

FOSS spatial DBMS and interoperability

Marco CiOLLI, Alfonso VITTI, Fabio ZOTTELE and Paolo ZATELLI

Dipartimento di Ingegneria Civile e Ambientale,
Università degli Studi di Trento,
Via Mesiano 77, 38100 Trento, Italy.
*tel. +39-0461-88-2618, fax +39-0461-88-2672,
e-mail Paolo.Zatelli@ing.unitn.it*

Abstract

The creation and management of centralised repository for GIS data offer many advantages but pose some problems. In particular, the interoperability of the server with a mixed set of clients is troublesome. Free and Open Source Software (FOSS) provide an appealing solution to this problem, but real usability must be checked. This paper describes the set up of a spatial data base management system and reports a series of tests of connection of various GIS clients.

1. Introduction

Geographic data are nowadays pervasive in many organisations, therefore proper practices for data management and dissemination must be set in place. In fact, while until recent days GIS data were used by specific, and usually small, parts of an organisation, georeferenced data are now directly or indirectly used by most, if not all, the subdivisions. When GIS data use is limited, their distribution is made using de facto standard file formats, which are accepted by all the systems involved in the data elaboration. However, as the use of geographic data becomes widespread, this model is no more feasible, since:

1. the number of software systems involved in data management and elaboration grows significantly;
2. the number and the type of user increases, most of the users are not GIS specialists anymore;
3. a way of controlling information fluxes must be deployed.

In particular, as GIS data become more relevant in volume and importance, procedures for data authoring, management and dissemination must be implemented, controlling the access for data users and providers, who must inevitably have different permissions on data. Moreover, data integrity and update must be guaranteed.

The best way of satisfying these needs is to use a central database management system (DBMS), where all the information, both geometric and semantic, is kept.

2. The spatial DBMS approach

The more suitable approach now available for the integrated management of geographic information is the use of a spatial DBMS, which offers several advantages in terms of flexibility, reliability and total control over information fluxes. Some of the leader DBMS projects offers the possibility of storing in a data base not only the traditional semantic (alphanumeric) data, but also geometric and topologic information. For this purpose specific data types are available, as well as functions working on these data types. These so-called “spatial extensions” allow the integrated management of the geographic data, combining the data management efficiency of DBMS with the ability to manage geometry and topology of GIS. There are several advantages in this approach, mostly due to the maintenance of an unique central data repository:

- different kinds of clients can concurrently access the system with different access levels, increasing security;
- standard interfaces to the database, such as WebGIS or DB browsing via web, can be provided for non specialist users;
- database updates are available in real time to the users and data duplication avoided;
- data production can be controlled, granting write access only to authorised operators;
- the system can automatically validate data geometry and topology.

This is a client-server approach, where a DBMS server can be accessed locally and remotely from different kinds of clients, working on different kinds of data with different permissions. The server can be accessed by different classes of clients, differing for type of data request and elaboration capacity. The range goes from the light GIS data viewer, currently available in any GIS product line usually for free, to the complete GIS suites, from DB clients to WebGIS. All these clients can be set up to access the data they need transparently to the final user.

Moreover, additional interfaces can be provided to occasional, non specialistic users, who needs to browse the archive without the knowledge of a particular client software. This can be done by setting up WebGISs based on the spatial DBMS, which can be tailored for different classes of users whom are provided different “views” of the database. This systems allow the simultaneous access to the spatial and semantic part of the spatial DB: sometimes, however, only the semantic (i.e. non spatial) part of the database is interesting for some users. Web-DBMS clients are available, allowing the publication of database table to the Web, still retaining full control of the information access.

As data volume and organisation complexity increase, one of the major problems is the guarantee of using consistent and up-to-date data through different users: this can be provided by an unique data repository accessed in real time, avoiding the need of asynchronuos data transfers from data providers to users. Updates are uploaded to the database from authorised users and become instantaneously available to all the users.

