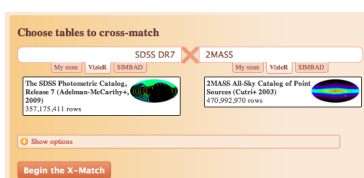


CDS xMatch service documentation

T.Boch, F.-X Pineau, S. Derriere

May 23, 2016



This document provides documentation for the CDS xMatch service. It is available as a PDF file at <http://cdsxmatch.u-strasbg.fr/xmatch/doc/CDSXMatchDoc.pdf>

1 Service

1.1 Overview

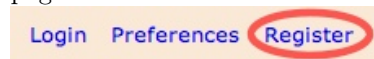
The CDS cross-match service is a new tool allowing astronomers to efficiently cross-identify sources between very large catalogues (up to 1 billion rows) or between a user-uploaded list of positions and a large catalogue.

1.2 Requirements

Users interact with the CDS xMatch service through a Web application which should run fine on any **recent Web browser** having **Javascript and cookies** enabled.

This includes Firefox, Chrome, Opera, Safari and Internet Explorer 8+.

The service is open to anonymous users, but we encourage people to register to the CDS services, using the **Register** link at the top right of the Web page.



We might in the future impose different usage policies for the xMatch service between anonymous and registered users.

We remind you that your account can also be used for the CDS Portal and the CDS Annotations service.

1.3 Available tables

Tables available are the following:

- **All 10,000+ VizieR tables with position.** For each VizieR table, we have kept the default VizieR columns plus positional errors, photometry columns and associated errors if they were not already included.
- **SIMBAD.** A view of the SIMBAD bibliographical database with the following fields: *main object name, position, errors on position, BVRJHK magnitudes, object type, proper motion, spectral type, and number of bibliographical references.* This view is updated each day, so that it reflects the latest updates in the database.
- **user-uploaded tables.** You can upload your own table in one of the following formats : VOTable, FITS table or CSV (Comma separated values) table (see section 4.2 for more details).

1.3.1 Large VizieR tables

Large tables (larger than 10 million rows) have been modified with respect to original VizieR tables. Positional errors have been reformatted to be expressed as an ellipse with 3 parameters: major axis, minor axis and position angle.

Below is the list of available fields for each large VizieR table:

- **2MASS:** *2MASS* identifier, position (*RAJ2000, DEJ2000*) and positional errors (*errHalfMaj, errHalfMin, errPosAng*), *Jmag, Hmag, Kmag, e_Jmag, e_Hmag, e_Kmag, Qfl, Rfl, X, MeasureJD*
- **2MASS6X:** *2MASS* identifier, position *ra, dec* and positional errors (*errHalfMaj, errHalfMin, errPosAng*), *Jmag, Hmag, Kmag, e_Jmag, e_Hmag, e_Kmag, Qflg, Rflg, Xflg, date, genr, l, b, Jsig, Hsig, Ksig, Jsnr, Hsnr, Ksnr, USNO-A2.0, Dopt, PAopt, Bmag, Rmag, Nopt, Bflg, Cflg, Aflg, Ndet, cat, rel*
- **CMC14:** *CMC14* identifier, position (*RAJ2000, DEJ2000*) and positional errors (*errHalfMaj, errHalfMin, errPosAng*), *r.mag, e_mag, Ep, Nt, Jmag, Hmag, Ksmag*
- **DENIS:** *DENIS* identifier, position (*RAJ2000, DEJ2000*) and positional errors (*errHalfMaj, errHalfMin, errPosAng*), *Imag, Jmag, Kmag, e_Imag, e_Jmag, e_Kmag, Image, Strip, Obs.JD*
- **GLIMPSE:** *GLIMPSE* identifier, position (*RAJ2000, DEJ2000*) and positional errors (*errHalfMaj, errHalfMin, errPosAng*), *3.6mag, 4.5mag, 5.8mag, 8.0mag, e_3.6mag, e_4.5mag, e_5.8mag, e_8.0mag*
- **GSC 2.3:** *GSC23* identifier, position (*RAJ2000, DEJ2000*) and positional errors (*errHalfMaj, errHalfMin, errPosAng*), *Fmag, jmag, Vmag, Nmag, Umag, Bmag, e_Fmag, e_jmag, e_Vmag, e_Nmag, e_Umag, e_Bmag, Epoch, Cl*

