



ΜΑΘΗΜΑ: **Χωρικές Βάσεις Δεδομένων**

[Άνοιξη 2014]

Εισαγωγή στο PostGIS 2.x

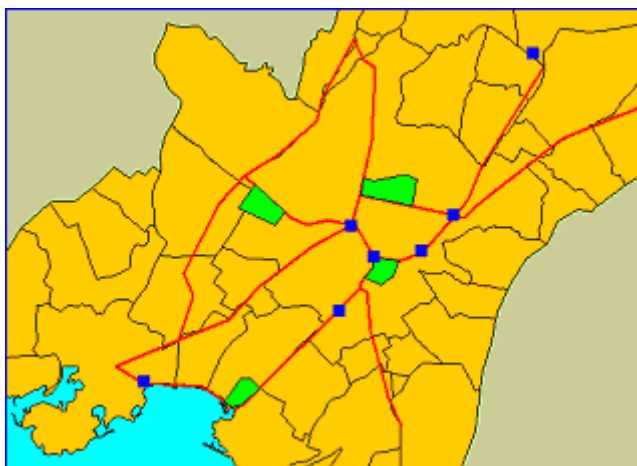
Κ. Πατρούμπας

kpatro@dblab.ece.ntua.gr

Σχεδιασμός απλουστευμένου σχήματος χωρικών στοιχείων, υλοποίηση σε περιβάλλον PostGIS 2.x (επέκταση της PostgreSQL 9.x), ορισμός ευρετηρίων, υποβολή τυπικών ερωτημάτων σε SQL.

Γενικό Πλαίσιο:

Ας θεωρήσουμε ένα σύνολο γεωγραφικών οντοτήτων που περιλαμβάνουν οδικό δίκτυο (**Roads**), διοικητική διαίρεση (**Zones**) και σημεία ενδιαφέροντος (**Locations**), λ.χ. για την περιοχή Αθηνών:



Οδηγίες εκτέλεσης:

Α. ΣΥΝΔΕΣΗ ΜΕ ΤΗΝ ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ

Βήμα 1^ο. Συνδεθείτε στην PostgreSQL 9.x (λ.χ. την έκδοση 9.2) με το συνθηματικό σας και επιλέξτε βάση δεδομένων. Η σύνδεση αυτή μπορεί να επιτευχθεί σε δύο εναλλακτικά περιβάλλοντα εργασίας:

- **Γραμμή εντολών (command prompt)**. Από το **Start -> Programs -> PostgreSQL 9.2 -> SQL shell (psql)**, ανοίγει μια κονσόλα. Για να συνδεθείτε με την PostgreSQL θα πρέπει να δώσετε στοιχεία για τον server (*default: localhost* για σύνδεση με β.δ. στο ίδιο μηχάνημα), το όνομα της β.δ. (*default: postgres*), το port (*default: 5432*), καθώς και το όνομα (username) και το συνθηματικό σας (password). Όταν θέλετε να κλείσετε την σύνδεση με την β.δ., εκτελείτε `\q` για έξοδο απ' το περιβάλλον psql της PostgreSQL.
- Το ίδιο περιβάλλον (**command prompt**) μπορείτε να έχετε ανοίγοντας μία κονσόλα (με **Start -> Programs -> Windows System -> Command Prompt**) και ακολούθως πηγαίνοντας στο μονοπάτι όπου υπάρχει εγκατεστημένη η PostgreSQL, λ.χ., `cd "C:\Program Files\PostgreSQL\9.2\bin"`. Εκεί μπορείτε ν' αλλάξετε την κωδικοσελίδα ώστε να απεικονίζονται σωστά στην οθόνη οι ελληνικοί χαρακτήρες (εφόσον χρησιμοποιούνται στα περιεχόμενα της β.δ.). Αυτή η αλλαγή γίνεται σε δύο βήματα:
 - i) εκτελώντας την εντολή `cmd.exe /c chcp 1253` στην κονσόλα (1253 είναι η ελληνική κωδικοσελίδα των Windows) και
 - ii) αλλάζοντας την γραμματοσειρά της κονσόλας σε **Lucida Console** από το μενού της κονσόλας (με το μενού **Properties -> Font** της κονσόλας).

Στην συνέχεια, μπορείτε να συνδεθείτε στην PostgreSQL εκτελώντας στην κονσόλα:

```
psql -U postgres
```

οπότε θα σας ζητηθεί και το συνθηματικό για τον χρήστη 'postgres' ή όποιον άλλον χρήστη έχετε δημιουργήσει για την σύνδεσή σας.

