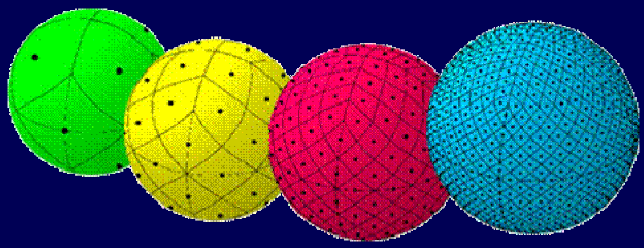
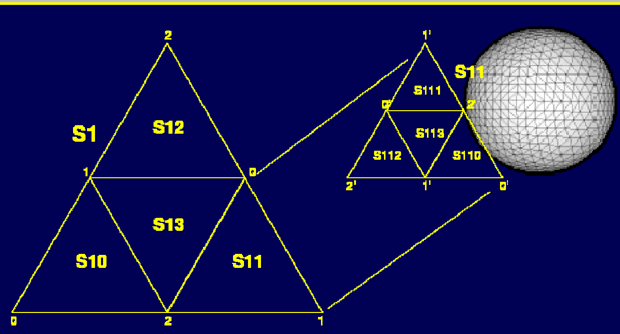
DB multi-depth sky pixelization

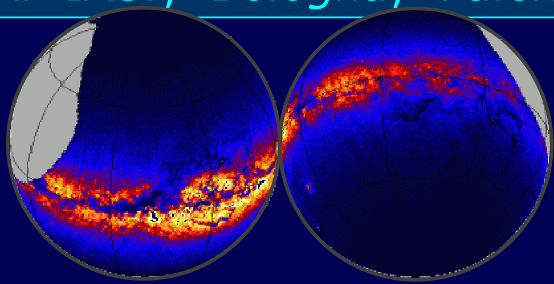
customizing MySQL with HEALPix and HTM - II





Luciano Nicastro[1] & Giorgio Calderone[2]

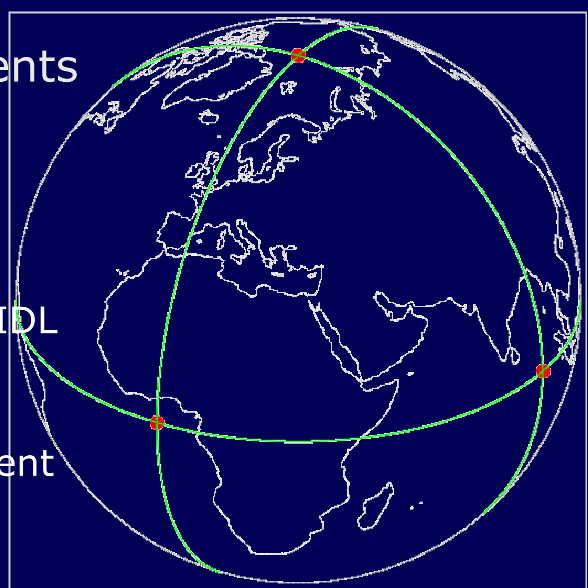
INAF-IASF, [1]*Bologna*, [2]*Palermo*



Malaga workshop, 18-21 May 2009

Summary

- Introduction
  - Relational database: MySQL
    - 1-d and 2-d data indexing
  - Tessellation of the sphere using HTM and HEALPix
  - MCS
- DIF:
  - Installation, system requirements
  - Usage and capabilities
  - Users' key issues
  - Tests and examples:
    - fake sky test results
    - demos on true catalogues using IDL
  - What's next?
    - X-match, full multi-index, …
    - FITS & VOTable direct management
    - Users' requirements/needs



RDB+DMS: Data Base Management System

A DBMS makes easier:

- Archiving,
- Accessing,
- Sharing,
- Protecting data (*of ANY sort!*)

But requires:

- Learning a “new” language (SQL)
- Must become familiar with the database “logic”

Still:

ANY modern project / experiment cannot avoid to use a DBMS in several or all its realization phases!

MySQL

The world's most popular open source database

www.mysql.com

- Open Source (!?)
- Works on more than 20 platforms included Linux, Windows, OS/X, HP-UX, AIX, Netware ...
- High performance, reliability, easy to install, maintain and use
- Flexible and configurable via Views, Triggers, Plug-ins, etc.
- Chosen DB for the new generation of application built on the *LAMP stack* (Linux, Apache, MySQL, PHP / Perl / Python)

Indexing DB tables

1-d

Col_1	Col_2	...	Col_N
v_1	...	...	...
v_2	...	...	...
...	...	...	...
v_N	...	...	...

2-d

Col_1	Col_2	...	Col_N	RA	Dec
v_1	...	...	...	α_1	β_1
v_2	...	...	...	α_2	β_2
...	...	...	...	...	...
v_N	...	...	...	α_N	β_N

Catalogue		Rows	Columns
Optical:	USNO-B	1,045,913,669	30
Optical:	GSC-2.3	945,592,683	53
Infrared:	2MASS	470,992,970	60
X-ray:	2XMM	246,897	379

USNO-B and 2MASS are distributed as files covering 0.1° in Dec each, RA ordered in the slice (*no unique index*).

Indexing DB tables

Spatial index needed for: *roi selection, self-join, spatial-join (multi-cats or multi-epoch cross-matching), spatial analysis.*
The gain in efficiency of an indexed table is 5 orders of mag!

DB availabile indexing include:

B-tree:

In computer science, a B-tree is a tree data structure that keeps data sorted and allows searches, insertions, and deletions in logarithmic amortized time. It is most commonly used in databases and filesystems.

R-tree:

R-trees are tree data structures that are similar to B-trees, but are used for spatial access methods i.e., for indexing multi-dimensional information; for example, the (X, Y) coordinates of geographical data. A common real-world usage for an R-tree might be: "Find all museums within 2 miles of my current location".

Indexing DB tables

Possible indexing of a 2-d (3-d, ...) table:

1. Split the sphere at your personal convenience – no index
2. Index (B-tree) on one single axis (e.g. declination)
3. Use an intrinsic (DB) spatial index like the R-tree
4. Use a function to map 2-d ⇔ 1-d then use B-tree

The most efficient is the last one, i.e. to have the possibility to allow the DB server to manage data on the sphere (2-d) exactly in the same way as it manages one dimensional data tables ⇒ using B-tree schema.

Various sphere coverage (tessellation – *Fig. 1*) functions have been proposed and used, but for Astronomical purposes the most common are *HTM* and *HEALPix* (excluding the geographic-like grids).

MCS: My Customizable Server

A flexible resource for astronomical projects

ross.iasfbo.inaf.it/MCS/

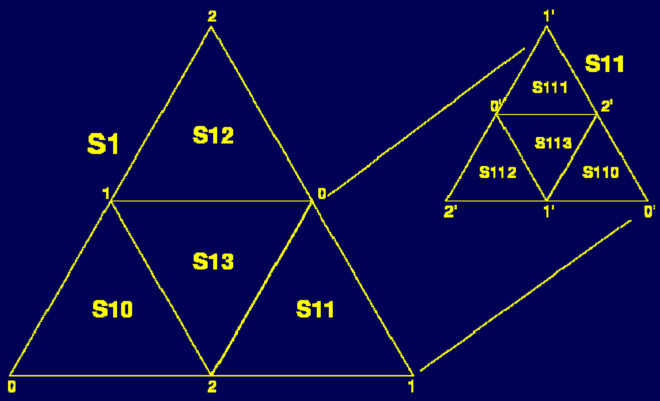
What is it for?

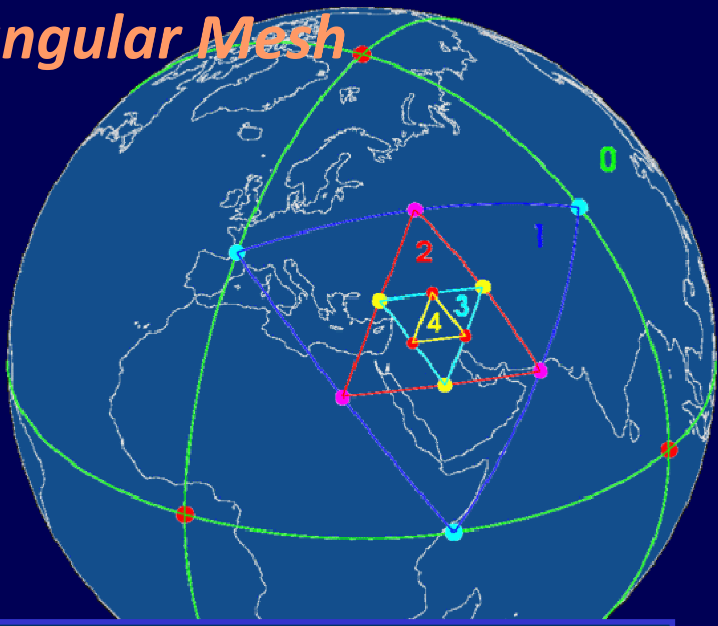
- Multi-thread applications
- Network applications (via TCP)
- Database applications (MySQL)
- Information servers