- **IRSF MCPS**: *IRSF* identifier, position (*RA*, *Dec*), *Jmag*, *Hmag*, *Ksmag*, *e_Jmag*, *e_Hmag*, *e_Ksmag*, *S*, *Qual*, *pflg*, *aflg*, *Jep*, *Jshrp*, *Jchi*, *XposJ*, *YposJ*, *Jregion*, *Hep*, *Hshrp*, *Hchi*, *XposH*, *YposH*, *Hregion*, *Ksep*, *Ksshrp*, *Kschi*, *XposKs*, *YposKs*, *Ksregion*, *MJD*
- **KIC**: *KIC* identifier, position (*RA*, *Dec*), *umag*, *gmag*, *rmag*, *imag*, *zmag*, *Jmag*, *Hmag*, *Kmag*, *kepmag*, *pmRA*, *pmDec*, *sg*, *v*, *aq*, *fc*
- **LMCPS**: position (*ra*, *dec*), *Umag*, *Bmag*, *Vmag*, *Imag*, *e_Umag*, *e_Bmag*, *e_Vmag*, *e_Imag*, *flag*
- **NOMAD**: *NOMAD1.0* identifier, position (*RAJ2000*, *DEJ2000*) and positional errors (*errHalfMaj*, *errHalfMin*, *errPosAng*), *Bmag*, *Vmag*, *Rmag*, *Jmag*, *Hmag*, *Kmag*, *YM*, *r*, *pmRA*, *pmDE*, *spRA*, *spDE*
- **PPMX**: *PPMX_Jname* identifier, position (*RAJ2000*, *DEJ2000*) and positional errors (*errHalfMaj*, *errHalfMin*, *errPosAng*), *Bmag*, *Vmag*, *Jmag*, *Hmag*, *Kmag*, *e_Bmag*, *e_Vmag*, *e_Jmag*, *e_Hmag*, *e_Kmag*, *pmRA*, *pmDE*, *epRA*, *epDE*, *epma*, *epmd*, *Cmag*, *Rmag*, *GoodLSQfit*, *SubSet*
- **PPMXL**: *PPMXL_ID* identifier, position (*RAJ2000*, *DEJ2000*) and positional errors (*errHalfMaj*, *errHalfMin*, *errPosAng*), *Jmag*, *Hmag*, *Kmag*, *e_Jmag*, *e_Hmag*, *e_Kmag*, *pmRA*, *pmDE*, *epRA*, *epDE*, *epma*, *epmd*, *b1mag*, *b2mag*, *r1mag*, *r2mag*, *imag*, *fl*
- **SDSS DR7**: *JNAME* identifier, position (*ra*, *dec*) and positional errors (*errHalfMaj*, *errHalfMin*, *errPosAng*), *JNAME*, *ra*, *dec*, *errHalfMaj*, *errHalfMin*, *errPosAng*, *u*, *g*, *r*, *i*, *z*, *u_err*, *g_err*, *r_err*, *i_err*, *z_err*, *cl*, *objID*, *ObsDate* . For this table, we only kept primary sources (ie with mode=1)
- **SDSS DR8**: position (*ra*, *dec*) and positional errors (*errHalfMaj*, *errHalfMin*, *errPosAng*), *u*, *g*, *r*, *i*, *z*, *u_err*, *g_err*, *r_err*, *i_err*, *z_err*, *objID*, *type*, *clean*, *flags*, *quality* and *mjd_med* which is the median value of *mjd_X* where X = u, g, r, i and z.
- **Tycho2**: position (*RA*, *DE*) and positional errors (*errHalfMaj*, *errHalfMin*, *errPosAng*), *BTmag*, *VTmag*, *TYC1*, *TYC2*, *TYC3*, *pmRA*, *pmDE*, *HIP*
- **UCAC3**: *3UC* identifier, position (*RAJ2000*, *DEJ2000*), *radec_err*, *f.mag*, *a.mag*, *e.f.mag*, *e.a.mag*, *EpRA*, *EpDE*, *ot*, *d*, *pmRA*, *pmDE*, *e_pmRA*, *e_pmDE*, *Jmag*, *Hmag*, *Kmag*, *Bmag*, *R2mag*, *Imag*, *Flag*
- **UCAC4**: *UCAC4* identifier, position (*RAJ2000*, *DEJ2000*), *radec_err*, *f.mag*, *a.mag*, *e.f.mag*, *e.a.mag*, *EpRA*, *EpDE*, *ot*, *d*, *pmRA*, *pmDE*, *e_pmRA*, *e_pmDE*, *Jmag*, *Hmag*, *Kmag*, *Bmag*, *Vmag*, *Gmag*, *Rmag*
- **UKIDSS-DR6 GPS**: *JName* identifier, position (*ra,dec*), *jAperMag3*, *hAperMag3*, *k_1AperMag3*, *k_2AperMag3*, *h2AperMag3*, *jAperMag3Err*, *hAperMag3Err*, *k_1AperMag3Err*, *k_2AperMag3Err*, *h2AperMag3*, *sourceID*, *mode*, *epoch*, *mergedClass*

- **UKIDSS-DR7 LAS:** position (ra, dec), $yAperMag3$, $j_1AperMag3$, $hAperMag3$, $kAperMag3$, $yAperMag3Err$, $j_1AperMag3Err$, $hAperMag3Err$, $kAperMag3Err$, $sourceID$, $epoch$, $priOrSec$, $mergedClass$
- **UKIDSS-DR8 DXS:** $JName$ identifier, position (ra, dec), $jAperMag3$, $kAperMag3$, $jAperMag3Err$, $kAperMag3Err$, $sourceID$, $mode$, $mergedClass$
- **UKIDSS-DR8 GCS:** $JName$ identifier, position (ra, dec), $jAperMag3$, $yAperMag3$, $jAperMag3$, $hAperMag3$, $k_1AperMag3$, $zAperMag3Err$, $yAperMag3Err$, $jAperMag3Err$, $hAperMag3Err$, $k_1AperMag3Err$, $sourceID$, $mode$, $epoch$, $mergedClass$
- **UKIDSS-DR8 LAS:** $JName$ identifier, position (ra, dec), $yAperMag3$, $j_1AperMag3$, $hAperMag3$, $kAperMag3$, $yAperMag3Err$, $j_1AperMag3Err$, $hAperMag3Err$, $kAperMag3Err$, $sourceID$, $mode$, $epoch$, $mergedClass$
- **USNO-A2.0:** *USNO-A2.0* identifier, position ($RAJ2000$, $DEJ2000$) and positional errors ($errHalfMaj$, $errHalfMin$, $errPosAng$), $Bmag$, $Rmag$, $GoodPhot$, $Epoch$
- **USNO-B1.0:** *USNO-B1.0* identifier, position ($RAJ2000$, $DEJ2000$) and positional errors ($errHalfMaj$, $errHalfMin$, $errPosAng$), $Bmag1$, $Rmag1$, $Bmag2$, $Rmag2$, $Imag$, $Epoch$, $pmRA$, $pmDE$, spA , spD , $Flags$
- **WISE:** $JNAME$ identifier, positions (ra , dec) and positional errors ($errHalfMaj$, $errHalfMin$, $errPosAng$), $W1mag$, $W2mag$, $W3mag$, $W4mag$, $Jmag$, $Hmag$, $Kmag$, e_W1mag , e_W2mag , e_W3mag , e_W4mag , e_Jmag , e_Hmag , e_Kmag , $cntr$, cc_flags , ext_flg , var_flg , ph_qual , r_2mass

Consult the VizieR service for a detailed description of each field above.

