

GIS MIDDLE EAST

DEVELOPMENT

THE MIDDLE EAST GIS BI MONTHLY MAGAZINE

24 TOWARDS A NATIONAL SPATIAL DATA INFRASTRUCTURE (NSDI)

Dr. Talal Al-Awadhi

28 GEODATABASE FOR URBAN DESIGN AND PLANNING

Simon Yanuar PUTRA

34 FREE, CHEAP, AND POWERFUL POSTGIS, THE OPEN SOURCE GEODATABASE

Paolo Cavallini

Interview

32 MR. RON LAKE

Galdos Systems Inc.

40 MR. SALAHALDEEN MOHAMMED AL MARZOUQI

Dubai Road and Transport Authority

www.gisdevelopment.net



20 Future Trends in Geospatial Data Management Process

Squadron Leader Mudit Mathur

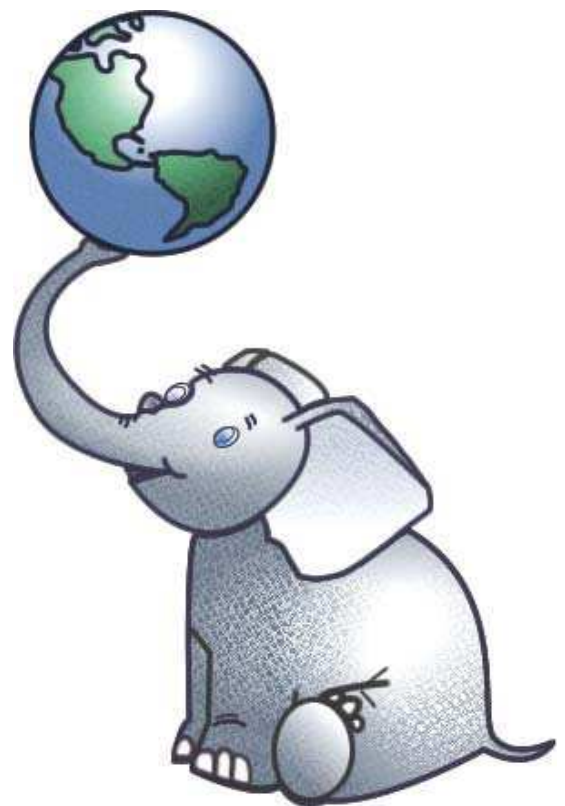
FREE, CHEAP, AND POWERFUL

PostGIS, the Open Source geodatabase

PAOLO CAVALLINI

The geographic database, or geodatabase for short, is an essential component of large and complex GIS environments, and especially for Spatial Data Infrastructure.

Several proprietary applications exist, but are generally complex and very expensive. In the Open Source/Free Software arena, the solution by far more powerful and reliable is PostGIS (<http://www.postgis.org>), a geographic extension of the rock-steady relational, object oriented, database management system PostgreSQL. PostGIS follows OGC standards, and assures the interoperability between GIS analyses, desktop applications, and web applications in a unified framework. It allows a wide variety of users to access large data sets (both geographic and alphanumeric), and the management of their relations, guaranteeing their integrity. It has full 3D support (i.e. volume, as opposed to 2.5D, i.e. draping). It is easy to install on a variety of platforms. It can be managed through many different interfaces, both graphical and character-based, both alphanumeric and geographic.



FROM DESKTOP TO GEODATABASE: DO I NEED A GEODB?

GIS is by definition based on geographic data structures. Desktop GIS applications have habituated users to use only one tool for the visualization of the data as well as for their filing. In the case of vector data, the most widely used format is ESRI shapefile, in which three files are used for every vector assemblages: one for geometries, one for attributes, and one for indexing the relation between the two. This schema can be useful for single users, or small groups, but for larger structures, data replication and security issues require a more robust infrastructure, such as a RDBMS (Relational Data Base Management System).