Η επιλογή της β.δ. στην οποία θα εργαστείτε (geodb) γίνεται με μια εντολή της μορφής:

```
\c geodb
```

Αν δεν επιλέξετε β.δ., τότε θεωρείται ότι εργάζεστε στην προεγκατεστημένη β.δ. (συνήθως με το όνομα postgres). Ακολούθως μπορείτε να εκτελείτε τις γνωστές εντολές SQL (CREATE, INSERT, DROP, SELECT,...).

Τέλος, εκτελείτε **\q** για την έξοδο απ' το περιβάλλον psql της PostgreSQL.

- **Παραθυρικό περιβάλλον**, εφόσον με την εγκατάσταση της PostgreSQL έχει εγκατασταθεί και η εφαρμογή pgAdmin. Στην περίπτωση αυτή, εκτελείτε την εφαρμογή από το **Start -> Programs -> PostgreSQL 9.x -> pgAdminIII**.

Στην οθόνη που εμφανίζεται, κάνετε ΔΕΞΙ κλικ πάνω στο PostgreSQL Database Server 9.x (πάνω αριστερά) και επιλέγετε **Connect** από το μενού. Ακολούθως, πρέπει να δώσετε το συνθηματικό (password) που ζητείται.

Τότε στην αριστερή πλευρά της οθόνης εμφανίζονται όλες οι διαθέσιμες βάσεις δεδομένων (databases). Χτυπάτε πάνω στην βάση δεδομένων στην οποία θα εργαστείτε (λ.χ. postgres ή geodb), η οποία *θα πρέπει να υποστηρίζει τις λειτουργίες του PostGIS*.

Από το μενού **Tools -> Query tool** θα ανοίξετε μια άλλη φόρμα, η οποία θα σας επιτρέψει να εκτελέσετε ερωτήματα.

ΠΡΟΣΟΧΗ: πάνω στο μενού της φόρμας αυτής θα πρέπει να αναγράφεται η βάση δεδομένων που έχετε επιλέξει προηγουμένως (λ.χ. geodb).

Βήμα 2^ο. (ΠΡΟΑΙΠΕΤΙΚΟ) Δημιουργία νέας βάσης δεδομένων με ονομασία **geodb**, κωδικοσελίδα **ISO 8859_7** (ή **UTF8**) για υποστήριξη ελληνικών (αντιστοίχως Unicode) και υποστήριξη λειτουργιών του PostGIS.

```
CREATE DATABASE geodb
  WITH OWNER = postgres
       ENCODING = 'ISO_8859_7'
       TABLESPACE = pg_default;
```

Η βάση δεδομένων **geodb** θα πρέπει να ενισχυθεί με τις λειτουργίες του PostGIS, ώστε να είναι δυνατή η εκτέλεση των εντολών. Το PostGIS απαιτεί την επέκταση με εντολές της γλώσσας PL/pgSQL, η οποία θα πρέπει να είναι διαθέσιμη για την συγκεκριμένη βάση δεδομένων (geodb). Αυτό μπορεί να γίνει έπειτα από σύνδεση στο περιβάλλον psql (από το μενού **Start -> Programs -> PostgreSQL 9.x -> SQL Shell (psql)** και πληκτρολογώντας στην κονσόλα:

```
createlang plpgsql geodb
```

Κατόπιν πρέπει να φορτωθούν στην βάση δεδομένων (geodb) οι ορισμοί των αντικειμένων και των συναρτήσεων του PostGIS με την εντολή από περιβάλλον psql:

```
psql -d geodb -f lwpostgis.sql
```

Αν επιπλέον τα δεδομένα αναφέρονται σε ένα γνωστό σύστημα αναφοράς (λ.χ. WGS84, ΕΓΣΑ'87) θα πρέπει να οριστούν συστήματα αναφοράς στον πίνακα SPATIAL_REF_SYS της συγκεκριμένης βάσης (geodb) με την εντολή από περιβάλλον psql:

```
psql -d geodb -f spatial_ref_sys.sql
```

Στο σημείο αυτό, θα πρέπει να έχετε δημιουργήσει μια σωστά ρυθμισμένη βάση δεδομένων σε PostgreSQL με υποστήριξη του PostGIS.

Παρατήρηση: Αυτό το βήμα ΔΕΝ χρειάζεται να το εκτελέσετε, εφόσον η βάση δεδομένων όπου θα εργαστείτε έχει ήδη δημιουργηθεί και υποστηρίζει τις λειτουργίες του PostGIS.