WARNING: some data have been modified from the original catalogues:

- Positions are given in FK5 J2000 (at epoch 2000 if proper motion available)
- Positional errors have been standardized: they are all given in arcseconds in the form of a circle or an ellipse with semi-minor, semi-major axis and position angle
- For catalogues containing proper motions such as the USNO-B1.0, although positions are given at epoch 2000, positional errors do not take into account proper motion uncertainties (will be done in a future release)
- For all catalogues, and especially the USNO-B1.0, a threshold of 40 has been put on magnitudes: higher values are set to 40. (If it is annoying for you, contact us, it can easily be changed).

If those limitations are a problem for you, contact us.

For an overview of the data before submitting a job, consult this page.

SDSS DR7 and DR8

- SDSS DR7 : Only primary source (mode=1) and object of type GALAXY or STAR (cl = 3 or 6) are available.
- SDSS DR8 : Only primary source (mode=1) are available.

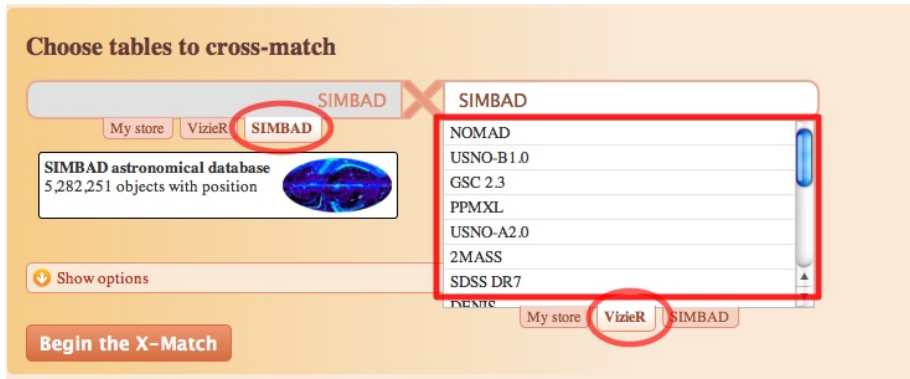
GALEX AIS and MIS Versions of Galex AIS and MIS are the catalogues available here: <http://dolomiti.pha.jhu.edu/uvsky/>.

UKIDSS DR5 For more informations on UKIDSS, see this page, and this page. For on overview of the parameters, browse this page.

1.4 Submitting a cross-match job

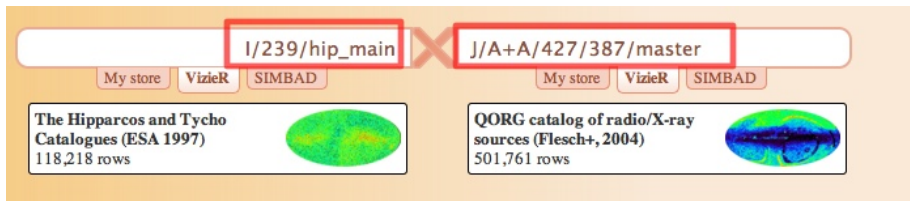
Point your browser to <http://cdsxmatch.u-strasbg.fr/>

1.4.1 Choose tables to cross-match



Select the two tables you wish to cross-match with the interface displayed above. The **My store** tab lists all your uploaded tables.

If you want to use a VizieR table which is not in the list, simply type its full name (eg *I/239/hip-main*) in the text box, as shown below:



1.4.2 Choose cross-match method

Click on **Show options** to open the panel with the cross-match options. This panel allows one to choose the cross-match method and its parameters.

You can perform a cross-match based only on the positions, or a cross-match based on positions taking into account error uncertainties.

Cross-match criteria

By position

Radius:

By position including error

Sigma: (completeness: 99.73 %)

Max. distance:

Details on these two methods can be found in section 2.1.

1.4.3 Choose sky area

Cross-match can be performed on all sources of both tables, or can be restricted to a cone around a given position or object name, or to a given HEALPix cell (in NESTED mode, ICRS frame). If an object name is given, its position will be resolved by Sesame.

Choose the sky area you wish from the panel shown below.

Cross-match area

All sky

Cone

Center:

Radius:

Healpix cell (ICRS, NESTED scheme)

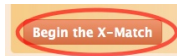
Nside:

Index:

See also section 2.2.

1.4.4 Submit !

Once the tables selected and the options set, click on



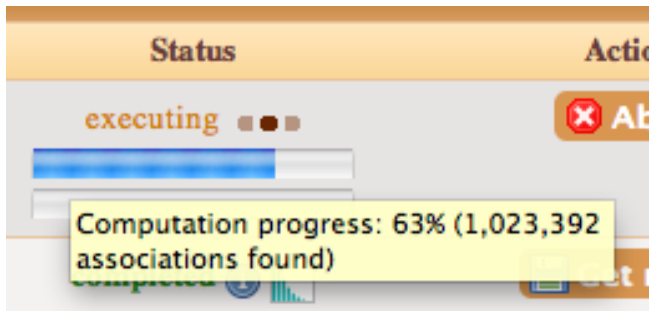
1.5 Monitoring job progress

Once your job is submitted, you can track its progress at the bottom of the page.

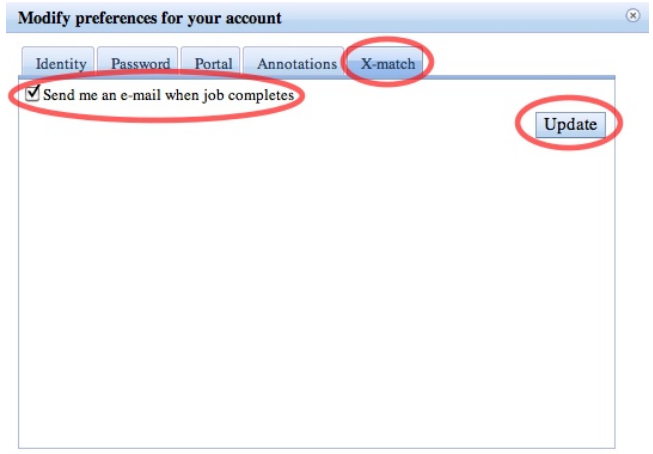


Your job might be queued. Queued jobs wait for a computation worker to be available.

