

DS4

Stage 04 CDS progress report

Thomas Boch

On behalf of : *Pierre Fernique, François Bonnarel,
Sébastien Derriere, François Ochsenbein,
Mark Allen*

List of CDS deliverables

- Simple footprint description
- Positional index for VizieR
- Cross-match web service
- Managing data staging in Aladin
- WCS information in JPEG header
- (Semi-)automatic astrometric (re)calibration
- Improvement of IDL/Aladin interface
- Aladin plugin mechanism

Simple footprint description (1)

- Follow-up
- Footprint description format
 - Lots of discussions with our partners from ESO
 - IVOA Note to be finalized during Wednesday Hackathon
- Collaboration with ESO (M. Dolensky, JC Malapert, F. Chéreau) and STScI (T. Donaldson)
- **APT** (Astronomer Proposal Tool) 16.0 uses footprint standard 'version 0.1' to display and manipulate HST instruments' footprints in Aladin

Simple footprint description (2)

- **FoV graphical editor**
 - Work in standby
 - Waiting for an agreement and a stable version of the standard
 - Postponed to stage 5

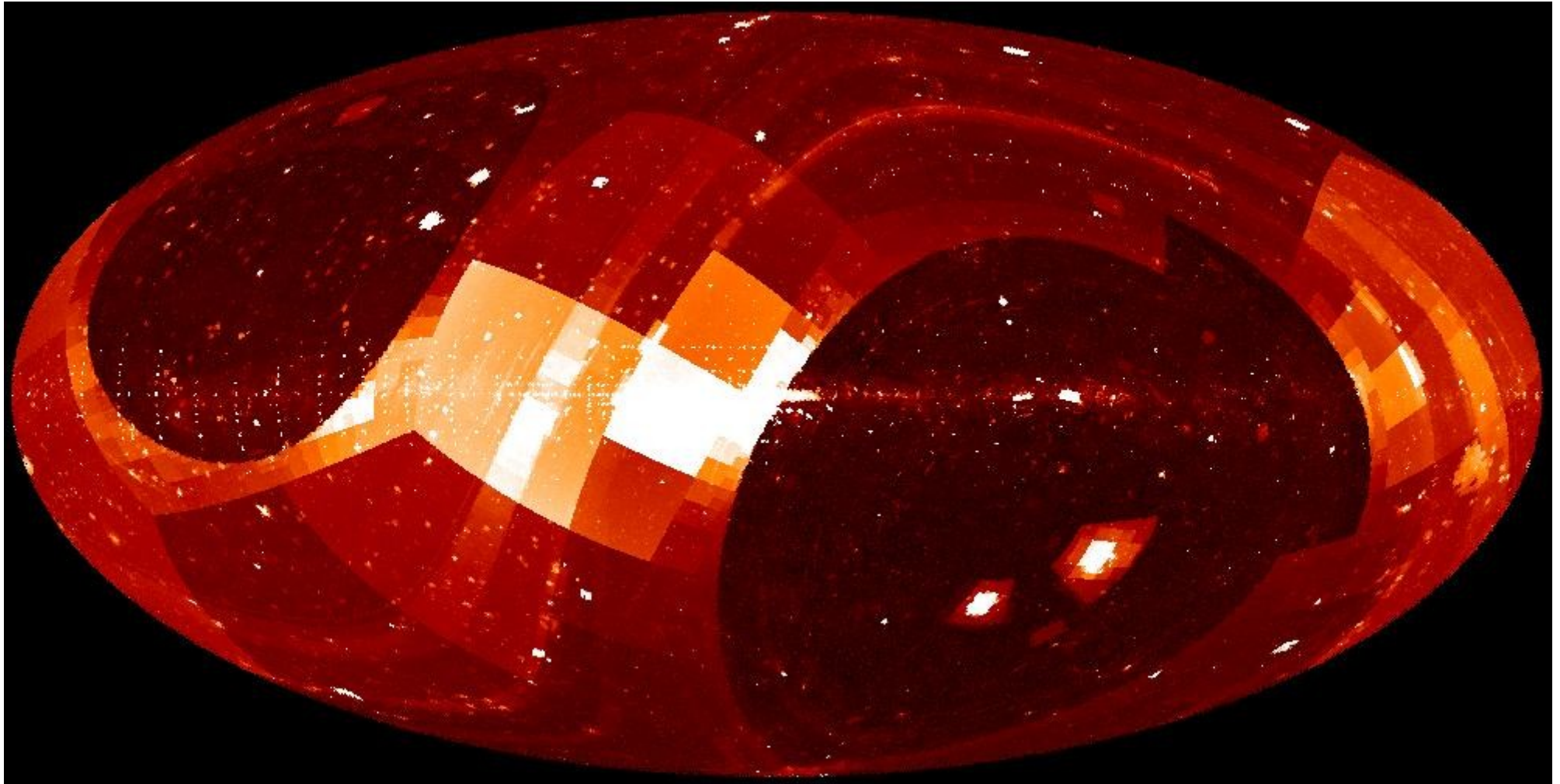
Positional index for VizieR (1)