Βήμα 3°. Δημιουργία νέου σχήματος με την ονομασία **network** :

```
CREATE SCHEMA network
  AUTHORIZATION postgres;
GRANT ALL ON SCHEMA network TO postgres;
GRANT ALL ON SCHEMA network TO public;
```

ΠΡΟΣΟΧΗ!!! Όλα τα δεδομένα μπορεί να εισαχθούν εναλλακτικά στο υπάρχον σχήμα **public**, προκειμένου να είναι αναγνωρίσιμα και εκτός περιβάλλοντος PostGIS (λ.χ. μέσω του λογισμικού **uDIG**). Σε περιβάλλον **uDIG**, για την β.δ. geodb δεδομένα σε WGS84 είναι αναγνωρίσιμα από οποιοδήποτε σχήμα. Ατυχώς όμως, δεδομένα σε ΕΓΣΑ'87 είναι προσβάσιμα μόνο απ' το σχήμα **public**. Εναλλακτικά, όλα τα δεδομένα μπορούν να φορτωθούν στην β.δ. **postgis** που δημιουργείται κατά την εγκατάσταση του PostGIS.

Βήμα 4°. Ορισμός του πίνακα σημειακών θέσεων σε **ΕΓΣΑ'87** (κωδικός **2100**):

```
CREATE TABLE network.locations(
  location_id      INTEGER PRIMARY KEY,
  location_name    VARCHAR(30) );
```

--Προσθήκη του πεδίου της γεωμετρίας (2-διάστατα σημεία)

```
SELECT AddGeometryColumn('network', 'locations', 'pos', 2100, 'POINT', 2);
```

--Καθορισμός ελέγχου εγκυρότητας της γεωμετρίας

```
ALTER TABLE network.locations
ADD CONSTRAINT location_valid_check CHECK (ST_isvalid(pos));
```

--Εναλλακτικό παράδειγμα με στοιχεία σε WGS84 (κωδικός 4326) στο σχήμα public!!!

```
CREATE TABLE locwgs(
  location_id      INTEGER PRIMARY KEY,
  location_name    VARCHAR(30) );
```

--Δημιουργία πεδίου γεωμετρίας, έλεγχος γεωμετριών και δημιουργία χωρικού ευρετηρίου:

```
SELECT AddGeometryColumn('public', 'locwgs', 'pos', 4326, 'POINT', 2);
ALTER TABLE locwgs ADD CONSTRAINT locwgs_valid_check CHECK (ST_isvalid(pos));
CREATE INDEX locwgs_idx ON locwgs USING GIST (pos);
```

--Εισαγωγή εγγραφών:

```
INSERT INTO locwgs VALUES (301, 'SYNTAGMA', ST_GeomFromText('POINT(23.7358
37.976)', 4326));
INSERT INTO locwgs VALUES (302, 'OMONIA', ST_GeomFromText('POINT(23.7286
37.9838)', 4326));
INSERT INTO locwgs VALUES (303, 'AMPELOKIPOI', ST_GeomFromText('POINT(23.7611
37.9866)', 4326));
```

Βήμα 5°. Δημιουργία ευρετηρίου με γενικευμένα δένδρα αναζήτησης (GiST) για τα σημεία:

```
CREATE INDEX locations_idx ON network.locations USING GIST (pos);
```

Βήμα 6°. Ορισμός του πίνακα γραμμικών αξόνων:

```
CREATE TABLE network.links(
  link_id          INTEGER PRIMARY KEY,
  roadname_gr     VARCHAR(50),
  roadname_lt     VARCHAR(50),
  weight          FLOAT );
```

--Προσθήκη του πεδίου της γεωμετρίας (2-διάστατες πολυγραμμές)

```
SELECT AddGeometryColumn('network', 'links', 'segment', 2100, 'LINESTRING', 2);
```

--Καθορισμός ελέγχου εγκυρότητας της γεωμετρίας

```
ALTER TABLE network.links
ADD CONSTRAINT segment_valid_check CHECK (ST_Isvalid(segment));
```

Βήμα 7^ο. Δημιουργία ευρετηρίου με γενικευμένα δένδρα αναζήτησης (GiST) για τους άξονες:

```
CREATE INDEX links_idx ON network.links USING GIST (segment);
```

Βήμα 8^ο. Δημιουργία πίνακα πολυγωνικών περιοχών:

```
CREATE TABLE network.zones (
zone_id      INTEGER PRIMARY KEY,
zone_name    VARCHAR(30) );
```

--Προσθήκη του πεδίου της γεωμετρίας (2-διάστατα πολύγωνα)

```
SELECT AddGeometryColumn('network', 'zones', 'sector', 2100, 'POLYGON', 2);
```

--Καθορισμός ελέγχου εγκυρότητας της γεωμετρίας

```
ALTER TABLE network.zones
ADD CONSTRAINT sector_valid_check CHECK (ST_Isvalid(sector));
```

