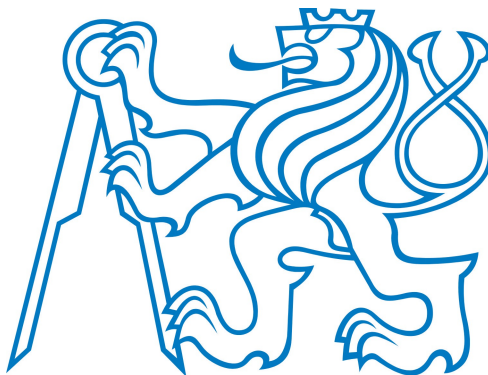


ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

FAKULTA STAVEBNÍ

KATEDRA MAPOVÁNÍ A KARTOGRAFIE



DOKUMENTACE

ÚVOD DO ZPRACOVÁNÍ PROSTOROVÝCH DAT

SEMESTRÁLNÍ PROJEK

SKUPINA A

ŠTĚPÁN BAMBULA, ADAM DLESK, ŽOFIE CIMBUROVÁ

AKADEMICKÝ ROK 2012/2013, LETNÍ SEMESTR

OBSAH

1. Úvod	2
1.1 Zadání	2
1.2 Výběr tématu	2
1.3 Použitý software	2
1.3.1 OSM	2
1.3.2 PostGIS	2
1.3.3 PgRouting	2
2. Vrstvy	3
2.1 Výběr vrstev a úprava	3
2.2 Transformace	4
2.3 Validace	5
2.4 PgRouting	6
2.4.1 Ohodnocení hran	7
2.4.2 Průjezdni místa	7
3. Dotazy	8
3.1 Atributové	8
3.2 Prostorové	9
3.3 Síťové	11
4. Závěr	15
5. Přílohy	15
6. Zdroje	15

1. Úvod

1.1 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.

1.2 Výběr tématu

V semestrální práci jsme se rozhodli věnovat zejména aplikaci síťových analýz. Proto jsme jako stěžejní vrstvu zvolili síť komunikací, na kterou je dobře aplikovatelná funkcionality rozšíření PgRouting (síťové analýzy pro vyhledání nejkratší trasy, problém obchodního cestujícího a dojezdovou vzdálenost). Tuto síť jsme doplnili polygonovými a bodovými vrstvami tak, aby bylo možno příslušné analýzy dobře představit a doplnit je dalšími atributovými a prostorovými dotazy.

1.3 Použitý software

1.3.1 OSM

Projekt Open Street Map slouží k vytváření volně dostupné a dále editovatelné mapy celého světa. Data jsou poskytována pod licencí Open Database License, která umožňuje bezplatný přístup ke kompletnímu datasetu.

Do programu PostGIS je možné tato data importovat pomocí aplikace *osm2pgsql*.

1.3.2 PostGIS

PostGIS je open source projekt představující rozšíření pro oběktově-relační databázový systém PostgreSQL, umožňující uložení a správu geoprostorových objektů.

1.3.3 PgRouting

PgRouting představuje rozšíření PostGISu pro využití síťových analýz (vyhledání nejkratší cesty, problém obchodního cestujícího, dojezdová vzdálenost).

2. Vrstvy

2.1 Výběr vrstev a úprava

Bodové

obce_body

pamatky

Liniové

cyklostezky

potoky

reky

silnice (*první, druhé a třetí třídy*)

Polygonové

lesy

obce

vodni_plochy

Vrstvy *obce* a *obce_body* byly převzaty ze schematu *gis1*, ostatní vrstvy ze schematu *uzpd*.

Jednotlivým vrstvám byly po vybrání definovány nové primární klíče (*gid*) a pomocí funkce *create index* byly vytvořeny prostorové indexy.

```
ALTER TABLE reky ADD COLUMN gid serial;  
ALTER TABLE reky ADD PRIMARY KEY (gid);  
CREATE INDEX reky_geom_gist ON reky USING gist (geom);
```

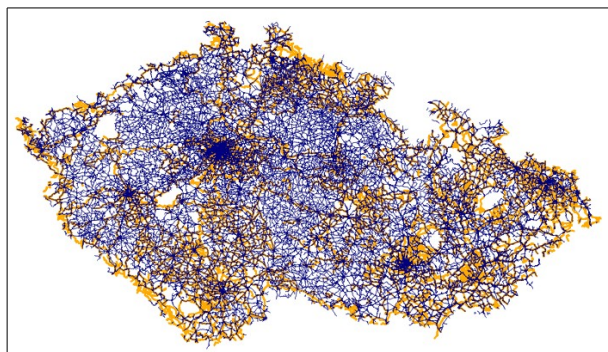
Aby bylo možno využít síťovou analýzu, byly spojeny tabulky *cyklostezky* a *silnice*.