The use of a central DBMS allows the validation of data, controlling their consistency. This issue is critical when different subjects are involved in GIS data creation and quality and consistency is not assured. Quality standards are difficult to be enforced, especially when data providers are not GIS specialists but perform this task occasionally. Automatic or semi-automatic procedures for data validation and consistency checking are available, but specialistic operators must be employed for this task: a unique central data repository allows the creation of a quality check team.

The design of the system must provide the guidelines for the setup of the server, determining which servers must run and which services must be provided, and the indication of the suitable clients for each task, after exhaustive tests of mutual compatibility.

3. The FOSS solution

All the mayor software houses developing GISs provide DBMSs with spatial extensions, however proprietary DBMSs inevitably “prefer” systems of the same software house, especially because no standard for data types and database transactions is in use. Many GIS softwares claim full or partial compatibility with spatially enabled DBMS from the mayor vendors, however field tests usually demonstrate that this compatibility most of the times is not really working. Moreover, proprietary systems support only few external DBMS and it is usually impossible to develop drivers and interfaces to unsupported DBMS.

While the external semantic data can usually be accessed, at the price of a loss of efficiency, through the ODBC driver, no similar protocol is available for geometric and topological data.

Free and Open Source Software (FOSS)¹ provides an alternative to proprietary systems in more than one way. First of all, FOSS systems for the management of geographic information offer good interoperability and adherence to open standards such as those of OGC (OpenGIS Consortium), despite the lack of central development coordination, since free information exchange is one of the FOSS movement goals. Therefore, the integration of different FOSS systems to develop a comprehensive and integrated system is usually straightforward. Moreover, the availability of the source code makes it possible to develop new interfaces to other systems, this possibility is of course more easily exploitable when all the involved systems are FOSS and therefore their source code is available.

FOSS systems exist for all the aspects of the elaboration, management and distribution of geographic information, with varying grade of maturity.

4. The server side

For representing a viable solution for real word applications, a server supporting access to a spatial DBMS must provide reliability, performance and flexibility. Reliability is needed as many different clients depend on the server for a multitude of tasks, therefore an outage can severely affect the organisation’s operations. Performance must be guaranteed since many clients are connecting at once

¹see <http://www.gnu.org> and <http://www.opensource.org> for reference.

and geographic data volume is usually high. Flexibility is essential because of the different clients' types and access protocols in use, as well as for the rapid evolution of the GIS tools, data types and access protocols. Reliability is provided by a proper hardware setup supplying redundancy and by the choice of dependable software: in this sense many FOSS systems are often referred to as references. Performance can be enhanced by the choice of modular software systems: modularity can also help flexibility because modules can be added or dropped as needed without service interruptions.

The server has been set up exclusively with FOSS software, as reported in table 1.

The Debian/GNU Linux distribution has been chosen for the operative system because of its reliability and its advanced software packaging system, allowing an easy and highly customisable maintenance.

The core services are provided by PostgreSQL, a DBMS with spatial extension. PostgreSQL is an object-relational database management system SQL compliant and highly scalable. It offers high efficiency in handling large datasets and allows the definition of access levels to databases and tables. The spatial extension is given by the PostGIS package, which "spatially enables" the PostgreSQL server, providing geographic data types and functions. PostGIS conforms to the OpenGIS Consortium "Simple features specification for SQL". PostGIS topological extension is provided by the GEOS library, allowing the use of topologic features and functions.

In this configuration the DBMS server can be accessed from any local or remote client with PostgreSQL connectivity or, for the semantic part only, via an ODBC driver. These are specialised clients, usually available to a few GIS trained personnel: to reach a wider and more generic audience additional servers, drawing data from the DBMS server, have been installed.

The most efficient way of publishing data on a Intranet or Internet network is to use a web server and to serve data as HTML pages. This method is efficient both on the server side, where proper server systems are available and widely used, and on the client side, where a web browser is used to present semantic and geographic data as tables and maps, i.e. in ways already familiar also to non-technical users.