This system allow a multiuser access to large data sets, guaranteeing security and integrity. It is also an ideal component for web mapping applications, especially when data are dynamic, and should be updated, either by hand or through automatic systems. The standard language to interact with a DBMS is the Structured Query Language (SQL), a standard, easy to learn language that allows to select, insert, modify, calculate, apply functions etc. on a data base represented by a number of related tables. A wide lit-

For larger structures, data replication insecurity robust infrastructure such as a RDBMS is required

Well Known Text

```
the_geom
SRID=26591;MULTIPOLYGON(((1719375.90049738 4815032.60850675...4815032.60850675)))
```

Well Known Binary

```
the_geom
0106000020DF6700000100000001030000000100...10DFF86E64F3C3A414CC6F1262E5E5241
```

Fig. 1 . The Well Known Text format is human-readable: you can see the Spatial Reference ID (SRID: a number which identifies unambiguously the projection of every single record), the type of geometry and the sequence of vertices as xy coordinates. The Well Known Binary is more compact, but unreadable by humans.

26591 EPSG	26591 PROJCS["Monte Mario (Rome) / Italy zone 1",GEOGCS[...	+proj=tmmerc +lat_0=0 +lon_0=-3.45233333333333 +k=...
26592 EPSG	26592 PROJCS["Monte Mario (Rome) / Italy zone 2",GEOGCS[...	+proj=tmmerc +lat_0=0 +lon_0=2.54766666666667 +k=0...
26632 EPSG	26632 PROJCS["M'poraloko / UTM zone 32N",GEOGCS["M'pora...	+proj=utm +zone=32 +a=6378249.2 +b=6356515 +units...
26692 EPSG	26692 PROJCS["M'poraloko / UTM zone 32S",GEOGCS["M'pora...	+proj=utm +zone=32 +south +a=6378249.2 +b=6356515...
26703 EPSG	26703 PROJCS["NAD27 / UTM zone 3N",GEOGCS["NAD27",DATUM...	+proj=utm +zone=3 +ellps=clrk66 +datum=NAD27 +uni...
26704 EPSG	26704 PROJCS["NAD27 / UTM zone 4N",GEOGCS["NAD27",DATUM...	+proj=utm +zone=4 +ellps=clrk66 +datum=NAD27 +uni...
26705 EPSG	26705 PROJCS["NAD27 / UTM zone 5N",GEOGCS["NAD27",DATUM...	+proj=utm +zone=5 +ellps=clrk66 +datum=NAD27 +uni...
26706 EPSG	26706 PROJCS["NAD27 / UTM zone 6N",GEOGCS["NAD27",DATUM...	+proj=utm +zone=6 +ellps=clrk66 +datum=NAD27 +uni...
26707 EPSG	26707 PROJCS["NAD27 / UTM zone 7N",GEOGCS["NAD27",DATUM...	+proj=utm +zone=7 +ellps=clrk66 +datum=NAD27 +uni...
26708 EPSG	26708 PROJCS["NAD27 / UTM zone 8N",GEOGCS["NAD27",DATUM...	+proj=utm +zone=8 +ellps=clrk66 +datum=NAD27 +uni...
26709 EPSG	26709 PROJCS["NAD27 / UTM zone 9N",GEOGCS["NAD27",DATUM...	+proj=utm +zone=9 +ellps=clrk66 +datum=NAD27 +uni...
26710 EPSG	26710 PROJCS["NAD27 / UTM zone 10N",GEOGCS["NAD27",DATU...	+proj=utm +zone=10 +ellps=clrk66 +datum=NAD27 +un...
26711 EPSG	26711 PROJCS["NAD27 / UTM zone 11N",GEOGCS["NAD27",DATU...	+proj=utm +zone=11 +ellps=clrk66 +datum=NAD27 +un...
26712 EPSG	26712 PROJCS["NAD27 / UTM zone 12N",GEOGCS["NAD27",DATU...	+proj=utm +zone=12 +ellps=clrk66 +datum=NAD27 +un...
26713 EPSG	26713 PROJCS["NAD27 / UTM zone 13N",GEOGCS["NAD27",DATU...	+proj=utm +zone=13 +ellps=clrk66 +datum=NAD27 +un...
26714 EPSG	26714 PROJCS["NAD27 / UTM zone 14N",GEOGCS["NAD27",DATU...	+proj=utm +zone=14 +ellps=clrk66 +datum=NAD27 +un...
26715 EPSG	26715 PROJCS["NAD27 / UTM zone 15N",GEOGCS["NAD27",DATU...	+proj=utm +zone=15 +ellps=clrk66 +datum=NAD27 +un...

