



ΜΟΝΑΔΕΣ ΑΡΙΣΤΕΙΑΣ
ΑΝΟΙΧΤΟΥ ΛΟΓΙΣΜΙΚΟΥ



ΜΟΝΑΔΕΣ ΑΡΙΣΤΕΙΑΣ ΑΝΟΙΧΤΟΥ ΛΟΓΙΣΜΙΚΟΥ

Μεταφορές - Ναυτιλία

3^η Σειρά Εκπαίδευσης

3^ο σεμινάριο

18 Μαΐου 2015



**ΙΝΣΤΙΤΟΥΤΟ
ΤΕΧΝΟΛΟΓΙΑΣ ΥΠΟΛΟΓΙΣΤΩΝ ΚΑΙ ΕΚΔΟΣΕΩΝ
ΙΜΙΣ**

Αντικειμενοστραφείς Σχεσιακές Βάσεις Δεδομένων

Σχεσιακές

- Τα δεδομένα οργανώνονται σε πίνακες
 - Κάθε γραμμή αντιπροσωπεύει ένα στιγμιότυπο μιας οντότητας
 - Π.χ. έναν συγκεκριμένο μαθητή
 - Κάθε στήλη αντιπροσωπεύει ένα χαρακτηριστικό μιας οντότητας
 - Π.χ. τηλέφωνο επικοινωνίας μαθητή
- Οι στήλες πινάκων περιγράφονται από μεταδεδομένα
- Οι σχέσεις μεταξύ οντοτήτων περιγράφονται από τιμές που αποθηκεύονται σε στήλες στους αντίστοιχους πίνακες (κλειδιά)
- Τα δεδομένα είναι προσβάσιμα μέσω κατάλληλης γλώσσας ερωτημάτων Standard Query Language (SQL)
 - `SELECT student_name, student_telephone WHERE class = 'B' δημοτικού`



Αντικειμενοστραφείς Σχεσιακές Βάσεις Δεδομένων

Σχεσιακές

- Βάσεις Δεδομένων (data bases)
- Σχήματα (schemas)
- Πίνακες (tables)
- Όψεις (views)
- Δείκτες (indexes)
- Περιορισμοί (constraints)
- Χρήστες (users)
- Δικαιώματα (privileges)



Αντικειμενοστραφείς Σχεσιακές Βάσεις Δεδομένων

Αντικειμενοστραφείς

Υλοποιούν ιδέες του αντικειμενοστραφούς προγραμματισμού:

- Αφηρημένοι τύποι δεδομένων
 - Ορισμός κλάσεων, επεκτάσεις για τη δημιουργία σύνθετων τύπων
- Ενθυλάκωση
 - Απόκρυψη της δομής και της λειτουργικότητας μέσα σε αντικείμενα
- Κληρονομικότητα
 - Διαμοιρασμός δεδομένων σε ιεραρχικό επίπεδο
- Πολυμορφισμός
 - Επαναχρησιμοποίηση τελεστών για διαφορετικούς τύπους δεδομένων
 - Υπερφόρτωση τελεστών



Γεωχωρικές Βάσεις Δεδομένων

Πλεονεκτήματα

- Διαχείριση μεγάλου όγκου δεδομένων
- Εύκολη συσχέτιση με περιγραφικά δεδομένα
- Μηχανισμοί ασφάλειας και τήρησης των στοιχείων
- Κεντρική υποστήριξη πολλαπλών εφαρμογών
- Ομοιογενής τρόπος αποθήκευσης και προσπέλασης στοιχείων
- Βελτιωμένες επιδόσεις στην επεξεργασία των στοιχείων

Προκλήσεις

- Εσωτερική αναπαράσταση γεωμετρίας
- Διαχείριση χρονικής εξέλιξης των στοιχείων
- Σταδιακή συσσώρευση όγκου δεδομένων (~ZBs)



PostgreSQL

Χαρακτηριστικά

- Ισχυρή, αξιόπιστη, δημοφιλής σχεσιακή βάση δεδομένων (ORDBMS) ανοικτού κώδικα
- Υποστηρίζει τα σημαντικότερα ΛΣ: Linux, UNIX και Windows
- Είναι σύμμορφη με το πρότυπο ANSI-SQL:2008.
- Υποστηρίζει πλήρως τις ιδιότητες **ACID** που εγγυώνται την αξιοπιστία των συναλλαγών:
 - **Ατομικότητα** (Atomicity): «Όλα ή τίποτα». Αν αποτύχει ένα στάδιο μιας συναλλαγής αποτυγχάνει όλη η συναλλαγή
 - **Συνέπεια** (Consistency): Κάθε συναλλαγή μεταφέρει το σύστημα από μια έγκυρη κατάσταση σε μια άλλη.
 - **Απομόνωση** (Isolation): Η ταυτόχρονη εκτέλεση διαφόρων σταδίων μιας συναλλαγής γίνονται με τέτοιο τρόπο ώστε το αποτέλεσμα να είναι ισοδύναμο σαν να εκτελούνταν σειριακά.
 - **Αντοχή** (Durability): Αν μια συναλλαγή υποβληθεί δεν αναιρείται από πτώση ή σφάλμα του συστήματος.



PostgreSQL

Χαρακτηριστικά

- Υποστηρίζει πλήρως χαρακτηριστικά όπως foreign keys, joins, views, triggers, και stored procedures
- Υποστηρίζει τους περισσότερους τύπους δεδομένων του πρότυπου ANSI-SQL:2008: INTEGER, NUMERIC, BOOLEAN, CHAR, VARCHAR, DATE, INTERVAL και TIMESTAMP
- Παρέχει δεικτοδότηση μέσω του **GiST** (Generalized Search Tree) το οποίο χρησιμοποιεί μια μεγάλη γκάμα διαφορετικών αλγόριθμων ταξινόμησης και αναζήτησης όπως τα B-tree, B+-tree, R-tree, partial sum trees, ranked B+-trees κλπ.
- Μπορεί να αποθηκεύσει σε μορφή μεγάλου δυαδικού αντικειμένου (BLOB) εικόνες, ήχο, βίντεο.
- Διαθέτει εγγενής προγραμματιστικές διεπαφές για γλώσσες όπως for C/C++, Java, .Net, Perl, Python, Ruby κλπ.



PostgreSQL

Εγγενή Γεωχωρικά Χαρακτηριστικά

- Ειδικοί τύποι δεδομένων για την αποθήκευση απλών γεωμετρικών οντοτήτων
- Μηχανισμούς δεικτοδότησης χωρικών δεδομένων (spatial indexing)
- Γεωμετρικούς τελεστές (operators) και συναρτήσεις (functions) για την διατύπωση χωρικών ή συνδυασμένων ερωτημάτων (queries)

Περιορισμοί:

- επιτρέπει μόνο 2-διάστατες γεωμετρικές οντότητες
- Δεν έχει δυνατότητα προσδιορισμού συστήματος αναφοράς για γεωγραφικά δεδομένα



Χρήσιμες εντολές

- CREATE DATABASE meteorological_db;
- DROP DATABASE meteorological_db;
- CREATE TABLE weather (city varchar(80),
temp_lo int, -- low temperature
temp_hi int, -- high temperature
prcp real, -- precipitation
date date
);
- DROP TABLE weather;
- INSERT INTO weather VALUES('Athens',12,19);
- DELETE FROM weather WHERE city = 'Athens';
- QUERIES
 - SELECT * FROM weather;
 - SELECT city, (temp_hi-temp_low)/2 as temp_avg FROM weather;
 - SELECT max(temp_hi) FROM weather;



PostGIS

Αποτελεί επέκταση της PostgreSQL

- Δίνει υποστήριξη για γεωχωρικά δεδομένα
- Αξιοποιεί τον PostgreSQL backend server
- Είναι λογισμικό ανοικτού κώδικα (open source)

Γεωμετρική αναπαράσταση

- Δεν ακολουθεί την τυπική μορφή της PostgreSQL
- Υπακούει στις προδιαγραφές του OGC
- Υποστηρίζονται χωρικές οντότητες 2-d, 3-d, 4-d
- Υποστηρίζονται συστήματα γραμμικής αναφοράς