Βήμα 9^ο. Κατασκευή ευρετηρίου για τα πολύγωνα με γενικευμένα δένδρα αναζήτησης (GiST):

```
CREATE INDEX zones_idx ON network.zones USING GIST (sector);
```

B. ΕΙΣΑΓΩΓΗ ΣΤΟΙΧΕΙΩΝ με εντολές INSERT σε WKT format [1^{ος} τρόπος]

Στην περίπτωση αυτή, κάθε γεωμετρική οντότητα διατυπώνεται βάσει της ακολουθούμενης από το OGC μορφής Well-Known Text (WKT) μαζί με προσδιορισμό συστήματος αναφοράς (2100):

Βήμα 10^ο. Εισαγωγή στοιχείων στον πίνακα των σημειακών θέσεων.

```
INSERT INTO network.locations VALUES (301, 'ΣΥΝΤΑΓΜΑ',
ST_GeomFromText('POINT(476654 4202904)', 2100));
```

```
INSERT INTO network.locations VALUES (302, 'ΟΜΟΝΟΙΑ',
ST_GeomFromText('POINT(476021 4203771)', 2100));
```

```
INSERT INTO network.locations VALUES (303, 'ΑΜΠΕΛΟΚΗΠΟΙ',
ST_GeomFromText('POINT(478878 4204071)', 2100));
```

```
INSERT INTO network.locations VALUES (304, 'ΦΙΕ',
ST_GeomFromText('POINT(475687 4201381)', 2100));
```

```
INSERT INTO network.locations VALUES (305, 'ΣΤΑΔΙΟ ΚΑΡΑΪΣΚΑΚΗ',
ST_GeomFromText('POINT(470229 4199447)', 2100));
```

```
INSERT INTO network.locations VALUES (306, 'ΟΛΥΜΠΙΑΚΟ ΣΤΑΔΙΟ',
ST_GeomFromText('POINT(481094 4208546)', 2100));
```

```
INSERT INTO network.locations VALUES (307, 'ΕΥΑΓΓΕΛΙΣΜΟΣ',
ST_GeomFromText('POINT(477963 4203061)', 2100));
```