Fig. 2 A sample from the 2671 standard projections (SRIDs) available in a PostGIS geodatabase. Any additional custom projection can be defined here.

erature is available to learn SQL (Google or other web crawlers are your friends here). The architecture of a geodatabase differs from a standard RDBMS not only because it can handle geographic (i.e., geometry+projections) data, but also for a larger set of commands that extend standard SQL language (e.g. distance calculations, buffers, overlay, conversion between coordinate systems, etc.). Open Geospatial Consortium (<http://www.opengeospatial.org>; OGC for short) has released a series of standards for geographic data formats (<http://www.opengeospatial.org/docs/99-049.pdf>), which define:

- which tables must be present in a geodatabase (geometry columns table and spatial reference system table);
- data formats, called Simple Features

(point, line, polygon, etc.), in one of two different formats (Fig. 1 [Insert image figura_1.png]):

- Well Know Text (WKT): easier to read for humans, but disk-space wasting
- Well Know Binary (WKB): unreadable for humans, but faster and more compact for a computer;
- a set of SQL-like instructions for geographic analysis.

The document is now the reference for any geodatabase; PostGIS is fully OGC compliant (at the time of writing, formal acceptance from OGC is under way).

POSTGIS AND POSTGRESQL: A BIT OF HISTORY

Back in 2001, the Canadian GIS firm Refractions (Victoria, British Columbia: <http://www.refractions.net>), analysed

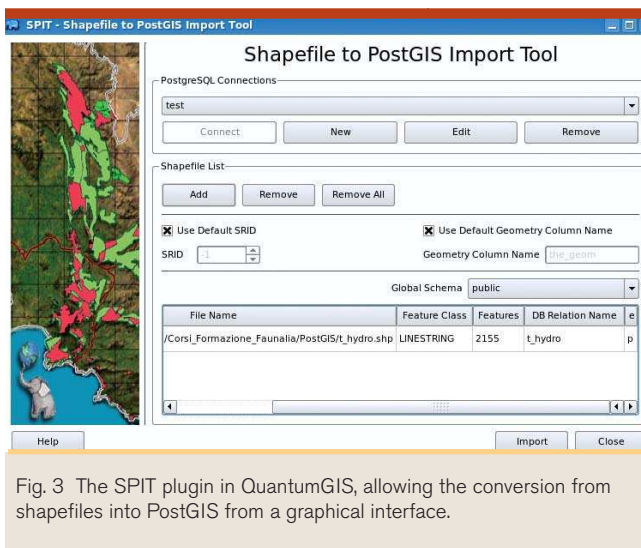


Fig. 3 The SPIT plugin in QuantumGIS, allowing the conversion from shapefiles into PostGIS from a graphical interface.