Γεωγραφικοί Τύποι

- **GEOMETRY**: αφηρημένη υπερκλάση όλων των παρακάτω κλάσεων
- **POINT**: Συντεταγμένες ενός σημείου όχι κατ' ανάγκη δυο διαστάσεων
- **LINESTRING**: Ένα σύνολο δυο ή περισσότερων σετ συντεταγμένων που αντιστοιχούν σε μια γραμμή. Η διάταξη παίζει ρόλο.
- **LINEARRING**: Ένα σύνολο τριών ή περισσότερων σετ συντεταγμένων όπου το αρχικό και τελευταίο σημείο συμπίπτουν κλείνοντας τη γραμμή και σχηματίζοντας δακτύλιο.
- **POLYGON**: Ένα σύνολο ενός ή περισσότερων δακτυλίων που σχηματίζουν ένα σύνθετο πολύγωνο. Ένας δακτύλιος παίζει το ρόλο της εξωτερικής περιμέτρου ενώ τα υπόλοιπα σημειώνουν εξαιρέσεις-τρύπες.
- **MULTIPOINT**: Συλλογή σημείων
- **MULTILINESTRING**: Συλλογή γραμμών
- **MULTIPOLYGON**: Συλλογή πολυγώνων
- **GEOMETRYCOLLECTION**: Συλλογή διαφορετικών γεωμετριών



Γεωγραφικοί Τύποι - WKT

- **POINT:** POINT (30 10)
- **LINESTRING:** LINESTRING (30 10, 10 30, 40 40)
- **POLYGON:** POLYGON ((35 10, 45 45, 15 40, 10 20, 35 10), (20 30, 35 35, 30 20, 20 30))
- **MULTIPOINT:** MULTIPOINT (10 40, 40 30, 20 20, 30 10)
- **MULTILINESTRING:** MULTILINESTRING ((0 0 0, 1 1 0, 1 2 1), (2 3 1, 3 2 1, 5 4 1))
- **MULTIPOLYGON:** MULTIPOLYGON (((0 0 0, 4 0 0, 4 4 0, 0 4 0, 0 0 0), (1 1 0, 2 1 0, 2 2 0, 1 2 0, 1 1 0)), ((-1 -1 0, -1 -2 0, -2 -2 0, -2 -1 0, -1 -1 0)))
- **GEOMETRYCOLLECTION:**
GEOMETRYCOLLECTION (POINT (4 6),
LINESTRING (4 6, 7 10))



Γεωγραφικά συστήματα αναφοράς

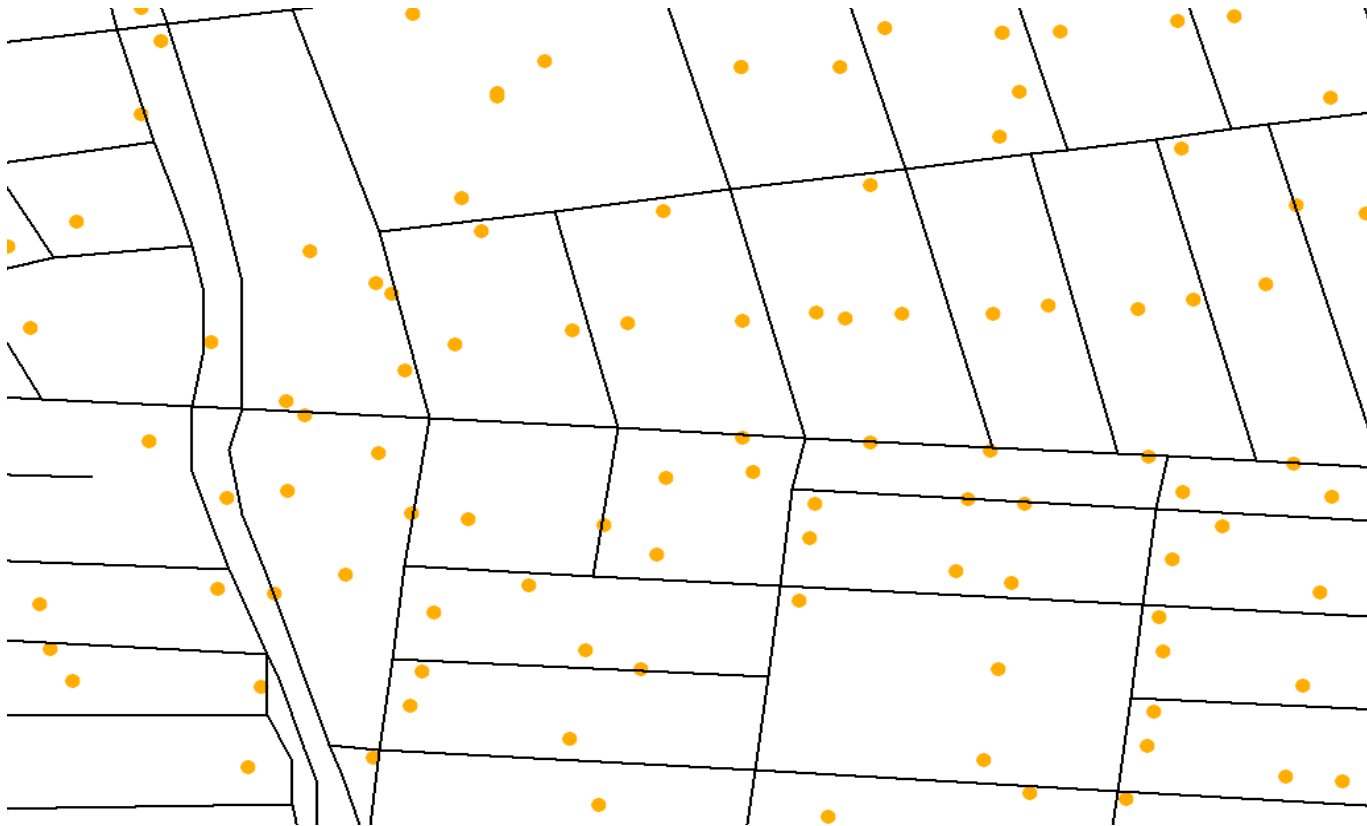
Υποστηρίζει έναν πολύ μεγάλο αριθμό (3910) συστημάτων γεωγραφικής αναφοράς περιλαμβάνοντας τόσο σφαιρικά όσο και καρτεσιανά συστήματα.

Μεταξύ αυτών τα:

- **WGS 84:** SRID 4326
- **ΕΓΣΑ 87:** SRID 2100
- **Google Spherical Mercator:** SRID 900913
- **UTM:** Variable SRID, location based



Παραδείγματα Δεδομένων



Δρόμοι (γραμμές) και Ταχυδρομικά Κουτιά
(σημεία)

