

Storing and searching for MOCs in relational databases



Markus Nullmeier

**Zentrum für Astronomie der Universität Heidelberg
Astronomisches Rechen-Institut**

`mnullmei@ari.uni.heidelberg.de`

`https://github.com/mnullmei`

For people new to MOCs...



Celestial Globe of Emperor 乾隆帝 (Qianlong),
Museum of the Imperial Palace, Beijing, with
star positions following the 儀象考成
(Astronomical Instruments and Star Maps)
by 明安圖 (Minggatu, 1692–1763).

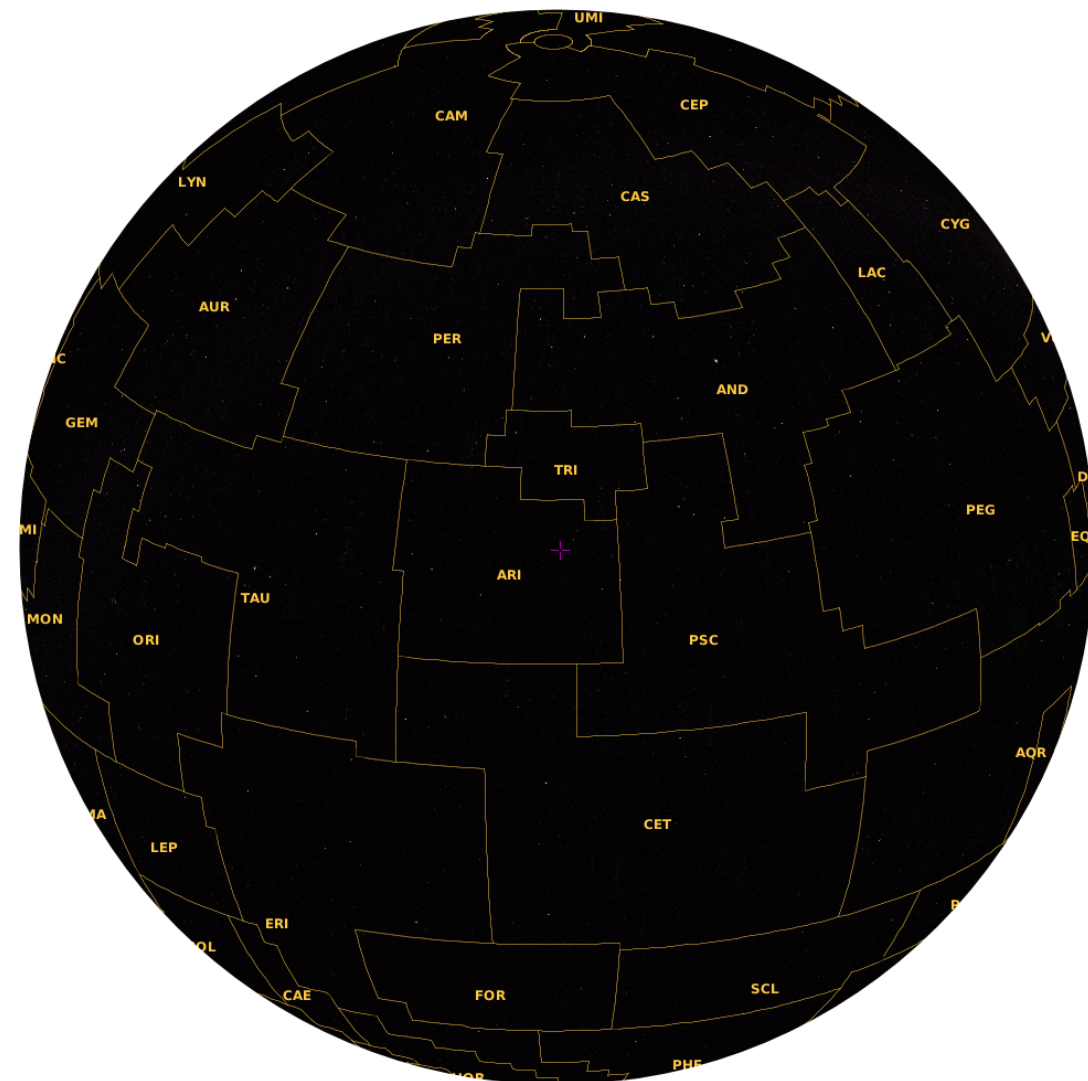
For people new to MOCs...

Constellations then ...



