Tutorial #5: Exploring Gaia data with TOPCAT and STILTS

Mark Taylor,
University of Bristol,
m.b.taylor@bristol.ac.uk

November 14, 2016

TOPCAT: http://www.starlink.ac.uk/topcat/

STILTS: http://www.starlink.ac.uk/stilts/

Mailing list: topcat-user@jiscmail.ac.uk

Contents

1 Pleiades distance determination

In this example we will determine the mean parallax of the stars in the Pleiades open star cluster (M45). A similar procedure is followed in the Gaia DR1 paper [1], sec 5.5.

1.1 Acquire TGAS data in Pleiades region

1. Open the 🔻 **VO|Cone Search** window
(i.e. use the **Cone Search** submenu of the **VO** menu in the main topcat window)

2. Fill in **Keywords**: "`tgas`", and hit **Find Services**

3. There is only one result(?) in this case: `ARI-Gaia`. Select it by clicking on it, and the partial URL of the service appears in the **Cone URL** field

4. **Object Name**: "`pleiades`", hit **Resolve** to fill in sky position fields

5. **Radius**: "5" (degrees)

6. Hit **OK**; new table is loaded into topcat main control window.

7. Use the ⊕ **Graphics|Sky Plot** window to see the positions on the sky
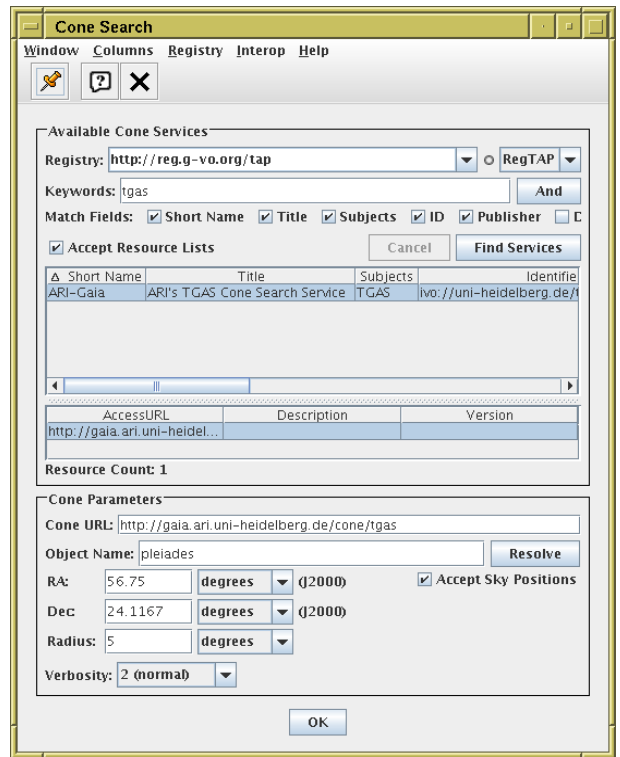
1.2 Identify comoving cluster

1. Plot sources in proper motion space:
☐ **Graphics|Plane Plot** window,
X: "`pmra`", **Y**: "`pmdec`"

2. Note overdensity far from (0,0); use mouse to navigate
(click little [?] button at bottom left for navigation help)

3. Graphically select this comoving cluster as new Subset:
🖉 **Subsets|Draw Subset Region** button,
drag mouse around cluster, hit same button again

4. A **New Subset** dialogue pops up: fill in **New Subset Name**: "`comoving`", **Add New Subset**

5. Look in **Subsets** tab of plot window; turn **All** and **comoving** subsets off/on

6. Look in ◉ **Views|Row Subsets** window to see new subset
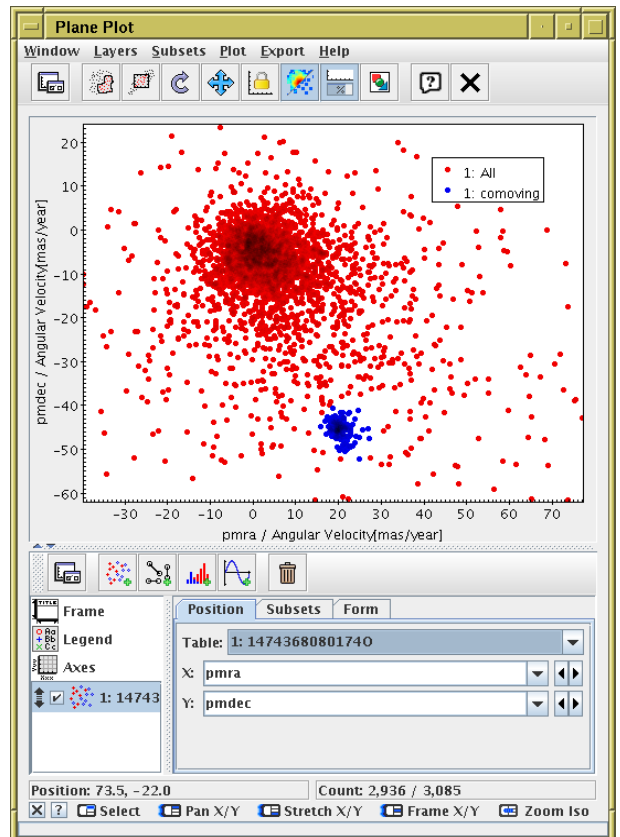
1.3 Refine selection using parallaxes