Βήμα 11^ο. Εισαγωγή στοιχείων στον πίνακα των γραμμικών αξόνων.

```
INSERT INTO network.links VALUES (101,'ΑΛΕΞΑΝΔΡΑΣ ΛΕΩΦ.','ALEXANDRAS',18.25,
ST_GeomFromText('LINESTRING( 476259 4204638 , 478895 4204061)', 2100));
```

```
INSERT INTO network.links VALUES (102,'ΒΑΣ. ΣΟΦΙΑΣ ΛΕΩΦ.','VAS. SOFIAS',24.69,
ST_GeomFromText('LINESTRING( 476638 4202908 , 477354 4202809 , 477997 4203023 ,
478895 4204061)', 2100));
```

```
INSERT INTO network.links VALUES (103,'ΠΑΝΕΠΙΣΤΗΜΙΟΥ','PANEPISTIMIΟΥ',12.87,
ST_GeomFromText('LINESTRING( 476638 4202908 , 476152 4203748 , 475995
4203798)', 2100));
```

```
INSERT INTO network.links VALUES (104,'ΠΑΤΗΣΙΩΝ','PATISSION',18.3,
ST_GeomFromText('LINESTRING( 475995 4203798 , 476259 4204638)', 2100));
```

```
INSERT INTO network.links VALUES (105,'ΑΜΑΛΙΑΣ ΛΕΩΦ.','AMALIAS',26.86,
ST_GeomFromText('LINESTRING( 476638 4202908 , 476598 4202603 , 476269 4202197 ,
476250 4202037)', 2100));
```

```
INSERT INTO network.links VALUES (106,'ΣΥΓΓΡΟΥ ΛΕΩΦ.','SYNGROU',62.34,
ST_GeomFromText('LINESTRING( 476250 4202037 , 472967 4198790)', 2100));
```

```
INSERT INTO network.links VALUES (107,'ΚΗΦΙΣΙΑΣ ΛΕΩΦ.','KIFISSIAS',29.51,
ST_GeomFromText('LINESTRING( 478895 4204061 , 479267 4204500 , 479641 4205387 ,
481368 4208164)', 2100));
```

```
INSERT INTO network.links VALUES (108,'ΠΑΤΗΣΙΩΝ','PATISSION',13.67,
ST_GeomFromText('LINESTRING( 476259 4204638 , 476560 4206110 , 476677 4208157 ,
476330 4208395 , 476072 4208960)', 2100));
```

```
INSERT INTO network.links VALUES (109,'ΠΟΣΕΙΔΩΝΟΣ ΛΕΩΦ.','POSEIDONOS',56.8,
ST_GeomFromText('LINESTRING( 472967 4198790 , 472674 4198735 , 471857 4199295 ,
470188 4199388 , 469418 4199867)', 2100));
```

```
INSERT INTO network.links VALUES (110,'ΒΟΥΛΙΑΓΜΕΝΗΣ ΛΕΩΦ.','VOULIAGMENIS',
51.72, ST_GeomFromText('LINESTRING( 476250 4202037 , 476437 4201916 , 476546
4200848 , 477056 4199017 , 477407 4198197)', 2100));
```

```
INSERT INTO network.links VALUES (111,'ΑΓ. ΚΩΝΣΤΑΝΤΙΝΟΥ','AG. KONSTANTINOY',
14.5, ST_GeomFromText('LINESTRING( 475995 4203798 , 475346 4203934 , 474764
4203806)', 2100));
```

```
INSERT INTO network.links VALUES (112,'ΙΕΡΑ ΟΔΟΣ','IERA ODOS',34.85,
ST_GeomFromText('LINESTRING( 474764 4203806, 474314 4204006 , 473014 4205166)',
2100));
```

```
INSERT INTO network.links VALUES (113,'ΠΕΙΡΑΙΩΣ','PIREOS',29.92,
ST_GeomFromText('LINESTRING( 475995 4203798 , 474926 4203181 , 473432 4202084,
472569 4201162 , 471193 4200596)', 2100));
```

```
INSERT INTO network.links VALUES (114,'ΠΕΙΡΑΙΩΣ','PIREOS',25.61,
ST_GeomFromText('LINESTRING( 471193 4200596 , 469418 4199867)', 2100));
```

```
INSERT INTO network.links VALUES (115,'ΚΗΦΙΣΙΟΥ ΛΕΩΦ.','KIFISSOU',45.82,
ST_GeomFromText('LINESTRING( 471193 4200596 , 471592 4201833 , 471313 4202478 ,
472323 4204393 , 473014 4205166)', 2100));
```

```
INSERT INTO network.links VALUES (116,'ΚΗΦΙΣΙΟΥ ΛΕΩΦ.','KIFISSOU',39.63,
ST_GeomFromText('LINESTRING( 473014 4205166 , 473991 4205758 , 475051 4206700 ,
475170 4207275 , 476072 4208960)', 2100));
```

```
INSERT INTO network.links VALUES (117, 'ΜΕΣΟΓΕΙΩΝ', 'MESOGEION', 41.76,
ST_GeomFromText('LINESTRING( 478895 4204061 , 479161 4203999 , 479518 4204319 ,
480144 4204811 , 481909 4206175 , 484065 4207065)', 2100));
```

Βήμα 12^ο. Εισαγωγή στοιχείων στον πίνακα των πολυγωνικών περιοχών.

```
INSERT INTO network.zones VALUES (501, 'ΑΘΗΝΑ', ST_GeomFromText('POLYGON((
473569 4202148 , 474778 4201575 , 474825 4201525 , 475240 4201082 , 474794
4200604 , 475644 4200030 , 478264 4201782 , 478582 4203294 , 481256 4203453 ,
481511 4204933 , 479426 4205618 , 477229 4206652 , 478216 4208546 , 476688
4208753 , 475176 4207305 , 475065 4206668 , 473091 4205172 , 472423 4203103 ,
473569 4202148 ))', 2100));
```

```
INSERT INTO network.zones VALUES (502, 'ΠΕΙΡΑΙΑΣ', ST_GeomFromText('POLYGON((
471076 4199236 , 471214 4200607 , 469779 4201255 , 467282 4201563 , 467133
4200192 , 465815 4199650 , 466591 4198927 , 467611 4199129 , 467930 4199841 ,
468377 4199863 , 468302 4199225 , 468005 4198683 , 467314 4198874 , 466602
4198534 , 467165 4197780 , 467867 4197376 , 468876 4197843 , 468759 4198534 ,
469004 4198821 , 469131 4198279 , 469790 4198417 , 470279 4198927 , 470353
4199310 , 471076 4199236 ))', 2100));
```

```
INSERT INTO network.zones VALUES (503, 'ΜΑΡΟΥΣΙ', ST_GeomFromText('POLYGON((
481370 4208169 , 481865 4209231 , 484259 4209302 , 485009 4210321 , 484882
4212389 , 482517 4213012 , 481696 4211100 , 481002 4211171 , 480166 4210180 ,
480662 4208820 , 481370 4208169 ))', 2100));
```

```
INSERT INTO network.zones VALUES (504, 'ΠΕΡΙΣΤΕΡΙ', ST_GeomFromText('POLYGON((
472852 4205012 , 473091 4205172 , 475065 4206668 , 475176 4207305 , 474036
4207759 , 472534 4208716 , 470864 4209475 , 470424 4208412 , 471228 4207198 ,
471000 4206393 , 472852 4205012 ))', 2100));
```

```
INSERT INTO network.zones VALUES (505, 'ΝΕΑ ΣΜΥΡΝΗ', ST_GeomFromText('POLYGON((
473437 4199247 , 474141 4198830 , 473993 4198470 , 474818 4197856 , 475453
4198957 , 476003 4199327 , 475644 4200030 , 474794 4200604 , 473437 4199247
))', 2100));
```

```
INSERT INTO network.zones VALUES (506, 'ΚΑΛΛΙΘΕΑ', ST_GeomFromText('POLYGON((
472678 4198742 , 472971 4198796 , 473437 4199247 , 474794 4200604 , 475240
4201082 , 474825 4201525 , 474491 4201444 , 474020 4201570 , 473001 4201386 ,
473043 4200819 , 472363 4200196 , 471939 4199262 , 472462 4198894 , 472533
4198844 , 472678 4198742 ))', 2100));
```