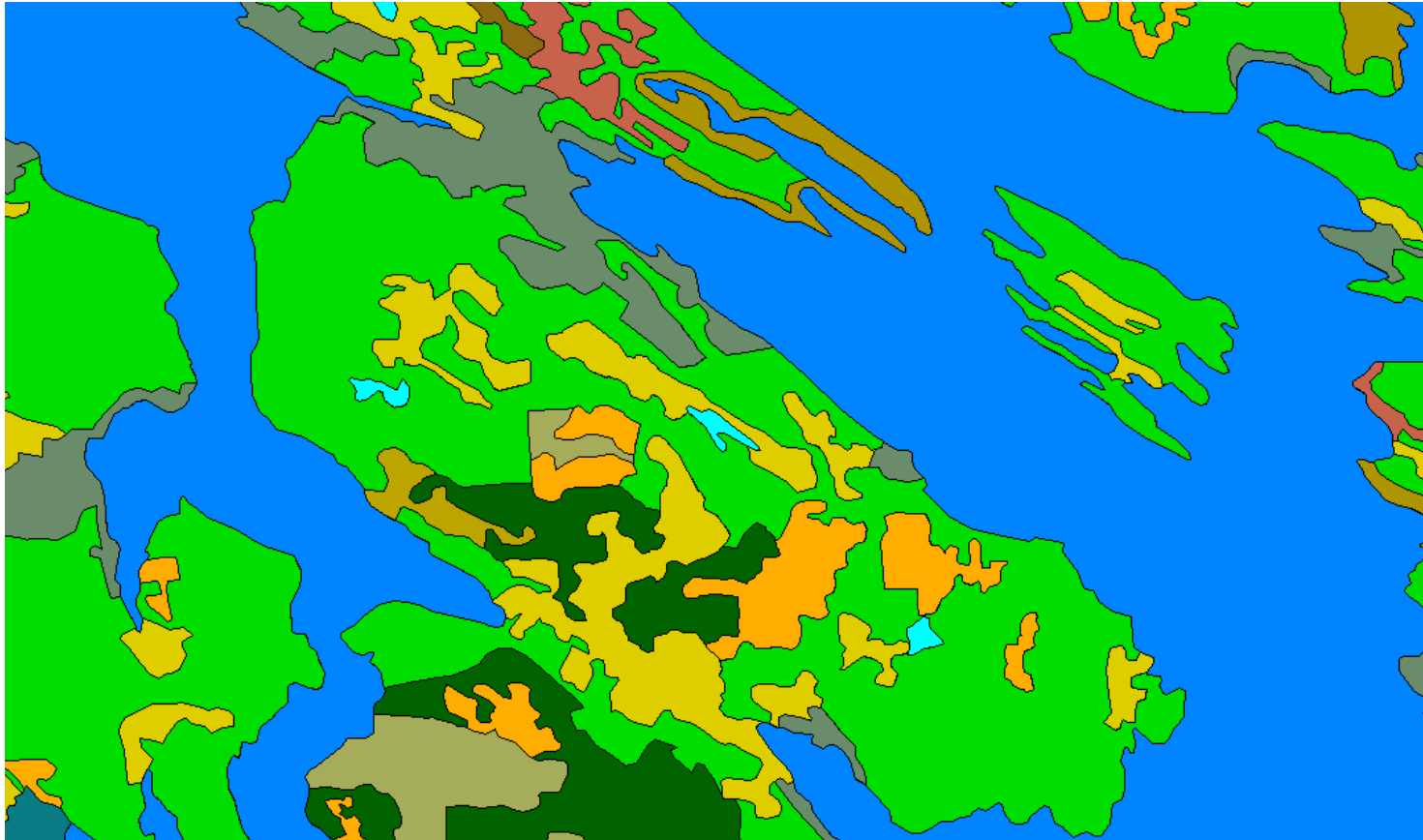


ΜΟΝΑΔΕΣ ΑΡΙΣΤΕΙΑΣ
ΑΝΟΙΧΤΟΥ ΛΟΓΙΣΜΙΚΟΥ



ΙΝΣΤΙΤΟΥΤΟ
ΤΕΧΝΟΛΟΓΙΑΣ

Παραδείγματα Δεδομένων



Καλύψεις γης (πολύγωνα)