1. Plot histogram of parallaxes:
   📊 **Graphics|Histogram Plot**,
   X: "`parallax`"

2. In **Subsets** tab, make sure only subset **comoving**, and not **All**, is plotted

3. Rescale plot (✥ or navigate with mouse)

There are some outliers visible, probably not cluster members. We want to create a new subset excluding those parallax outliers.

4. Open 🔴 **Views|Row Subsets** window

5. Create new algebraic subset ➕ **Subsets|New Subset**,
   Subset Name: "cluster",
   Expression: "`comoving && parallax > 6 && parallax < 9`"

6. Plot only these points using **Subsets** tab in histogram plot window

1.4 Examine cluster data

Now we have identified the cluster members (**cluster** row subset) we can do useful things with them.

1. Use the ∑ **Views|Column Statistics** window: select **cluster** in the **Subset for calculations** selector at the bottom, and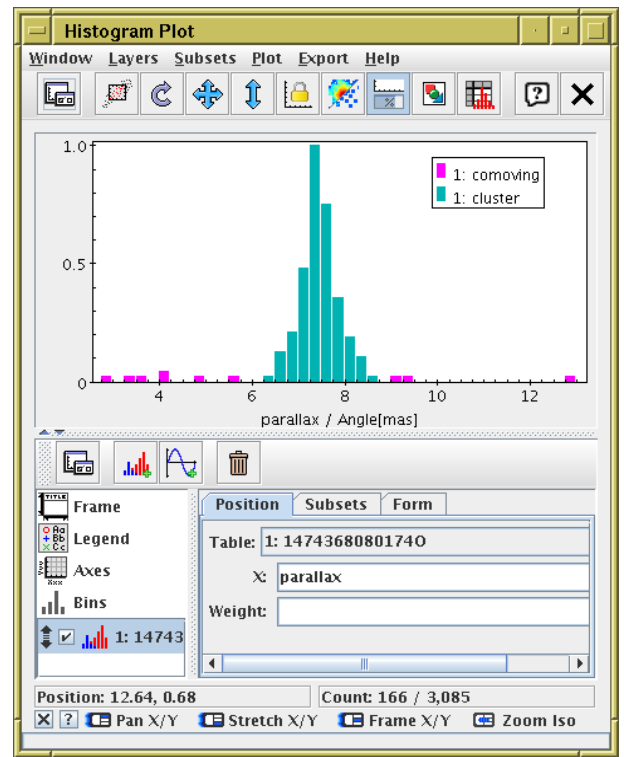 read off **Mean** and **SD** (standard deviation) of columns `parallax`, `pmra`, `pmdec`. How far away are the Pleiades? Hint: distance in parsec is the reciprocal of parallax in arcsec. However, inverting parallax to get distance is problematic if parallax errors are large, say $> 10\%$ (see [2]).

2. Plot the cluster positions in real space. Use the ◯ **Graphics|Sphere Plot** window with **Lon**: "ra", **Lat**: "dec" (or you can use galactic coords "l", "b") and **Radius**: "`1000/parallax`" to show coordinates in 3d space with units of parsec. 3d navigation is a bit harder; click the ? button at the bottom left for help.

3. Plot the proper motions directly as little arrows: in the 🌐 **Graphics|Sky Plot** window use the ↗ button in the **Form** tab, with **Delta Longitude**: "pmra", **Delta Latitude**: "pmdec".

Extras

- As a bonus, you can try instead to identify the cluster members in one step (i.e. combining sections 1.2 and 1.3) by doing a 3-d plot in 2 dimensions of proper motion and one dimension of parallax. Use the ⬡ **Graphics|Cube Plot**, and fill in **X**: "pmra", **Y**: "pmdec", **Z**: "parallax". Set **Shading Mode**: "`transparent`" in the **Form** tab. Use 3d navigation, especially the *re-center* action (right-click) — use the (? button at the bottom left for help).

- View the cluster members on the sky. Open Aladin with all-sky imagery (DSS?), send the whole table to Aladin (select the pleiades table in control window and 📡 **Interop|Broadcast Table**) via SAMP); then send the cluster subset to Aladin (in the **Subsets Window** select the **cluster** subset and use 📡 **Interop|Broadcast Row Subset**).