When the job is being executed, a progress bar shows the computation progress and another shows the progress of the result generation.

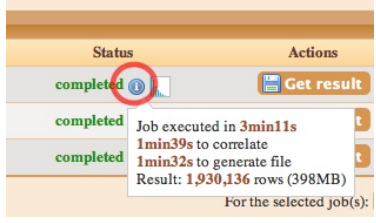


If you are a registered user, you might choose, from your preferences, to receive an email when the computation is over (see below).

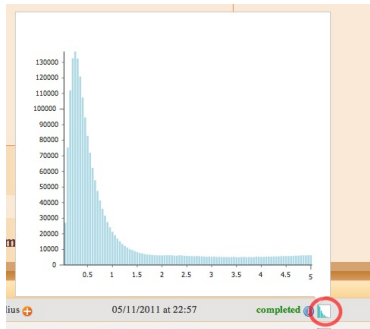


1.6 Retrieving result

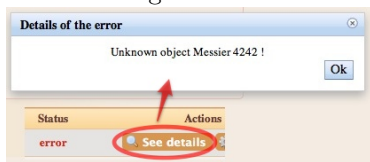
When a job has successfully completed, a short summary of its execution is available by clicking on the info icon .



The histogram of the distances between the computed associations can also be displayed:



If the job has failed, click on **See details** button to get more information on the error origin.



In order to download the result, click on **Get result**. The result can be retrieved either as a CSV (comma-separated values), an ASCII (column-aligned) or a VOTable file. In the result, each row is a concatenation of the following fields: distance between catalogue A and catalogue B sources, distance in sigma (if positional errors were taken into account), fields from catalogue A, fields from catalogue B.

1.7 Limitations

The following limitations apply to the submitted jobs:

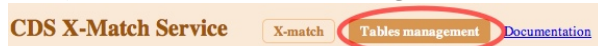
- **Long jobs** are aborted if computation exceeds 100 minutes. This should be sufficient for any cross-match job, even for the largest catalogues, if no error happens.
- **Short jobs** are aborted if computation exceeds 15 minutes. Short jobs are jobs which are restricted to a cone search of Healpix cell. If the largest table of the job is smaller than 20M rows or if the smallest table is smaller than 150,000 rows, the job is also considered as a short job. All other jobs are long jobs.
- The maximum search radius is 120 arcsec for a simple cross-match by position

- When restricting the sky area to a cone-search, the maximum cone radius is 15 degrees.
- Results are removed from our servers 7 days after their submission.

1.8 Uploading a table

You can upload your own table with positions, and use it later in a cross-match job. Your table can be in VOTable, FITS table or CSV format (see section 4.2 for an example).

First, click on the **Tables management** tab

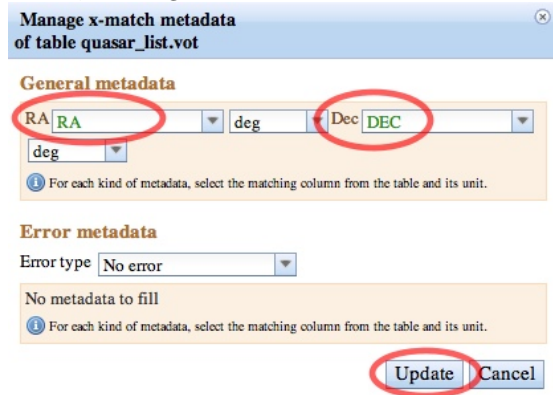


At the bottom of the page, select on your local disk the table to upload and click on **Upload to the store**.



When the upload process is done, the new table will appear in your store list. Before using the uploaded table in a cross-match job, positional metadata must

be set, clicking on  and selecting the relevant fields:.



You will also need to set the error metadata if you plan to use this table for cross-match jobs with positional errors.

Please note that the total size of uploaded tables is limited to 100 MB for anonymous users, and 500 MB for registered users.

If you use the storage space, you must comply to the following guidelines:

- *only data related to astronomical research might be saved or uploaded to the CDS Portal storage space*
- *CDS does not assume any responsibility for loss of data. Please backup your important results locally !*
- *the user storage space must not be considered as a long-term backup facility, but as a convenient facility to facilitate the use of CDS services*
- *in order to preserve disk space, saved data might be removed if the associated account is inactive for more than 6 consecutive months*

1.9 Pre-computed cross-identifications

In order to speed up the computation, the following popular cross-identifications have been pre-computed:

- SDSS DR7 vs 2MASS
- SDSS DR7 vs GSC 2.3
- SDSS DR7 vs PPMXL
- SDSS DR7 vs USNO-B1.0

1.10 xMatch service Release notes

November 20 2013

Hardware problems (resulting in poor execution times) are fixed

January 20 2013

API released

November 28 2012

Web interface now works fine under IE8+

August 31 2012

Added VOTable as output format

March 15 2012

Added ASCII as output format

Version 1.0 - *November 7 2011*

First public release

2 Algorithms

This section describe the parameters the user can use for his cross-match. For more information on the cross-match service internals, see Pineau et al. (2011a) and http://adass2010.cfa.harvard.edu/ADASS2010/incl/presentations/001_2.pdf.

