# Cross-matching the sky with database server clusters

László Dobos

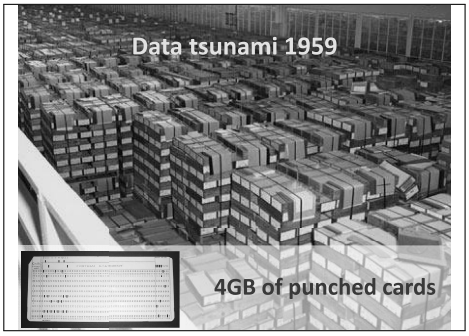
*Department of Physics of Complex Systems, Eötvös University, Budapest, Hungary*

**Abstract**

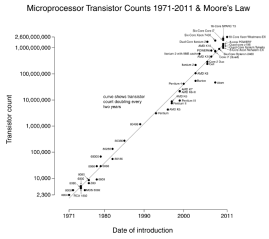
*Multi-wavelength astronomical studies require cross-identification of detections of the same celestial objects in multiple catalogs based on spherical coordinates and other properties. Because of the large data volumes and spherical geometry, the symmetric N-way association of astronomical detections is a computationally intensive problem, even when sophisticated indexing schemes are used to exclude obviously false candidates. Legacy astronomical catalogs already contain detections of more than a hundred million objects while ongoing and future surveys will produce catalogs of billions of objects with multiple detections of each at different times. One time, pair-wise cross-identification of these large catalogs is not sufficient for many astronomical scenarios. Consequently, a novel system is necessary that can cross-identify multiple catalogs on-demand, efficiently and reliably. In this paper, we present our solution based on a cluster of commodity servers and ordinary relational databases. The cross-identification problems are formulated in a language based on SQL, but extended with special clauses. These special queries are partitioned spatially by coordinate ranges and compiled into a complex workflow of ordinary SQL queries. Workflows are then executed in a parallel framework using a cluster of servers hosting identical mirrors of the same data sets.*

Cross-matching the sky with database server cluster

László Dobos
Eötvös University
Dept. of Physics of Complex Systems




## Driving force: Moore's law




## Exponential growth

Electronics → Detectors → Data size

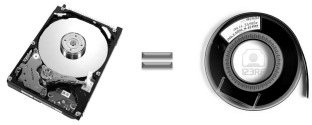
## Tractable algorithms

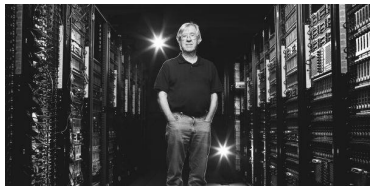
- Data size $\propto e^t$
- Computing power $\propto e^t$
- Algorithms must be o(N)
- o(N log N) at most!


## Reading a 1TB disk

- **Sequentially:** 4.5 hours
- **Random access:** 15-150 day

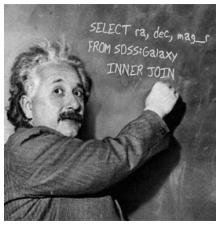
## Main limitation: Disk = tape = ☠



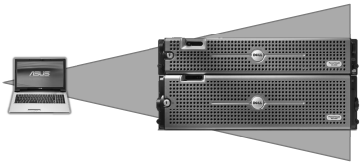


## Relational databases

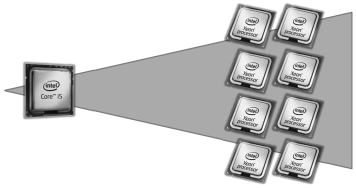
- Optimized disk access
- Declarative prog. Model
- Ad hoc query support
- User-defined functions
- But: interpreted



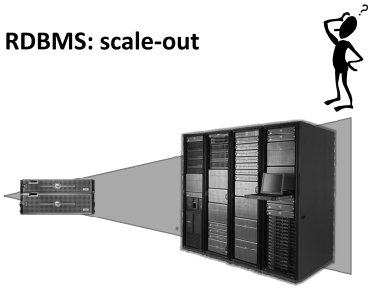

## RDBMS: scale-up



## RDBMS: scale-out



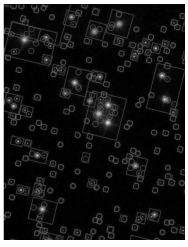
## RDBMS: scale-out



## Algorithms

| | |
|---|---|
| scan | ~~random seek~~ |
| sort | N log N |
| join | only local! (sorted in the same order) |

## Sky surveys, catalogs

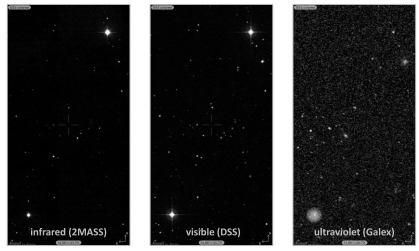


```
1 SELECT TOP 100
2   ra, dec, g, r, i
3 FROM Star
4 WHERE g - r > 2
```