- Work in progress
- Goals : to allow
 - Fast search by position
 - **Fast cross-match between any 2 VizieR catalogues**
 - Easy creation of density maps for any catalogue with position
- Indexing will be performed using **Qbox**, a hierarchical partitioning scheme allowing to split the celestial sphere into Qbox cells

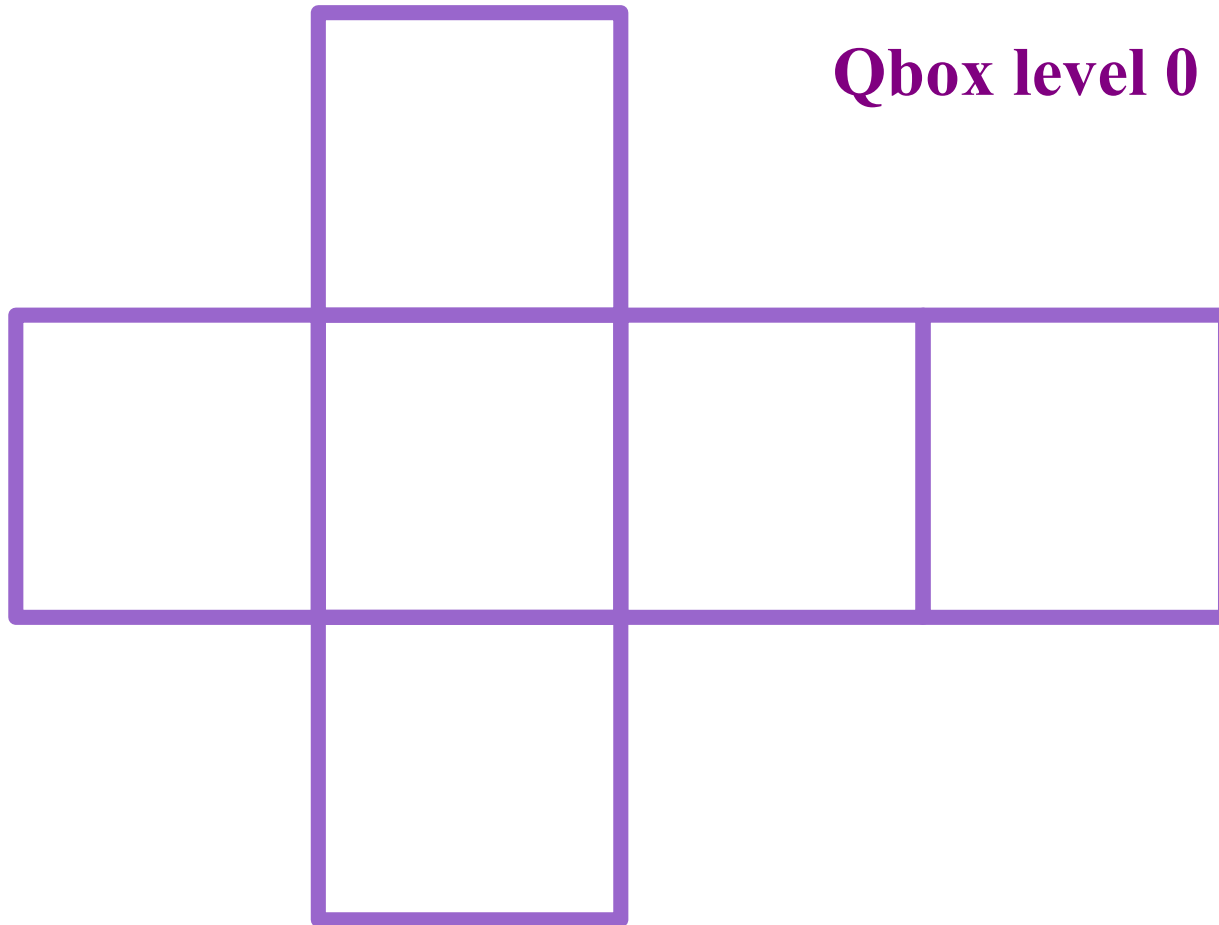
Positional index for VizieR (2)