2.1 Cross-match criteria

2.1.1 By position

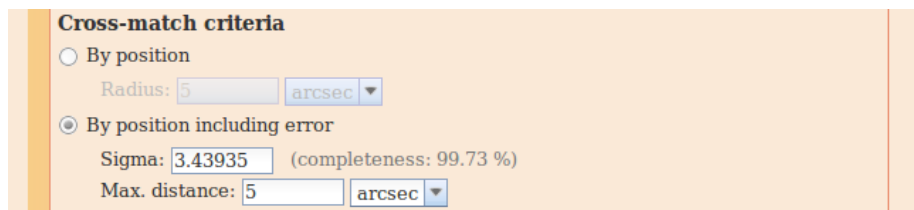


The screenshot shows a web form titled "Cross-match criteria". It has two radio buttons: "By position" (selected) and "By position including error". Under "By position", there is a text input field for "Radius" containing the value "5" and a dropdown menu set to "arcsec".

This cross-match is a simple cross-match: for each source of the catalogue A, it returns the sources of the catalogue B lying at an angular distance less than $Radius$ (θ) from the catalogue A source.

In other words, for each source of the catalogue A, it returns the sources of the catalogue B which are inside the cone of aperture 2θ having for apex (also called vertex) the center of the ICRS frame and for axis the direction of the catalogue A source in the ICRS frame.

2.1.2 By position including error



The screenshot shows the same "Cross-match criteria" form, but with the "By position including error" radio button selected. The "By position" option is now unselected. The "Radius" field is still present but is disabled. The "By position including error" section has a "Sigma" field with the value "3.43935" and the text "(completeness: 99.73 %)" next to it. The "Max. distance" field has the value "5" and a dropdown menu set to "arcsec".

This cross-match is a cross-match based on the positional uncertainties of both catalogue A and catalogue B sources. The selection of the candidates is the same as described in section 3.1.1 of Pineau et al. (2011b).

- $Sigma$ correspond to k_γ in eq. 3 of Pineau et al. (2011b).
- $Max. distance$ is a safeguard preventing from correlating sources having a too high angular separation.

To look for candidates, we first estimate the maximum of the semi-major axis of errors on the positions of catalogue B sources by a boxplot: $e_{Bmax} = q_3 + 1.5(q_3 - q_1)$ where q_1 and q_3 are the lower and upper quartiles respectively.

For each source of the catalogue A, the search radius we use to look for catalogue B source candidates is the minimum between $k_\gamma \sqrt{e_A^2 + e_{Bmax}^2}$ (in which e_A is the semi-major axis of the error on the source of the catalogue A), and the watchdog *Max. distance*. We then keep only candidates satisfying eq. 3 of Pineau et al. (2011b).

INFO:

- USNO-B1.0: a systematic error of 0.2 arcsec is quadratically added to positional errors on both e_RAJ2000 and e_DEJ2000

WARNING:

1. Sources having for positional error a NULL (or a NaN) value are ignored.
2. Because of the use of a boxplot on positional errors to avoid outliers, the result of the cross-match of 2 catalogues is not necessarily symmetric: the number of associations between A and B can be different from the number of associations between B and A.

2.2 Cross-match area

2.2.1 All-sky

The screenshot shows a web form titled "Cross-match area". It has three radio button options: "All sky" (which is selected), "Cone", and "Healpix cell (ICRS, NESTED scheme)". Under "Cone", there are input fields for "Center:" and "Radius:" with a "deg" dropdown menu. Under "Healpix cell", there are dropdown menus for "Nside:" (set to 16) and "Index:" (set to 0).

All sources of the catalogue A are cross-matched.

2.2.2 Cone

The screenshot shows the same "Cross-match area" form, but with the "Cone" radio button selected. The "Center:" field now contains the text "NGC 1313" and the "Radius:" field contains the number "3". The "Nside:" and "Index:" fields remain the same as in the previous screenshot.

Only the sources of the catalogue A lying at an angular distance less than *Radius* from the given position (*Center*) are cross-matched.

2.2.3 Healpix Cell

Cross-match area

All sky

Cone

Center:

Radius:

Healpix cell (ICRS, NESTED scheme)

Nside:

Index:

Only the sources of the catalogue A which are inside the HEALPix pixel of given *Index* and *Nside* are cross-matched. The given *Index* is the index defined by the HEALPix Nested scheme in the ICRS frame. For more information on HEALPix, see <http://healpix.jpl.nasa.gov> or Górski et al. (2005).

3 HTTP API

Beside the web interface described in section 1, the cross-match service can be accessed by scripts or programs through an API (Application Program Interface) described in this section.

This API is partly compliant with concepts developed in the IVOA DALI document.

3.1 Base URL

The base URL for the API is:

<http://cdsxmatch.u-strasbg.fr/xmatch/api/v1/sync>.

3.2 Calls

Calls to the API can be performed using either GET or POST methods.

However, if your request involves file uploads, you will be enforced to use the POST method.

3.2.1 Submitting a cross-match job

Compulsory parameters

- REQUEST - only allowed value is `xmatch` (REQUEST=xmatch)
- cat1 - identifier of the first table.
It can either be a URL, the payload of a local file being uploaded or a CDS table identifier (either `simbad` for a view of SIMBAD data, or to point out a given VizieR table. Examples: `vizier:II/311/wise` or `vizier:II/246/out`).

If the table is uploaded or accessed through a URL, it must be in VOTable or CSV format with the positions in J2000 equatorial frame and as decimal degrees numbers.

Here is an example of a CSV file to be uploaded:

```
ra,dec
267.22029,-20.35869
274.83971,-25.42714
275.92229,-30.36572
283.26621,-8.70756
306.01575,33.86756
322.493,12.16703
```

Total size of uploaded tables can not be larger than 100 MB.

- **cat2** - identifier of the second table.
Follows the same rules as **cat1**.
- **distMaxArcsec** - maximum distance in arcsec to look for counterparts.
Maximum allowed value is 180.
- **RESPONSEFORMAT** - format of the result.
Possible values are `votable|json|csv|tsv`.