Moreover:

- Client interfaces for (almost) all languages (e.g. IDL)
- VOTable and FITS file access ⇒ ***VOTPP***
- Privilege system at the record level for MySQL tables ⇒ ***MyRO***
- Automatic indexing of tables with spherical coordinates ⇒ ***DIF***

HTM: Hierarchical Triangular Mesh





HTM: **www.sdss.jhu.edu/htm/**

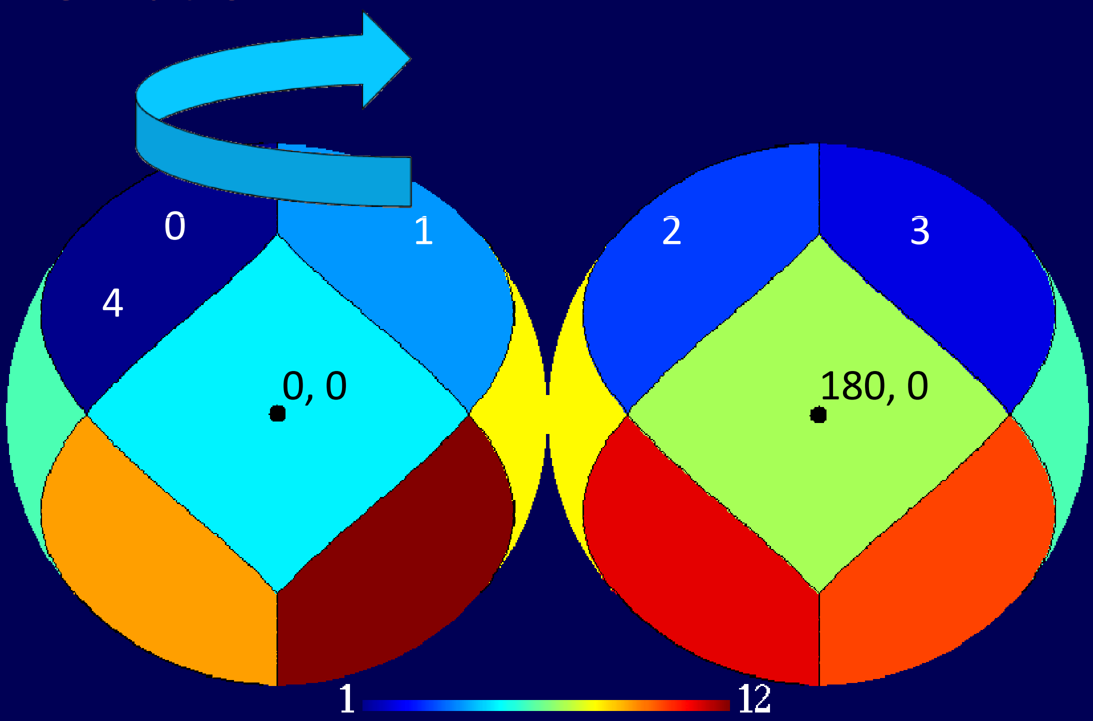
Invented at Johns Hopkins University for the SDSS survey.
The total Nr of pixels (*trixels*) in a map is set by *depth*:

d ∈ [0 , 25] *(up to 30 possible!)* $N_{pix} = 8 \times 4^d$

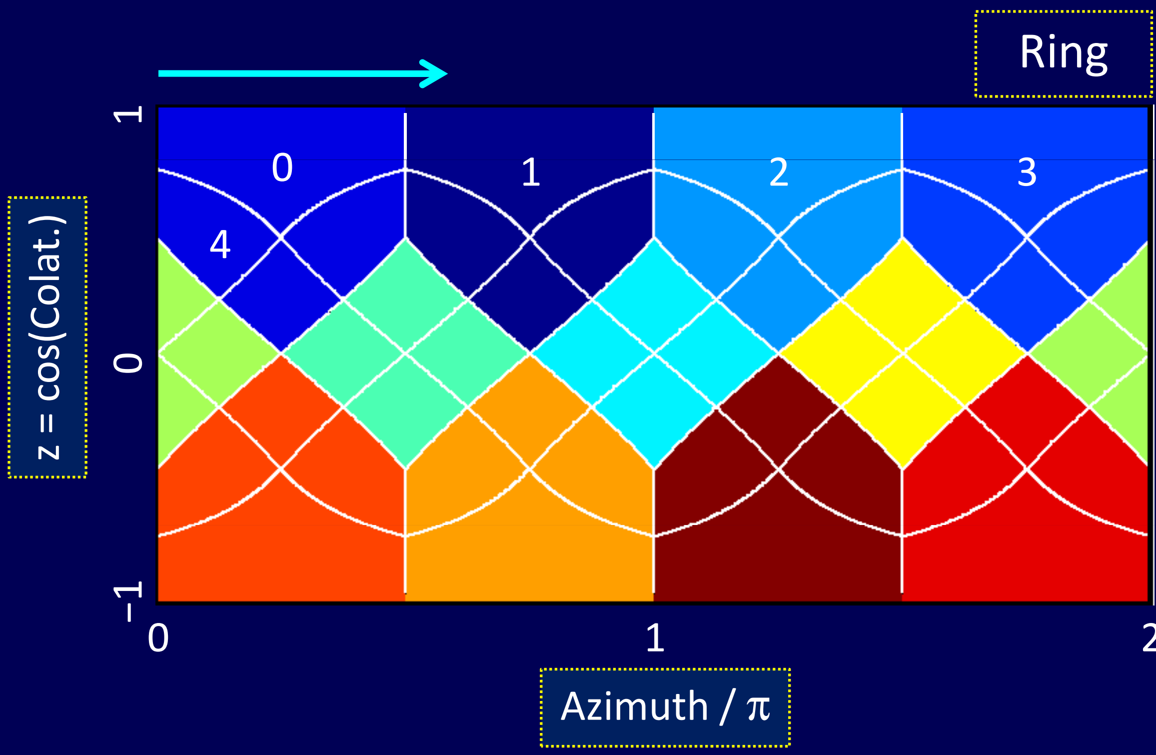
ID_{range}: $[N_{pix} , 2 \times N_{pix} - 1]$

Max res.: 9.0×10^{15} px → 7.7'' $\times 10^{-3}$ (*24 cm on Earth*)

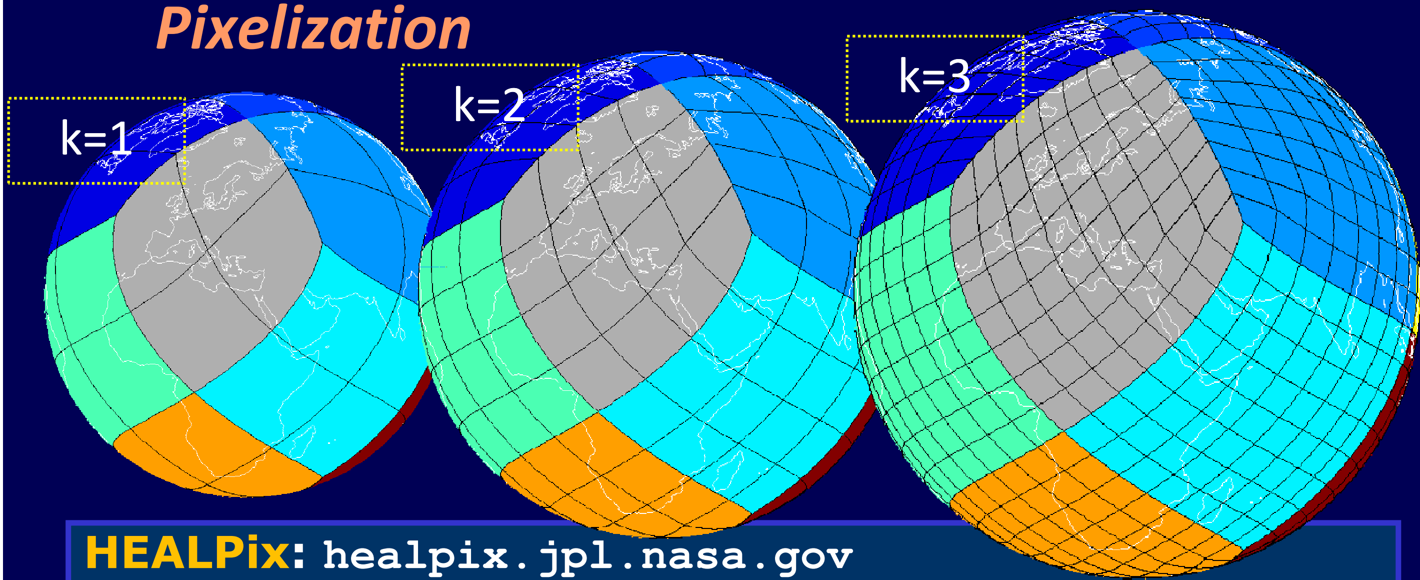
HEALPix: Hierarchical Equal Area isoLatitude Pixelization