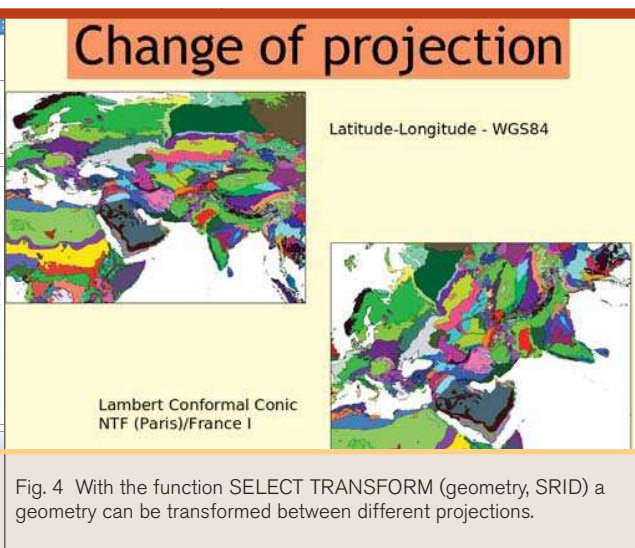


Fig. 4 With the function SELECT TRANSFORM (geometry, SRID) a geometry can be transformed between different projections.

the geodatabase systems available on the market, and concluded that all were either very expensive, very slow, not expandable and customizable, or some combination of the above. Therefore, they decided to start the development of a new tool.

Much in the line of the Open Source strategy, they built on existing blocks, and in particular on the most advanced, powerful and stable Open Source RDBMS available: PostgreSQL (<http://www.postgresql.org>). PostgreSQL has very interesting features, paralleled only by the most powerful (and expensive) proprietary systems, among which:

- it is fully ACID compliant (guarantees the integrity of transactions);
- it supports a wide set of different data types;
- allows the management of very large data bases (hundreds of Terabytes), even with computer of limited power;
- allows clustering, load balancing, efficient use of multiprocessor computers, and data redundancy;
- has strong security, based on several cryptographic modes
- it has ODBC and JDBC drivers, so it can be accessed from a variety of clients, even proprietary (like the popular MicroSoft Access).

See <http://www.postgresql.org/about/advantages> for further details. Starting

from current main version (8.x), it is available natively for many different operating systems (Solaris, Linux, HPUX, FreeBSD, and even Microsoft Windows).

This choice was a wise one, and allowed PostGIS to grow rapidly, and to fulfil all the requirements of OGC specifications within a couple of years.

The integration with the Java Topology Suite (ported in C++ as GEOS library: <http://geos.refractor.net>) allowed the analysis of geographic data directly within the database, without any need for additional, external tools. The integration with the PROJ library (<http://proj.maptools.org>) allowed the management and conversion of thousands of different projection systems.

HOW DO I GET IT?

Installation and setup of PostgreSQL +PostGIS is very straightforward on the majority of systems: most Linux distribution have it packaged (thus, it is installable with a couple of clicks, or one simple command). Windows users have a clickable installer available. Of course, being Open Source and Free Software, its download and installation is free, in both senses of not having to pay anything and being free to use and modify the program at your will. Once installed, the first con-

figuration is easy, but requires the correct understanding of some basic concept (what is a database function, what is the spatial reference system table, etc.). The most advanced Linux distribution (in particular our preferred, Debian GNU/Linux: <http://www.debian.org>) do automatically most of the necessary configurations.

Once installed and configured, the management of the database can be done, either with a graphic client (see below) or through the native PostgreSQL command line interface.

HOW DO I GET MY DATA IN?

PostGIS allows for seven different geographic data types:

POINT, LINESTRING, POLYGON, MULTYPOINT (collection of different point elements), MULTILINESTRING (collection of different linear elements), MULTYPOLYGON (collection of different areal elements), GEOMETRYCOLLECTION (collection of elements of different types: points, lines, polygons).

Each of them can be either two (x,y) or three-dimensional (x,y,z, i.e. fully three-dimensional, not simply draping). Every record has its own Spatial Reference ID (SRID), one code from the spatial_ref_sys table (Fig. 2 [Insert image figura_2.png]), identifying uniquely its projection sys-

tem; user can therefore mix data from different sources, keeping each in the original projection format, or converting them during the importation process.