If *cat1* is an uploaded table or a pointer to a URL, the following parameters are also compulsory:

- **colRA1** - name of the column holding the right ascension. *Example:* `colRA1=ra`.
- **colDec1** - name of the column holding the declination. *Example:* `colDec1=dec`.

If *cat2* is an uploaded table or a pointer to a URL, the following parameters are also compulsory:

- **colRA2** - name of the column holding the right ascension. *Example:* `colRA2=ra`.
- **colDec2** - name of the column holding the declination. *Example:* `colDec2=dec`.

Optional parameters

- **selection** - match selection. Allowed values are `best|all`. Default selection mode is `all`.
`all`: all matches are kept. In this mode, the cross-match is symmetric.
`best`: keep best match in `cat2` for each row of `cat1`. In this mode, the cross-match is not symmetric.
- **MAXREC** - maximum number of rows to return. The server has a hard upper limit (currently set at 2 000 000 records) which can not be overridden by the user.

- cols1 - comma-separated list of columns of cat1 to be kept in output.
Example: cols1=ra,dec,main_id
- cols2 - comma-separated list of columns of cat2 to be kept in output.

Have a look at section 3.7 for some typical usage examples.

3.2.2 Retrieving the available VizieR tables

To get the list of the VizieR tables which are available in the service, make a call to:

```
http://cdsxmatch.u-strasbg.fr/xmatch/api/v1/sync/tables?
action=getVizieRTableNames&RESPONSEFORMAT=txt
```

You can also get back this list formatted as a JSON document:

```
http://cdsxmatch.u-strasbg.fr/xmatch/api/v1/sync/tables?
action=getVizieRTableNames&RESPONSEFORMAT=json
```

Optionally you can ask for *all* tables (default) or only *large*, *notlarge*, *medium* or *small* tables using option *select*. Example:

```
http://cdsxmatch.u-strasbg.fr/xmatch/api/v1/sync/tables?
action=getVizieRTableNames&select=large&RESPONSEFORMAT=json
```

You can then retrieve more information about a specific table:

```
http://cdsxmatch.u-strasbg.fr/xmatch/api/v1/sync/tables?
action=getInfo&tabName=TABLENAME
```

And the list of its columns:

```
http://cdsxmatch.u-strasbg.fr/xmatch/api/v1/sync/tables?
action=getColList&tabName=TABLENAME&RESPONSEFORMAT=votable|json|csv|tsv|txt
```

You can also directly download the information of all tables with one call to:

```
http://cdsxmatch.u-strasbg.fr/xmatch/api/v1/sync/tables
```

Finally, you can also call:

```
http://cdsxmatch.u-strasbg.fr/xmatch/api/v1/sync/tables?action=getPrettyNames
```

```
http://cdsxmatch.u-strasbg.fr/xmatch/api/v1/sync/tables?action=getAliases
```


3.3 Successful queries

If the query was successful, the service will return a HTTP code 200. In addition, if the `RESPONSEFORMAT` was set to `VOTable`, the `VOTable` result document will contain, in the `RESOURCE` element and before the `TABLE` element, the following snippet: `<INFO name="QUERY_STATUS" value="OK" />`.

Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<VOTABLE ... >
  ....
  <RESOURCE type="results">
    <INFO name="QUERY_STATUS" value="OK" />
    <TABLE>
      ...
    </TABLE>
  </RESOURCE>
</VOTABLE>
```

3.4 Overflows

If the number of rows matching exceeds the value of the `MAXREC` parameter, the result will be truncated to `MAXREC` rows.

If the `RESPONSEFORMAT` was set to `VOTable`, this truncation is indicated in the `VOTable` result document by the following snippet: `<INFO name="QUERY_STATUS" value="OVERFLOW" />`.

Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<VOTABLE ... >
  ....
  <RESOURCE type="results">
    <INFO name="QUERY_STATUS" value="OVERFLOW" />
    <TABLE>
      ...
    </TABLE>
  </RESOURCE>
</VOTABLE>
```

3.5 Error handling

If an error occurred during the processing of the cross-match job, the service will return one of the following HTTP codes:

- a code 400 if there was an error in the input parameters (missing parameter, unknown table name, etc), or if something went wrong during the processing of the job.

- a code 503 if the service is overloaded. In this case, try to resubmit your job later.

In addition, a VOTable document is returned, including the following snippet: `<INFO name="QUERY_STATUS" value="ERROR">...</INFO>`. The text value of this `INFO` element provides with an explicit explanation of the origin of the problem.

Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<VOTABLE ... >
  ...
  <RESOURCE type="results">
    <INFO name="QUERY_STATUS" value="ERROR">
      Message: Missing parameter RESPONSEFORMAT.
    ...
  </INFO>
</RESOURCE>
</VOTABLE>
```

3.6 API limitations

Following limitations apply:

- The maximum size of uploaded files can not exceed 100 MB. An error message is raised otherwise.
- If the cross-match computations returns more than 2 million matches, the result will be truncated to 2 million rows. For larger jobs, use the web interface.
- The maximum search radius is 180 arcsec.

3.7 Usage examples

3.7.1 curl

The curl command line below submits a list of position from the file `posList.csv` (positional columns being `RAJ2000` and `DEJ2000`) and looks for counterparts in SIMBAD with a maximum search radius of 10 arcsec. The result is retrieved as a VOTable document.

```
curl -X POST -F request=xmatch -F distMaxArcsec=10 \
-F RESPONSEFORMAT=votable \
-F cat1=@posList.csv -F colRA1=RAJ2000 -F colDec1=DEJ2000 \
-F cat2=simbad http://cdsxmatch.u-strasbg.fr/xmatch/api/v1/sync \
> res.vot
```

3.7.2 wget

The following command line will perform the cross-match between VizieR catalogues V/123A/cv and V/106/lmxbcat with a maximum search radius of 22 arcsec and will retrieve the result as a CSV file.