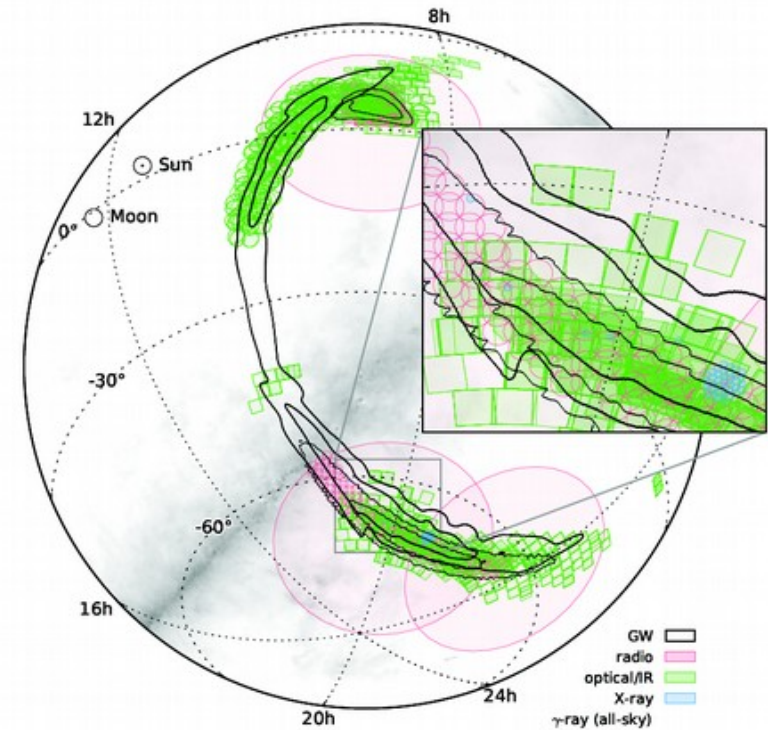
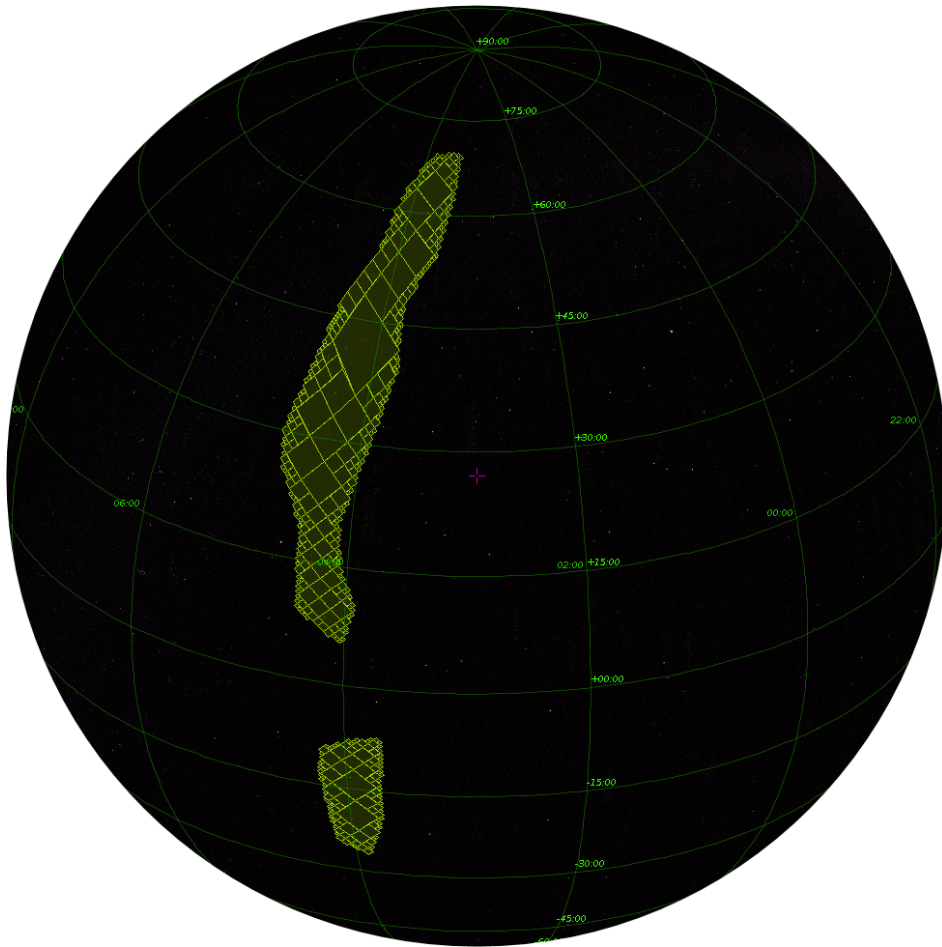
... and today

Useful for naming purposes,
but too coarse for denoting
specific sky regions 😊



For people new to MOCs...