(100 row(s))

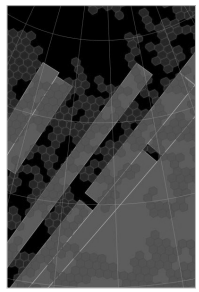
| ra | dec | g | r | i |
|---|---|---|---|---|
| 50.5527889011988 | 77.1230611213438 | 23.26467 | 21.11166 | 19.9525 |
| 51.0932611363351 | 77.0424525775233 | 24.52098 | 21.29801 | 19.943 |
| 50.9922132732151 | 77.0008094614706 | 21.46784 | 21.27179 | 19.76744 |
| 50.8725562056018 | 77.0127142143097 | 24.42376 | 22.28052 | 20.57266 |
| 50.3553840311531 | 77.0020026952045 | 24.10777 | 21.91023 | 20.56079 |
| 50.7264938781833 | 77.0093540770062 | 23.64728 | 21.18413 | 20.14919 |
| 51.1089525896638 | 77.0194358666908 | 22.48647 | 20.46075 | 18.77669 |
| 50.1949918869567 | 77.0331196473319 | 22.27824 | 20.14604 | 18.83638 |
| 50.7624455384169 | 77.0463890033524 | 22.03737 | 19.60554 | 18.64096 |
| 50.5320487964634 | 77.0881821099662 | 23.13912 | 20.68158 | 19.58165 |
| 50.3897627545917 | 77.0648611021876 | 23.60388 | 21.34204 | 19.96223 |
| 50.3931435705621 | 77.0583094752454 | 23.37035 | 21.05236 | 19.38886 |
| 50.4120082736208 | 77.0787322698702 | 25.2165 | 23.17204 | 21.07352 |

## The multiwavelength sky



## Crossmatching

- Astronomical catalogs today
  - in RDBMS
  - o(100 million) objects
  - o(1TB – 10TB) DB size
- Match by coordinates
  - RA, Dec
  - Astrometric error
  - Different sky coverage
  - Different wavelength range
  - Moving objects etc.



## Sample XMatch query

```
SELECT s.objId, g.objID, t.objID,
       s.ra, s.dec, g.ra, g.dec, t.ra, t.dec,
       x.ra, x.dec
FROM       SDSSDR7:Galaxies AS s
           WITH (POINT(s.cx, s.cy, s.cz), ERROR(0.1))
CROSS JOIN Galex:Galaxies AS g
           WITH (POINT(g.cx, g.cy, g.cz), ERROR(0.2))
CROSS JOIN TwoMASS:ExtendedSources AS t
           WITH (POINT(t.ra, t.dec), ERROR(0.5))
XMATCH BAYESIAN AS x
     MUST EXIST g, MAY EXIST t
     HAVING LIMIT 1e3
REGION 'CIRCLE J2000 165.7 0.3 60'
```

Standard SQL

Probabilistic crossmatch

Spatial constraint

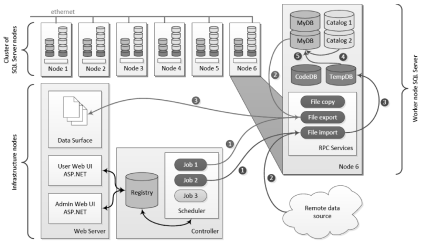
## How to extend SQL

- SQL is declarative
  - everything can be executed that can be expressed
  - extensions must be executable in any case
- Query optimization is hard
- Design language for easy optimization in mind
  - constrain on the level of the grammar
  - custom clauses instead of complex where clause logic

## Scaling out: query partitioning

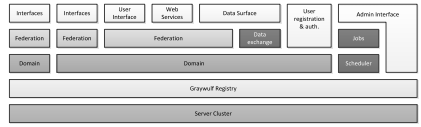
- Zone algorithm to index sphere
- Mirror everything
  - go for speed, old catalogs are small
  - be able to partition small queries by RA
  - avoid „sweet spot” problems
- Statistics gathered on the fly
  - use sampled data with cardinality of $\sqrt[3]{N}$
  - apply WHERE clause and REGION

## New generation SkyQuery




## Graywulf software stack

- Generic infrastructure based on MSSQL
- Distributed queries for scientific data
- Import, export, scheduling, web access
- SQL parser and name resolver
- Schema browser, extended meta-data




## SkyQuery features (ongoing work)

- Fetch data from remote sources
  - Protocol plug-ins
  - Pre-filtering using WHERE clause
  - Works with MySQL, Postgres, no VO TAP yet
- VO compliancy
  - via plug-ins
  - Keep system generic and domain independent
- Region-based filtering
  - Drop-out detection

## SkyQuery (ongoing work)

- Storage optimizations
  - Access to data columns is not equally distributed
  - Move less-accessed data to big but slow storage
  - Lazy joins
- Full SQL support
  - Currently limited to single SELECT queries
  - Add support for full T-SQL scripting