```
wget -O xmatch-result.csv \  
"http://cdsxmatch.u-strasbg.fr/xmatch/api/v1/sync?REQUEST=xmatch\  
&cat1=vizier:V/123A/cv&cat2=vizier:V/106/lmxbcat\  
&distMaxArcsec=22&RESPONSEFORMAT=csv"
```

3.7.3 Python

The Python script below consumes the API by submitting a list of positions from a VOTable file, looking for counterparts in the 2MASS catalogue (table II/246/out in VizieR).

It requires the Requests library (that we recommend when dealing with HTTP in Python).

```
#!/usr/bin/env python  
  
import requests  
  
r = requests.post(  
    'http://cdsxmatch.u-strasbg.fr/xmatch/api/v1/sync',  
    data={'request': 'xmatch', 'distMaxArcsec': 5, 'RESPONSEFORMAT': 'csv',  
        'cat2': 'vizier:II/246/out', 'colRA1': 'RAJ2000', 'colDec1': 'DEJ2000'},  
    files={'cat1': open('posList.vot', 'r')})  
  
h = open('results.csv', 'w')  
h.write(r.text)  
h.close()
```

3.7.4 Ruby

This Ruby script submits a list of positions stored in the VOTable file `posList.vot` and looks for counterparts in SIMBAD. We limit the number of rows in the result (using the `MAXREC` parameter).

The script requires the `'net/http/post/multipart'` gem

```
#!/usr/bin/ruby  
  
require 'rubygems'  
require 'net/http/post/multipart'  
  
url = URI.parse('http://cdsxmatch.u-strasbg.fr/xmatch/api/v1/sync')  
File.open("./posList.vot") do |posList|
```

```

req = Net::HTTP::Post::Multipart.new url.path,
  "request" => "xmatch",
  "cat1" => UploadIO.new(posList, 'application/x-votable+xml', 'posList.vot'),
  "cat2" => "simbad",
  "distMaxArcsec" => "9",
  "RESPONSEFORMAT" => "votable",
  "colRA1" => "RAJ2000",
  "colDec1" => "DEJ2000",
  "MAXREC" => "10"
req.add_field("User-Agent", "Ruby")
res = Net::HTTP.start(url.host, url.port) do |http|
  response = http.request(req)
  puts response.body
end
end

```

3.7.5 Java

This example requires the `HttpCore` and `HttpClient` modules from the Apache `HttpComponents` project.

It submits a list of positions extracted from the CSV file `postList.csv` and looks for counterparts in the Hipparcos catalogue (designated I/239/hip_main in VizieR) with a maximum search radius of 15 arcsec.

```

import java.io.*;

import org.apache.http.*;
import org.apache.http.client.*;
import org.apache.http.client.methods.*;
import org.apache.http.impl.client.DefaultHttpClient;
import org.apache.http.entity.mime.*;
import org.apache.http.entity.mime.content.*;

public class SubmitJob {

  public static void main(String[] args) {
    HttpClient httpClient = new DefaultHttpClient();
    HttpPost httpPost = new HttpPost("http://cdsxmatch.u-strasbg.fr/xmatch/api/v1/sync");
    MultipartEntity entity = new MultipartEntity( HttpMultipartMode.BROWSER_COMPATIBLE );

    try {
      entity.addPart( "request", new StringBody("xmatch"));
      entity.addPart( "distMaxArcsec", new StringBody("15"));
    }
  }
}

```

```

entity.addPart( "RESPONSEFORMAT", new StringBody("csv"));
entity.addPart( "colRA1", new StringBody("RAJ2000"));
entity.addPart( "colDec1", new StringBody("DEJ2000"));

entity.addPart( "cat1", new FileBody(new File("posList.csv")));
entity.addPart( "cat2", new StringBody("vizier:I/239/hip_main"));

httpPost.setEntity(entity);

HttpResponse response = httpClient.execute(httpPost);

BufferedInputStream in = new BufferedInputStream(
    response.getEntity().getContent());
BufferedOutputStream out = new BufferedOutputStream(
    new FileOutputStream(new File("result.vot")));

// copy result to file
byte[] buffer = new byte[1024];
int len = in.read(buffer);
while (len != -1) {
    out.write(buffer, 0, len);
    len = in.read(buffer);
}
out.close();

}
catch(Exception e) {
    e.printStackTrace();
}
}
}

```

4 FAQ

4.1 Do I need to register to use this service ?

No, the service is open both to registered and anonymous users. However, we recommend to register. Registration process takes only a few minutes and registered users can retrieve their results from whatever computer they want. Moreover, we might decide in the future to restrict usage of the xMatch service for anonymous users.

4.2 Can I cross-match my own list of positions with a VizieR catalogue ?

Yes. Use the **Tables management** tab on the cross-match web page to upload your own table.

Currently, we support VOTable, FITS table and CSV as the table input format. If your table is in another format, we suggest using TOPCAT to convert it to VOTable.

Here is an example of a simple table you can upload :