Γ. ΕΙΣΑΓΩΓΗ ΣΤΟΙΧΕΙΩΝ με μετατροπή δεδομένων από shapfile [2^{ος} τρόπος]

Πρόκειται για εναλλακτικό τρόπο μαζικής εισαγωγής δεδομένων που προέρχονται από αρχεία ESRI shapfiles. Συνιστάται ιδιαίτερα για πολύπλοκες γεωμετρίες με μεγάλο πλήθος κορυφών. Παρατηρήστε ότι οι γεωμετρίες μετατρέπονται και τελικά αναπαρίστανται σε **WKB format** συμβατό με PostGIS (βάσει των προδιαγραφών του OpenGIS Consortium)

Βήμα 13^ο. Ανοίγετε το **command prompt** και στην γραμμή εντολών πηγαίνετε στον κατάλογο με τα εκτελέσιμα της PostgreSQL, λ.χ.:

```
cd C:\Program Files\PostgreSQL\9.2\bin\
```

Έστω ότι έχετε το ESRI shapfile **parks.shp** στον κατάλογο **C:\Temp** και ότι τα αποτελέσματα της μετατροπής θα μπουν στον ίδιο κατάλογο με το όνομα **results.sql**, οπότε:

```
shp2pgsql -s 2100 -t 2D C:\Temp\parks.shp network.parks > C:\Temp\results.sql
```

Οι παράμετροι **-s** και **-t** δηλώνουν αντιστοίχως τον κωδικό του συστήματος γεωαναφοράς (ΕΓΣΑ87) και τις διαστάσεις των γεωμετριών (2). Αν ανοίξετε το προκύπτον αρχείο **results.sql** στο Notepad, τότε θα έχει την ακόλουθη μορφή:

```
BEGIN;
CREATE TABLE "network"."parks" (gid serial PRIMARY KEY,
"id" int4,
"name" varchar(30));
```

```

SELECT AddGeometryColumn('network','parks','the_geom','2100','MULTIPOLYGON',2);
INSERT INTO "network"."parks" ("id","name",the_geom) VALUES ('201','ΠΕΔΙΟΝ
ΑΡΕΩΣ','SRID=2100;010600000001000000010300000001000000070000000315E0A778D111D411
20FC0AB170A5041E01F606BF0121D4165724CA0840A5041528DC4E3BA171D41ECFFEECE750A5041
8C12F1D67E251D41983DBBF4A40A50413AA39398402A1D41D378DFF3830A50412C1E71FCE7271D4
1C5F1C253C9095041315E0A778D111D41120FC0AB170A5041');
INSERT INTO "network"."parks" ("id","name",the_geom) VALUES ('202','ΕΘΝΙΚΟΣ
ΚΗΠΟΣ','SRID=2100;01060000000100000001030000000100000007000000E384BCDE77171D414
2784E02670850418BCA643EAA221D41689DD34C4E085041BE0121B0E41F1D41B9E4D959DC075041
F7D3229D821A1D41B6859987A2075041F60E2BA05C131D41933AAABED50750411D24C1D4DA161D4
1695E4FC61A085041E384BCDE77171D4142784E0267085041');
INSERT INTO "network"."parks" ("id","name",the_geom) VALUES
('203','ΙΠΠΟΔΡΟΜΟΣ','SRID=2100;01060000000100000001030000000100000007000000B32A
FB39B4E51C41A2426B1ADA04504189E71C1118DE1C419546DDA061045041BC3BCE8683D91C41367
62A0454045041E8E678001FD61C41146D080D79045041A81F120075DD1C41C307C31A1C055041B8
44CD209BE11C411FDD794A1F055041B32AFB39B4E51C41A2426B1ADA045041');
INSERT INTO "network"."parks" ("id","name",the_geom) VALUES ('204','BOTANIKΟΣ
ΚΗΠΟΣ','SRID=2100;010600000001000000010300000001000000060000000B5BB80E64CE31C415
47E23775A0A504165FC9AD992F01C41581EB52A9E095041C1B8B80027EC1C41BDCEB5D140095041
BAC181BFA2E61C4154091F3A6F095041A1C2848257DC1C41636602EFB8095041B5BB80E64CE31C4
1547E23775A0A5041');
END;