Nejprve byla v tabulce *cyklostezky* sloupci *highway* přiřazena jednotná hodnota '*cycleway*'.

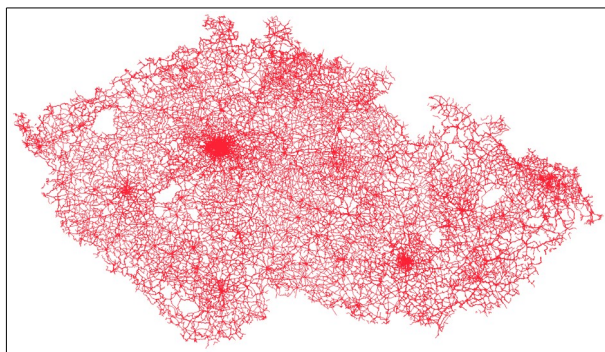
```
UPDATE cyklostezky SET highway = 'cycleway';
```

Následně byly tabulky *silnice* a *cyklostezky* spojeny do jedné tabulky *komunikace*.

```
CREATE TABLE komunikace AS
(SELECT osm_id, highway, geom FROM cyklostezky UNION
SELECT osm_id, highway, geom FROM silnice);
```



síť silnic a cyklostezek před spojením



síť komunikací po spojení

2.2 Transformace

Data ze schématu *uzpd* jsou v souřadnicovém systému Google Mercator.

Data ze schématu *gis1* jsou v souřadnicovém systému S-JTSK, proto bylo potřeba je před vlastním použitím pro dotazy transformovat.

```
ALTER TABLE a13.obce RENAME COLUMN geom TO geom1;
SELECT AddGeometryColumn('obce', 'geom', 900913, 'multipolygon', 2);
UPDATE obce SET geom = ST_Transform(geom1, 900913);
SELECT DropGeometryColumn('obce', 'geom1');
```

2.3 Validace

Pro práci s daty je třeba zajistit jejich konzistenci. Nevalidní prvky byly zjišťovány pomocí funkce *ST_IsValid()* a důvod chyby pomocí funkce *ST_IsValidReason()*.

```
SELECT osm_id, ST_IsValidReason(geom) AS duvod FROM reky
WHERE ST_IsValid(geom) = FALSE;
```

Po spuštění funkcí pro všechny vrstvy byl zjištěn jeden nevalidní prvek ve vrstvě *obce*.

```
ogc_fid | reason
-----+-----
6       | Ring Self-intersection[1670500.05469174 6586747.41733279]
```



před opravou

Problém jsme se pokoušeli opravit pomocí funkce *ST_MakeValid()*, která převádí polygony na multipolygony, validace se ale nezdařila a protože se do termínu odevzdání nepodařilo najít vhodné řešení, ponechali jsme tento jeden prvek invalidní.

2.4 PgRouting

Aby bylo možné využít vrstvu *komunikace* pro síťové analýzy, bylo třeba provést několik kroků.

Nejprve byly každé linii přiřazeny počáteční a koncové uzly. Do tabulky byly přidány sloupce *source*, *target* a *length*.

```
ALTER TABLE komunikace ADD COLUMN source INTEGER;  
ALTER TABLE komunikace ADD COLUMN target INTEGER;  
ALTER TABLE komunikace ADD COLUMN length FLOAT;
```

Dále byly funkcí *create index* vytvořeny prostorové indexy.

```
CREATE INDEX source_idx ON komunikace(source);  
CREATE INDEX target_idx ON komunikace(target);
```

Nakonec byly vytvořeny počáteční a koncové uzly a spočtena délka jednotlivých úseků.

```
SELECT assign_vertex_id('a13', 'komunikace', 1, 'geom', 'gid');  
UPDATE komunikace SET length = ST_Length(geom);
```

Pro aplikaci algoritmů *A-Star* a *Shooting Star* bylo potřeba přidat další sloupce.