- Qbox splitting scheme
 - Sphere is projected on a cube (level 0)
 - Then, each face is recursively splitted into 4 boxes

Positional index for VizieR (3)

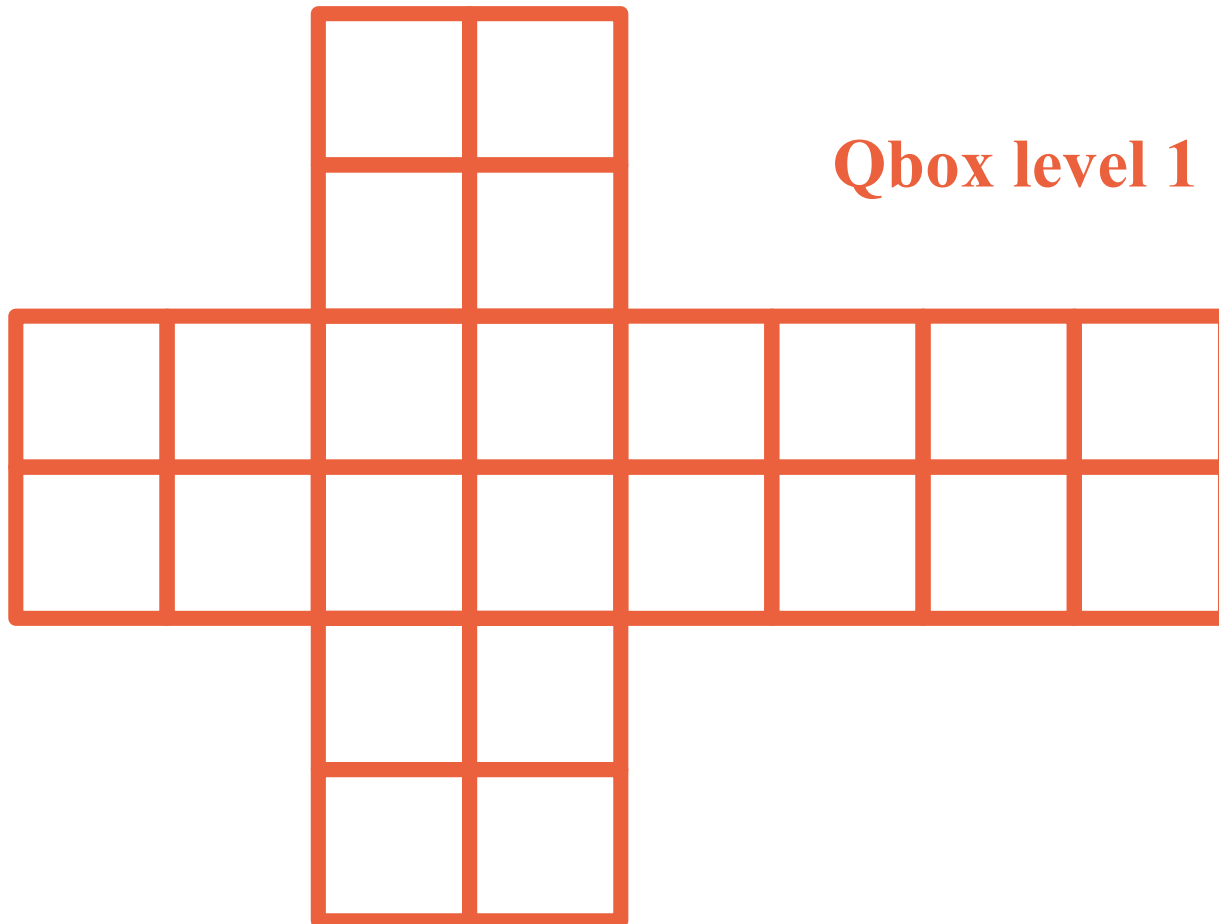


Positional index for VizieR (4)



Qbox level 0 : 6 cells

Positional index for VizieR (5)



Qbox level 1 : 24 cells

Positional index for VizieR (6)