For migrating large and complex sets of data, it is possible to use directly the Simple Feature (OGC) SQL, but several tools make the life easier:

- PostGIS itself comes bundled with its own command line tool (shp2pgsql: from shape to postgis) who converts ESRI shapefiles into the necessary set of SQL commands, to be fed to PostgreSQL
- the program (also command-line driven) ogr2ogr, included in the distribution of the library gdal (<http://www.gdal.org>) is capable of converting between over 20 vector formats (a list at: http://www.gdal.org/ogr/ogr_formats.html), including shapefile, ArcInfo, Oracle Spatial, etc.
- if you do not like the command line, QGIS (<http://qgis.org>), an easy-to-use free desktop GIS, has a plugin that allows importation of shapefiles into PostGIS entirely from the graphical interface (Fig. 3 [Insert image figura_3.png]).

Of course, the design of a complex data base structure, with normalization, optimization, constraints, triggers etc. is a fairly complex issue, and requires specialized personnel.

POSTGIS FUNCTIONS: WHAT CAN I DO WITH IT?

Describing all the functions, and their combination, of PostGIS/PostgreSQL is a very large and complex task. In addition to the basic OGC functions, PostGIS offers a set of additional functions, that allow complex tasks and analyses. Here we offer a brief overview of both. Open Geospatial Consortium standard functions

Base Functions

Allow the creation and deletion of geometric columns and the setting of a geographic reference system.

Geometry Relationship Functions

Calculate the distances and evaluate the topological relations between objects (touches, intersects, contains, etc.).

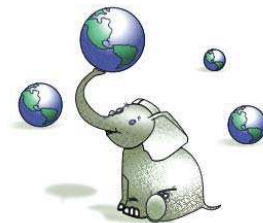
Geometry Processing Functions

Calculate area, perimeter, centroid, buffer, convex hull, intersection, union, etc. between geometries.

Geometry Information

Return a variety of information on the

PostGIS
Geographic Objects for PostgreSQL



A Spatial Database for the Rest of Us

geometry, e.g. its reference system, the number of points contained, the value (x,y,z) of a single object, etc. Allow also the visualization in WKT (text, human-readable) of a binary geometry.

Geometry Creation

Build a geometry (points, lines and polygons), starting from coordinate sets. Useful for converting an alphanumeric data-

WHAT IS FREE SOFTWARE? WHAT IS OPEN SOURCE?

Free software, as defined by the Free Software Foundation, is software which can be used, copied, studied, modified and redistributed without restriction. Freedom from such restrictions is central to the concept, with the opposite of free software being proprietary software (a distinction unrelated to whether a fee is charged). The usual way for software to be distributed as free software is for the software to be licensed to recipient with a free software license, and the source code of software to be made available (for a compiled language).

Open-source software is computer software whose source code is available under a copyright license that permits users to study, change, and improve the software, and to redistribute it in modified or unmodified form.

In practice, the two terms are used interchangeably, even though the first is more correct and general, while the second

is regarded more as a marketing name.

One of the many interesting readings on the subject is also: Why Open Source Software / Free Software? Look at the Numbers! by David A. Wheeler: http://www.dwheeler.com/oss_fs_why.html

In the FS/OSS sectors, the business revolves around providing services (installation, configuration, migration, support, etc.) rather than selling the programs. A number of commercial firms show that this is a viable approach.

Among the main advantages for the user of Free Software, other than lower total cost of ownership and easier customizability, are a higher security, because the open code cannot contain unnoticed viruses, trojan horses, spyware and other hidden features. This is especially important for strategic data.