A-Star

```
ALTER TABLE komunikace ADD COLUMN x1 DOUBLE PRECISION;  
ALTER TABLE komunikace ADD COLUMN y1 DOUBLE PRECISION;  
ALTER TABLE komunikace ADD COLUMN x2 DOUBLE PRECISION;  
ALTER TABLE komunikace ADD COLUMN y2 DOUBLE PRECISION;  
  
UPDATE komunikace SET x1 = ST_x(ST_startpoint(geom));  
UPDATE komunikace SET y1 = ST_y(ST_startpoint(geom));  
UPDATE komunikace SET x2 = ST_x(ST_endpoint(geom));  
UPDATE komunikace SET y2 = ST_y(ST_endpoint(geom));
```

Shooting Star

```
ALTER TABLE komunikace ADD COLUMN to_cost double precision;  
ALTER TABLE komunikace ADD COLUMN rule text;
```

2.4.1 Sloupec ohodnocení hran

Při jízdě na kole je jednoznačně mnohem příjemnější a výhodnější využít cyklostezku než silnici 3. třídy a tu než silnici 2. třídy a následně 1. třídy (bereme-li v úvahu kromě kvality vozovky zároveň hustotu provozu, která jízdní komfort značně snižuje). Nehledáme tedy nejkratší cestu, ale v jistém slova smyslu 'nejlevnější' cestu.

S přihlédnutím k těmto faktům jsme volili hodnocení hran sítě. Cyklostezkám (*cycleway*), jakožto nejpríjemnějším, byla přiřazena cena 1, silnicím 3. třídy (*tertiary*) cena 2, silnicím 2. třídy (*secondary*) cena 3 a silnicím 1. třídy (*primary*), jakožto nejrizikovějším, cena 4.

O ohodnocování hran by se jistě dalo polemizovat, zejména pokud by tato práce měla vést k reálným výstupům, nicméně vzhledem k tomu, že její účel je pouze ukázkový, nemusíme se o ohodnocování hran déle rozepisovat.

```
ALTER TABLE komunikace ADD COLUMN cost INTEGER;
UPDATE komunikace SET cost=1 ;
UPDATE komunikace SET cost=2 WHERE highway IN ('tertiary');
UPDATE komunikace SET cost=3 WHERE highway IN ('secondary');
UPDATE komunikace SET cost=4 WHERE highway IN ('primary');
```

Hodnocení hran se v syntaxi funkcí projeví jednoduchým způsobem:

```
SELECT * FROM shortest_path_shooting_star('SELECT gid AS id, source,
target, length*cost AS naklad, x1, y1, x2, y2, rule, to_cost, FROM
komunikace', A, B, false, false);
```

2.4.2 Průjezdní místa

Při hledání požadované trasy je vhodné jako průjezdní místa zadávat obce, nikoliv uzly komunikací. Pro vyhledání uzlu, který leží nejbližší zadané obci byly použity následující funkce z knihovny PgRouting:

```
find_node_by_nearest_link_within_distance() (pro Dijkstrův algoritmus a A Star)
find_nearest_link_within_distance() (pro algoritmus Shoting Star)
```

Zdrojové kódy funkcí byl upraven tak, aby byly použitelné pro příslušné tabulky - parametr '*the_geom*' byl nahrazen za '*geom*'.

3. Dotazy

3.1 Atributové

1. Jaká je celková délka všech cyklostezek [km] ?

```
SELECT round(sum(length)/1000) AS delka_km
FROM komunikace WHERE highway = 'cycleway';
```

```
delka_km
-----
38534
```

[1 row]

2. Jak početné jsou jednotlivé typy památek? Seřadte od nejvyššího.

```
SELECT distinct historic AS typ, count (*) AS pocet
FROM pamatky GROUP BY historic ORDER BY pocet desc;
```

typ	pocet
wayside_cross	1676
memorial	1502
boundary_stone	721
wayside_shrine	680
ruins	297
monument	243
castle	240
...	