```

Παρατηρήστε ότι οι εντολές πρώτα δημιουργούν την δομή κι έπειτα εισάγουν τα περιεχόμενα του πίνακα, οπότε μπορείτε να τις εκτελέσετε όλες μαζί στην PostgreSQL.

Δ. ΔΙΑΤΥΠΩΣΗ ΕΡΩΤΗΜΑΤΩΝ ΜΕ ΧΩΡΙΚΟΥΣ ΤΕΛΕΣΤΕΣ

Βήμα 14°. Ερώτημα σημείου (*point query*):

--Βρες τις ζώνες που περιλαμβάνουν μια συγκεκριμένη θέση ενδιαφέροντος

```

SELECT zone_id, zone_name
FROM network.zones
WHERE ST_WITHIN( ST_GeomFromText('POINT(482000 4210000)',2100), sector);

```

Βήμα 15°. Ερώτημα απόστασης από σημείο (*distance query*):

--Βρες τα σημεία ενδιαφέροντος βρίσκονται εντός απόστασης 200 μέτρων από συγκεκριμένη θέση

```

SELECT location_id,
       ST_DISTANCE(pos, ST_GeomFromText('POINT(476500 4202800)',2100)) AS dist
FROM network.locations
WHERE ST_DISTANCE(pos, ST_GeomFromText('POINT(476500 4202800)',2100))<=200;

```

Βήμα 16°. Ερώτημα περιοχής (*range query*):

--Βρες ποια σημεία τέμνουν συγκεκριμένη πολυγωνική περιοχή ενδιαφέροντος:

```

SELECT location_id, location_name
FROM network.locations
WHERE ST_INTERSECTS(pos, ST_GeomFromText('POLYGON( (475000 4202000, 480000
4202000, 480000 4206000 , 475000 4206000 , 475000 4202000 )',2100) );

```

Βήμα 17°. Ερώτημα εγκλεισμού (*containment query*):

--Βρες ποια σημεία ενδιαφέροντος βρίσκονται εντός του δήμου Αθηναίων:

```

SELECT location_id, location_name
FROM network.locations
WHERE ST_WITHIN(pos, (SELECT sector FROM network.zones
                       WHERE zone_name='ΑΘΗΝΑ' ) );

```

Βήμα 18°. Ερώτημα περιφραξης (*enclosure query*):

--Βρες ποια ζώνη περικλείει το κτίριο ΦΙΞ

```
SELECT zone_name
FROM network.zones
WHERE ST_CONTAINS( sector, (SELECT pos FROM network.locations
                             WHERE location_name='ΦΙΞ'));
```

Βήμα 19°. Ερωτήματα χωρικής σύνδεσης

(α) με εγκλεισμό (*spatial containment join*)

--Βρες τα σημεία ενδιαφέροντος και τις ζώνες όπου ανήκουν

```
SELECT location_id, location_name, zone_id, zone_name
FROM network.locations, network.zones
WHERE ST_WITHIN(pos, sector);
```

(β) με περιφραξη (*spatial enclosure join*)

--Βρες τα σημεία ενδιαφέροντος και τις ζώνες που τα περικλείουν

```
SELECT location_id, location_name, zone_id, zone_name
FROM network.locations, network.zones
WHERE ST_CONTAINS(sector, pos);
```

(γ) με τομή (*spatial intersection join*)

--Βρες από ποιες ζώνες περνά κάθε δρόμος

```
SELECT roadname_gr, zone_name
FROM network.links, network.zones
WHERE ST_INTERSECTS(segment, sector);
```

(δ) με επικάλυψη (*spatial overlap join*)

--Βρες σε ποιες ζώνες υπάρχει επικάλυψη με πάρκα

```
SELECT name, zone_name
FROM network.parks, network.zones
WHERE ST_OVERLAPS(the_geom, sector);
```

Βήμα 20°. Ερώτημα χωρικής (αυτο-)σύνδεσης:

(α) με βάση την απόσταση (*distance self-join*):

