

Semestrální projekt

153UZPD Úvod do zpracování prostorových dat

T. Fiedlerová, P. Klimunda, J. Špička

16. května 2012

Obsah

- 1 O projektu
 - Zadání
 - Náš cíl
 - Základní pojmy
- 2 Vytvoření a validace vrstev
 - Vytvoření vrstev
 - Validace dat
- 3 Síťové analýzy
 - Příprava dat
 - Vyhledání nejkratší cesty
- 4 Dotazy
 - Atributové dotazy
 - Prostorové dotazy
 - Síťové analýzy
- 5 Konec

Zadání

- Navrhnete a vytvořte tématické vrstvy (např. vodní toky, vodní plochy, lesy, silnice, železnice a pod.) na základě dat OSM (viz cvičná databáze `pgis_student` schéma `osm`). Pro tento účel byla na serveru 'geo102' založena databáze `pgis_uzpd`.
- Aplikujte testy datové integrity a odstraňte případné nekonzistence v datech.
- Vytvořte tutoriál pro výuku PostGIS - tj. sadu atributových a prostorových dotazů nad databází *pgis_uzpd*.

Náš cíl

- Vytvoření tématických vrstev souvisejících především s motorismem.
- Základní validace dat.
- Vytvoření různorodých atributových a prostorových dotazů.
- Sestavení dotazů využívajících PgRouting.

Základní pojmy

- OSM - Open Street Map
- PostGIS - prostorová nadstavba PostgreSQL
- PgRouting - rozšíření PostGISu pro provádění síťových analýz



Vybrané tématické vrstvy

- Bodové vrstvy: benzinky, motorismus, parkoviste, obce_bod
- Liniové vrstvy: silnice
- Polygonové vrstvy: trava, prum_zony, obce
- Výsledná data v SS Google Mercator - u obcí nutná transformace

Vytvoření vrstev

```
CREATE TABLE silnice AS
SELECT osm_id, name AS nazev, highway AS popis, geom
FROM osm.czech_line
WHERE highway IN ('motorway', 'motorway_link', 'trunk',
                  'trunk_link', 'primary', 'primary_link',
                  'secondary', 'secondary_link');
ALTER TABLE silnice ADD PRIMARY KEY (osm_id);
CREATE INDEX silnice_geom ON silnice USING gist (geom);
```

Převedení polygonů na body

- Vrstvy benzinky, parkoviste a motorismus - kombinace dat z *czech_polygon*, *czech_point*
- Nutné převedení všech prvků na body

```
CREATE TABLE benzinky_pol AS  
SELECT osm_id, name AS nazev, amenity AS popis, geom  
FROM czech_polygon  
WHERE amenity = 'fuel';
```

```
ALTER TABLE b12.benzinky_pol ADD PRIMARY KEY (osm_id);
```


Převedení polygonů na body

- Kontrola validity dat
- Odstranění duplicitních dat
- *ST_PointOnSurface()* × *ST_Centroid()*

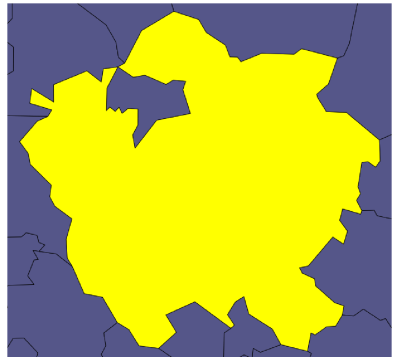
```
DELETE FROM benzinky_pol WHERE osm_id IN  
(SELECT b.osm_id FROM benzinky_bod AS a  
JOIN benzinky_pol AS b  
ON ST_Contains(b.geom, a.geom));
```

```
INSERT INTO benzinky_bod  
(SELECT osm_id, nazev, popis,  
ST_PointOnSurface(geom) AS geom FROM benzinky_pol);
```

```
DROP TABLE IF EXISTS benzinky_pol;
```

Validace dat

- Bodové vrstvy - validní
- Liniové vrstvy - validní
- Polygonové vrstvy - chyby
Self-Intersection,
Ring Self-Intersection
- Funkce *ST_MakeValid()*

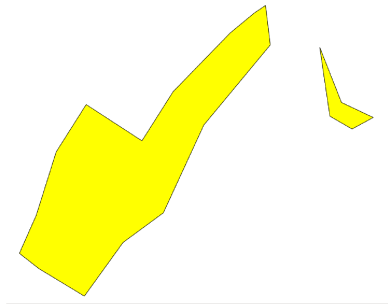
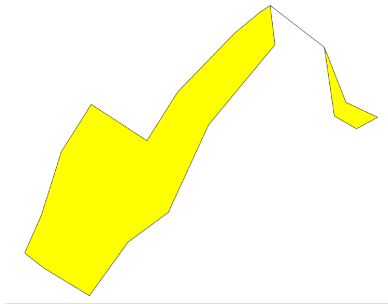