- VizieR : 5.5 billion sources with positions
- Index will be made up of $\approx 25,000$ files
(Qbox level 6 : $6 * 4^6$)
 - In dense fields, indexation might go deeper, up to level 12
 - Index size : $\approx 100\text{GB}$
- Link with characterization
 - For each source position, we store the corresponding position accuracy of the table
- Managing mirrors
 - Re-creation of the whole index very slow
 - Index will most likely stand in a centralized place
- First implementation for stage 5

Cross-match web service (1)

- Goal
 - Create web service version of Aladin's positional cross-match tools
 - ◆ Allows more flexibility
 - ◆ Increases visibility to users
- First prototype developed
 - Allows basic positional cross-match of 2 VOTables available through a URL
 - Improvement of the algorithm
 - ◆ Uses Qbox sky indexing to improve performance

Cross-match web service (2)

- Available as an asynchronous REST service
 - ◆ POST to *xmatch/jobs* to initiate a new xmatch job --> gets back a job identifier
 - ◆ GET *xmatch/jobs/<job-id>* to retrieve current status of the job
 - ◆ When result is available, GET *xmatch/jobs/<job-id>/result* to retrieve it

- Demo

Cross-match web service (3)

- Future developments
 - Stage 5
 - ◆ Provide a SOAP layer for submitting a job and retrieving a xmatch result
 - ◆ Improve scalability
 - Distribute xmatch task on a cluster of PCs
 - ◆ Implement cross-match service taking into account error ellipses
 - ◆ Better documentation, better error reporting
 - Stage 6
 - ◆ Integrate Vizier index in the xmatch scheme
 - ◆ Study links with VOSpace to execute the cross-match near the data

Managing data staging in Aladin

- Goal
 - Adding support to Aladin for asynchronous services (includes a shopping basket functionality, interface with VOSpace)
- Still in an exploratory phase
- Looking for use cases
 - AIDA (image processing) tools
 - Cross-match on large tables
- Postponed to stage 5

WCS information in JPEG headers

- Goal
 - Including WCS information in JPEG headers, allowing bundling of astrometry with quick look images
- Work done
 - Aladin application
 - ◆ Able to read WCS bundled in a JPEG
 - ◆ Successfully tested on SDSS JPEG images
 - Aladin image server
 - ◆ Able to provide JPEG images with embedded WCS info
- FITS header is written/read from JPEG comment field
- Available in Aladin v4

(Semi-)automatic astrometric (re)calibration

- Goal
 - Automatically calibrate (in Aladin) an image using a reference image and/or an astrometric reference catalogue
- Work in progress
- Developed as an Aladin plugin (see above)
- First version released in a few weeks
- Improved version for stage 5

IDL/Aladin interface improvements (1)

- Set of functions allowing to control Aladin from IDL
 - Relies on IDL Java Bridge
- Much better documentation
 - <http://eurovotech.org/twiki/bin/view/VOTech/AladinIDL>
 - All functions are fully described and documented
 - Flash tutorial [available](#)
- 2 new functions:
 - Get table from Aladin to IDL
 - Select objects in Aladin from IDL

IDL/Aladin interface improvements (2)

```
;+
; NAME:
;   SELECT_OBJECTS
;
; PURPOSE:
;   select in Aladin objects of a given table according to their row indexes
;
; CALLING SEQUENCE:
;   SELECT_OBJECTS, aladin_ref, planename, indexes
;
; INPUTS:
;   aladin_ref - reference to the aladin object (return value of launch_aladin
function)
;   planename - name of the plane containing objects to be selected.
;               The name may contain the wildcard character '*'
;   indexes - array of row indexes of each object to be selected in plane plan
ename
;
; EXAMPLE:
;   load_table, aladin, ra, dec, jmag, hmag, kmag, VECNAMES=['ra','dec','jmag'
,'hmag','kmag'], PLANENAME='2MASS'
;   t = where(jmag ne 0 AND jmag gt 18.0)
;   select_objects, aladin, '2MASS', t
;
; MINIMUM IDL VERSION:
;   v6.0 (relies on IDL Java Bridge)
;
; HISTORY
;   Written      Thomas Boch [CDS]      February 2007
```

- Demonstrated to some scientists at Strasbourg observatory
 - Several showed interest

Aladin plugin mechanism

For your information :

- Aladin v4 can be extended by plugins written by external developers
- Motivations
 - Users can easily develop their extensions to Aladin for their own purposes without interacting with Aladin developers
 - Main code stays untouched
- Aladin provides a set of methods allowing to access to its internal objects
- Documentation and plugin repository :
<http://aladin.u-strasbg.fr/java/nph-aladin.pl?frame=plugins>
- Demo