HEALPix: Hierarchical Equal Area isoLatitude Pixelization



HEALPix: Hierarchical Equal Area isoLatitude Pixelization



HEALPix: `healpix.jpl.nasa.gov`

Invented at ESO for COBE. Also used by WMAP, Planck, etc.
The total Nr of pixels in a map is set by *order*: $k \in [0\ ,\ 29]$

$N_{side} = 2^k$ $N_{pix} = 12 \times N^2_{side}$

$ID_{range}: [0\ ,\ N_{pix} - 1]$ $\Omega_{pix} = \pi / (3 \times N^2_{side})$

Max res.: 3.46×10^{18} pix $\rightarrow$ 3.93'' $\times 10^{-4}$ (*1.2 cm on Earth*)

DIF: Dynamyc Indexing Facility

DIF is a C++ library + a set of tools aimed at indexing and managing MySQL tables which include spherical coordinates.
It uses the *HTM* and *HEALPix* C++ libraries to implement these tessellation. It's usage does not impact on the accessibility of a table but adds new features to the SQL language which are then usable from any language. For Astronomical data, possible usage include:

1. Catalogues indexing and objects cross-matching (*Fig. 2*)
2. Observations logging and management (e.g. astrometry – *Fig. 3*)
3. Single and multi-frequency data analysis
4. Maps production (*Fig. 4*)
5. All sorts of parameters selection on the table(s) columns

System requirements:
– MySQL version >= 5.1.20
– Perl with DBD::mysql
IDL library available! Requires:
– IDL version >= 5.6, HEALPix lib. (>= 2.0), *MCS user lib.*

DIF: installation

The typical Linux source packages procedure:

• To configure: `./configure --with-mysql-sorce=…`
(`configure --help` to see all the options).
Note: the "dif" script uses Perl so the Perl module DBD::mysql must be present on the system.
• To compile/install: `make` and `make install`
• To add the facility to MySQL: `dif --install` (see `dif --help`).

To index a table using HTM give a command like:

```
dif --index-htm MyDBname MyCatname 6
        "RAcs/3.6E5" "DECcs/3.6E5"
```

6 is the depth parameter ($N_{pix} = 8 \times 4^6 = 32768$).
To index a table using HEALPix give a command like:

```
dif --index-healpix-nested MyDBname MyCatname
    8 "RAcs/3.6E5" "DECcs/3.6E5"
```

8 is the HEALPix *order* ($N_{pix} = 12 \times (2^8)^2 = 786432$).

DIF: things you must consider

- The choice on the pixel resolutions to use depends on the table usage and typical size (or range) of the queried regions. For a catalogue with up to billion objects having 100s – 1000s objects per pixel is OK, for a cross-match usage should stay below 100. Benchmarks!
- The DB response – "elapsed" – time is dominated, in order, by:
  - File access time, i.e. HDs and file-system, in particular ⇒ ***seek time***
  - DB server configuration ⇒ available RAM
  - System & server load
  - (CPU)

MySQL can distribute a single table over various physical file and still see it as a single table ⇒ use `MERGE` DB engine or the `PARTITION BY` statement. However if the files reside on the same HD the advantages are minimal. The only way to mitigate seek-time is:

- Use a multi-disk RAID system (in the order: 10, 0, 1, 5)
- Tune the filesystem parameters: optimization for SELECT, INSERT, UPDATE
- Consider adding "manually" the coordinates to the pixel index

DIF: MySQL extensions

UDF Functions:

DIF.getHTMDepth	(db, tab)
DIF.getHEALPOrder	(db, tab)
DIF.getHEALPNested	(db, tab)
DIF.getRa	(db, tab)
DIF.getDec	(db, tab)
HTMBary	(d, ID)
HTMBaryC	(d, Ra,Dec)
HTMBaryDist	(d, ID, Ra,Dec)
HTMNeighb	(d, ID)
HTMNeighbC	(d, Ra,Dec)
HEALPBary	(s, k, ID)
HEALPBaryC	(s, k, Ra,Dec)
HEALPBaryDist	(s, k, ID, Ra,Dec)
HEALPNeighb	(s, k, ID)
HEALPNeighbC	(s, k, Ra,Dec)
Sphedist	(R1,D1, R2,D2)

HTMLookup	(d, Ra, Dec)
HEALPLookup	(s, k, Ra, Dec)

DIF engine Functions:

DIF_HTMCircle	(Ra, Dec, R)
DIF_HTMRect	(Ra, Dec, S1, *S2*)
DIF_HTMRectV	(Ra1,Dec1, ...)
DIF_HEALPCircle	(Ra, Dec, R)
DIF_HTMNeighbC	(Ra, Dec)
DIF_HEALPNeighbC	(Ra, Dec)
DIF_setHTMDepth	(d)
DIF_setHEALPOrder	(s, k)
DIF_FineSearch	(*var*)
DIF_Sphedist	(Ra1,Dec1,Ra2,Dec2)

```
Select HEALPLookup(0,8,Ra,Dec)
From V Where DIF_HTMRect(0,0,8);
```

DIF: a look from a MySQL session

```
% mysql -u root -p DIF [ --local-infile=1 ]
Enter password:
mysql> show tables;
+----------------+
| Tables_in_DIF |
+----------------+
| dif            |
| tbl            |
+----------------+
mysql> describe dif;
+-------+------------+------+-----+---------+-------+
| Field | Type       | Null | Key | Default | Extra |
+-------+------------+------+-----+---------+-------+
| param | int(11)    | YES  |     | NULL    |       |
| id    | bigint(20) | YES  |     | NULL    |       |
| full  | tinyint(1) | YES  |     | NULL    |       |
+-------+------------+------+-----+---------+-------+
mysql> select count(*) from dif;
+----------+
| count(*) |
+----------+
|        0 |
+----------+
```

DIF: a look from a MySQL session

```
mysql> describe tbl;
+-----------+-----------+------+-----+---------+-------+
| Field     | Type      | Null | Key | Default | Extra |
+-----------+-----------+------+-----+---------+-------+
| db        | char(50)  | NO   | PRI |         |       |
| name      | char(50)  | NO   | PRI |         |       |
| id_type   | int(11)   | YES  |     | NULL    |       |
| id_opt    | int(11)   | YES  |     | NULL    |       |
| param     | int(11)   | YES  |     | NULL    |       |
| Ra_field  | char(100) | YES  |     | NULL    |       |
| Dec_field | char(100) | YES  |     | NULL    |       |
+-----------+-----------+------+-----+---------+-------+
mysql> select * from tbl;
+--------+---------+---------+--------+-------+-----------+------------+
| db     | name    | id_type | id_opt | param | Ra_field  | Dec_field  |
+--------+---------+---------+--------+-------+-----------+------------+
| MyCats | GSC_23  |       1 |      0 |     6 |RAcs/3.6e5| DECcs/3.6e5|
| MyCats | UCAC_2  |       1 |      0 |     6 |RAcs/3.6e5| DECcs/3.6e5|
| MyCats | ASCC_25 |       1 |      0 |     6 |RAcs/3.6e5| DECcs/3.6e5|
| TEST   | fakesky |       2 |      1 |     8 |RAdeg     | DECdeg     |
| TEST   | fakesky |       2 |      1 |    12 |RAdeg     | DECdeg     |
+--------+---------+--------+-------+-------+-----------+------------+
```

DIF: demo::Messier

```
mysql> DROP TABLE IF EXISTS Messier;
mysql> CREATE TABLE Messier (
  M          INT    NOT NULL,
  Type       CHAR(2) DEFAULT '**',
  Const      CHAR(3) DEFAULT '***',
  Mag        FLOAT,
  Ra         FLOAT,
  Decl       FLOAT,
  Dist       CHAR(20),
  App_size   CHAR(20) DEFAULT 'unknown',
  PRIMARY KEY (M));

mysql> LOAD DATA LOCAL INFILE './messier' INTO TABLE Messier;
mysql> select count(*) from Messier;
+----------+
| count(*) |
+----------+
|      110 |
+----------+

mysql> SELECT * FROM Messier;
...
```

DIF: demo::Messier

```
% dif --index-htm test Messier 6 “RA*15E0” Decl

mysql> use test;
mysql> describe Messier;
+----------+---------------------+------+-----+---------+-------+
| Field    | Type                | Null | Key | Default | Extra|
+----------+---------------------+------+-----+---------+-------+
| M        | int(11)             | NO   | PRI |         |       |
| Type     | char(2)             | YES  | MUL | **      |       |
| Const    | char(3)             | YES  |     | ***     |       |
| Mag      | float               | YES  |     | NULL    |       |
| Ra       | float               | YES  |     | NULL    |       |
| Decl     | float               | YES  |     | NULL    |       |
| Dist     | char(20)            | YES  |     | NULL    |       |
| App_size | char(20)            | YES  |     | unknown |       |
| htmID_6  | smallint(5)unsigned | YES  | MUL | 0       |       |
+----------+---------------------+------+-----+---------+-------+
```