Postup validace

- Zjištění nevalidních prvků:

```
SELECT osm_id, ST_IsValidReason(geom)
FROM trava
WHERE ST_IsValid(geom) = FALSE;
```

Ukázka validace



Řešení

- Řešení:

```
ALTER TABLE trava RENAME COLUMN geom TO geom1;  
SELECT AddGeometryColumn('trava','geom',900913,'multipolygon',2,true);  
UPDATE trava SET geom = ST_Multi(geom1);  
UPDATE trava SET geom = ST_CollectionExtract(ST_MakeValid(geom),3)  
  WHERE ST_IsValid(geom) = false  
  AND ST_GeometryType(ST_MakeValid(geom)) = 'ST_GeometryCollection';  
UPDATE trava SET geom = ST_MakeValid(geom)  
  WHERE ST_IsValid(geom) = FALSE;  
SELECT DropGeometryColumn('trava', 'geom1');
```

Příprava dat

- Vybraná vrstva - silnice
- Přidání sloupců - source, target, length, x1, y1, x2, y2, reverse_cost, to_cost, rule
- Vytvoření topologie sítě - *assign_vertex_id()*
- *ST_X()*, *ST_Y()*, *ST_Length()*

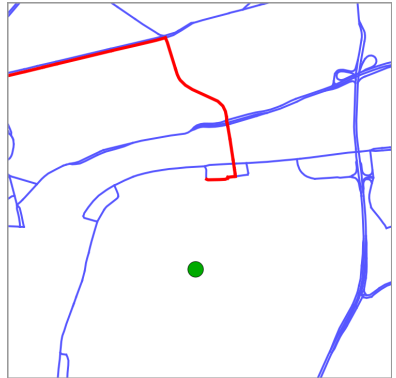
```
ALTER TABLE silnice ADD COLUMN length FLOAT;  
...  
SELECT assign_vertex_id('b12', 'silnice', 1, 'geom', 'osm_id');
```

Vyhledání nejkratší cesty

- Djistkra
 - první algoritmus pro pgRouting
- A-Star
 - vyžaduje sloupce s x-ovými a y-ovými souřadnicemi uzlových bodů
- Shooting Star
 - výpočet výběrem hran, ne spojnic uzlů
- Další možnosti PgRouting
 - Traveling Salesperson Problem (TSP)
 - Driving Distance calculation (Isolines)

Vyhledání nejkratší cesty

- Vyhledání 'správného' uzlu
find_node_by_nearest_link_within_distance()
- Jiné možné řešení -
vytvoření uzlů k daným
místům



Atributové dotazy

Dotaz

Které obce v Olomouckém kraji nezačínají na 'D' a končí na 'ov'?
Vypište id a název.

```
SELECT ogc_fid, nazev_eng FROM obce  
WHERE nazev_eng LIKE '%ov' AND nk='OL'  
AND nazev_eng NOT LIKE 'D%';
```

ogc_fid	nazev_eng
4087	Grymov
794	Bezuchov
3075	Komarov
333	Cisarov
...	

(50 rows)

Prostorové dotazy

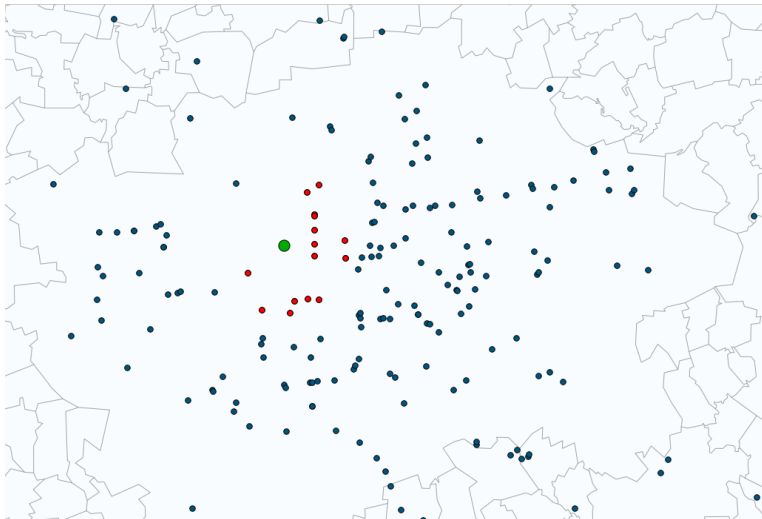
Dotaz

Vypište název, popis a geometrii benzínek, které jsou v okruhu 5 km od centra Prahy (bodová vrstva).

```
SELECT nazev, popis, ST_Astext(geom) FROM benzinky  
WHERE ST_Intersects(geom,ST_Buffer((SELECT geom FROM obce_bod  
WHERE nazob_eng = 'Praha')),5000));
```

nazev	popis	st_astext
	fuel	POINT(1603198.86042138 6459632.03752206)
	fuel	POINT(1604175.31046669 6457073.08983793)
	fuel	POINT(1607275.45551884 6465159.795)
Rohanské nábřeží	fuel	POINT(1607750.55121804 6462543.105)
...		
(15 rows)		