2 Match Gaia and HST observations

In this example we have a local catalogue from a publication by Gouliermis et al. 2006 [3], available in VizieR as J/ApJS/166/549. This contains about 100,000 sources observed by the ACS instrument on the Hubble Space Telescope at epoch ≈ J2004.6 of stars in NGC346, a cluster in the Small Magellanic Cloud. We match these positions with positions in the main Gaia catalogue at J2015.0 (not just TGAS this time, since we're not using proper motions or parallaxes) and see what we can find out about the relative astrometry of the two catalogues.

2.1 Acquire HST observations

There are various ways to do this, but here we will use TOPCAT's VizieR dialogue window, which talks directly to the VizieR catalogue service.

1. Open 🔴 **VO|VizieR Catalogue Service** window

2. **Object Name**: "ngc346", and **Resolve** to fill in **RA** and **Dec**

3. **Radius**: "1" (degrees)

4. **Maximum Row Count**: "100000" (or some large number)

5. Catalogue selection panel: **By Keyword** tab

6. **Keywords**: "Gouliermis"

7. Select "J/ApJS/166/549"

8. Hit the **OK** button at the bottom. A new table with 99,079 rows should be loaded

2.2 Crossmatch with Gaia

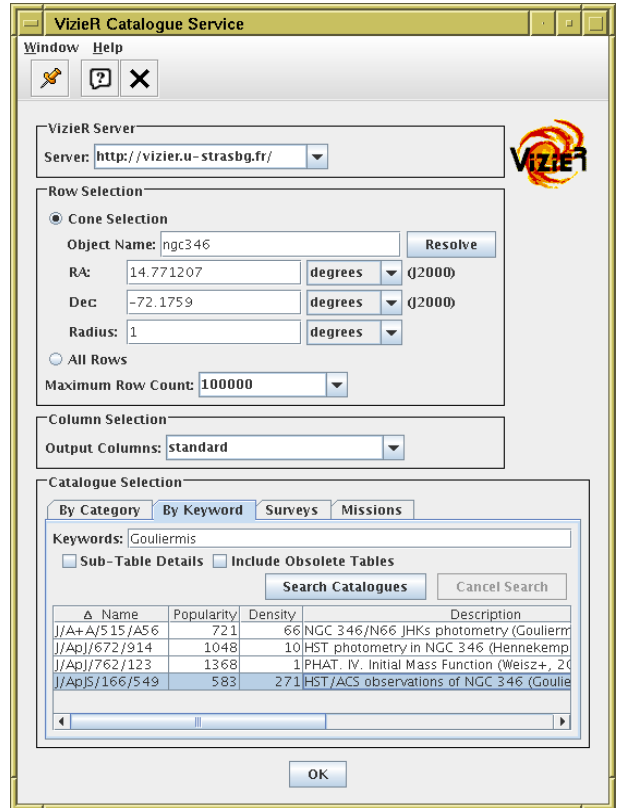Now we want to find associations of the HST objects with sources from Gaia DR1. Use the CDS X-Match service from TOPCAT. This uploads a local table to the CDS X-Match service, where the match is made against the Gaia DR1 catalogue (or any other catalogue in VizieR). The resulting matched catalogue is then received as a new table in TOPCAT.

1. Open the ❌ **VO|CDS Upload X-Match** window

2. **VizieR Table ID/Alias**: "GAIA DR1"

3. **Input Table**: "J_ApJS_166_549_table2" (or whatever the HST table is called)

4. **RA column**: "_RAJ000", **Dec column**: "_DEJ2000" (should be filled in automatically)

5. **Radius**: "1" (arcsec)

6. **Find Mode**: "All"

