

## Who's Using PostGIS?

The **City of Boston** has an internal web-based property analysis application, based on PostGIS, which performs spatial queries on land parcels and displays results using Mapserver.

The **Department of Natural Resources** in Queensland, Australia uses PostGIS in their internal Forestry Permitting web application. The database is queried by attribute and a map of the parcel area is displayed along with supporting information.

The **Ministry of Sustainable Resource Management** in British Columbia, Canada uses PostGIS to store, manage and analyze their Digital Road Atlas, a very large and complex road network database.

**I-Cubed Inc**, USA, has a large automated image processing system which uses PostGIS to track the spatial extents of incoming requests, and coordinate processing tasks among their network of workstations.

**Intevation Gmbh**, Germany, is developing a massively multiplayer online game where players will have their position stored in PostGIS, and the game engine will query the server to generate the game display.

The **Finnish Geodetic Institute** is creating a regional distributed GIS network using GeoServer OpenGIS Web Feature Server with PostGIS as the database back-end.

## PostGIS Team

**Dave Blasby** is the principal developer of the PostGIS spatial database extension to PostgreSQL. Dave maintains the core objects and functions, and Mapserver connectivity.

**Paul Ramsey** is the PostGIS project evangelist and release manager. Paul maintains the documentation, web site, and the JDBC extensions.

**Chris Hodgson** maintains the GiST indexing support functions.

**Jeff Lounsbury** maintains the Shape file data loaders and dumpers.

## Refractions Research

Refractions Research is a GIS and database consulting and development corporation, specializing in data integration and open-source spatial solutions.

Refractions offers consulting services in GIS, internet mapping with Mapserver, and PostGIS/PostgreSQL database implementation, as well as custom enhancements to PostGIS.

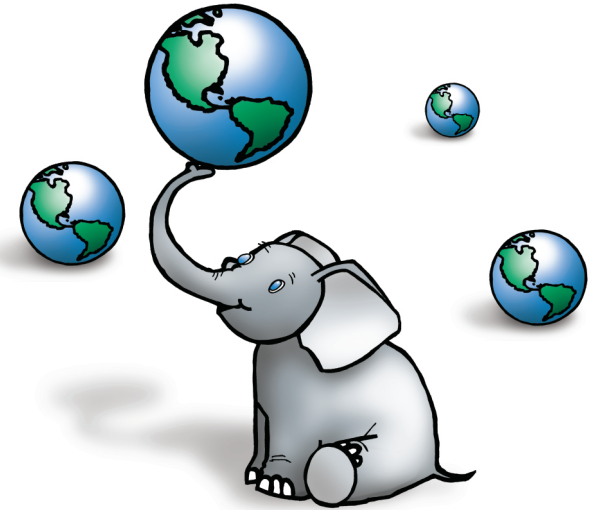
**For more information, contact:**



Paul Ramsey, President  
pramsey@refractions.net  
Phone: (250) 885-0632  
Fax: (250) 383-2140

# PostGIS

Geographic Objects for PostgreSQL



## A Spatial Database for the Rest of Us

Refractions Research  
Victoria, BC, Canada  
<http://postgis.refractions.net>  
<http://www.refractions.net>

## About PostGIS

PostGIS adds support for geographic objects to the PostgreSQL object-relational database. In effect, PostGIS “spatially enables” the PostgreSQL server, allowing it to be used as a backend spatial database for geographic information systems (GIS), much like ESRI’s SDE or Oracle’s Spatial extension.

### PostGIS/PostgreSQL includes the following functionality:

- Simple Features as defined by the OpenGIS Consortium (OGC)
- Support for Well-Known Text and Well-Known Binary representations of GIS objects
- High performance spatial searching using GiST (Generalized Search Tree) indexes
- Geospatial analysis functions
- JDBC extension objects corresponding to the PostGIS geometries
- Support for OGC spatial access and manipulation functions as defined by the Simple Features Specification

Open-source GIS tools are relatively new, and an open-source spatial database is important to the open-source GIS community. PostGIS source code is released under the GNU General Public Licence (GPL).

PostGIS is developed and maintained by the staff of Refractions Research, in Victoria, BC, Canada.



## History

Adding spatial capabilities to PostgreSQL is a project which seems redundant – after all, PostgreSQL already has “geometric types”. Unfortunately the native geometric types, while interesting, are not suitable for true GIS data: none of the types support “aggregate features”, the polygon type does not support holes, and the indexing is restricted by the old 8Kb row size limit.

PostGIS became a feasible project when PostgreSQL 7.1 removed the 8Kb row size limitation. Using the OpenGIS spatial database specification as a guide, we created new spatial types, and tied them into the GiST indexing system. Version 0.1 was released in late June, 2001, featuring the spatial objects, spatial indexing, and little else.

Subsequent development has focused on implementing the OpenGIS “Simple Features for SQL” specification more completely.

Version 0.5 (July 2001) featured a major breakthrough – the ability to use PostGIS as a data source for the University of Minnesota Internet Mapserver. For the first time, it became possible to graphically view the spatial data stored in PostGIS.

Versions 0.6 and 0.7 have continued the trend of OpenGIS compatibility, adding support for embedded spatial reference systems, coordinate re-projection, and even more functions.



## Development Roadmap

Future development will concentrate on making PostGIS more OpenGIS conformant and on improving connectivity to external software:

- **Port the topological functions in the “Java Topology Suite” (JTS) to C++ and PostGIS.** Full topology operations are a major feature necessary to be fully compliant with the OpenGIS “Simple Features for SQL” specification. JTS provides a robust suite of open-source topology algorithms.
- **Implement an OpenGIS “Web Feature Server” (WFS) on top of PostGIS.** WFS will be a major interoperability breakthrough, allowing sophisticated desktop GIS software to access any GIS data source through a single open API.

### More Information

#### PostGIS

<http://postgis.refractions.net>

#### OpenGIS

<http://www.opengis.org>

#### GiST for PostgreSQL

<http://www.sai.msu.su/~megeera/postgres/gist/>

#### Java Topology Suite

<http://www.vividsolutions.com/jts/jtshome.htm>

#### Minnesota Mapserver

<http://mapserver.gis.umn.edu>