Prostorové dotazy



Prostorové dotazy

Dotaz

Jaká je rozloha všech travnatých ploch v ha ve Znojmě?

```
SELECT SUM(St_Area(St_Intersection(t.geom, o.geom)))/10000  
AS plocha_ha FROM trava AS t  
JOIN obce AS o ON St_Intersects(t.geom, o.geom)  
WHERE nazev_eng = 'Znojmo';
```

plocha_ha

```
-----  
104.307467119909  
(1 row)
```

Prostorové dotazy

Dotaz

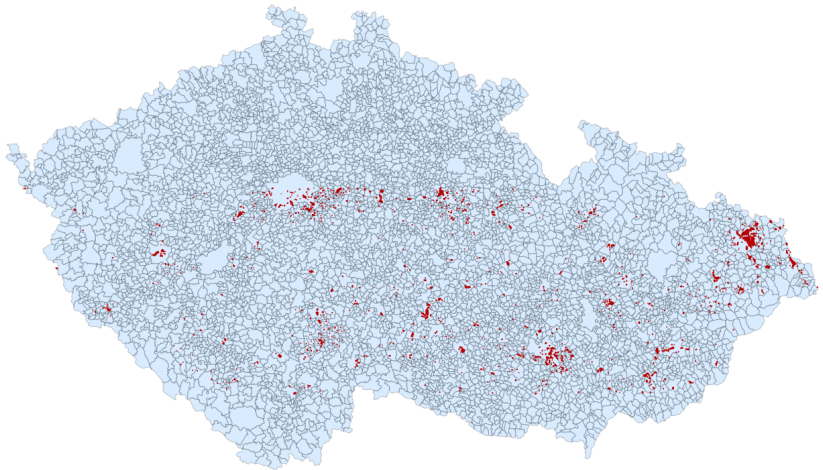
Jaká je průměrná rozloha (ha) a počet průmyslových zón, které jsou jižněji než Praha a severněji než České Budějovice?

```
SELECT DISTINCT (SUM(ST_Area(prum_zony.geom))/1e4)/COUNT(*)  
AS prumer, COUNT(*) AS pocet FROM prum_zony  
JOIN obce_bod AS ob1  
ON (ST_y(ST_centroid(prum_zony.geom)) < ST_y(ob1.geom))  
JOIN obce_bod AS ob2  
ON (ST_y(ST_centroid(prum_zony.geom)) > ST_y(ob2.geom))  
WHERE ob1.nazob_eng='Praha' AND ob2.nazob_eng='Česke Budejovice';
```

prumer	pocet
13.4854556402118	2533

(1 row)

Prostorové dotazy



Síťové analýzy

Dotaz

O kolik delší bude cesta z uzlového bodu 5626 do 12073, jestliže nepojedeme po dálnici?

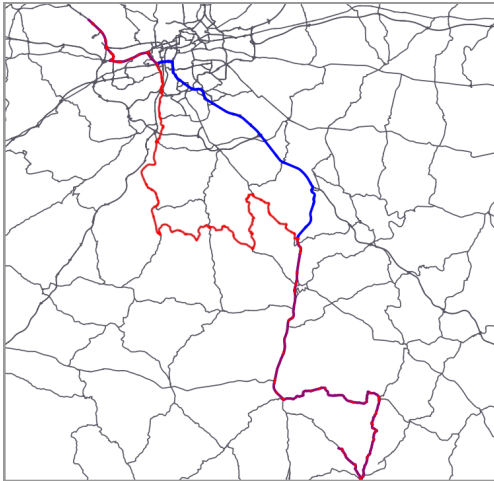
```
SELECT (b.delka-a.delka)/1000 AS rozdil_km FROM (SELECT SUM(length)
AS delka FROM shortest_path('SELECT osm_id AS id, source, target,
length AS cost FROM silnice', 5626, 12073, FALSE, FALSE) AS cesta2
JOIN silnice ON cesta2.edge_id = silnice.osm_id) AS a
JOIN
(SELECT SUM(length) AS delka FROM shortest_path('SELECT osm_id AS id,
source, target, length AS cost FROM silnice WHERE popis NOT IN
(''motorway'', ''motorway_link'')', 5626, 12073, FALSE, FALSE) AS cesta
JOIN silnice ON cesta.edge_id = silnice.osm_id) AS b
ON a.delka != b.delka;
```

rozdil_km

47.8367686689323

(1 row)