The Apache web server, the most used web server on the Internet and one of the most known FOSS projects, has been installed. The modular version, MPM (Multi Process Modules), has been chosen for performance's sake. The link between the DBMS and the web server is given by the php language, a scripting language for the creation of dynamical HTML pages which is able to directly access PostgreSQL tables. In this way specialised servers can provide database data as web pages.

Database tables are rendered as HTML tables by phpPgAdmin, a web front-end to PostgreSQL which exploits PostgreSQL capabilities to allow selective access to the database, giving different users or groups of users different permissions for reading and/or modifying data tables.

Geometric data are published on the web using the Mapserver WebGIS, which allows the development of sophisticated applications for the distribution of geographic data. Mapserver is able to build maps and tables on the fly, using data pulled in real time from the DBMS, providing a large number of tools for viewing and querying the dataset. Development tools and widgets libraries, such as Chameleon, are available for Mapserver allowing a fast and high quality WebGIS development.

This server configuration provides connections to GIS clients, using proper protocols and data types, as well as to non technical users via a standard web connections, allowing a full availability of

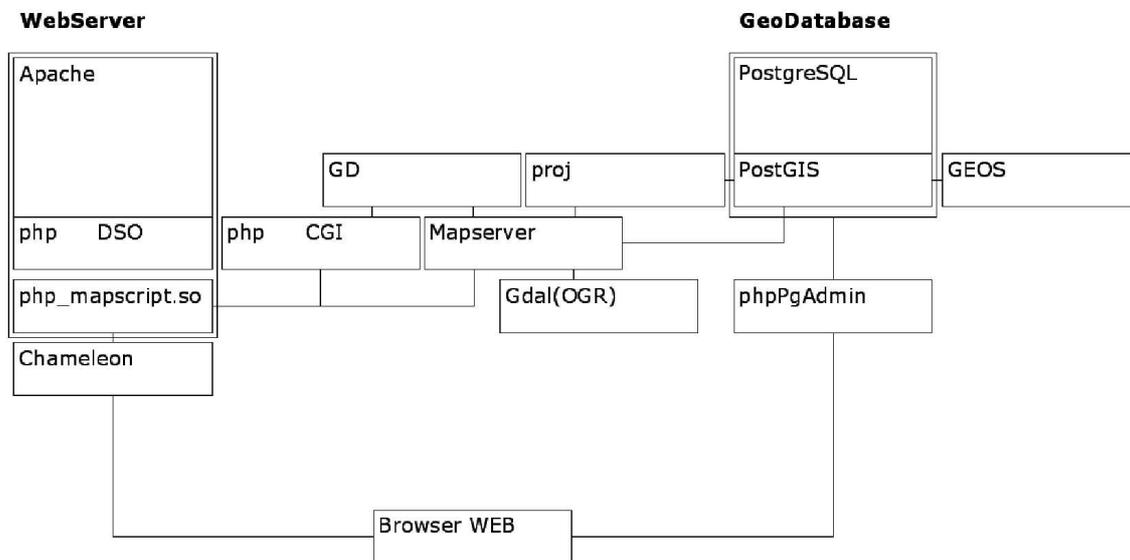


Figure 1: Server software relationships.

the semantic and geometric part of the dataset to both.

5. The client side

The main goal of this study is to test the actual interoperability between a FOSS spatial DBMS server and the most common GIS clients. The interoperability can be on different levels: clients can access both semantic and geometric (and topologic) data or just the semantic part, data can be accessed in read and write or in read only mode.

The choice of the client systems to be tested has been made with reference to their diffusion in public administrations and professional users. A complete list of the tested GIS softwares can be found in table 2. Note that these GISs are not FOSS. Some of these softwares are modular and their connectivity depends on the available modules. Moreover, it is possible to use external connectivity tools, which act as interfaces between a GIS client and data server, such as FME (Feature Manipulation Engine) from Safe Software.

While evaluating the connectivity of GIS and spatial DBMS servers, it must be kept in mind that this is one of the parts more rapidly evolving, since the client-server approach is becoming more appealing.