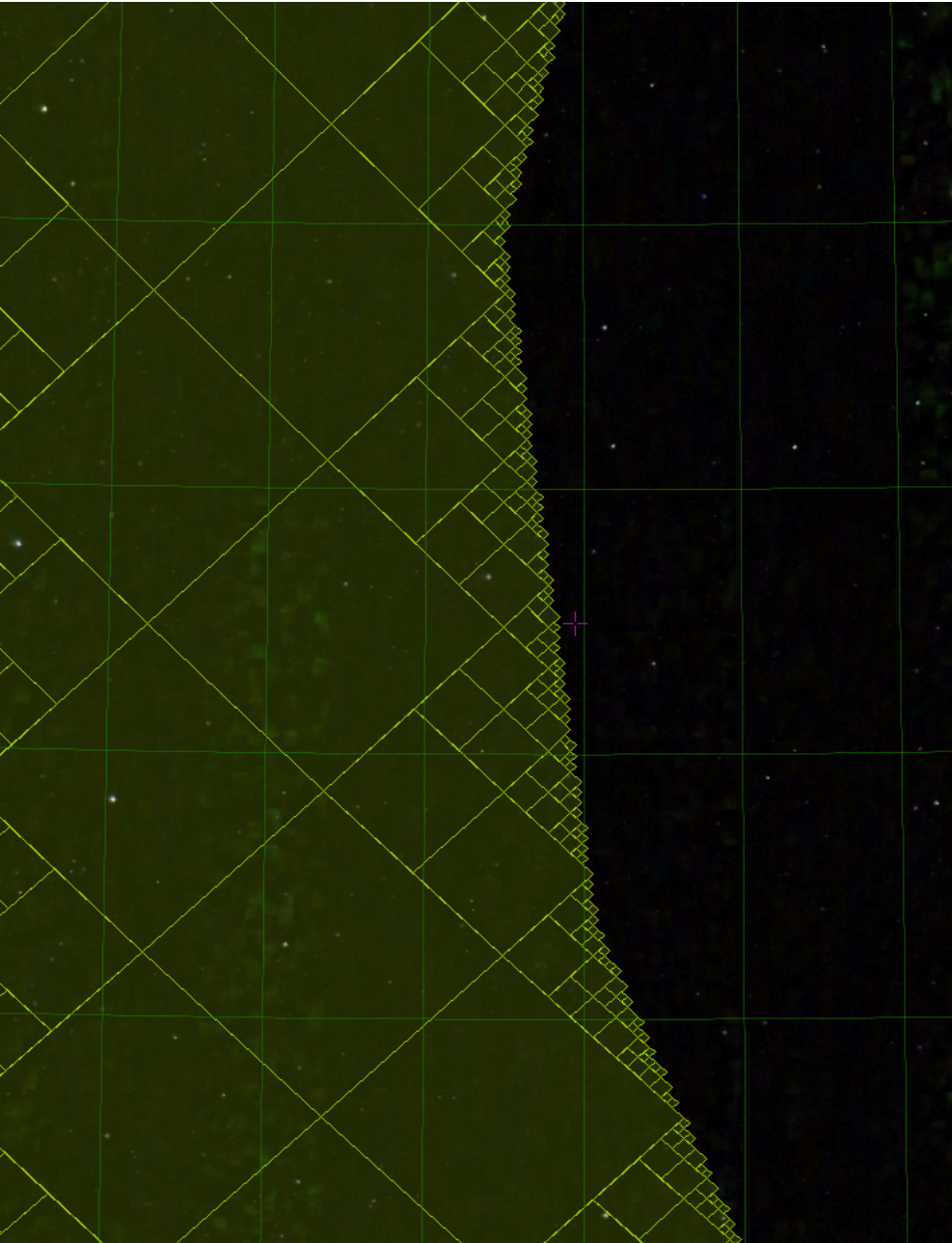
To describe arbitrary sky regions such as those of gravitational wave events, we need something else:



MOC: Multi-Order Coverage

= set of Healpix sphere elements (diamond-shaped) of different orders

MOC internals



MOC: Multi-Order Coverage

= set of Healpix sphere elements (diamond-shaped) of different orders

**1 diamond element
= 1 integer interval**

**1 MOC object
= 1 list of intervals**

**{ [2, 6) [17, 30) [33, 40)
[123, 124) [332, 438), ... }**

MOC + RDBMS = ?

1 MOC object = 1 list of intervals

{ [2, 6) [17, 30) [33, 40) [123, 124) [332, 438), ... }

- **[start, end) as table columns?**

- Maybe OK for one or a few MOCs
- ≥ 1 MOC with extra ID column
- Queries relative to MOC contents:
 - Use hand-crafted SQL code

start	end
2	6
17	30
33	40
...	...