Síťové analýzy



Síťové analýzy

Dotaz

Kterými obcemi se projíždí na nejkratší trase mezi danými dvěma body?

```
SELECT DISTINCT nazev_eng
FROM (SELECT geom, silnice.osm_id AS osm_id
FROM shortest_path_astar('SELECT osm_id AS id,source,target,
length AS cost,x1,y1,x2,y2 FROM silnice', 1099,8316, false, false) AS path
JOIN silnice ON path.edge_id=silnice.osm_id) AS A
JOIN obce ON ST_intersects(A.geom,obce.geom) order by nazev_eng;
```

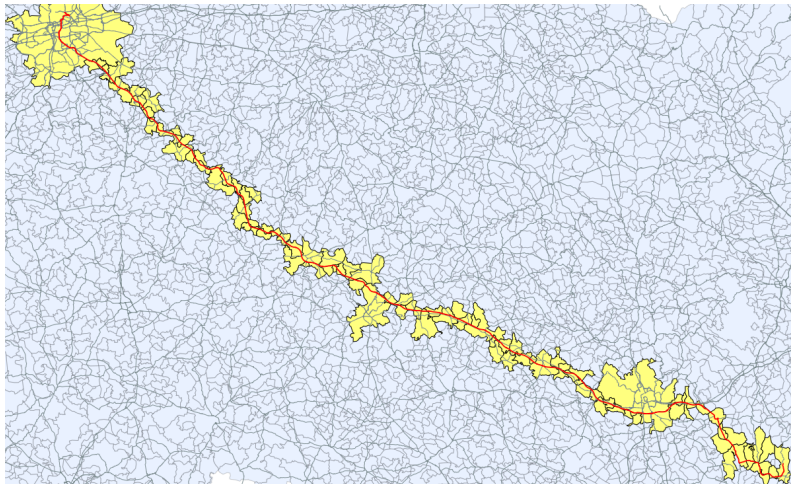
nazev_eng

Archlebov
Bernartice
Blizkov

...

(96 rows)

Síťové analýzy



Síťové analýzy

Dotaz

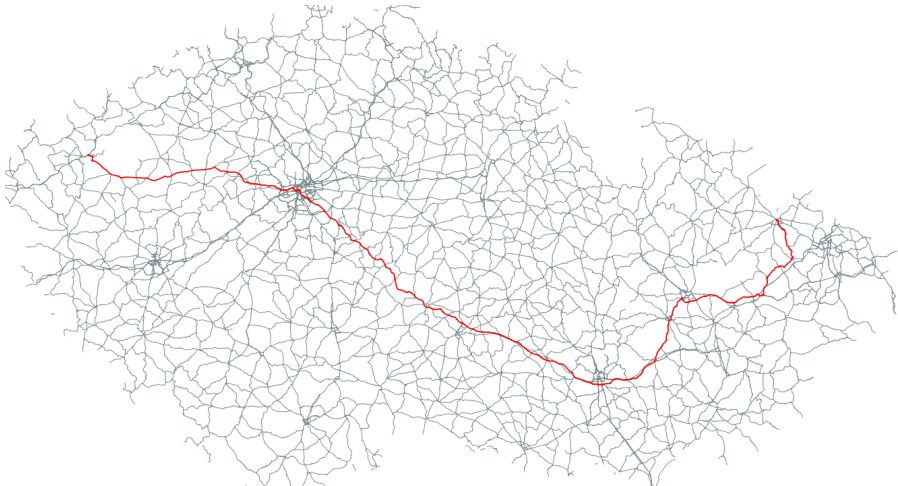
Jaká je nejkratší cesta z Karlových Varů do Opavy? Řešte pomocí algoritmu Shooting Star.

```
SELECT * FROM shortest_path_shooting_star('SELECT osm_id as id,source,
target,length as cost,x1,y1,x2,y2,rule,to_cost FROM silnice',
(SELECT * from find_nearest_link_within_distance((SELECT st_astext(geom)
FROM obce_bod WHERE nazob_eng = 'Karlovy Vary'),5000,'silnice')),
(SELECT * from find_nearest_link_within_distance((SELECT st_astext(geom)
FROM obce_bod WHERE nazob_eng = 'Opava'),5000,'silnice')),false,false);
```

vertex_id	edge_id	cost
21009	5046149	2040.77321110597
21052	5083061	431.150051875099
...		
14326	49302669	1036.98283951956

(508 rows)

Síťové analýzy



Děkujeme za pozornost.
Skupina b12