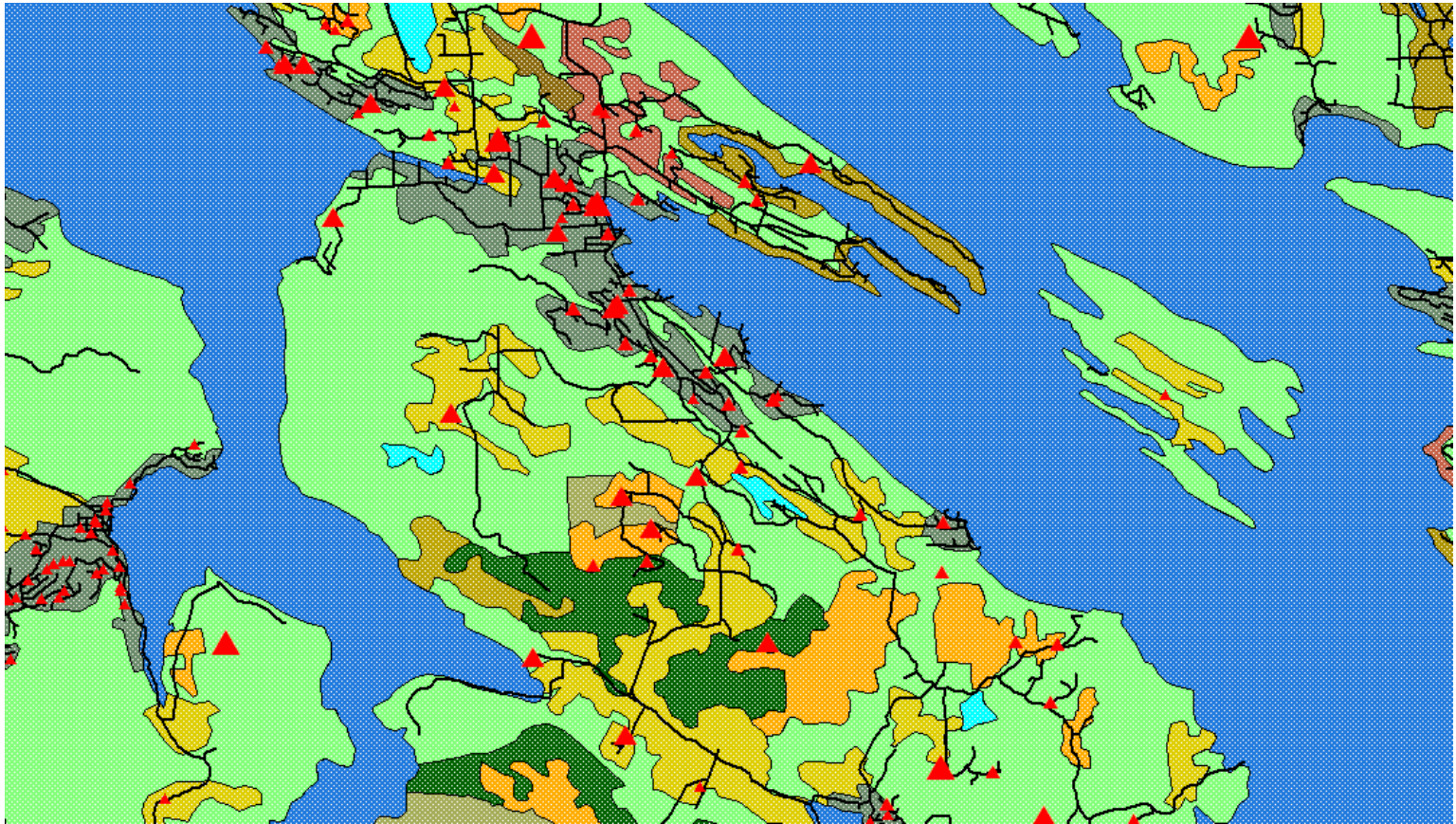


ΜΟΝΑΔΕΣ ΑΡΙΣΤΕΙΑΣ
ΑΝΟΙΧΤΟΥ ΛΟΓΙΣΜΙΚΟΥ



ΙΙΣΥ ΙΜΙΣ

Παραδείγματα Δεδομένων



Συνδυασμοί



ΜΟΝΑΔΕΣ ΑΡΙΣΤΕΙΑΣ
ΑΝΟΙΧΤΟΥ ΛΟΓΙΣΜΙΚΟΥ



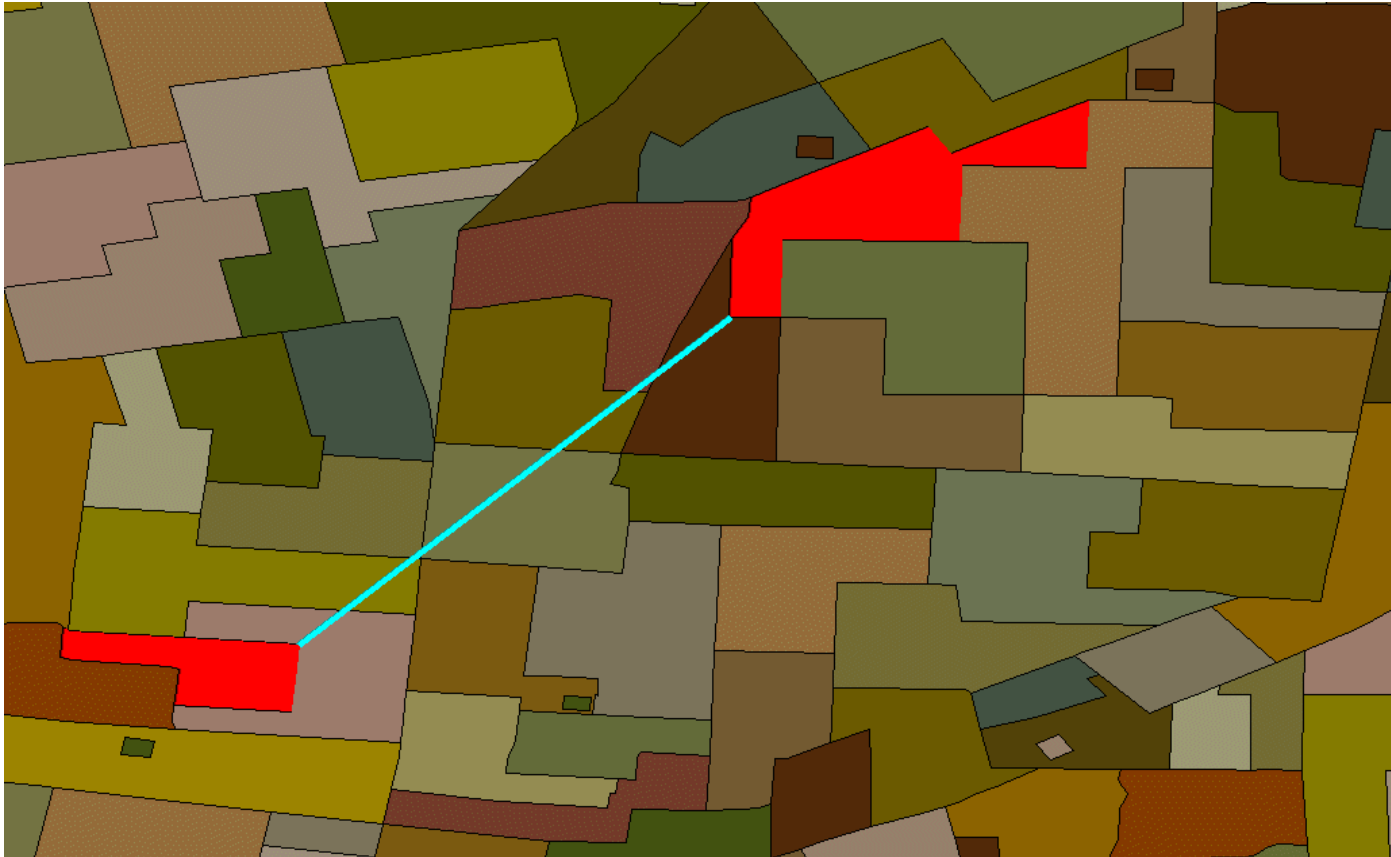
ΙΙΣΥ ΙΜΙΣ

Γεωχωρικές σχέσεις

- Τοπολογικές σχέσεις: γειτνίαση, περίληψη
- Σχέσεις κατεύθυνσης: «από πάνω», «από κάτω»...
- Μετρικές σχέσεις: απόσταση