Each GIS software has a different protocol to connect to a spatial database and it is characterised by a peculiar, more or less complete, proprietior spatial extension. The vendors which have a more specific and restricted market and diffusion prefer the “de facto” standards like Oracle DB, while

Software	version	service
Debian/GNU www.debian.org	3.1	Operative system
PostgreSQL www.postgresql.org	7.4.7	DBMS
PostGIS postgis.refrations.net	1.0	DBMS spatial extension
GEOS geos.refrations.net	2.1.1	DBMS topological extension
Proj4 www.remotesensing.org/proj	4.4.9	Datum management
GDAL/OGR www.remotesensing.org/gdal	1.2.6	Data file format support
Apache www.apache.org	2.0.54	Web server
php www.php.net	4.3.10	DBMS/Web interface
phpPgAdmin phppgadmin.sourceforge.net	3.5.2	PostgreSQL/Web interface
Mapserver mapserver.gis.umn.edu	4.6.2	WebGIS
Chamaleon chameleon.maptools.org	2.2	Mapserver development library

Table 1: Servers and services.

Software	version	vendor	web site
ArcView	3.2	ESRI Inc.	www.esri.com
ArcGIS	8 & 9	ESRI Inc.	www.esri.com
Geomedia	5.0.2	InterGraph Inc.	www.intergraph.com
AutoDesk Land	2005	AutoDesk Inc.	www.autodesk.com
MapInfo	6.0	MapInfo Corporation	www.mapinfo.com

Table 2: Trial clients specifications.

leader companies try to impose their own standards creating new ones, deliberately more tied to their GIS product.

Software systems of table 2 have been set up performing an installation as complete as possible, checking the availability of all the connectivity tools provided by each GIS vendor. Windows 2000 Professional has been the operative system in use.

6. Tests

Connectivity tests consist in uploading some thematic layers to the spatial database and trying to access them directly from client GISs on a remote machine. Data access can happen at different levels:

1. semantic data (attributes) only;
2. semantic data and geocoding of point features by using values in two database columns as coordinates;
3. full access to both geometric and semantic data, with the retrieving of all geometric primitives.

On any remote machine all data can be displayed and modified through the web using the additional services installed on the server: phpPgAdmin for the semantic part and Mapserver for the full geometry and attribute access, see section 4.. All GISs of section 5. have been installed on the same computer, so no difference exists in terms of driver availability and connection settings.

Test data consist in different layers vector data containing all the geometric primitives and complex topology, linked to several database tables. The upload of these layers, originally in *shape* format, to PostgreSQL/PostGIS requires them to be topologically correct: a FOSS tool, JUMP², is available to check and correct geometry and topology errors.

All clients have been able to connect through the ODBC protocol, accessing all the database tables but, of course, the geometry, stored in a database column in the WKB (Well Known Binary) format, is not interpreted as such. Moreover, ArcGIS, ArcView and Geomedia are not able to edit the database tables, even if write access is granted. For establishing an ODBC connection to a PostgreSQL

²<http://www.jumpproject.org>

Software	attribute	geometry	geocoding	add-ons
ArcView	yes	read-only with pgArc	yes	PgArc
ArcGIS 8 & 9	yes	read-only with pgArc	yes	PgArc
Geomedia	ODBC	no	yes	none
AutoDesk Land	ODBC	no	no	none
MapInfo	ODBC	no	yes	none

Table 3: Trial results.

DBMS the PostgreSQL ODBC driver, available under GPL license for all the software platforms on the PostgreSQL web site, must be installed on the client machine. Clients show different behaviours with reference to data commit from the client to the server: some require explicit commit, some auto commit any change, some allow to choose between these commit modes.

The test results are summarised in table 3: most of the GISs have been able to perform geocoding, using the values stored in two database columns as coordinates. These two columns must be inserted in a table by extracting the proper values from the geometry columns, as these contain not only coordinates but also geometric primitives in WKB format.