7. Hit **Go**; within a few seconds, it should inform you that a new table has been loaded, with 8717 rows.

8. Look at the columns of the new table (all HST followed by all Gaia) in the 🔢 **Views|Column Info** window

9. View the new table in a 🌐 **Graphics|Sky Plot**
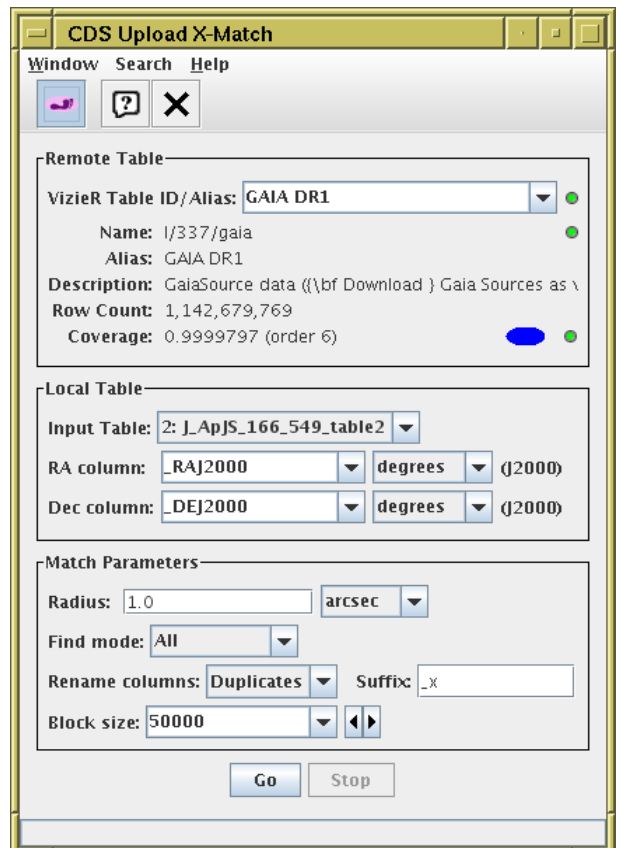
2.3 Visualise the crossmatch

1. Open a ⊕ **Graphics|Sky Plot** window

2. Plot the HST observations: **Table**: "J_ApJS_166_549_table2",
   Lon: "_RAJ2000", **Lat**: "_DEJ2000"

3. Overplot the actual matches. Add a new Pair layer:
   ⚯ **Layers|Add Pair Control** and fill in:
   Table: "2xGAIA DR1" (or whatever the xmatch result table is
   called)
   and both sets of coordinates:
   Lon(1): "_RAJ2000", **Lat(1)**: "_DEJ2000" (HST)
   Lon(2): "ra", **Lat(2)**: "dec" (Gaia)

4. Zoom in to look at the associations. There are too many!

5. You can fiddle around with the tab to make the plot clearer, e.g.
   add a ⬚ **Mark2** layer

Visualising the results of a crossmatch is very often a good idea, unless
you're pretty sure what you're going to get. Here, you can see it was
crucial to understand the results.
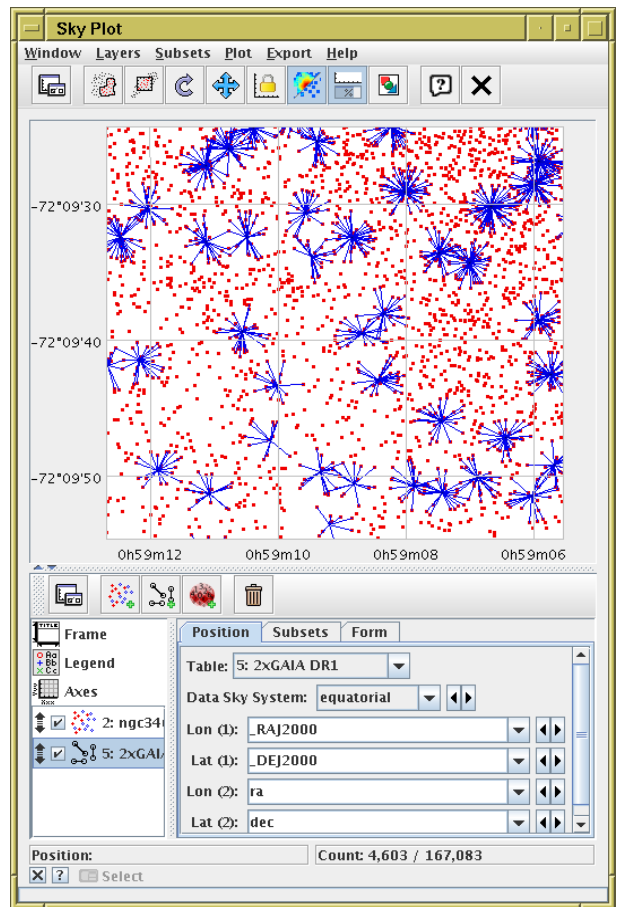
2.4 Investigate and identify matches

1. Add new columns giving RA/Dec discrepancies between HST and
   Gaia positions:
   Open ▦ **Views|Column Info** window,
   then define new columns using ➕ **Columns|New Synthetic
   Column**:
   Name: "deltaRa", **Expression**: "3600*(ra - _RAJ2000)",
   Units: "arcsec"
   Name: "deltaDec", **Expression**: "3600*(dec - _DEJ2000)",
   Units: "arcsec"

2. Use ⬚ **Graphics|Plane Plot** window,
   plot **X**: "deltaRa", **Y**: "deltaDec"

3. Identify overdense region, select as in section 1.2, define new
   subset **true_match**.

4. Go back to the sky associations plot from the previous section,
   and use the **Subsets** tab to visualise which are the true matches.

5. Use the Σ **Views|Column Statistics** window as in section 1.4
   to determine mean and standard deviation for **deltaRa** and
   deltaDec, giving apparent movement of NGC346 between epoch
   of Gouliermis observations (J2004.6) and Gaia (J2015.0). What,
   if anything, can you conclude?