```
RA,DEC
2.514167,+46.39083
4.107500,+53.945278
4.172083,+54.660556
4.626250,+26.139167
7.465000,+38.324167
9.805417,+3.134167
10.665417,+71.366944
233.190417,+68.866667
113.448750,+52.689444
```

4.3 Which format do I retrieve the result in ?

The result of a cross-match jobs consists in a list of rows, one for each association found.

Each row is a concatenation of the following fields:

- distance between the catalogue A and the catalogue B sources
- distance in sigma (if xmatch by positional errors) between the catalogue A and the catalogue B sources
- fields of the catalogue A
- fields of the catalogue B

The result can be retrieved in one of the following formats : CSV, ASCII or VOTable.

4.4 Where do data from large catalogues come from ?

It comes from special binary files with limited number of columns. Those files have been created from VizieR data.

The selected columns include (at least) VizieR default columns.

4.5 Do I need to let the Web page open during a cross-match computation ?

No, the computation is done remotely, and the whole process is asynchronous. Thus, you can safely close the cross-match Web page after firing up a new job, and come back later to check its status.

4.6 How long are the results of cross-match jobs kept ?

Job results are kept for a duration of 7 days following their submission.

4.7 How can I acknowledge usage of the CDS cross-match service ?

See section 5.1.

4.8 My favourite catalogue/table is not available for cross-matching. What should I do ?

Please contact us (see section 6) and give us some details about the table. We might be able to integrate it in our service.

4.9 Who is the team behind the service ?

The CDS cross-match service is developed and maintained by François-Xavier Pineau, Thomas Boch and Sébastien Derrière.

We also acknowledge the valuable participation of Brice Gassmann and Grégory Mantelet.

5 Acknowledgements

5.1 Cross-match service acknowledgement

If this service was useful to your research, we would appreciate the following acknowledgement:

This research made use of the cross-match service provided by CDS, Strasbourg.

If you integrated the cross-match API into your application or web page, we would appreciate if you could use one of the following icons:





5.2 SIMBAD acknowledgement

If the Simbad database was helpful for your research work, the following acknowledgment would be appreciated:

This research has made use of the SIMBAD database, operated at CDS, Strasbourg, France.

5.3 VizieR acknowledgement

If the access to catalogues with VizieR was helpful for your research work, the following acknowledgment would be appreciated:

This research has made use of the VizieR catalogue access tool, CDS, Strasbourg, France.

Note that acknowledgements for specific catalogues might be found in the ReadMe file.

5.4 Large catalogues acknowledgement

Some of the large catalogues have specific acknowledgements, as listed below.

5.4.1 2MASS

The primary journal reference for 2MASS and its image and catalogue data products is:

The Two Micron All Sky Survey (2MASS)
M.F. Skrutskie, R.M. Cutri, R. Stiening, M.D. Weinberg, S. Schneider, J.M. Carpenter, C. Beichman, R. Capps, T. Chester, J. Elias, J. Huchra, J. Liebert, C. Lonsdale, D.G. Monet, S. Price, P. Seitzer, T. Jarrett, J.D. Kirkpatrick, J. Gizis, E. Howard, T. Evans, J. Fowler, L. Fullmer, R. Hurt, R. Light, E.L. Kopan, K.A. Marsh, H.L. McCallon, R. Tam, S. Van Dyk, and S. Wheelock, 2006, AJ, 131, 1163.

Please include the following standard acknowledgment in any published material that makes use of 2MASS data products:

”This publication makes use of data products from the Two Micron All Sky Survey, which is a joint project of the University of Massachusetts and the Infrared Processing and Analysis Center/California Institute of Technology, funded by the National Aeronautics and Space Administration and the National Science Foundation.”

5.4.2 GALEX

See http://archive.stsci.edu/prepds/bianchi_gr5xdr7/

5.4.3 SDSS DR8

Funding for SDSS-III has been provided by the Alfred P. Sloan Foundation, the Participating Institutions, the National Science Foundation, and the U.S. Department of Energy Office of Science. The SDSS-III web site is <http://www.sdss3.org/>.

SDSS-III is managed by the Astrophysical Research Consortium for the Participating Institutions of the SDSS-III Collaboration including the University of Arizona, the Brazilian Participation Group, Brookhaven National Laboratory, University of Cambridge, Carnegie Mellon University, University of Florida, the French Participation Group, the German Participation Group, Harvard University, the Instituto de Astrofísica de Canarias, the Michigan State/Notre Dame/JINA Participation Group, Johns Hopkins University, Lawrence Berkeley National Laboratory, Max Planck Institute for Astrophysics, Max Planck Institute for Extraterrestrial Physics, New Mexico State University, New York University, Ohio State University, Pennsylvania State University, University of Portsmouth, Princeton University, the Spanish Participation Group, University of Tokyo, University of Utah, Vanderbilt University, University of Virginia, University of Washington, and Yale University.

5.4.4 UCAC4

The upcoming paper

The fourth U.S. Naval Observatory CCD Astrograph Catalog (UCAC4)

by *Zacharias N., Finch C.T., Girard T.M., Henden A., Bartlet J.L., Monet D.G., Zacharias M.I.*

should be cited whenever UCAC4 data is utilized.

5.4.5 UKIDSS

The teams which built WFCAM, processed the data, and implemented the UKIDSS surveys have no proprietary time. When you use UKIDSS data, please credit their efforts with something similar to the following wording:

”The UKIDSS project is defined in Lawrence et al (2007). UKIDSS uses the UKIRT Wide Field Camera (WFCAM; Casali et al, 2007). The photometric system is described in Hewett et al (2006), and the calibration is described in Hodgkin et al. (2009). The pipeline processing and science archive are described in Irwin et al (2009, in prep) and Hambly et al (2008).”

5.4.6 WISE

Published material that makes use of WISE data products should include the following acknowledgement:

”This publication makes use of data products from the Wide-field Infrared Survey Explorer, which is a joint project of the University of California, Los Angeles, and the Jet Propulsion Laboratory/California Institute of Technology, funded by the National Aeronautics and Space Administration.”

6 Feedback

If you have troubles using the CDS xMatch service, or if you have suggestions to improve it, do not hesitate to send us some feedback by email to cds-question@unistra.fr

References

- Górski, K. M., Hivon, E., Banday, A. J., Wandelt, B. D., Hansen, F. K., Reinecke, M., & Bartelmann, M. 2005, *ApJ*, 622, 759
- Pineau, F.-X., Boch, T., & Derriere, S. 2011a, in *Astronomical Society of the Pacific Conference Series*, Vol. 442, *Astronomical Data Analysis Software and Systems XX*, ed. I. N. Evans, A. Accomazzi, D. J. Mink, & A. H. Rots, 85
- Pineau, F.-X., Motch, C., Carrera, F., Della Ceca, R., Derrière, S., Michel, L., Schwobe, A., & Watson, M. G. 2011b, *A&A*, 527, A126