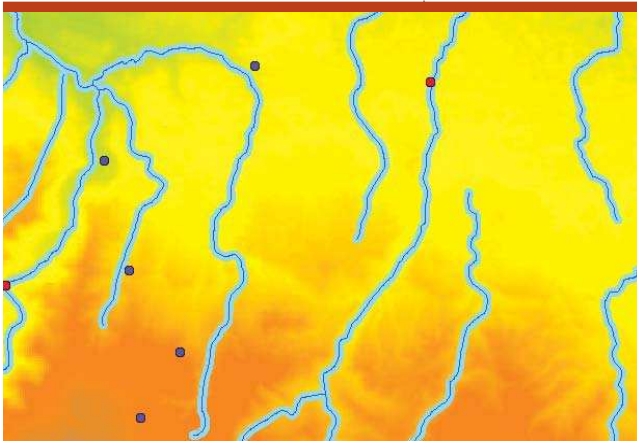


Fig. 5 From PostGIS we can select records on the basis of spatial relationships: in this case, only points within 50 meters of rivers are selected (in red), the others are in blue.

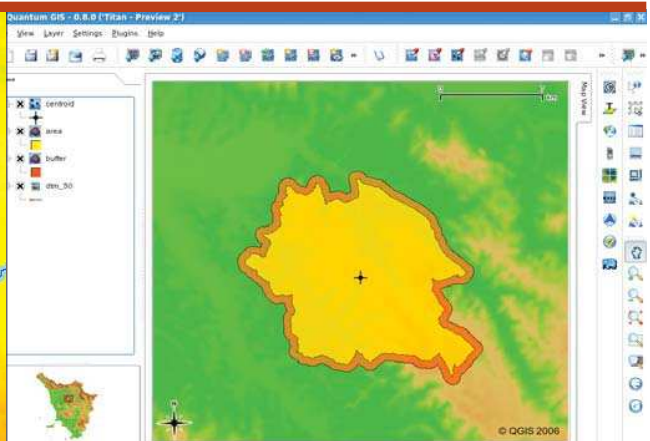


Fig. 6 A simple example of interaction between PostGIS and QuantumGIS: a shapefile (in green), imported in the database with the SPIT plugin, is displayed in QuantumGIS, together with its buffer (in orange) and its centroid (the cross), calculated by PostGIS

base (where coordinates are stored as 'x', 'y' and possibly 'z') into a geographic one.

POSTGIS EXTENSIONS

Measurement Functions

Calculate the areas, perimeters, lengths, etc. in relation to the reference system used and along the earth's spheroid, in 2 or 3 dimensions.

Output Functions

Export the data as WKB and WKT, as SVG (Scalable Vector Graphics; an XML format for the description of geometries, standard W3C: www.w3.org/TR/SVG/; can be used in vector graphic programs) or GML (Geographics Markup Language; an XML format for GIS, standard OGC).

PostGIS integrates well with a wide number of programs

Geometry Creation and Editing

Allow a number of transformation of the geometries: change of reference system, translation, affine, rotation, scaling, simplify, snap to grid, etc.; (Fig. 4)

Logical Operators

Allow to check coincidence, overlap etc. between geometries (Fig. 5).

Linear Referencing Functions

A new addition to PostGIS, this set of functions (especially useful for the management of roads, pipelines, and other linear systems) allow to refer to a point on a line by its distance from the origin and from other points. For a complete and updated description of all the functions, please refer to the official PostGIS documentation, available at: <http://postgis.refractor.net/docs>.

POSTGIS AND OTHER FREE SOFTWARE

Many task may result tedious or difficult for the novice by the direct use of standard, command-line SQL interface. Even advanced users find it easier to perform some task with something fancier. Being a product of the Open Source and Free Software community, PostGIS integrates well with a wide number of other pro-

grams; the base library OGR (<http://www.gdal.org/ogr>, part of the larger GDAL) allows most of other free software to read and write from/to PostGIS. Among the most important we can cite:

- GRASS (<http://grass.itc.it>), a powerful analytical GIS
- QuantumGIS (<http://qgis.org>; Fig. 6 [Insert image figura_6.jpg]), an easy-to-use desktop GIS, can also digitize points, lines and polygons directly into PostGIS
- uDIG (<http://udig.refractor.net>), a JAVA desktop/web client GIS
- JUMP (<http://jumpilot.sourceforge.net>), a JAVA desktop GIS
- DeeGree <http://www.deegree.org>, a JAVA framework for Spatial data Infrastructures
- UMN Mapserver (<http://mapserver.gis.umn.edu>), the fastest and most powerful web mapping engine
- GEOserver (<http://geoserver.sourceforge.net/html>), an alternative (mostly vector) web mapping engine.