Παραδείγματα Συσχέτισης



Απόσταση

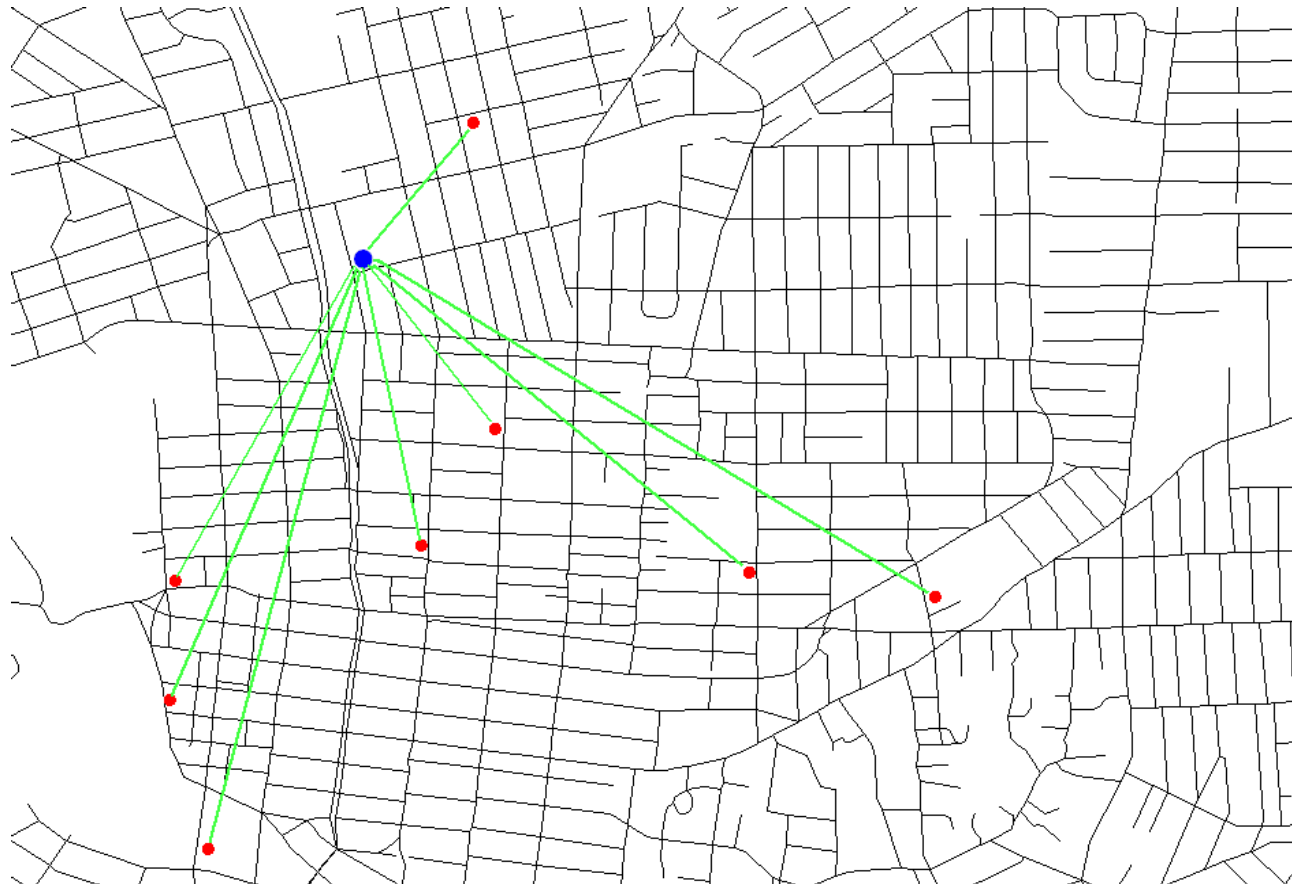


ΜΟΝΑΔΕΣ ΑΡΙΣΤΕΙΑΣ
ΑΝΟΙΧΤΟΥ ΛΟΓΙΣΜΙΚΟΥ



ΙΝΣΤΙΤΟΥΤΟ
ΤΕΧΝΟΛΟΓΙΑΣ

Παραδείγματα Συσχέτισης



Απόσταση

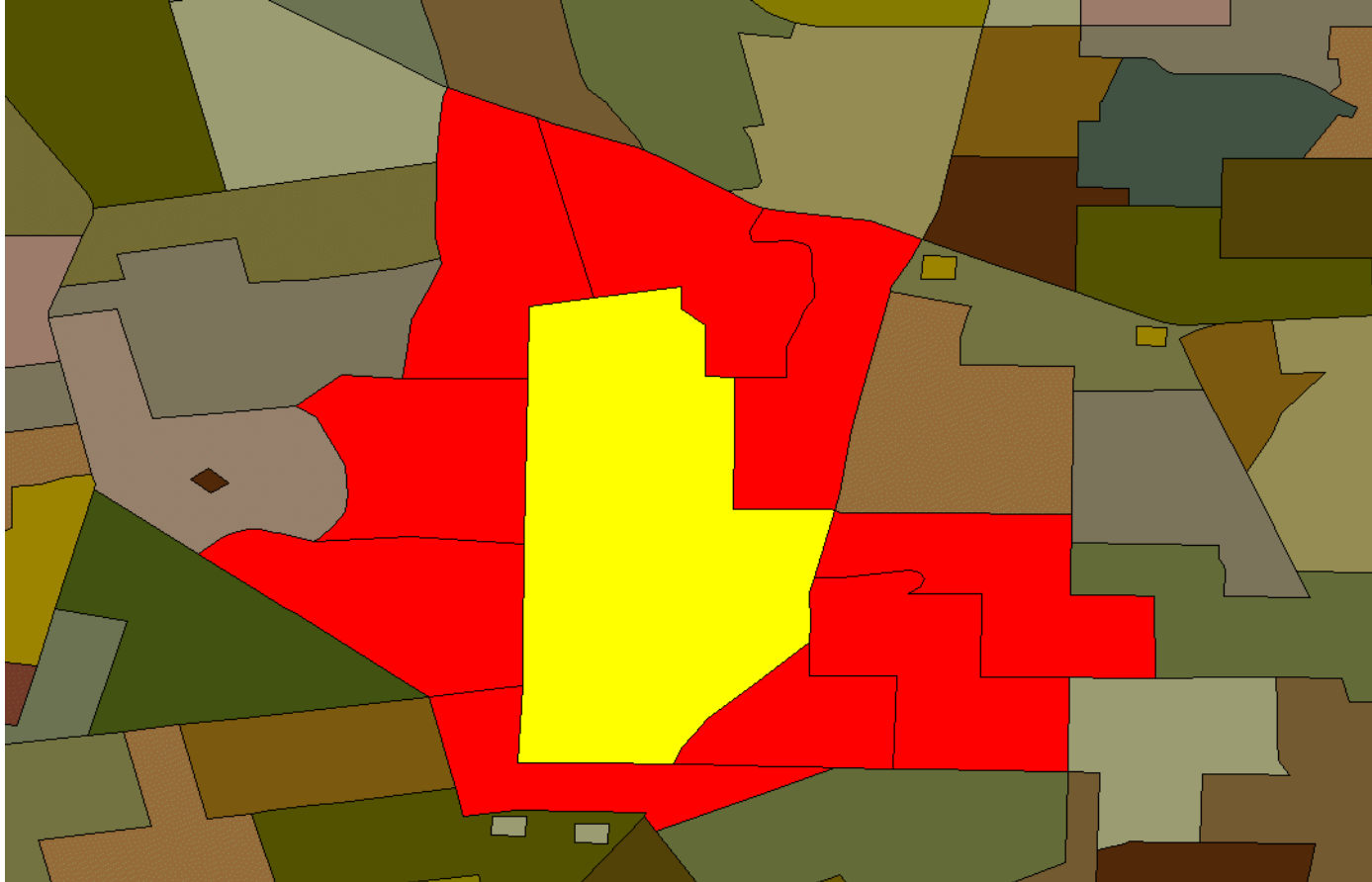


ΜΟΝΑΔΕΣ ΑΡΙΣΤΕΙΑΣ
ΑΝΟΙΧΤΟΥ ΛΟΓΙΣΜΙΚΟΥ



ΙΝΣΤΙΤΟΥΤΟ ΜΕΣΕΩΣ ΕΚΠΑΙΔΕΥΣΗΣ

Παραδείγματα Συσχέτισης



Γειτνίαση οικοπέδων

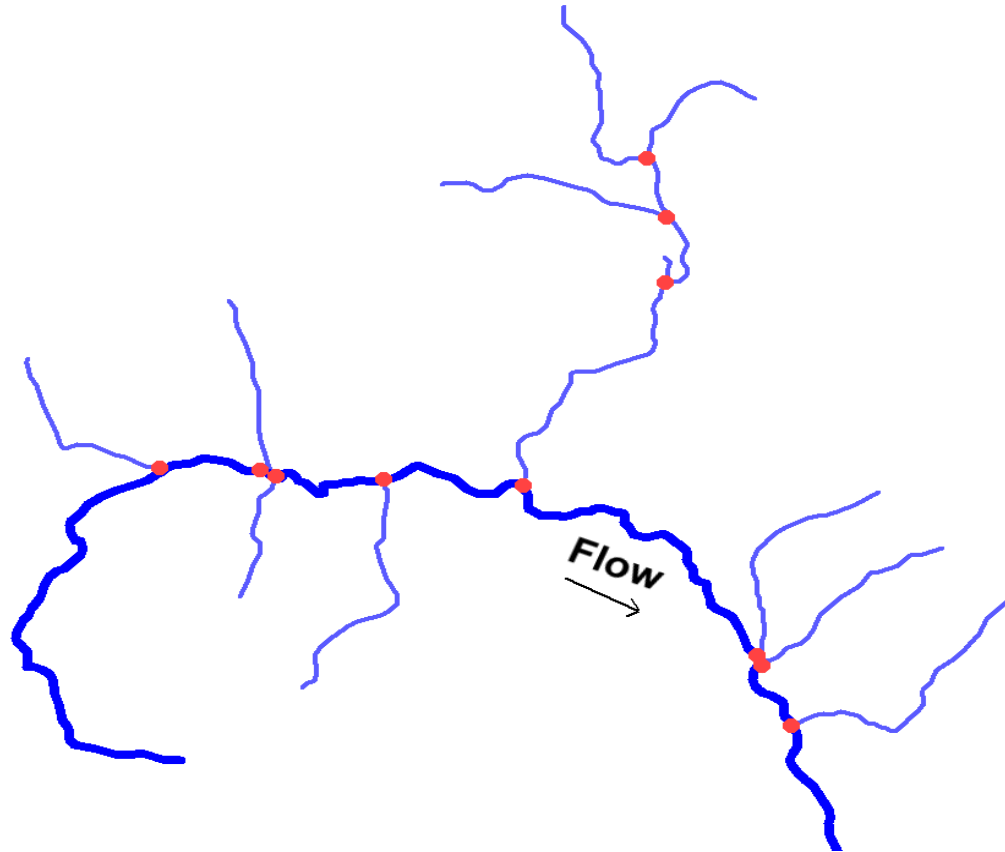


ΜΟΝΑΔΕΣ ΑΡΙΣΤΕΙΑΣ
ΑΝΟΙΧΤΟΥ ΛΟΓΙΣΜΙΚΟΥ



ΙΝΣΤΙΤΟΥΤΟ ΜΕΣΕΩΣ ΕΚΠΑΙΔΕΥΣΗΣ

Παραδείγματα Συσχέτισης



Συνδεσιμότητα (υδρογραφικό δίκτυο)



ΜΟΝΑΔΕΣ ΑΡΙΣΤΕΙΑΣ
ΑΝΟΙΧΤΟΥ ΛΟΓΙΣΜΙΚΟΥ



ΙΤΣΥ ΙΜΙΣ

Παραδείγματα Συσχέτισης



Περίληψη (ποτάμια σε λεκάνες απορροής)