6. Make a colour-magnitude diagram combining HST and Gaia
   photometry:
   Use ⬚ **Graphics|Plane Plot** window, plot **X**: "Vmag-Imag",
   Y: "phot_g_mean_mag", display **true_match** subset only.
   Use the ▦ **Axes** control, **Coords** tab, **Y Flip** checkbox to flip
   it the right way round. Are there outliers? What can you find
   out about them?
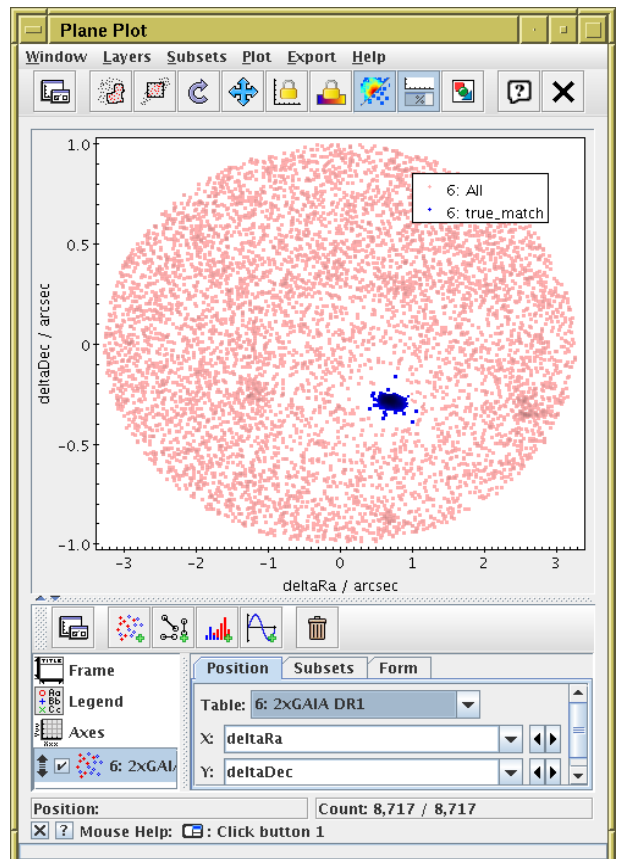
3   TGAS-Hipparcos colour-magnitude diagram

This example uses TOPCAT's TAP window to look at the Gaia DR1 archive, and obtain colour-magnitude information from combined TGAS and Hipparcos observations. Without having to understand how you did it!

3.1   Locate Gaia TAP service

1. Open the TAP window:
   ![icon] **VO|Table Access Protocol (TAP) Query**

2. Fill in **Keywords**: "`gaia`" and hit **Find Services** button

3. Select the **ARI-Gaia** service (there are other Gaia TAP services containing similar but different datasets with similar but different implementations) The service URL appears in the field at the bottom of the window.

4. Hit the **Use Service** button at the bottom

5. Try out the editing buttons (**Edit** menu or toolbar above the ADQL Text entry area): multiple tabs, undo/redo, paste in table/column names from selected items
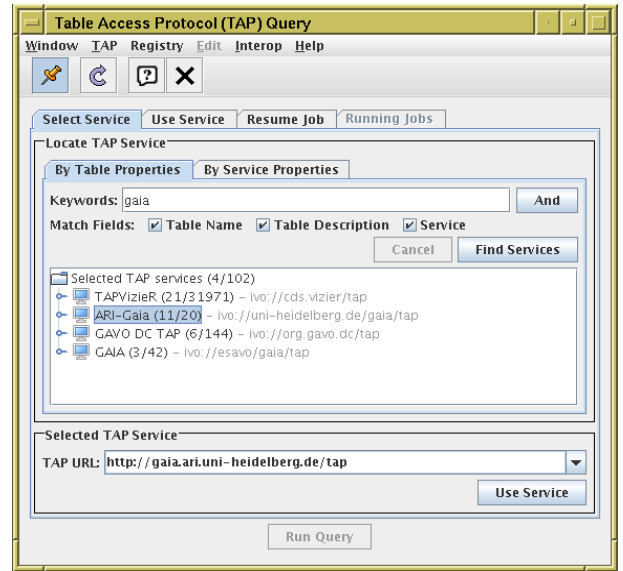
3.2   Explore the TAP service

Use the TAP window to explore the tables that are present and their metadata.

1. Browse the table list on the left, The tables in the `gaiadr1` schema are the ones with Gaia data. The `TAP_SCHEMA` tables are internal housekeeping ones.

2. Highlight tables and look at the **Table** and **Columns** tabs, to see information about what tables are available and what columns they have.