DIF: demo::Messier

Circular region about M31 and rectangle in Leo:

```
mysql> SELECT * FROM Messier_htm_6 WHERE
         DIF_HTMCircle(0.7*15, 41.3, 30);
+-----+------+-------+-----+--------+--------+---------+--------------+-------+
| M   | Type | Const | Mag | Ra     | Decl   | Dist    | App_size     |htmID_6|
+-----+------+-------+-----+--------+--------+---------+--------------+-------+
|  31 | GX   | And   | 4.8 | 0.7116 | 41.268 | 2.2 Mly | 192.4'x62.2' | 64538|
|  32 | GX   | And   | 8.7 | 0.7116 | 40.860 | 2.2 Mly | 8.7'x6.4'    | 64539|
| 110 | GX   | And   | 9.4 | 0.6733 | 41.686 | 2.2 Mly | 21.9'x10.9'  | 64571|
+-----+------+-------+-----+--------+--------+---------+--------------+-------+
mysql> SELECT * FROM Messier_htm_6 WHERE
         DIF_HTMRect(0.7*15, 41.3, 60);
...same result...
mysql> SELECT * FROM Messier_htm_6 WHERE
         DIF_HTMRectV(180.,28,  142.5,32,  180.,10,  142.5,6);
+-----+------+-------+------+---------+---------+--------+-----------+-------+
| M   | Type | Const | Mag  | Ra      | Decl    | Dist   | App_size  |htmID_6|
+-----+------+-------+------+---------+---------+--------+-----------+-------+
|  66 | GX   | Leo   |  8.2 | 11.3367 | 12.9917 | 35 Mly | 9.1'x4.1' | 57426 |
|  95 | GX   | Leo   | 10.4 | 10.7333 | 11.7033 | 38 Mly | 7.5'x5.0' | 58216 |
|  96 | GX   | Leo   |  9.1 |   10.78 | 11.8217 | 38 Mly | 7.6'x5.2' | 58216 |
|  65 | GX   | Leo   |  9.3 |  11.315 | 13.0933 | 35 Mly | 8'x1.5'   | 58273 |
| 105 | GC   | Leo   |  9.2 | 10.7967 | 12.5817 | 38 Mly | 5.4'x4.8' | 58356 |
+-----+------+-------+------+---------+---------+--------+-----------+-------+
```

DIF: test::fake_sky

Several fake objects catalogues with up to 3 billion entries:

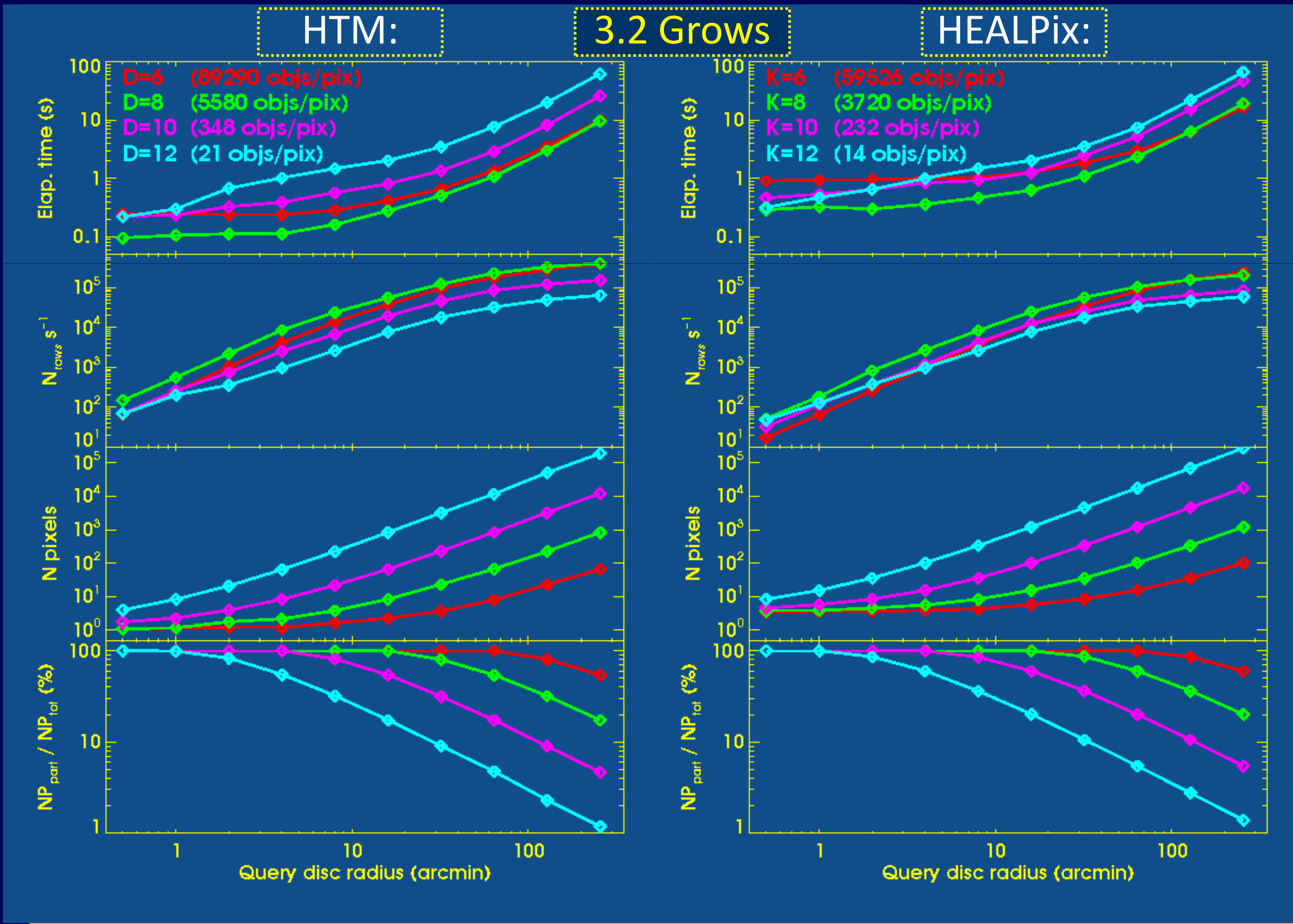
- Random on 4 π
- Random on HTM depth 6 pixelized sphere (32768 trixels) ⇒ ~1° resolution

Used pixelization resolutions:

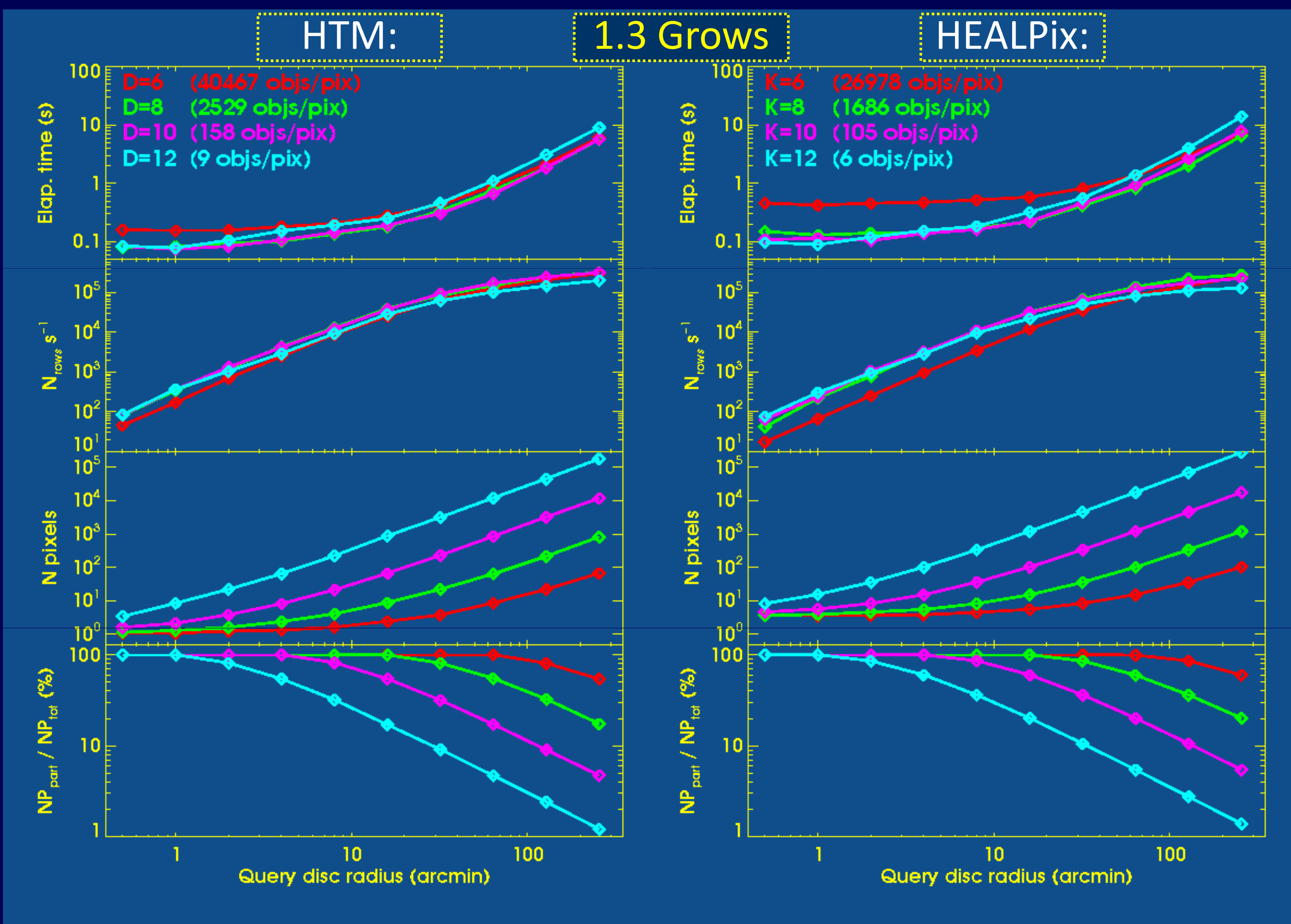
– HTM: 6, 8, 10, 12 – HEALPix: 6, 8, 10, 12