ΜΟΝΑΔΕΣ ΑΡΙΣΤΕΙΑΣ
ΑΝΟΙΧΤΟΥ ΛΟΓΙΣΜΙΚΟΥ



ΙΤΑΥ ΙΜΙΣ

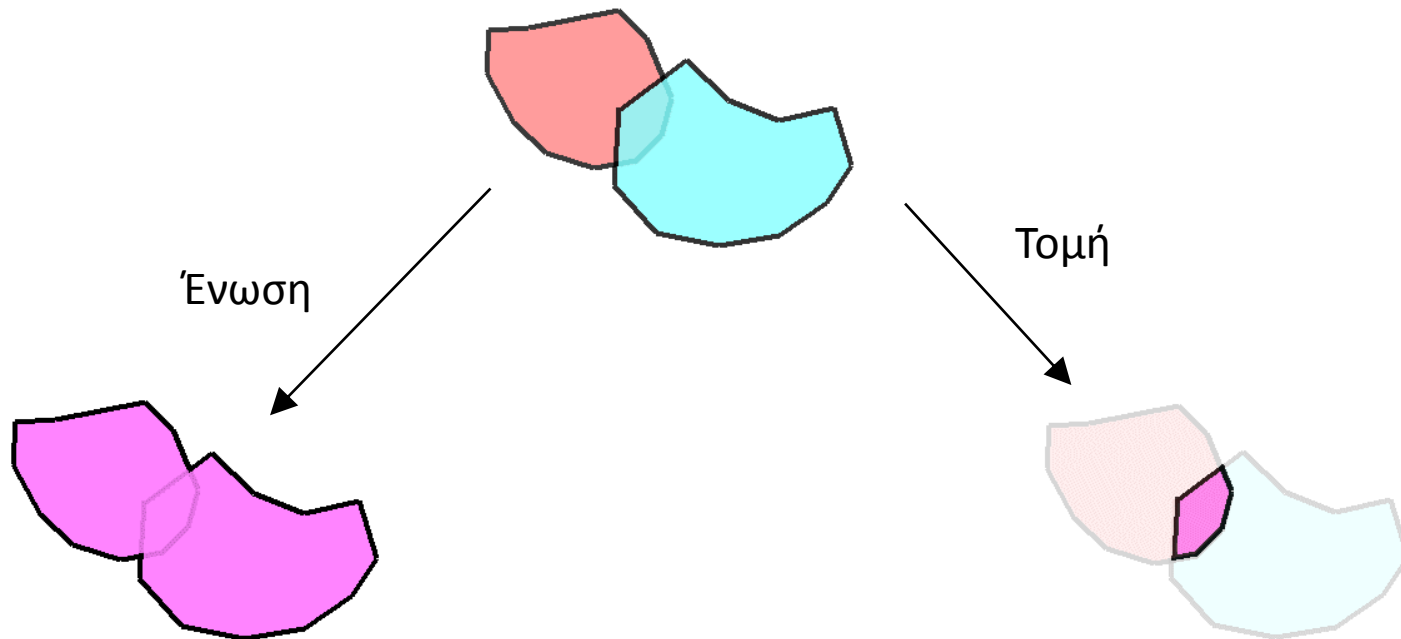
Χωρική Συσχέτιση

Relationship	Inputs	Output
inside	geo, regions	bool
intersect, meets	line/pol1, line/pol2	bool
adjacent, encloses	regions, regions	bool
intersection	lines, lines	points
intersection	regions, regions	regions
plus, minus	geo, geo	geo
dist	geo1, geo2	real
perimeter, area	regions	real



Χωρικές πράξεις

Εμβαδό, Μήκος, Τομή, Ένωση, Buffer

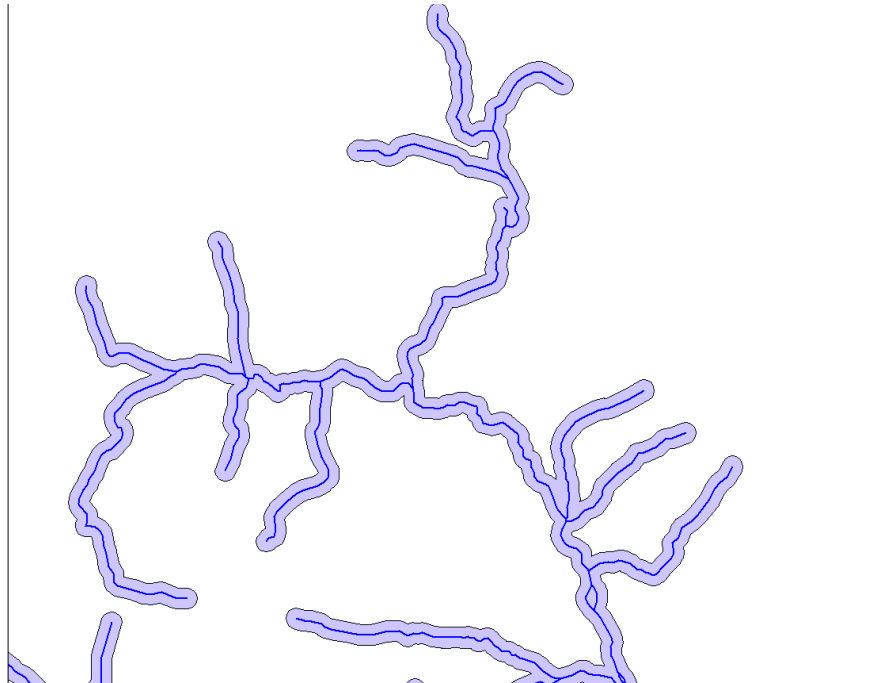


ΜΟΝΑΔΕΣ ΑΡΙΣΤΕΙΑΣ
ΑΝΟΙΧΤΟΥ ΛΟΓΙΣΜΙΚΟΥ



ΙΝΣΤΙΤΟΥΤΟ ΜΕΣΕΩΣ ΕΚΠΑΙΔΕΥΣΗΣ

Χωρικές πράξεις



Υδρογραφικό δίκτυο με buffer



ΜΟΝΑΔΕΣ ΑΡΙΣΤΕΙΑΣ
ΑΝΟΙΧΤΟΥ ΛΟΓΙΣΜΙΚΟΥ



ΙΝΣΤΙΤΟΥΤΟ
IMIS

Συναρτήσεις

Συναρτήσεις Κατασκευής

- **ST_GeomFromText:** Κατασκευάζει γεωμετρίες από κείμενο (well known text)
 - `SELECT ST_GeomFromText('LINESTRING(-71.160 42.259,-71.1637 42.253,-71.1614 42.252)',4269);`
- **ST_GeomFromWKB:** Κατασκευάζει γεωμετρίες από κείμενο (well known text)
 - `SELECT ST_GeomFromWKB(E'\\001\\002\\000\\000\\000\\002\\000\\000\\000\\037\\205\\353Q\\270~\\\\\\\\300\\323Mb\\020X\\231C@\\020X9\\264\\310~\\\\\\\\300)\\\\\\\\217\\302\\365\\230C@',4326);`
- **ST_MakePoint:** Κατασκευή ειδικής γεωμετρίας (π.χ. σημείο)
 - `SELECT ST_MakePoint(-71.10, 42.31);`
- **ST_GeomFromKML:** Κατασκευή γεωμετρίας από KML κείμενο
 - `SELECT ST_GeomFromKML(' <LineString> <coordinates>-71.1663,42.2614 -71.1667,42.2616</coordinates> </LineString>');`



Συναρτήσεις

Συναρτήσεις Ερώτησης της Γεωμετρίας

- **ST_Boundary:** Επιστροφή περιμέτρου της γεωμετρίας

```
SELECT ST_AsText(ST_Boundary(ST_GeomFromText('POLYGON((1 1,0 0, -1 1, 1 1))')));
```

- **ST_Envelope:** Επιστροφή bounding box της γεωμετρίας

```
SELECT ST_AsText(ST_Envelope('LINESTRING(0 0, 1 3)::geometry));
```

- **ST_GeometryType:** Επιστροφή του τύπου της γεωμετρίας

```
SELECT ST_GeometryType(ST_GeomFromText('POINT (77.29 29.07)'));
```

- **ST_Ndims:** Επιστροφή των διαστάσεων της γεωμετρίας

```
SELECT ST_NDims(ST_GeomFromText('POINT(1 1 1)'));
```

- **ST_Npoints:** Επιστροφή για το πλήθος σημείων της γεωμετρίας

```
SELECT ST_NPoints(ST_GeomFromText('LINESTRING(77.29 29.07,77.42 29.26,77.27 29.31,77.29 29.07)'));
```

- **ST_SRID:** Ερώτηση για το αναγνωριστικό του συστήματος γεωγραφικής αναφοράς της γεωμετρίας

```
SELECT ST_SRID(ST_GeomFromText('POINT(-71.1043 42.315)',4326));
```

- **ST_X:** Ερώτηση για την τετμημένη x της γεωμετρίας

```
SELECT ST_X(ST_GeomFromEWKT('POINT(1 2 3 4)'));
```



Συναρτήσεις

Συναρτήσεις Επεξεργασίας

- **ST_AddPoint:** Προσθήκη σημείου σε γραμμή

```
SELECT ST_AddPoint(ST_GeomFromText('LINESTRING(0 0 1, 1 1 1)', 4326),  
ST_MakePoint(1, 2, 3));
```

- **ST_Force_2D:** Μετατροπή διάστασης γεωμετρίας σε 2

```
SELECT ST_Force_2D(ST_GeomFromEWKT('LINESTRING(1 1 2, 2 3 2, 4 5 2, 6  
7 2, 5 6 2)'));
```