--Βρες όλα τα ζεύγη σημείων που απέχουν μεταξύ τους απόσταση μικρότερη από 2000 μέτρα

```
SELECT L1.location_name AS loc1, L2.location_name AS loc2,
       ST_DISTANCE(L1.pos, L2.pos) AS dist
FROM network.locations L1, network.locations L2
WHERE ST_DISTANCE(L1.pos, L2.pos) <= 2000
AND L1.location_id < L2.location_id;
```

(β) με βάση την γειννίαση (*adjacency self-join*):

--Βρες όλα τα ζεύγη ζωνών που γειννιάζουν

```
SELECT Z1.zone_name, Z2.zone_name
FROM network.zones Z1, network.zones Z2
WHERE ST_TOUCHES(Z1.sector, Z2.sector)
AND Z1.zone_id < Z2.zone_id;
```

Βήμα 21°. Γεωμετρικές συναρτήσεις.

(α) Εμβαδόν πολυγωνικής περιοχής

```
SELECT zone_name, ST_area(sector) AS calc_area
FROM network.zones
ORDER BY calc_area DESC;
```


(β) Μήκος γραμμικών αξόνων

```
SELECT roadname_gr, SUM(ST_length(segment)) AS calc_length
FROM network.links
GROUP by roadname_gr
ORDER BY calc_length DESC;
```

(γ) Πλήθος κορυφών ανά πολύγωνο

```
SELECT zone_id, zone_name, ST_npoints(sector) AS vertex_count
FROM network.zones;
```

(δ) Ελάχιστα περιβάλλοντα παραλληλόγραμμα για τις περιοχές

--Μετατροπή σε μορφή κειμένου

```
SELECT zone_name, ST_AsText(ST_envelope(sector)) AS MBBText
FROM network.zones;
```

--Εναλλακτικά, μετατροπή σε GML, χρησιμοποιώντας 10 ψηφία για τις συντεταγμένες

```
SELECT zone_name, ST_AsGml(ST_envelope(sector), 10) AS MBBGml
FROM network.zones;
```

--Εναλλακτικά, μετατροπή σε KML, χρησιμοποιώντας 10 ψηφία για τις συντεταγμένες

```
SELECT zone_name, ST_AsKml(ST_envelope(sector), 10) AS MBBGml
FROM network.zones;
```

Βήμα 22°. Εύρεση γεωμετριών εντός αποστάσεως.

--Όλες οι σημειακές θέσεις που βρίσκονται εντός Ευκλείδειας αποστάσεως 1000 μέτρων από συγκεκριμένες συντεταγμένες:

```
SELECT location_name, ST_AsText(pos) AS loc
FROM network.locations
WHERE ST_DWITHIN(pos, (ST_GeometryFromText('SRID=2100;POINT(476000 4204000)'),
1000));
```

Βήμα 23°. Μετασχηματισμοί σε άλλο σύστημα γεωαναφοράς.

(α) Από το ΕΓΣΑ87 (όπου είναι ορισμένα τα στοιχεία του πίνακα) στο WGS84 (κωδικός: 4326):

```
SELECT ST_AsText(pos), ST_AsText(ST_Transform(pos,4326)) AS loc_wgs84
FROM network.locations;
```

(β) Τοπολογικός έλεγχος μεταξύ γεωμετριών (όπως στο βήμα 19β), αλλά τα στοιχεία είναι σε διαφορετικά συστήματα γεωαναφοράς. Παρατηρήστε ότι ο πίνακας locwgs είναι σε WGS84, οπότε οι γεωμετρίες του πρέπει να μετασχηματιστούν ειδικά για τον έλεγχο:

```
SELECT location_id, location_name, zone_id, zone_name
FROM locwgs, network.zones
WHERE ST_CONTAINS(sector, ST_SetSRID(ST_Transform(pos,2100), 2100));
```

Βήμα 24°. Διαγραφή όλων των πινάκων (**μαζί με τις εγγραφές τους!**) από την βάση δεδομένων:

```
DROP TABLE locwgs CASCADE;
DROP TABLE network.locations CASCADE;
DROP TABLE network.links CASCADE;
DROP TABLE network.zones CASCADE;
DROP TABLE network.parks CASCADE;
DROP SCHEMA network;
```

Παραπομπές

- PostGIS documentation & software: <http://postgis.net/>
- OGC Simple Features Specification for SQL http://portal.opengeospatial.org/files/?artifact_id=829
- Quantum GIS (QGIS): <http://www.qgis.org/>
- User-friendly Desktop Internet GIS: <http://udig.refractions.net/>
- Display PostGIS layers from QGIS: <http://trac.osgeo.org/postgis/wiki/UsersWikiQGIS>