	d / k	Npix	<Area>	Obj/Pix	Bytes
HTM	6	32,768	1.26 deg^2	89,290	2
	8	524,288	283 arcmin^2	5,580	3
	10	8,388,608	18 arcmin^2	348	3
	12	134,217,728	1 arcmin^2	21	4
HEALPix	6	49,152	0.84 deg^2	59,526	2
	8	786,432	189 arcmin^2	3,720	3
	10	12,582,912	12 arcmin^2	232	3
	12	201,326,592	0.7 arcmin^2	14	4

DIF: test::fake_sky – cone queries results



DIF: test::fake_sky – cone queries results



DIF: IDL_demo::ASCC / UCAC / GSC

Catalogues adapted from:

- ASCC-2.5: (*Fig. 5*)
  - astro-photometric catalogue (B, V) of 2.5 Mobjs
- UCAC-2 (2): (*Fig. 6, 7*)
  - astrometric catalogue (mag A in [V,R]) of 48.3 Mobjs
- GSC-2.2 (2): (*Fig. 8, 9*)
  - photometric catalogue (I, R, B, V) of 455 Mobjs

DIF: what's next

- Automatic multi-depth management (*Fig. 10*)
- Cross-matching UDF
- More DB engine functions ⇒ data type related
- FITS and VOTable DB engine
  - x-ray, γ-ray photon-event lists, pixel level image saving, etc.
- *Usage in experiments* ⇒ new features *added*
- *Active involvement of the users*
- *Collaboration with the HTM, HEALPix groups (and MySQL?)*

Contact us reporting criticisms, comments or simply your needs. Have a look at the web page:

MCS, DIF, MyRO, VOTPP
ross.iasfbo.inaf.it/MCS/

Sky tessellation: some examples

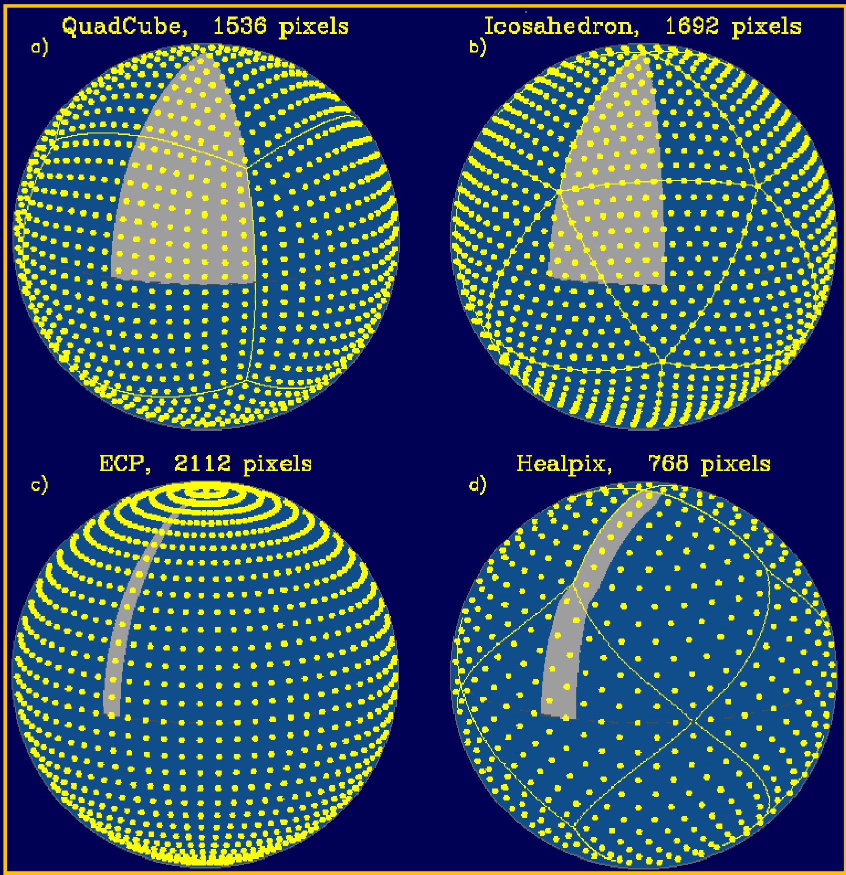


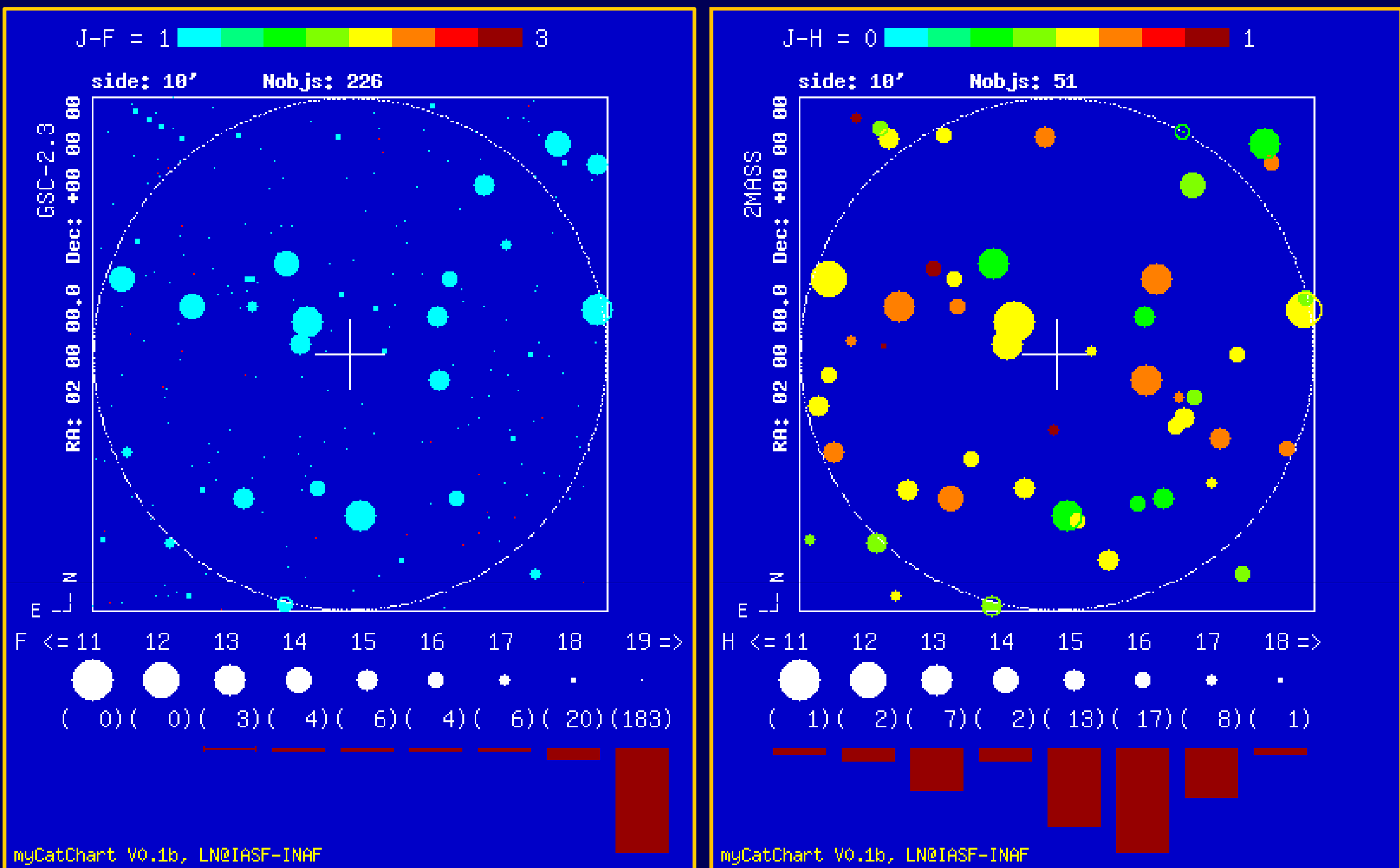
Fig. 1

example: GSC2.3 vs 2MASS

GSC2.3

2MASS

Fig. 2



example: catalogue matching

ROSS – astrometry

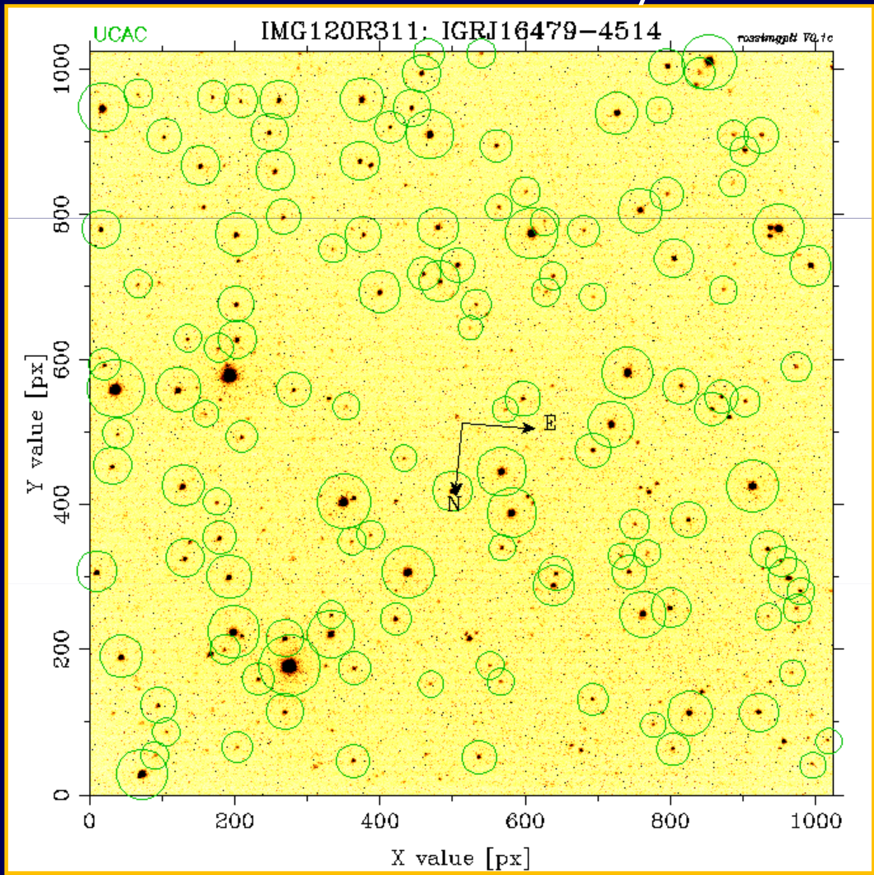


Fig. 3

example: ASCC2.5 density

K = 8

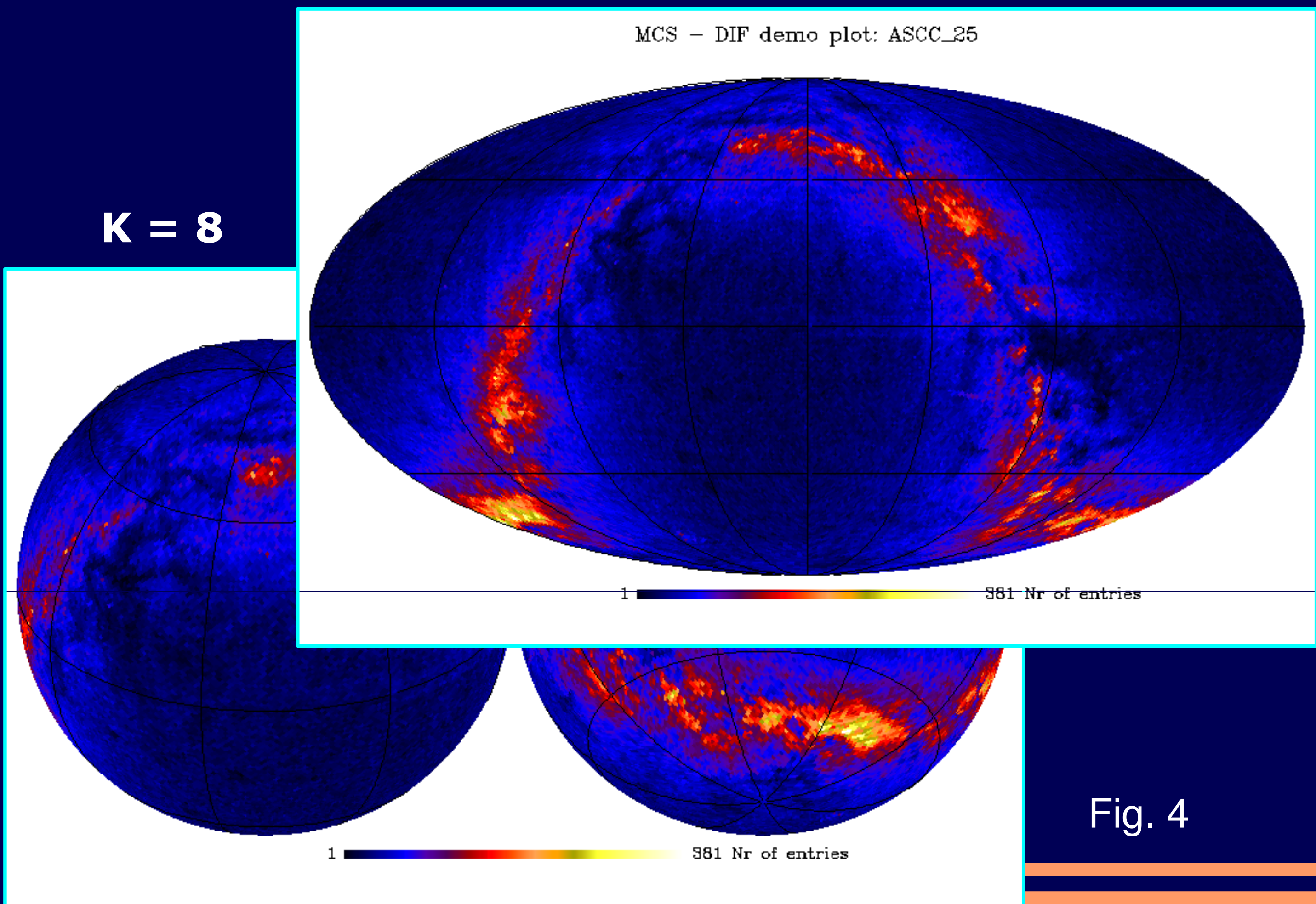


Fig. 4

example: ASCC2.5 B-V

K = 4
(3072 px.)