[25 rows]

3. Která je rozlohou největší obec, která má v názvu 'Týn'? Vypište název a rozlohu v ha.

```
SELECT nazev, round(round(area)/10000) AS rozloha
FROM obce WHERE nazev like '%Týn%' ORDER BY rozloha desc limit 1;
```

nazev	rozloha
Horšovský Týn	72788051

[1 row]

4. Z kolika úseků se skládá Vltava?

```
SELECT distinct name, count (*) AS pocet
FROM reky WHERE name = 'Vltava' GROUP BY name;
```

name	pocet
Vltava	40

[1 row]

5. Které jsou 3 nejsevernější obce v ČR?

```
SELECT nazob AS nazev
FROM obce_body ORDER BY sxob asc limit 3;
```

```
nazev
-----
Lobendava
Lipová
Šluknov
```

[3 rows]

3.2 Prostorové

1. Jaké procento obcí na svém území nemá cyklostezku?

```
SELECT COUNT(distinct(obce.name))*100./(SELECT COUNT(*) FROM obce)
AS maji
FROM obce AS obce
JOIN cyklostezky AS cyklo ON
ST_Intersects(obce.geom, cyklo.geom);
```

```
--- -
57 %
```

2. Kolik procent cyklostezek prochází lesem?

```
SELECT sum(ST_Length(ST_Intersection(cyklo.geom, lesy.geom)))*100./ (
SELECT sum(ST_Length(cyklostezky.geom)) FROM cyklostezky)
FROM cyklostezky AS cyklo
JOIN lesy AS lesy ON
ST_Intersects(cyklo.geom, lesy.geom);
```

```
-----
20.7 %
```

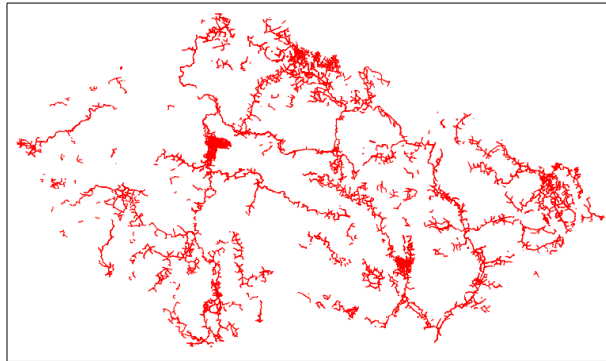


Cyklostezky v lese

3. Které cyklostezky leží do 5 km od řeky?

```
CREATE TABLE reky_buffer AS
SELECT ST_Buffer(reky.geom, 5000) AS geom1 FROM reky;

CREATE TABLE dotaz_cykloreky AS
SELECT ST_Intersection(reky_buffer.geom1, cyklostezky.geom)
FROM   cyklostezky
JOIN   reky_buffer
ON     ST_Intersects(reky_buffer.geom1, cyklostezky.geom);
```



Cyklostezky u řeky

4. Kolik procent Jihočeského kraje zabírají vodní plochy?

```
SELECT sum(ST_Area(vodni_plochy.geom))*100/(
SELECT sum(ST_Area(obce.geom))
FROM   obce
WHERE  nk = 'JC') from vodni_plochy;
```

```
-----
6.7 %
```

5. Kolik cyklostezek přejíždí přes řeku?

```
SELECT COUNT(*) FROM cyklostezky AS cyklo
JOIN reky AS reky ON
ST_Intersects(reky.geom, cyklo.geom);
```

```
---
963
```

3.3 Síťové

Kvůli zjednodušení byly předem zjištěny průjezdní uzly pomocí funkce

find_node_by_nearest_link_within_distance()

a linie pomocí funkce

find_nearest_link_within_distance()

pro používaná startovní a cílová místa.

```
SELECT * FROM find_nearest_link_within_distance(  
  (SELECT st_astext(geom) FROM obce_body WHERE nazob_eng = 'Litomerice'),  
  2000, 'komunikace')
```

```
SELECT * FROM find_node_by_nearest_link_within_distance(  
  (SELECT st_astext(geom) FROM obce_body WHERE nazob_eng = 'Jihlava'),  
  2000, 'komunikace')
```

	nearest node	nearest link

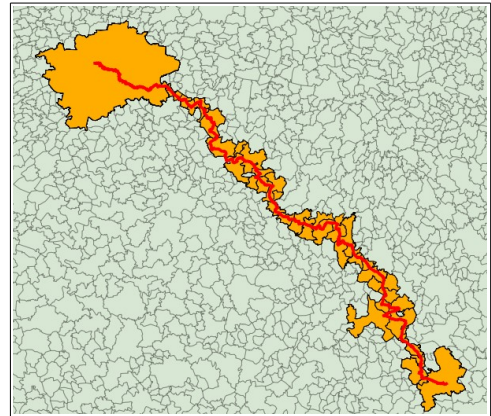
Jihlava	79907	1456
Praha	24486	1581
Železný Brod	46860	
Víchová nad Jizerou	16708	