3. Look in the **Service** tab to see information about the service

4. Look in the **Hints** tab for a very basic ADQL cheat sheet

5. Type in some very simple ADQL:
   "`SELECT TOP 10 * FROM gaiadr1.tgas_source`"

6. Note that syntax errors (including partial or misspelt tables/columns) are highlighted in red.

7. Hit **Run Query** to run the query; if successful a new table is loaded
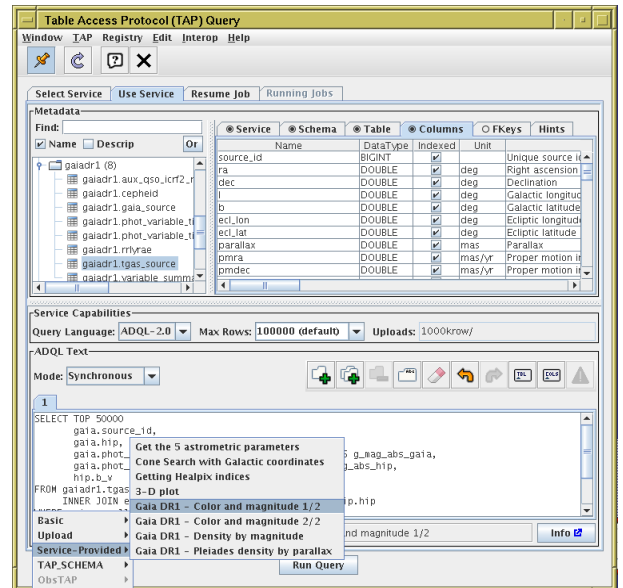
3.3 Use TAP Examples menu

The **Examples** button at the bottom right gives you various lists of example ADQL queries to use as they stand, or to use as starting points for your own queries.

1. Try looking at, and maybe running, some of the **Basic** examples. These are provided by TOPCAT, but the details change depending on which table is currently highlighted in the table browser at the left.

2. If you have a table loaded in TOPCAT, you can use the **Upload** examples too. These are also sensitive to which table is currently selected in topcat's loaded table list

3. The **Service-Provided** examples are more interesting. These (if present) are provided by the TAP service, so can represent scientifically meaningful queries on specific data available.

4. Choose **Examples|Service-Provided|Gaia DR1 - Color and magnitude 1/2** and run it

5. Plot the output table as a scatter plot:
   ▢ **Graphics|Plane Plot** window,
   X: "b_v", **Y**: "g_mag_abs_gaia".
   Use the ▦ **Axes** control, **Coords** tab, **Y Flip** checkbox to make it appear the right way up. This is figure 3 of [1] (the ADQL is also given in Table B.1 of that paper). It's almost too easy.

6. To recover an image more like that in the paper, you can adjust the visualisation:
   In **Form** tab set **Shading Mode**: "Density", **Density Shader**: "Viridis", **Shader Flip**: "True", add ◉ contour layer, increase **Smoothing**, decrease **Level Count**, ...

7. Go back to the ▦ TAP window, and adjust the example a bit: add some more columns to the selection from the `gaia` or `hip` tables and re-run the query

8. Do the plot again, colour-coding the points
   (use **Shading Mode Aux** or **Weighted**, filling in the **Aux** or **Weight** fields with some additional value.
   Fiddle with ▮ **Aux Axis** settings.
   Can you find any interesting relationships?
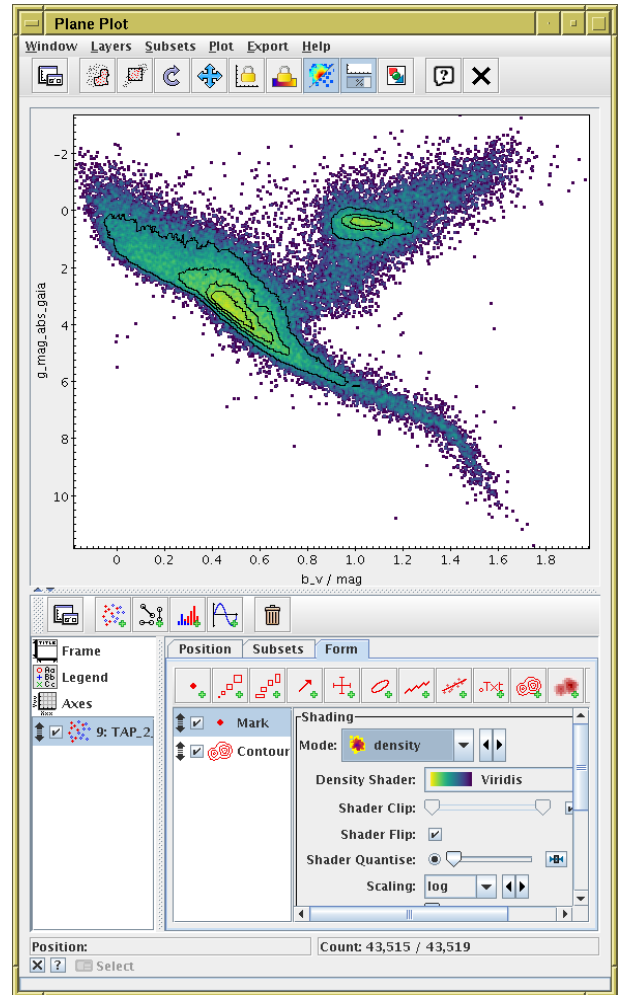
Extra