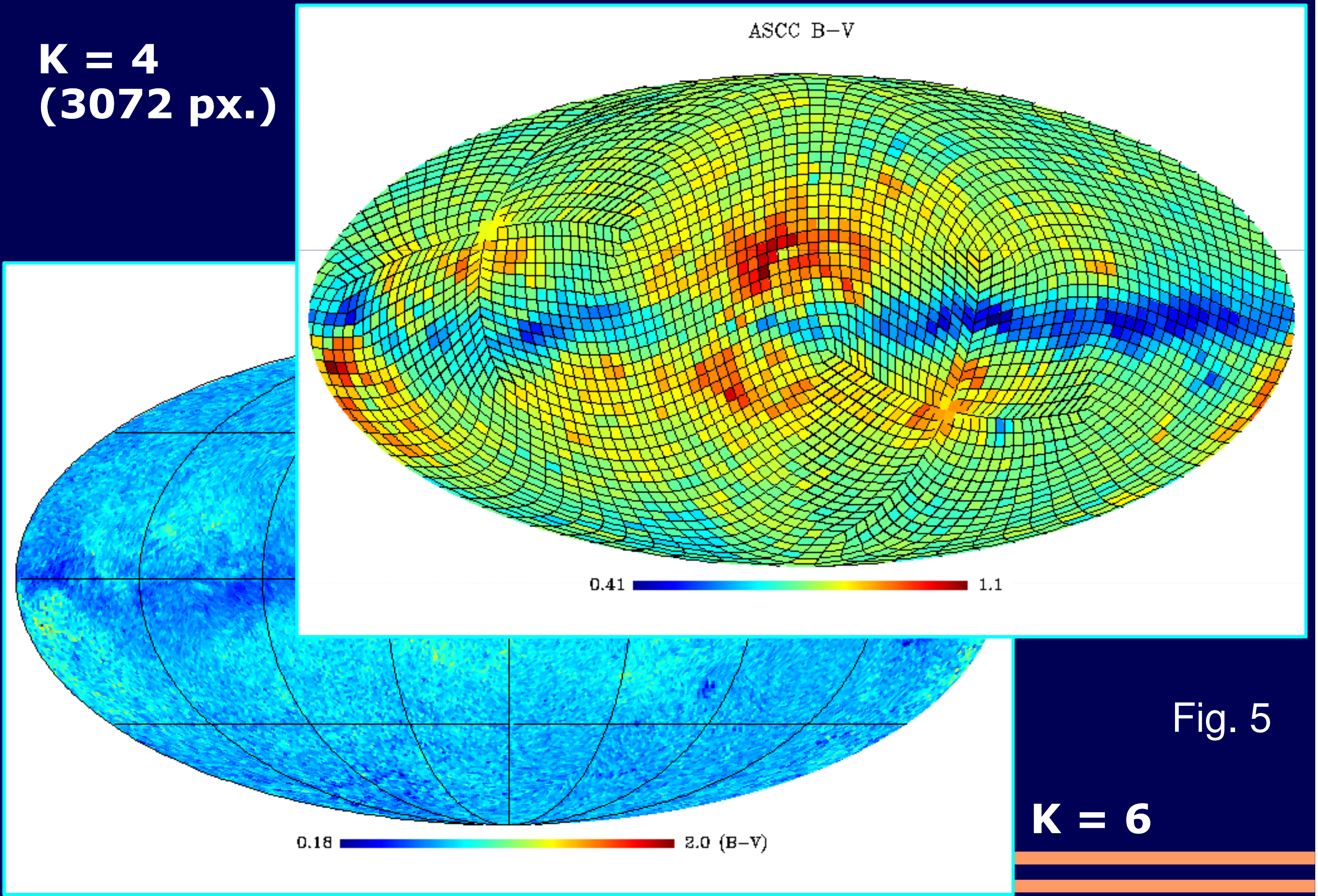


Fig. 5

K = 6

Example: UCAC2 density full sky

Ortographic
Equatorial

K = 8

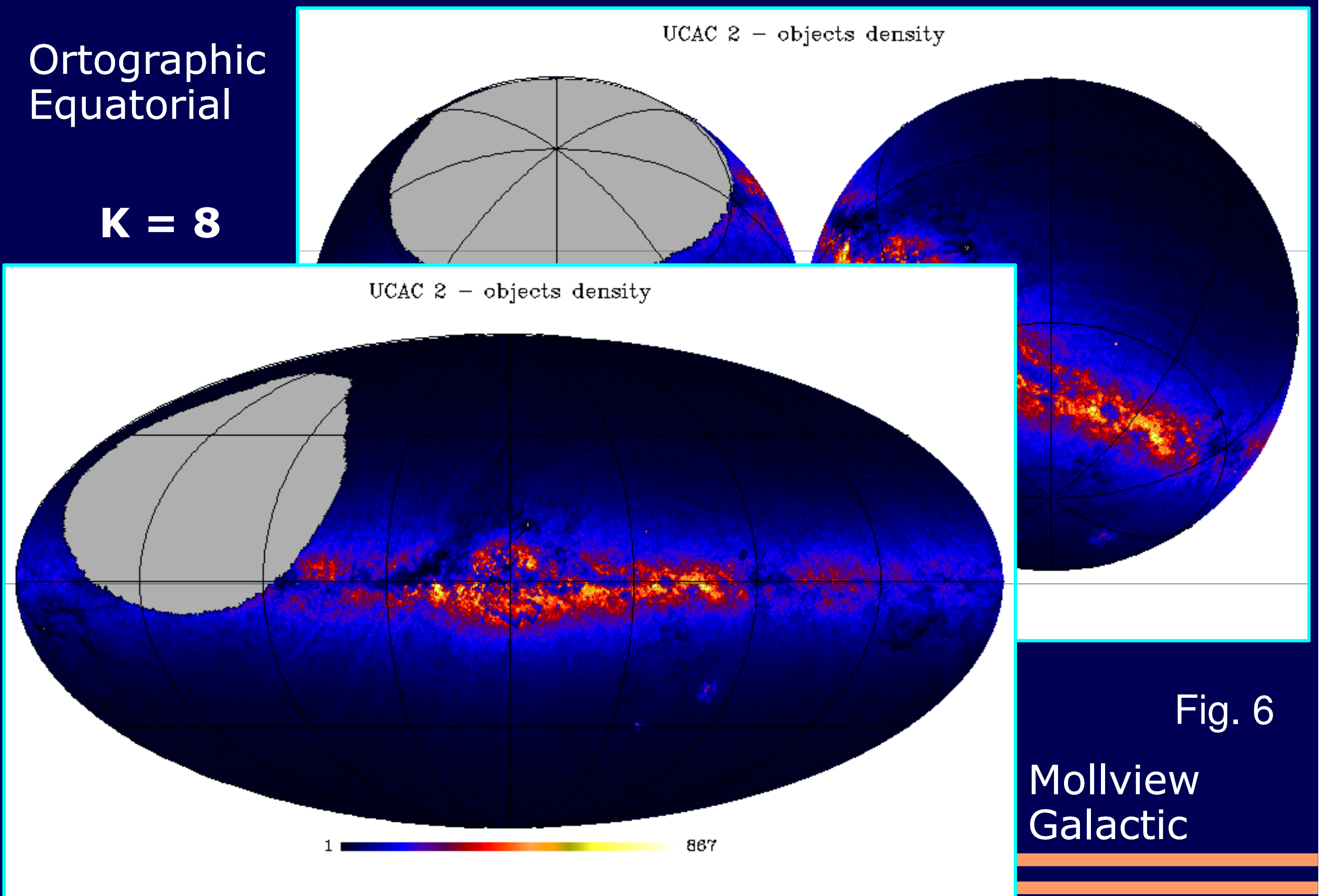
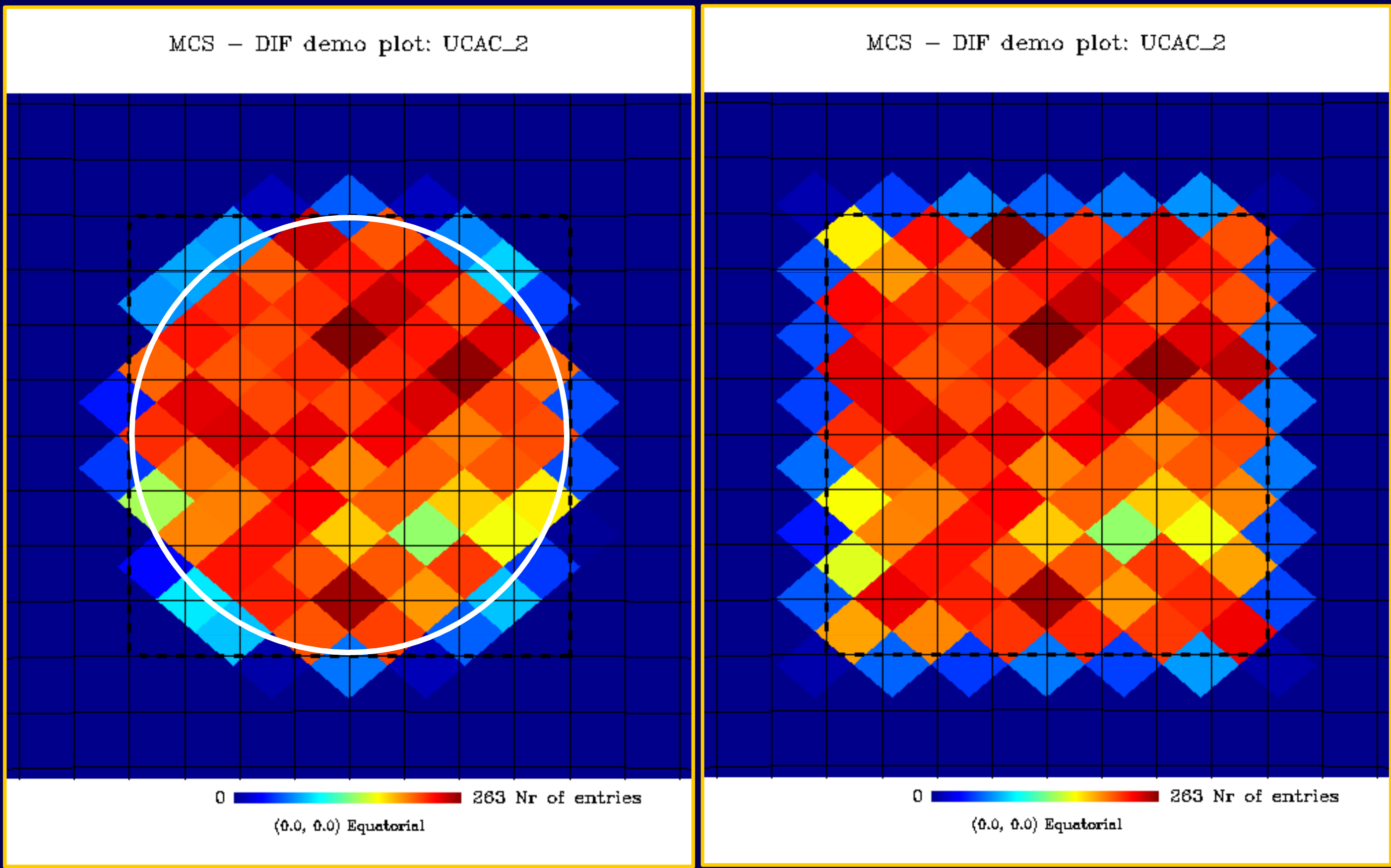