1. Kterými obcemi projedete z Jihlavy do Prahy, pokud pojedete na kole? (*Shooting Star*)

```
SELECT distinct nazev FROM
(SELECT * FROM shortest_path_shooting_star
('SELECT gid AS id,source, target, length*cost AS cost, x1, y1, x2,
y2, rule, to_cost
FROM komunikace', 1456, 1581, false, false) AS cesta
JOIN komunikace ON cesta.edge_id=komunikace.gid) as a
JOIN obce ON ST_Intersects(a.geom,obce.geom);
```

```
-----
Babice
Bela
Bily Kamen
Bojiste
Brezi
Cejov
Cerne Voderady
...
```

[45 rows]



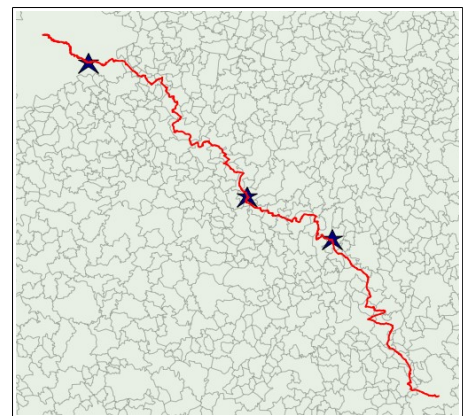
Průjezdní obce

2. Kolik hradů leží v bufferu 1 km od trasy z Jihlavy do Prahy? (*A Star*)

```
SELECT distinct p.osm_id, name
FROM (SELECT ST_Buffer(geom,1000) AS buf
FROM shortest_path_astar
('SELECT gid AS id, source, target, length*cost AS cost,
x1, y1, x2, y2
FROM komunikace', 79907, 24486, false, false) AS cesta
JOIN komunikace AS k ON cesta.edge_id=k.gid) as a
JOIN pamatky AS p ON ST_Intersects(a.buf, p.geom)
WHERE historic='castle';
```

osm_id	name
429455025	
296767926	Zámek Petrovice
477278863	

[3 rows]



Hrady u trasy

3. Porovnání metod A Star, Shooting Star a Dijkstrova algoritmu z hlediska délky a tvaru trasy

Dijkstrův algoritmus

```
SELECT round(sum(length)/1000) FROM shortest_path('SELECT gid AS
id, source, target, length as cost
FROM komunikace' , 79907, 24486, false, false) AS cesta
JOIN komunikace AS kom ON cesta.edge_id = kom.gid
```

```
---
239
```

A Star

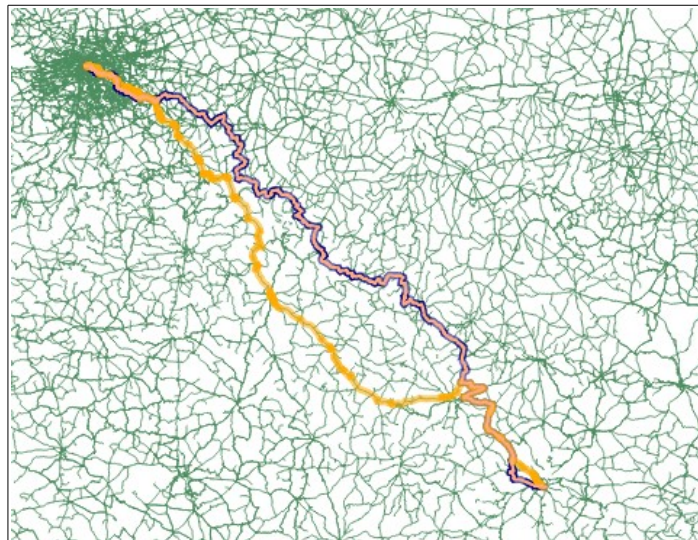
```
SELECT round(sum(length)/1000) FROM shortest_path_astar
('SELECT gid AS id, source, target, length*cost AS cost, x1, y1,
x2, y2
FROM komunikace', 79907, 24486, false, false) AS cesta
JOIN komunikace AS kom ON cesta.edge_id = kom.gid
```

```
---
275
```