Go back to the TAP window **Select Service** tab and look for other TAP services containing your favourite data sets. Note: not all data is currently available from TAP (though increasing amounts are), and not all TAP services are equal: some offer more functionality than others, and some don't work very well.

4 Play with TGAS

In this section we will acquire the whole of the TGAS catalogue. It's 2 million rows by 60 columns ($\approx 650\,\text{Mbyte}$). This size is normally fine for TOPCAT or STILTS. But if you don't have a couple of Gbyte free on your computer, or your installation of java/topcat struggles with this much data (or the workshop network won't handle the load), you can just download a subsample.

4.1 Get STILTS working

This section uses STILTS. On a Unix-like OS (e.g. Linux or MacOS) the best thing is to download the files `stilts.jar` and `stilts` from http://www.starlink.ac.uk/stilts/ into the same directory, execute `chmod +x stilts`, and then put that directory on your path or set up an alias to that `stilts` script.

On MS Windows you can just get `stilts.jar` and where the examples say "`stilts ...`" you can write "`java -jar stilts.jar ...`" from a DOS prompt (? or whatever it's called these days).

If it's working, you should be able to use the `calc` command:

```
stilts calc 1+1
    2
```

To make good use of STILTS, you will need the manual at hand. You can find it at http://www.starlink.ac.uk/stilts/sun256/. Using it for reference, you should start from the command you're interested in (Appendix A), and then follow links.

If you can't get STILTS working, you can still download TGAS and play with it in TOPCAT (see section 4.4).

4.2 Get TGAS as single FITS file

TGAS is available from the Gaia archive at ESA, but in the form of 16 separate files. We will download the data and use the STILTS `tcat` command to glue them together. For large amounts of data FITS (or maybe "colfits") is much the best format for TOPCAT and STILTS to use. You can do it in one of the following ways:

Un*x (linux/MacOS) with `wget`:

First use `wget` to download the 16 FITS files:

```
% wget -r -np -nd -N -A fits http://cdn.gea.esac.esa.int/Gaia/tgas_source/fits/
```

(careful to get the flags right — if it seems to be taking a long time it may be downloading far too many files, it should just get 16 FITS files). Then use stilts `tcat` to glue them into one file:

```
% stilts tcat in="`echo TgasSource_000-000-*.fits`" out=tgas_source.fits
```

JyStilts:

Get http://www.starlink.ac.uk/stilts/jystilts.jar and run `java -jar jystilts.jar`, which should give you the interactive prompt ">>>".
Then execute

```
>>> from stilts import *
>>> basename = ("http://cdn.gea.esac.esa.int/Gaia/tgas_source/fits/TgasSource_000-000-%03d.fits")
>>> chunks = [tread(basename % (i)) for i in range(16)]
>>> tgas = tcat(chunks)
>>> tgas.write("tgas_source.fits")
```

Backup:

Download the subsample http://andromeda.star.bris.ac.uk/data/tgas_source.fits

Backup for limited disk space:

Download http://cdn.gea.esac.esa.int/Gaia/tgas_source/fits/TgasSource_000-000-000.fits
(and maybe rename it `tgas_source.fits`)

4.3 Investigate with STILTS

Here are a few things you can try with the `tgas_source.fits` file:

See how big it is:

```
stilts tpipe in=tgas_source.fits omode=count
```

Count how many Hipparcos sources it contains:

```
stilts tpipe in=tgas_source.fits cmd='select hip>=0' omode=count
```

Calculate statistics for some columns:

```
stilts tpipe in=tgas_source.fits
             cmd='keepcols "phot_g_mean_mag parallax pmra pmdec"'
             omode=stats
```

Find the 5 sources with largest parallax:

```
stilts tpipe in=tgas_source.fits cmd='keepcols "source_id ra dec parallax"'
                                 cmd='sorthead -down 5 parallax'
```

Identify hypervelocity stars:

```
stilts tpipe in=tgas_source.fits
             cmd='select parallax_error/parallax<0.2'
             cmd='addcol -units km/h vel 4.74*hypot(pmra,pmdec)/parallax'
             cmd='keepcols "source_id ra dec pmra pmdec vel"'
             cmd='select vel>500'
```

You can do plots as well. Note that these plots are interactive: you can resize the windows and navigate with the mouse just like in topcat. If instead you add the parameter `out=foo.png` it will write the image directly to a PNG file instead. Similarly for `pdf`, `eps`, `jpeg`, `gif`.