Fig. 6

Mollview
Galactic

example: UCAC2 – gnomic 0,0

Fig. 7

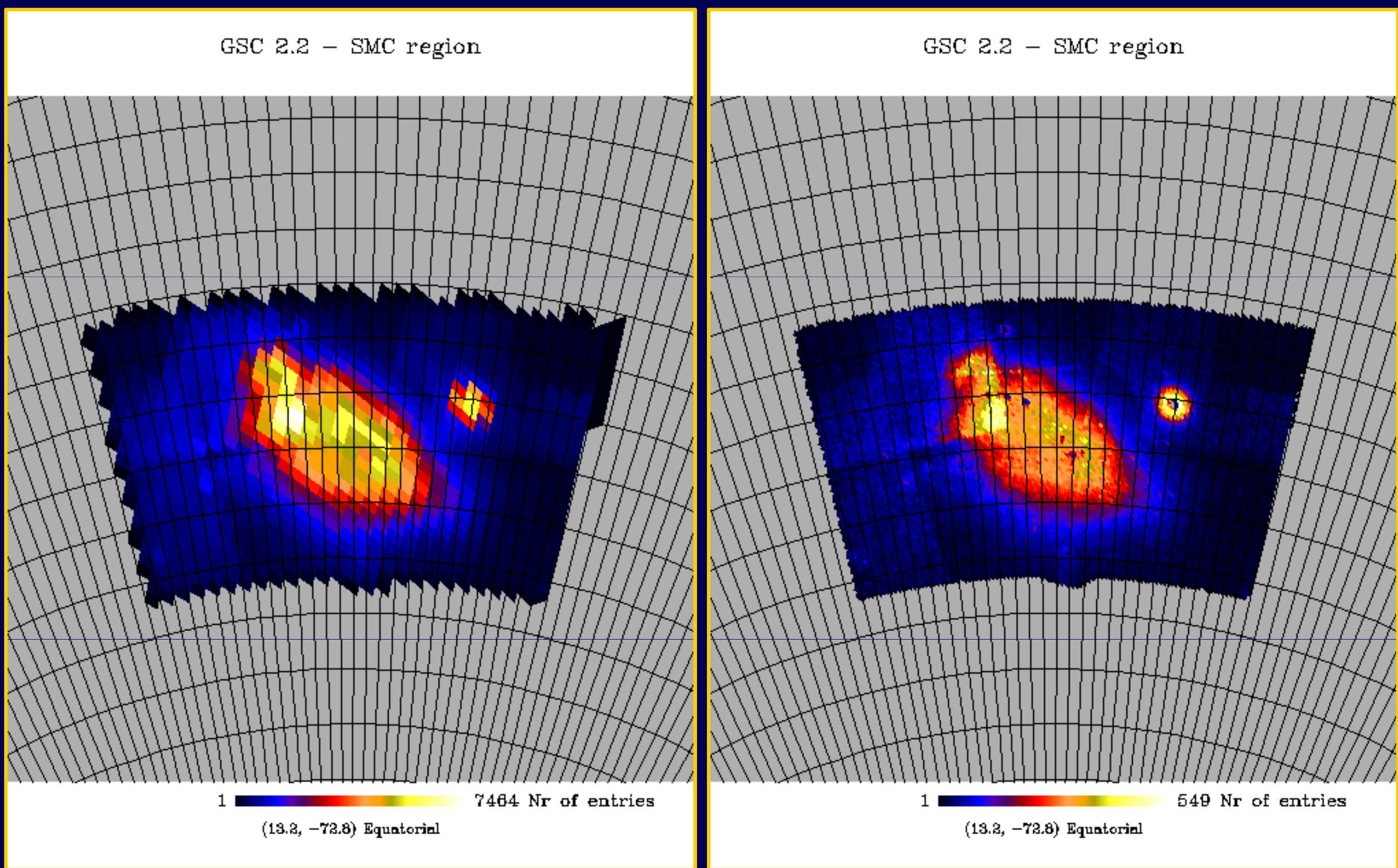


Circle

Square

example: GSC2.2 SMC

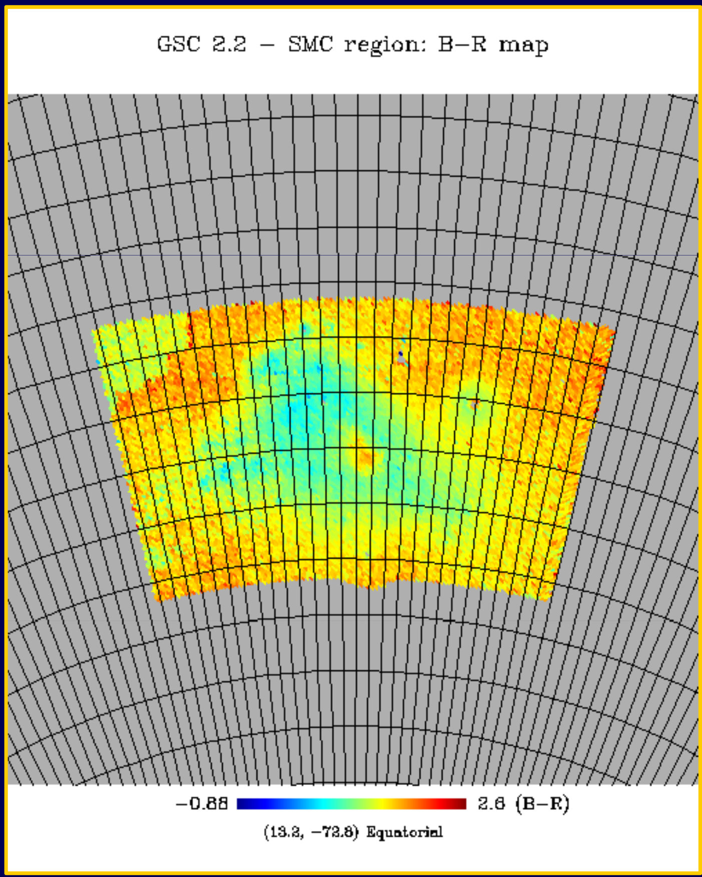
Fig. 8



K = 8

K = 10

example: GSC2.2 SMC



B - R

K = 10

Fig. 9

Multi-depth: progressive erosion

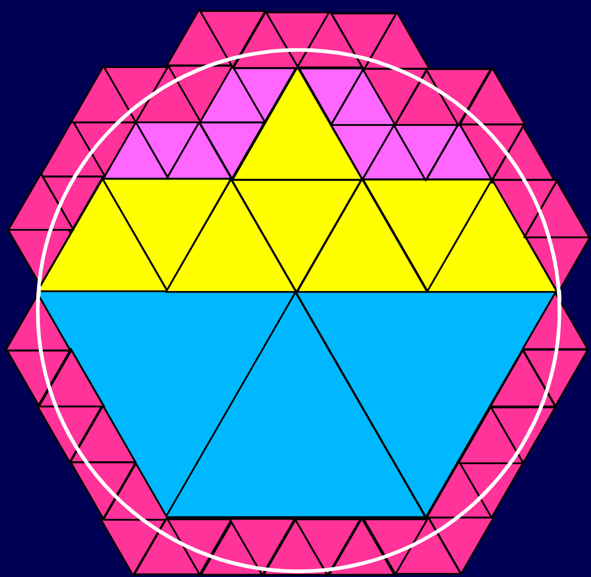


Fig. 10