner of this window to keep it open even while the query is executing (ico background must be blue).
- Enter Gaia in the "Keywords" box . Click "Find Services"
- Click on GAIA (ivo://esavo/gaia/tap). The corresponging TAP URL will appear in the TAP URL box (at the bottom of the window). Hit "Use Service".
- In the tab "Use service" you will see all the tables available from the Gaia database. Select **gaiadr2.gaia_source**. If you click on the "Columns" tab you will get information of all the columns of this particular table.
  - **Select / TOP**
    - In the bottom box enter: Select top 5 * from gaiadr2.gaia_source
    - Hit "Run query"
    - "top" is just an integer giving how many rows you want returned. Once the query is finished a new table should have been created in TOPCAT with the following information:



1http://www.ivoa.net/documents/TAP/

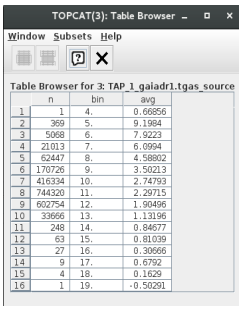
- **Select / Order by**
  - In the bottom box enter: Select top 5 source_id,phot_g_mean_mag from gaiadr2.gaia_source order by phot_g_mean_mag
  - Hit “Run query”. You will get the 5 brightest stars in the Gaia DR2 source catalogue.
  - If you now enter: Select top 5 source_id,phot_g_mean_mag from gaiadr2.gaia_source order by phot_g_mean_mag **desc** and click “Run query” you will get the 5 faintest stars in the Gaia DR2 source catalogue.

- **Select / where**
  - In the bottom box enter: Select source_id, parallax, parallax_error from gaiadr2.gaia_source where parallax>100 AND parallax_error/parallax<0.1 order by parallax desc
    - This query returns the 1722 objects observed with Gaia at less than 10 pc and good parallax determinations. Closest objects come first.

- **Select / count**
  - Use count(*) to figure out how many rows there are in a table
    - SELECT COUNT(*) FROM gaiadr1.tgas_source

- **Creating new columns**
  - In the bottom box enter:
    - Select top 5 source_id, pmra,pmdec,sqrt(power(pmra,2)+power(pmdec,2)) as pm_tot from gaiadr2.gaia_source
    - NOTES:
      - “AS” can be used to rename a column.

- **Grouping**
  - For histogram-like functionality, you can compute factor sets, i.e., subsets that have identical values for one or more columns, and you can compute aggregate functions for them.
  - In the bottom box enter:
    - SELECT COUNT(*) AS n, ROUND(phot_g_mean_mag) AS bin, AVG(parallax) FROM gaiadr1.tgas_source GROUP BY bin ORDER BY bin
    - Here we have grouped all objects with the same G magnitude (taken as an integer). For each bin we have calculated the average parallax. A new table will be created in TOPCAT with the following information.



	n	bin	avg
1	1	4.	0.66856
2	369	5.	9.1984
3	5068	6.	7.9223
4	21013	7.	6.0994
5	62447	8.	4.58802
6	170726	9.	3.50213
7	416334	10.	2.74793
8	744320	11.	2.29715
9	602754	12.	1.98496
10	33666	13.	1.13196
11	248	14.	0.84677
12	63	15.	0.81039
13	27	16.	0.30666
14	9	17.	0.6792
15	4	18.	0.1629
16	1	19.	-0.50291

As expected, brighter stars tend to have larger parallaxes.

- **Joining:**
  - So far, we had a single table. To work with more than one table we need to use the "JOIN" command.
  - In the bottom box enter:
    - SELECT TOP 10 h1.ra, h1.dec, h1.hip, t1.hip FROM public.hipparcos_newreduction AS h1 JOIN public.tycho2 AS t1 USING (hip)
  - NOTES: JOIN is a combination of cartesian product and a select. It yields the cartesian product of the hipparcos and tycho2 tables but only retains the rows in which the hip columns in both tables agree.

All the above is just a very basic introduction to TAP / ADQL. If you want to know more, the following URLs can be useful:

- http://docs.g-vo.org/adql-gaia/html/twoup.pdf
- http://tapvizier.u-strasbg.fr/adql/help.html

Also, more examples of queries can be found by clicking "Examples" in the "Table Access Protocol (TAP) Query" window of TOPCAT.