Shooting Star

```
SELECT round(sum(length)/1000) FROM shortest_path_shooting_star
('SELECT gid AS id,source, target, length*cost AS cost, x1, y1, x2,
y2, rule, to_cost
FROM komunikace', 1456, 1581, false, false) AS cesta
JOIN komunikace AS kom ON cesta.edge_id = kom.gid
```

```
---
276
```

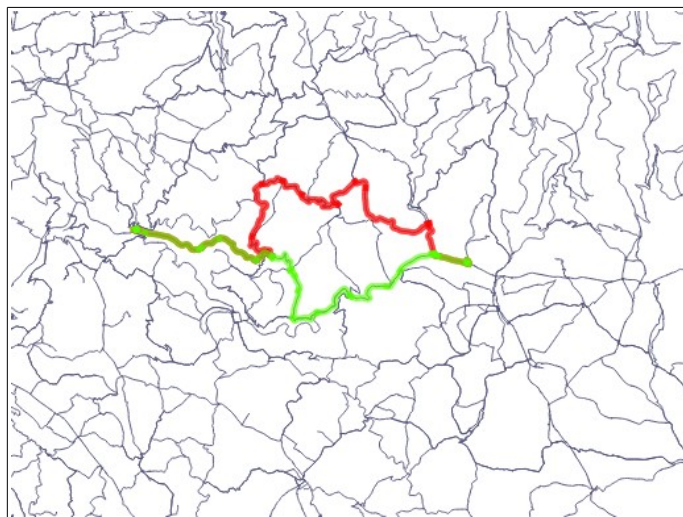


Porovnání Dijkstrova algoritmu (žlutě), A Star (modře) a Shooting Star (růžově)

4. Porovnání cesty mezi Železným Brodem a Víchovou nad Jizerou na kole a autem
(Dijkstrův algoritmus)

```
SELECT
(
  (SELECT SUM(kolo.cost) FROM
  (SELECT * FROM shortest_path('SELECT gid AS id, source, target,
length*cost as cost
FROM komunikace where highway != ''cycleway'' , 46860, 16708,
false, false)) AS kolo)
-
  (SELECT SUM(auto.cost) FROM
  (SELECT * FROM shortest_path('SELECT gid AS id, source, target,
length*cost as cost
FROM komunikace' , 46860, 16708, false, false)) AS auto)
)/1000 AS rozdil_km;

-----
34.5 km
```



Rozdíl mezi jízdou autem (červeně) a na kole (zeleně)

4. Závěr

Cílem semestrální práce bylo zejména zpracovat a důkladně se seznámit s problémem síťových analýz a dále pak představit jejich možné využití pro práci s geoprostorovými daty.

Jako největší problém se jevila naše nezkušenost s danou tematikou, práce byla první příležitostí, kdy jsme se se síťovými analýzami setkali a tento fakt se velkou mírou projevil na řešení problémů, které nastaly v průběhu zpracování (hledání funkce pro nalezení průjezdního bodu, ohodnocování hran, sestavování dotazů). V rámci síťových dotazů jsme chtěli aplikovat i úlohu obchodních cestujících (*Traveling Sales Person Problem*), ale nepodařilo se nám najít použitelnou syntaxi.

Tématice síťových analýz bychom se nicméně rádi věnovali i nadále.

Závěrem bychom rádi podotkli, že podobnost se semestrální prací z letního semestru 2012 je čistě náhodná, téma ke zpracování jsme zvolili bez vědomí, že již bylo zpracováno a podobnost síťových dotazů je dána pouze omezenou možností volby, jak síťové analýzy představit.

5. Přílohy

V elektronické podobě bude odevzdáno:

uzpd_2013_a.zip dokumentace.pdf
prezentace.pdf
davka.sql

6. Zdroje

<http://wiki.openstreetmap.org>

<http://workshop.PgRouting.org>

http://geo.fsv.cvut.cz/gwiki/153UZPD_Úvod_do_zpracování_prostorových_dat