PostgreSQL: the Suitable DBMS Solution for Astronomy and Astrophysics

Chilingarian, I. 1) 2); Bartunov, O. 3) 4); Richter, J. 4); Sigarev, T. 3) 4)
1) Sternberg Astronomical Institute of the Moscow State University, 2) Special Astrophysical Observatory of the Russian Academy of Sciences, 3) DataEase LLC, 4) International Memo Group. *A participant of MGSA collaboration, responsible for DBMS subsystem.* An official member of PostgreSQL development team.

**Abstract:**
PostgreSQL, the open-source ORDBMS, is probably one of the best database solutions for astronomy and astrophysics. Compared to several available commercial and free-open source databases engines, it appears to be the most versatile. At present PostgreSQL is being used in several well-known astronomical projects, for example in the HyperLEDA database, http://leda.univ-lyon1.fr/ created by a team lead by Gborg. We have developed pgSphere contribution module, http://gborg.postgresql.org/ for PostgreSQL using backend programming and GiST interface. It is distributed under BSD license. It introduces data types for geometrical objects on a sphere and access methods for them.

**Keywords:** PostgreSQL, GiST, SP-GiST, pgSphere, Sky Survey, Space Partitioning Tree, spherical data types, astronomical catalogs.

**References:**

**2. Extensibility of PostgreSQL**
As noted before, the extensibility becomes the most important feature of the DBMS to be used in science. PostgreSQL allows to very wide possibilities extending the database and adapting it to the raised objective.

- PostgreSQL allows to create user-defined functions and aggregates in the upper layer using SQL, or one of the available procedural languages. This feature is quite common for the most of the DBMSs. Also it is possible to create completely different types and new access methods for dealing with them.
- PostgreSQL provides a powerful functionality for so called back-end programming. This allows developer to create functions using low-level language (i.e. C). Compile them and load dynamically into the running database server as shared objects. Binary code mode increases the performance dramatically. Moreover, the standard interface to GiST (Generalized Search Tree) is provided to create custom data types with indexed access methods and extensible set of queries for specific domain experts not a database one.
- PostgreSQL implemented in an earlier version of PostgreSQL by J. Hellesenstien and P.Aaik, more details is available from "The GiST Indexing Project" hosted by http://gborg.postgresql.org/.

**4. pgAstro**
We have developed pgSpHERE contribution module, http://www.postgresql.org/cgi-bin/wikp/pgSpHERE for PostgreSQL using backend programming and GiST interface. It is distributed under BSD license. It introduces data types for geometrical objects on a sphere and access methods for them. The project is hosted by Gborg, http://gborg.postgresql.org/

- pgSpHERE provides the following types and functions:
  - input and output of “spherical” data in several formats (radius, degrees etc., containing, overlapping and other geometrical operations for different types of objects on a sphere; some of these types are shown below
  - variable input and converting functions
  - calculation of circumference and area of an object on a sphere
  - indexed data access methods for “spherical” data types

Hence it is possible to do a fast search and analysis for objects with spherical physical models, using PostgreSQL. This functionality may be very useful for different types of astronomical and geo-science applications. For instance it makes possible management of data for geographical objects on the Earth or astronomical data like stellar and other objects conveniently using a SQL interface.

The aim of pgSpHERE is to provide a uniform access to spherical data. PostgreSQL itself supports a lot of software interfaces; therefore one can use the same database for access with different utilities and applications, also using the pgSpHERE.

Several performance tests were made with different datasets. We used TYCHO-1 catalogue and its parts to compare the performance of GiST-R-tree based algorithm implemented in pgSpHERE to 2-column B-tree index and other well-known sort-based index methods. The complete catalogue includes 105115 stars. Four sub-catalogues were created using the simple schema:

```
SELECT * INTO TABLE tycho07 FROM tycho WHERE mag<07;
SELECT * INTO TABLE tycho08 FROM tycho WHERE mag<08;
SELECT * INTO TABLE tycho09 FROM tycho WHERE mag<09;
```

The number of objects in the sub-samples varies from ~1500 to ~40000. After construction of indices these two queries had been executed for the whole catalogue and all 30 sub-catalogues for 10 times.

```
SELECT count(*) FROM tycho WHERE dec between -3 AND 3;
```

The execution times were averaged for each group of queries. Three series of queries were executed for the area of sphere, 66,10 and 20,20 degrees. The results are demonstrated below. Figure has exactly the same meaning as fig. 1 in [2]. All lines are the absolute running time decreased by a factor of 50 due to progress in hardware and software, but the relative performance is the same or less the same.

**5. Conclusions**
From the given examples PostgreSQL appears to be the most versatile DBMS solution for astronomical and astrophysical data. It is easily extensible, has powerful set of features well comparable to leading commercial database solutions. The fact that PostgreSQL is freely distributed open source database allows it to be used in many different applications and it is a very important advantage. Many projects including the Postgres project create contributions useful for scientists, who is hardly possible with commercial databases.

The further features of PostgreSQL will include XML support. It may be very useful for many VO applications and tools.

**Acknowledgments**
Our development is supported by the Russian Foundation for Basic Research, projects 02-07-9222 and 03-07-06116. Also we greatly appreciate PostgreSQL community, TASS Amateur Sky Survey working group especially Robert Cooper and Chris Altherston. Great thanks to ADASS-XIII organizing committee for financial support.

Any feedback is welcome at chil@saik.msu.ru