No one of the client softwares of table 3 has been able to directly connect to the DBMS, either for semantic or geometric data, but their connectivity can be enhanced using add on software. These add-ons are either generic, i.e. able to act as a bridge between a GIS application and a server, such as the FME suite, or specific for a particular GIS software.

In particular, there exist the pgArc ArcGIS extension that allows the direct connection of ArcGIS and ArcView to PostgreSQL. PgArc³ is a FOSS project developed for ArcGIS 8.1 and aims to enable ArcGIS to directly read a spatial database from a PostgreSQL/PostGIS server, with the possibility to upload to the server geometry and semantic data. The software version 4 beta available at time of writing needs some modifications to work with ArcGIS 8.1 and ArcGIS 9.0. It is possible to read geometry except for the *MULTIPOINT* data type, which is incorrectly supported, while it seems not possible to commit to the database changes to the geometry, even if this feature is in theory available.

7. Conclusions

In a short time span, the adoption of spatial DBMSs will be inevitable for medium to large scale GIS datasets, especially when heterogeneous client systems must access geographic data at the same time and selective access must be provided. However, no fully interoperable solution is available today.

Tests have shown still a significant distance between full interoperability and the real use of the systems available today.

Future developments may take two different directions: the development of PostgreSQL/PostGIS driver specific for each client or the development of a PostgreSQL/PostGIS to Oracle Spatial real time

³<http://pgarc.sourceforge.net>

translating server, exploiting the availability of Oracle Spatial drivers for most of the GISs softwares. A PostgreSQL bridge to Oracle is already in development, but not for the spatial part of the data.

While major GIS software providers are focused on creating a full range suite of products covering all the aspects of geographic data management, the coding and adoption of standard formats and protocols seems slower than the users' demand. With regard to this, FOSS systems can provide some advantages because of their "openness" and their natural preference of open standard. FOSS is somehow a natural environment for integration and interoperability in every research and application field: this remains true also for geographic data management.

Acknowledgements

This work was partially supported by grants supplied by the Italian Ministry for School, University and Scientific Research (MIUR) in the frame of the project MIUR-COFIN 2004 - "Strutture evolute della cartografia numerica per i GIS l'ambiente Web" - National Principal Investigator: R. Galetto (Subproject: "Developments of Open Source systems for the distribution of environmental data via WEB" - Local Principal Investigator: P. Zatelli).

References

- [Vallée2004] VALLÈE R., 2004, SIG et SGBD : Etude sur l'interconnexion et l'interopérabilité de Oracle 9i et de PostgreSQL. http://www.carto.net/papers/romain_vallee/romain_vallee_interoperabilite_oracle9i_PostgreSQL_070704.pdf.
- [ESRI1998] ESRI, 1998, ESRI Shapefile Technical Description - An ESRI White Paper - July 1998, <http://www.esri.com/library/whitepapers/pdfs/shapefile.pdf>.
- [OpenGIS Consortium1999] OpenGIS Consortium, 1999, OpenGIS Simple Features Specification for SQL, <http://www.opengis.org/docs/99-049.pdf>.
- [Lake2006] LAKE R., 2006, The Future of Interoperability, <http://www.geoplance.com/gw/2004/0405/0405int.asp>.
- [Smith2004] SMITH S., 2004, Jack Dangermond on Standards and Interoperability, http://www10.giscafe.com/nbc/articles/view_weekly.php?newsletter=1&run_date=30-Aug-2004.
- [Buehler2005] BUEHLER K., 2005, GIS Technology Designed for User Benefit, ESRI White Paper, http://www10.giscafe.com/nbc/articles/view_weekly.php?newsletter=1&run_date=30-Aug-2004.
- [Betancourt2003] BETANCOURT T. L., WILHELMI O., 2003, Final Report on the GIS Demonstration Project, National Center for Atmospheric Research, USA, <http://www.gis.ucar.edu/FinalReport.pdf>.