Spatial indexes in PostgreSQL for astronomy

Alexander Korotkov\textsuperscript{1}, Oleg Bartunov\textsuperscript{12}

\textsuperscript{1}Postgres Professional

\textsuperscript{2}SAI MSU
Alexander Korotkov, Teodor Sigaev, Oleg Bartunov

**PostgreSQL CORE**
- Locale support
- PostgreSQL extendability:
  - GiST(KNN), GIN, SP-GiST
  - Full Text Search (FTS)
  - NoSQL (hstore, jsonb)
  - Indexed regexp search
  - Custom AM & Generic WAL
  - Pluggable table engines (WIP)

**Extensions:**
- Intarray
- Pg_trgm
- Ltree
- Hstore
- plantuner

- Major contributors to PostgreSQL
- Co-founders of Postgres Professional
Why PostgreSQL?

- Mature DBMS with all necessaries
- Open Source product distributed under BSD-like license
- Big and responsible community
- Outstanding extendability
- Rich features to work with
WSDB Whole Sky DataBase

- Database of astronomic catalogues in Cambridge University
- ~5 dbs, ~40 users, up to ~10^7 queries per day, size 40Tb
- pg 9.4 + q3c + hstore
Gaia Alerts Database

• Real time Detection of alerts in the Gaia
• ~10 dbs, 10 users, up to ~ $10^6$ queries per day, size 30Tb
• pg 9.3 + synchronous replication + q3c
MASTER database

• Robotic net of telescopes by SAI MSU
• 8 observatories (5 in Russia, 3 outside)
• total size ~100TB
• pg 9.0-9.4 + pgsphere + replication
• See:
Radial query
Crossmatch
Indexing methods

• haversine (haive)
• Q3C
• pgSphere
• PostGIS
q3c structure

- Cube is inscribed into sphere
- Central projection of cube into sphere
- Quad-tree on cube faces transforms spherical coordinates into integer (IPIX)
Querying using q3c
Generalized search tree (GiST)

• Generalization over R-tree and its variances
• PostgreSQL GiST is the only full featured implementation of GiST
• A lot of applications (operator classes) to various tasks
pgsphere versions

- pgSphere 1.1.1 – last version before this research
- pgSphere 1.1.2 – bug fixes, new node splitting algorithm
- pgSphere 1.1.5 – new operator class spoint2, which able to store source points in index leaves
Crossmatch by two indexes
Time required for insertion
Time required for radial queries
Number of used index blocks
Number of read index blocks with shared_buffers = 512 МБ
Number of read heap blocks
Time of crossmatch execution
Z-curve
3d-zcurve
Experiments on real-life dataset

- Nomad catalogue
- $10^9$ points
- Size of table without indexes = 182 GB
Index size, GB

- pgsphere: 80 GB
- postgis: 90 GB
- q3c: 20 GB
- zcurve: 30 GB
Build time, sec

- pgsphere
- postgis
- q3c
- zcurve
Radial query blocks used

![Graph showing the number of blocks returned vs. number of points returned for different query methods: q3c, pgsphere, zcurve, and postgis. The graph has a log-log scale on both axes.]
Conclusion

СУБД PostgreSQL provides rich set of features for working with spatial data including spatial objects in spherical system of coordinates. It allows to effectively process various types of search among them.
Future work

• Improve z-curve
• Index using GiST and z-curve in spherical coordinates
• Implement HTM using SP-GiST
Thank you for attention!
Any questions?