Also several non-free systems can interface with PostGIS.

The database can also be accessed by a variety of non-geographic graphical clients, including desktop programs (e.g. PgAdmin: <http://www.pgadmin.org>, ReCALL <http://www.recallrevealed.org>, and others), web interfaces (phpPgAdmin: <http://phpPgAdmin.sourceforge.net>, and also the popular OpenOffice.org suite

(<http://www.openoffice.org>). The same data (with appropriate read/write privileges) can therefore be useful for all members of a structure, from the secretary who will prepare a report without knowing anything about GIS, to the GIS analyst who will extensively manipulate the data.

CAN WE TRUST IT?

The software at the base of the project, the RDBMS PostgreSQL, is notorious for its stability and reliability, and it is deployed in thousands of mission-critical applications. PostGIS is actively developed by a healthy community, spread worldwide. A number of new features are being developed or planned; at the time of writing, its main limitation is the lack of support for rasters, but given the rate of development we can be confident that this will be overcome in a reasonable time frame. It has proved himself capable of handling

very complex situations, with data base in the order of Terabytes, with tens of millions records.

Considering the very young age of the project, it has a surprisingly large base of professional users.

Support is available, both for free, through the very responsive mailing list associate to the project, and commercially, through a number of companies (including Faunalia, of course!).] As typical for Open Source software, the eventual bugs are fixed with surprising speed (often a matter of hours to days); this marks a striking difference with proprietary software, for which the entire cycle of bug fixing usually takes months. Missing features can be implemented with limited investment (often much lower than the cost of a single licence for a proprietary equivalent).

In conclusion, our suggestion is just start testing and using it: most probably

will fit your needs. Do not forget to seek community and professional support instead of stopping at the first problem, and if you will adopt it for serious business, do not forget to give back to the community some of the benefits you got for free.

Happy free mapping!



Paolo Cavallini

Faunalia, Piazza Garibaldi
5 - Pontedera (PI), Italy

cavallini@faunalia.it



Emilia Venturato

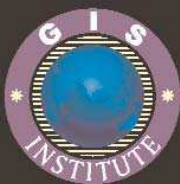
Faunalia
Piazza Garibaldi 5 - Pontedera (PI)
Italy

venturato@faunalia.it



Maurizio Napolitano

ITC-irst, Divisione SRA,
Via Sommarive 18 - Povo (TN)
Italy



BUILDING CAREER IN GEO-SPATIAL TECHNOLOGY

Diploma in Geo-informatics

Date of commencement 13 November 2006

Last date for registration 03 November 2006

No. of seats 15

Minimum qualification

Graduate / Post Graduate in Engineering | Science | Geography | Geology, Environment | Earth Science | Civil | Planning | Computer science or related fields with some knowledge / experience in the field of mapping.

Duration 8 weeks full time (5hrs/day, five days in a week)

FIELDWORK

- GPS based Mobile Mapping

SOFTWARE

- AutoCAD/AutoCAD Map
- ArcView/ArcInfo/Arc GIS
- MapInfo Professional
- ERDAS Imagine



Get **UNIGIS** Professional Diploma



Career Orientation



100% Placement Record

www.GISinstitute.net

For more details please contact: **Neeraj Budhari** Course Co-ordinator

Venue: GIS Institute, G-4, Sector-39, Noida-201 301, INDIA, **Tel** +91-120-2502180 to 87, **Fax** + 91-120-2500811, **Mobile** 9810495813, 9871725868

Email info@GISinstitute.net, **Web** www.GISinstitute.net