- **ST_LineMerge:** Συρραφή γραμμών για τη δημιουργία μιας ενιαίας

```
SELECT ST_AsText(ST_LineMerge(ST_GeomFromText('MULTILINESTRING((-29  
-27,-30 -29.7,-36 -31,-45 -33), (-45 -33,-46 -32)')));
```

- **ST_Scale:** Αλλαγή της κλίμακας μιας γεωμετρίας

```
SELECT ST_AsEWKT(ST_Scale(ST_GeomFromEWKT('LINESTRING(1 2 3, 1 1  
1)'), 0.5, 0.75, 0.8));
```

- **ST_Translate:** Μετατόπιση γεωμετρίας στο δισδιάστατο επίπεδο

```
SELECT ST_AsText(ST_Translate(ST_GeomFromText(  
'LINESTRING(-71.1 42.3,-71.1 42.3)', 4326), 1, 0.5));
```

- **ST_Transform:** Μετασχηματισμός γεωμετρίας σε σύστημα αναφοράς

```
SELECT ST_Transform(ST_GeomFromText(POINT(743238  
2967416)', 2249), 4326);
```



Συναρτήσεις

Συναρτήσεις εξαγωγής γεωμετρίας

- **ST_AsText:** Εξαγωγή ως WKT κείμενο

```
SELECT ST_AsText(ST_GeomFromText('POLYGON((0 0,0 1,1 1,1 0,0 0))',4326));
```

- **ST_AsBinary:** Εξαγωγή σε δυαδική μορφή (WKB)

```
SELECT ST_AsBinary(ST_GeomFromText(' POLYGON((0 0,0 1,1 1,1 0,0 0))',4326));
```

- **ST_AsGeoJSON:** Εξαγωγή σε μορφή GeoJSON

```
SELECT ST_AsGeoJSON(ST_GeomFromText(' POLYGON( (0 0,0 1,1 1,1 0,0 0))',4326));
```

- **ST_AsGML:** Εξαγωγή σε μορφή GML

```
SELECT ST_AsGML('POLYGON( (0 0,0 1,1 1,1 0,0 0))',4326));
```

- **ST_AsKML:** Εξαγωγή σε μορφή KML

```
SELECT ST_AsKML('POLYGON( (0 0,0 1,1 1,1 0,0 0))',4326));
```

- **ST_AsLatLonText:** Εξαγωγή σε μορφή σφαιρικών συντεταγμένων

```
SELECT ST_AsLatLonText(ST_GeomFromText ('POINT(23.3456 38.5562)',4326));
```



Συναρτήσεις

Συναρτήσεις χωρικής συσχέτισης

- **ST_Equals:** Ελέγχει αν δυο γεωμετρίες ταυτίζονται.
- **ST_Contains:** Ελέγχει αν τα σημεία μιας γεωμετρίας A δε βρίσκονται στο εξωτερικό μιας γεωμετρίας B και τουλάχιστον ένα σημείο βρίσκεται στο εσωτερικό της B.
- **ST_Covers:** Ελέγχει εάν κανένα από τα σημεία μιας γεωμετρίας A δε βρίσκονται στο εξωτερικό μιας γεωμετρίας B.
- **ST_Crosses:** Ελέγχει εάν δυο γεωμετρίες έχουν κάποια (αλλά όχι όλα) κοινά εσωτερικά σημεία.
- **ST_Touches:** Ελέγχει εάν δυο γεωμετρίες έχουν τουλάχιστον ένα κοινό σημείο στην περιφέρειά τους και κανένα κοινό εσωτερικό σημείο.
- **ST_Dwithin:** Ελέγχει εάν δυο γεωμετρίες απέχουν μεταξύ τους το πολύ όσο μια δηλωμένη απόσταση.
- **ST_Intersects:** Ελέγχει εάν δυο γεωμετρίες δυο διαστάσεων τέμνονται μεταξύ τους.



Συναρτήσεις

Συναρτήσεις υπολογισμών

- **ST_Distance:** Επιστρέφει την απόσταση μεταξύ δυο δισδιάστατων γεωμετριών.
- **ST_Area:** Επιστρέφεται το εμβαδό ενός πολυγώνου ή ενός πολυπολυγώνου.
- **ST_Length:** Επιστρέφει το μήκος μιας δισδιάστατης γραμμής ή πολύ-γραμμής.
- **ST_Perimeter:** Επιστρέφει το μήκος της περιμέτρου μιας επιφάνειας (π.χ. πολύγωνο ή πολυπολύγωνο).
- **ST_ShortestLine:** Επιστρέφει την κοντύτερη γραμμή που ενώνει δυο γεωμετρίες.
- **ST_Centroid:** Επιστρέφεται το κεντροειδές σημείο μιας γεωμετρίας.
- **ST_ClosestPoint:** Επιστρέφεται το σημείο μιας Α γεωμετρίας που βρίσκεται κοντύτερα σε μια Β γεωμετρία.



Συναρτήσεις

Γενικές Συναρτήσεις Διαχείρισης

- **AddGeometryColumn:** Προσθέτει σε έναν πίνακα μια στήλη γεωμετρίας.

```
SELECT AddGeometryColumn('tablename',  
    'columnname', 2100, POINT, 2);
```
- **DropGeometryColumn:** Καταργεί τη στήλη γεωμετρίας ενός πίνακα.
- **UpdateGeometrySRID:** Ανανεώνει το αναγνωριστικό του συστήματος γεωγραφικής αναφοράς μιας στήλης γεωμετρίας.
- **PostGIS_Full_Version:** Ενημερώνει αναλυτικά για την τρέχουσα έκδοση της PostGIS.
- **PostGIS_PROJ_Version:** Ενημερώνει την για την τρέχουσα έκδοση της βιβλιοθήκης PROJ4.



Τελεστές

- **&&**: Αληθές όταν τα δισδιάστατα περικλείοντα κουτιά (ΠΚ) δυο γεωμετριών τέμνονται.
- **&&&**: Αληθές όταν τα τρισδιάστατα ΠΚ δυο γεωμετριών τέμνονται.
- **&<**: Αληθές όταν το ΠΚ της γεωμετρίας A υπερκαλύπτει ή βρίσκεται αριστερά του ΠΚ της γεωμετρίας B.
- **&>**: Αληθές όταν το ΠΚ της γεωμετρίας A υπερκαλύπτει ή βρίσκεται δεξιά του ΠΚ της γεωμετρίας B.
- **<<**: Αληθές όταν το ΠΚ της γεωμετρίας A βρίσκεται αυστηρά αριστερά του ΠΚ της γεωμετρίας B.
- **>>**: Αληθές όταν το ΠΚ της γεωμετρίας A βρίσκεται αυστηρά δεξιά του ΠΚ της γεωμετρίας B.
- **@**: Αληθές όταν το ΠΚ της γεωμετρίας A περιέχεται από το ΠΚ της γεωμετρίας B.
- **~=**: Αληθές όταν το ΠΚ της γεωμετρίας A ταυτίζεται με το ΠΚ της γεωμετρίας B.