- **Less impedance mismatch if your database offers**

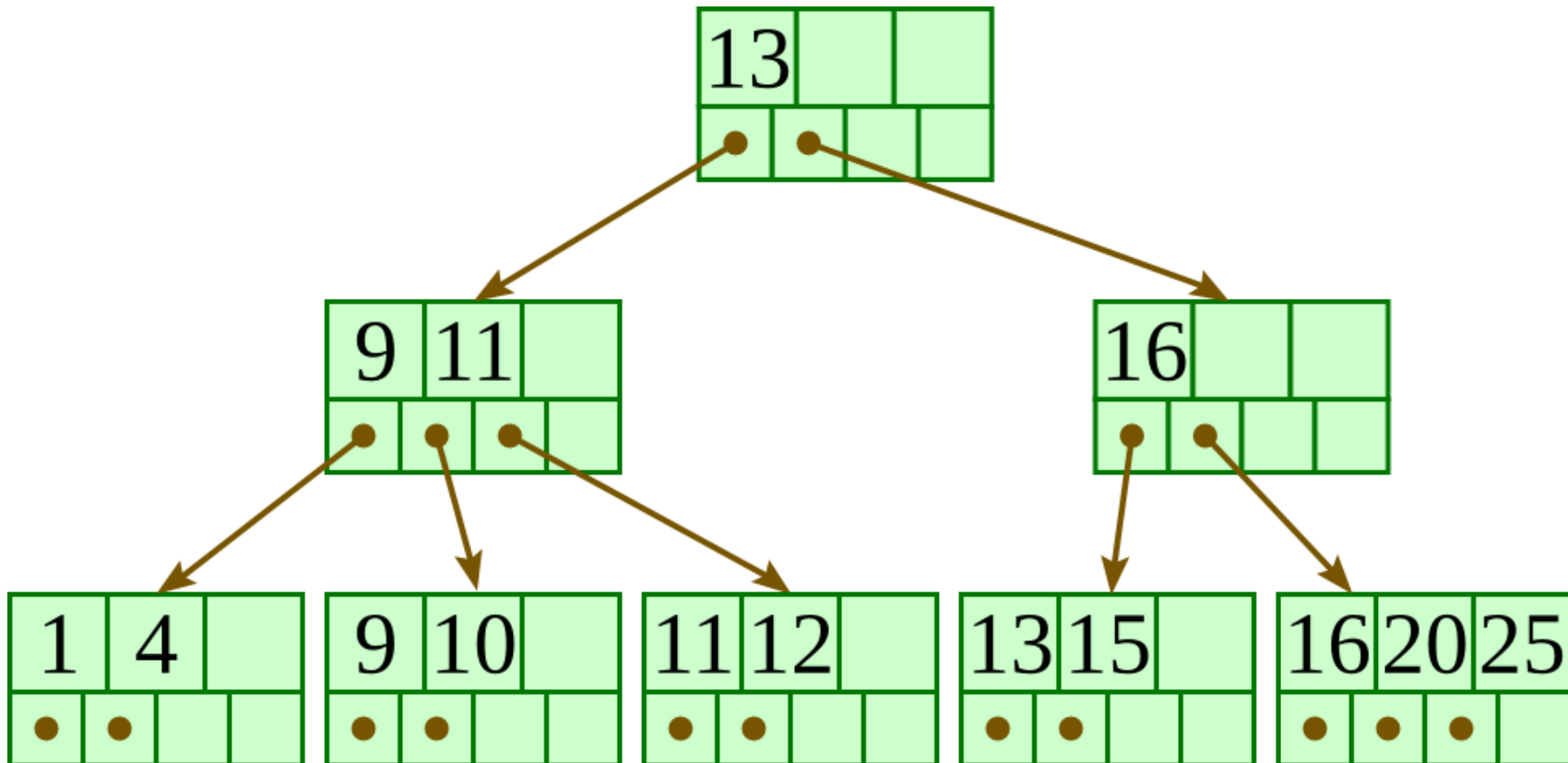
- Some kind of SQL-native collection types, such as JSON
- Better yet: SQL-native Array types

The real solution: custom SQL-level MOC data type

- **GAVO data centers uses PostgreSQL**
 - ... also many other VO sites
- **Feature of custom PostgreSQL data types:**
 - Custom PostgreSQL may be quit large (1GB)
 - No need to load the whole MOC
for a query such as “MOC contains point”
 - Creating of a MOC from a list of points
returned by an SQL query with
a single funtion call (aggregate)
 - Any kind of advanced custom indexing possible

Internals of the PostgreSQL custom SQL-level MOC data type

- On-Disk serialisation of a single MOC
 - “read-only B-tree”



Sample SQL query

- **Implementation will be part of Pgsphere**
 - `http://pgsphere.github.io`
- **Sample SQL queries with MOCs**
 - **Creation:** `select smoc('11/2-5');`
 - **Point inside MOC:**
`select '(0.7, 0.8)::spoint)`
`<@ smoc('11/2-5');`
 - **MOC from table column: pgsp**
`select smoc(point, 6) from t where ... ;`