Plot all the points on the sky:

```
stilts plot2sky in=tgas_source.fits layer1=mark lon1=ra lat1=dec
```

Plot points weighted by observation count, Aitoff, galactic::

```
stilts plot2sky in=tgas_source.fits
                projection=aitoff datasys=equatorial viewsys=galactic
                layer1=mark lon1=ra lat1=dec shading1=weighted weight1=phot_g_n_obs
```

Plot histogram of parallaxes:

```
stilts plot2plane layer1=histogram in1=tgas_source.fits x1=parallax ylog=true
```

STILTS does a ton of other stuff that there's not space to explain here. Here is an example of downloading some data from the VizieR TAP service and crossmatching it locally against TGAS, to identify Gliese sources with no apparent TGAS counterpart (not necessarily the best way to do it):

```
stilts tapquery tapurl='http://tapvizier.u-strasbg.fr/beta/TAPVizieR/tap' sync=true
                adql='select * from "V/70A/catalog"'
                out=gliese.fits
stilts tskymatch2 in1=gliese.fits ra1=_ra_icrs dec1=_de_icrs
                  in2=tgas_source.fits ra2=ra dec2=dec
                  error=5 join=1not2
                  out=match.fits
```

4.4　All-sky plots with TOPCAT

If you've got this far, or you're bamboozled by STILTS, you can load `tgas_source.fits` into TOPCAT and play around with visualising the whole dataset interactively.

1. Use the ⊕ **Graphics|Sky Plot** window

2. Use the ⊞ **Axes** control to change the **Projection** (e.g. Aitoff) and **View Sky System** (e.g. Galactic)

3. Experiment with the settings in **Form** tab; Marker **Shape** and **Size** (may only be visible when you zoom in), different **Shading Modes** and their options. When you use **Aux** or **Weighted** shading modes, use the 🔲 **Aux Axis** control to adjust settings.

4. Note some of the layer types may be slow with 2 million rows

5. Try using **Aux** or **Weighted** mode with `phot_g_n_obs`. What are you seeing? What's the difference between the two modes? Hint: zoom in/out.
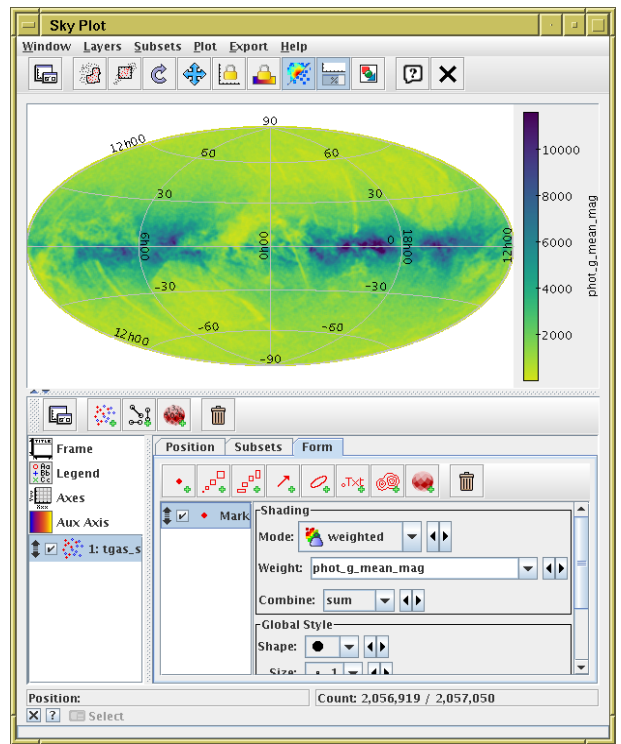
4.5　More fun with STILTS

Anything TOPCAT can do, STILTS can do (sometimes better or more scalably). You can find command-line STILTS or JyStilts scripts for the earlier sections of the tutorial at http://www.star.bristol.ac.uk/~mbt/gaia/tutorial.html. Try them!

References

[1] Gaia Collaboration: A.G.A.Brown et al., "Gaia Data Release 1. Summary of the astrometric, photometric and survey properties", Astronomy and Astrophysics Gaia special issue, `arXiv:1609.04172`

[2] C.L.Bailer-Jones, "Estimating distances from parallaxes" PASP 127, p.994 (2015) `2015PASP..127..994B`

[3] D.A.Gouliermis, A.E.Dolphin, W.Brandner and Th.Henning, "The Star-forming Region NGC 346 in the Small Magellanic Cloud with Hubble Space Telescope ACS Observations. I. Photometry", ApJS, 166 p.549 `2006ApJS..166..549G`