Εργαλεία

Φόρτωση Δεδομένων

- **shp2pgsql**: Φορτώνει ένα shaperefile σε σχεσιακή βάση δίνοντας σαν παραμέτρους
 - το όνομα του πίνακα όπου θα φορτωθεί
 - το SRID
 - προαιρετικά το encoding των μη γεωμετρικών δεδομένων

```
$ shp2pgsql -s 2270 jacksonco_schools.shp  
jacksonco_schools_table | psql -d ma.ellak.db
```



Παραδείγματα

Δημιουργία πίνακα με στήλη γεωμετρίας

```
CREATE TABLE lines (name varchar);
```

```
SELECT AddGeometryColumn('lines', 'line', -1,  
'LINESTRING', 2);
```

```
INSERT INTO lines VALUES ('North West', 'LINESTRING(0 0,-1  
1)'),  
('North East', 'LINESTRING(0 0, 1 1)'),  
('South West', 'LINESTRING(0 0,-1 -1)'),  
('South East', 'LINESTRING(0 0,1 -1)');
```

```
SELECT name, ST_AsText(line) FROM lines;
```



Παραδείγματα

Δημιουργία πίνακα με στήλη γεωμετρίας (2)

```
CREATE TABLE points (name varchar, point geometry);
```

```
INSERT INTO points VALUES ('Origin', 'POINT(0 0)'),  
    ('North', 'POINT(0 1)'),  
    ('East', 'POINT(1 0)'),  
    ('West', 'POINT(-1 0)'),  
    ('South', 'POINT(0 -1)');
```

```
SELECT name, ST_AsText(point) FROM points;
```



Παραδείγματα

Μετρήσεις

Μήκος

```
SELECT ST_Length(ST_GeomFromText('LINESTRING(0 0,5 0,5 10)',2100));
```

Μήκος σε 3 διαστάσεις

```
SELECT ST_3DLength(ST_GeomFromEWKT('LINESTRING(0 0 0,5 0 3,5 10 5)'));
```

Μήκος στο σφαιροειδές

```
SELECT ST_Length2D_Spheroid(  
    ST_GeomFromEWKT('LINESTRING(151.1205 -33.7145  
0,151.1218 -33.7087 54)'),  
    'SPHEROID["GRS 1980",6378137,298.257222101]');
```

Περίμετρος

```
SELECT ST_Perimeter(  
    ST_GeomFromEWKT('POLYGON((-2 -2 0,2 -2 1,2 2 2,-2 2 1,-2  
-2 0))'));
```



Παραδείγματα

Μετρήσεις

Περίμετρος - 1ος τρόπος

```
SELECT ST_Perimeter(ST_GeomFromEWKT('POLYGON((-2 -2,2 -2,2 2,-2 2,-2 -2),(1 1,-1 1,-1 -1,1 -1,1 1))'))
```

Περίμετρος - 2ος τρόπος

```
SELECT
  ST_Length(
    ST_ExteriorRing(
      ST_GeomFromEWKT(
        'POLYGON((-2 -2,2 -2,2 2,-2 2,-2 -2),(1 1,-1 1,-1 -1,1 -1,1 1))'
      )
    )
  ) ;
```



Παραδείγματα

Μετρήσεις

Εμβαδόν

```
SELECT ST_Area(ST_GeomFromEWKT(g))
FROM (
VALUES
    ('POLYGON((-2 -2 0,2 -2 1,2 2 2,-2 2 1,-2 -2 0))'),
    ('POLYGON((-2 -2,2 -2,2 2,-2 2,-2 -2),(1 1,-1 1,-1 -1,1 -1,1 1))')
) AS query(g);
```

Απόσταση

```
SELECT ST_Distance(ST_GeomFromEWKT('POINT(0 5)'),
ST_GeomFromEWKT('LINESTRING(-2 2,2 2)'));
```



Παραδείγματα

Μετρήσεις

Υπολογισμός των σχολείων που βρίσκονται σε απόσταση το πολύ 5 KM από το πάρκο «Hawthorne Park / Pool»

```
SELECT jacksonco_schools.name
      FROM jacksonco_schools, medford_parks
     WHERE ST_Distance(medford_parks.geom,
jacksonco_schools.geom) < 5000 / 0.3048 AND
medford_parks.name = 'Hawthorne Park / Pool';
```



Παραδείγματα

Μετρήσεις

Εμβαδόν και περίμετρος

```
SELECT ST_Area((SELECT geom FROM my_polygons WHERE name = 'polygon 1')),  
        ST_Area((SELECT geom FROM my_polygons WHERE name = 'polygon 2'));
```

```
SELECT ST_Perimeter((SELECT geom FROM my_polygons WHERE name =  
'polygon 1')) ,  
        ST_Perimeter((SELECT geom FROM my_polygons WHERE name =  
'polygon 2')) ,  
        ST_Length(ST_ExteriorRing((SELECT geom FROM my_polygons  
WHERE name = 'polygon 1'))));
```



Παραδείγματα

Υπολογισμοί Γεωμετριών

Κοντινότερη Γραμμή

```
INSERT INTO my_lines (name,geom)
VALUES ('shortest line between polygons',
        ST_ShortestLine((SELECT geom FROM my_polygons WHERE name = 'polygon
1'),
                        (SELECT geom FROM my_polygons WHERE name =
'polygon 2'))
        );
```

Κεντροειδές

```
INSERT INTO my_points (name,geom)
VALUES ('polygons' '1 centroid',
        ST_Centroid((SELECT geom FROM my_polygons WHERE name = 'polygon 1'))
        );
```

Κοντινότερο Σημείο

```
INSERT INTO my_points (name,geom)
VALUES ('closest point on line 2 between line 2 and polygon 2',
        ST_ClosestPoint((SELECT geom FROM my_lines WHERE name = 'line 2'),
                        (SELECT geom FROM my_polygons WHERE name =
'polygon 2'))
        );
```



Παραδείγματα

Υπολογισμοί συσχέτισης

```
SELECT id,name FROM my_polygons WHERE
    ST_Equals(geom,(SELECT geom FROM my_polygons WHERE name = 'polygon 1'));

SELECT id,name FROM my_lines WHERE
    ST_Contains((SELECT geom FROM my_polygons WHERE name = 'polygon 1'),geom);

SELECT id,name FROM my_lines WHERE
    ST_Crosses((SELECT geom FROM my_polygons WHERE name = 'polygon 1'),geom);

SELECT id,name FROM my_lines WHERE
    ST_Covers((SELECT geom FROM my_polygons WHERE name = 'polygon 1'),geom);

SELECT id,name FROM my_lines WHERE
    ST_Touches((SELECT geom FROM my_polygons WHERE name = 'polygon 1'),geom);
```



Παραδείγματα

Υπολογισμοί συσχέτισης

```
SELECT id,name FROM my_lines WHERE  
      ST_Intersects((SELECT geom FROM my_polygons WHERE name = 'polygon  
1'),geom);
```

```
SELECT id,name FROM my_lines WHERE  
      ST_Disjoint((SELECT geom FROM my_polygons WHERE name = 'polygon 1'),geom);
```

```
SELECT id,name FROM my_lines WHERE  
      ST_Dwithin((SELECT geom FROM my_polygons WHERE name = 'polygon 1'),geom, 0.1);
```

```
INSERT INTO my_polygons (name, geom) VALUES ('polygon 1's buffer',  
      ST_Buffer((SELECT geom FROM my_polygons WHERE name = 'polygon 1'),0.1));
```



Παραδείγματα

Προβολή γεωμετρίας

```
SELECT ST_AsText((SELECT geom FROM my_polygons WHERE name = 'polygon 2'));  
SELECT ST_AsBinary((SELECT geom FROM my_polygons WHERE name = 'polygon 2'));  
SELECT ST_AsGeoJSON((SELECT geom FROM my_polygons WHERE name = 'polygon 2'));  
SELECT ST_AsGML((SELECT geom FROM my_polygons WHERE name = 'polygon 2'));  
SELECT ST_AsKML((SELECT geom FROM my_polygons WHERE name = 'polygon 2'));  
SELECT ST_AsLatLonText((SELECT geom FROM my_points LIMIT 1));
```



Παραδείγματα

Χρήση Τελεστών

```
SELECT (SELECT geom FROM my_polygons WHERE name = 'big C') &&  
      (SELECT geom FROM my_polygons WHERE name = 'little o');
```

```
SELECT ST_Intersects(  
      (SELECT geom FROM my_polygons WHERE name = 'big C'),  
      (SELECT geom FROM my_polygons WHERE name = 'little o'));
```

```
SELECT (SELECT geom FROM my_polygons WHERE name = 'little o') @  
      (SELECT geom FROM my_polygons WHERE name = 'big C');
```

```
SELECT (SELECT geom FROM my_polygons WHERE name = 'right triangle') &>  
      (SELECT geom FROM my_polygons WHERE name = 'big C');
```

```
SELECT (SELECT geom FROM my_polygons WHERE name = 'left triangle') <<  
      (SELECT geom FROM my_polygons WHERE name = 'big C');
```

```
SELECT (SELECT geom FROM my_polygons WHERE name = 'left triangle') &<  
      (SELECT geom FROM my_polygons WHERE name = 'big C');
```



Χρήσιμοι σύνδεσμοι

- <http://www.postgresql.org/download/>
- <http://postgis.net/install>

