Semestrální projekt

J. Klíma, J. Urik, M. Krejčí

Obsah

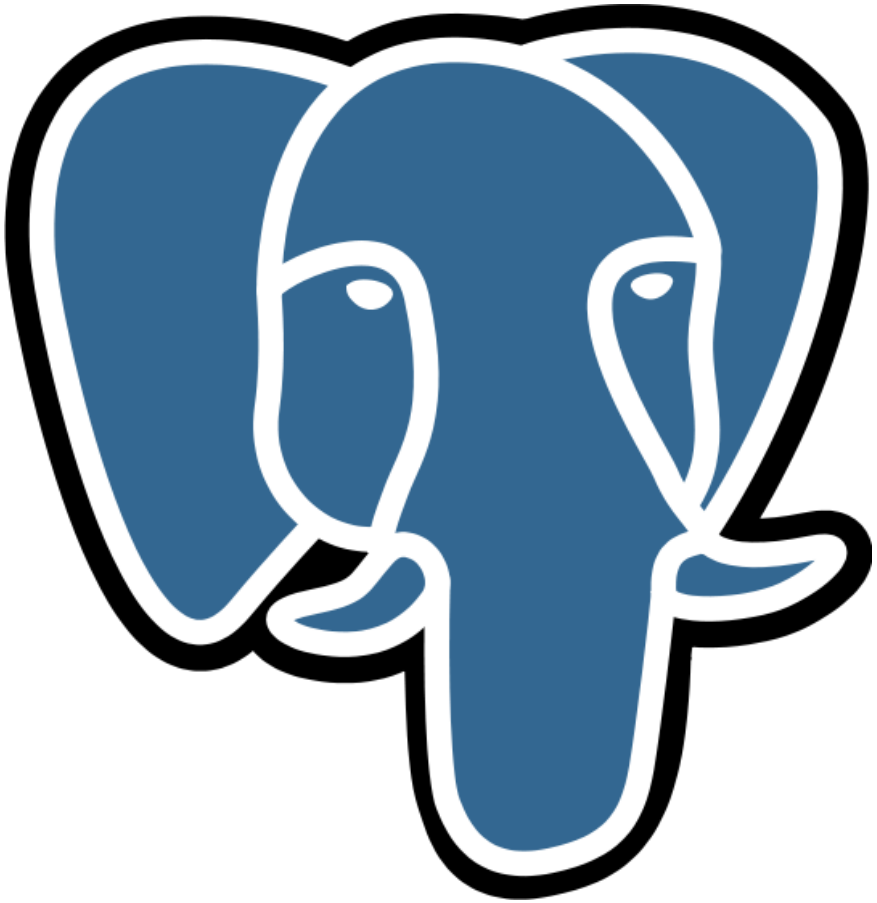
- O projektu
  - Zadání
  - Cíl
- Příprava dat
  - Vytvoření vrstev
  - Úprava dat
  - Příprava pro síťové analýzy
- Dotazy
  - Atributové dotazy
  - Prostorové dotazy
  - Síťové analýzy

Zadání

- Navrhněte a vytvořte tématické vrstvy (např. vodní toky, vodní plochy, lesy, silnice, železnice a pod.) na základě dat OpenStreetMap
- 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**.

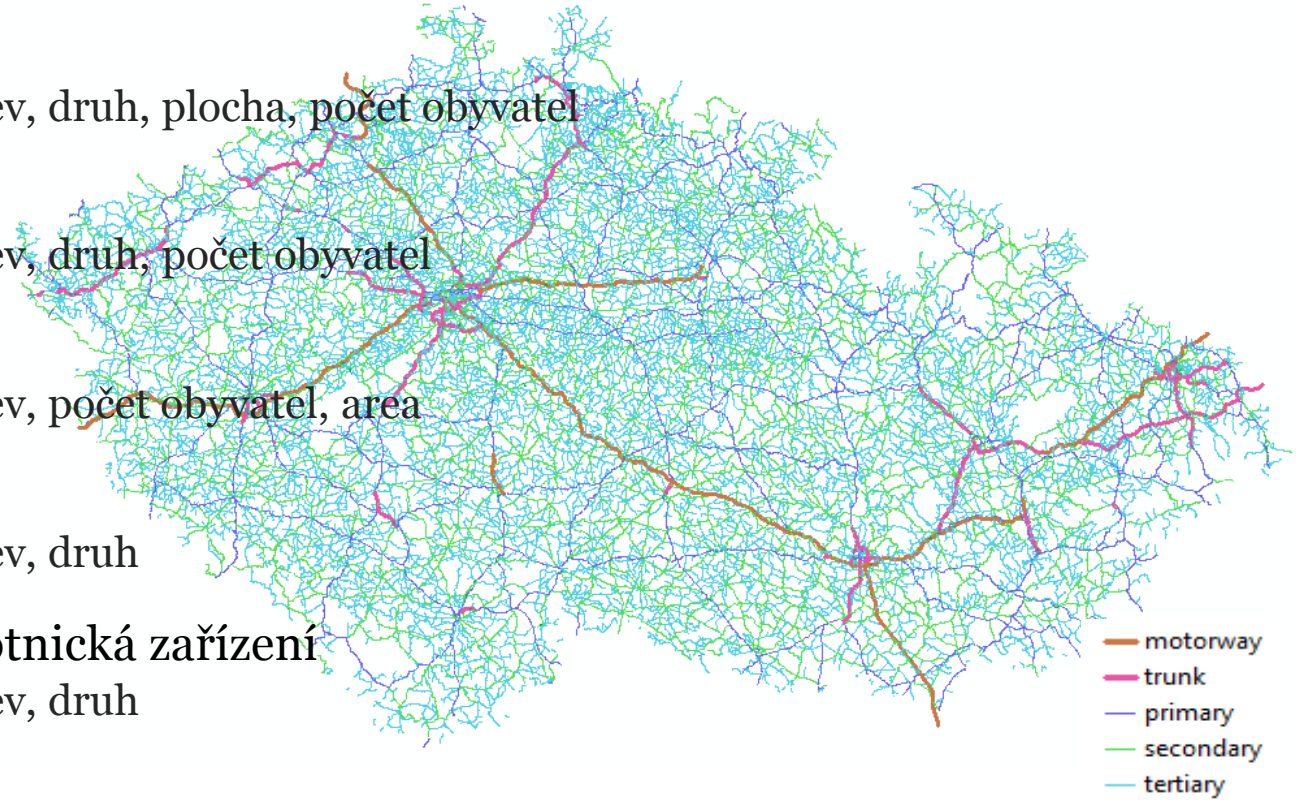
Cíl

- Vytvoření vrstev pro realné analýzy
- Integrita dat
- Tutorial
- Postgres



Vytvoření vrstev

- Benzinové stanice + elektrické stanice
  - Název, druh
- Kraje
  - Název, druh, plocha, počet obyvatel
- Města
  - Název, druh, počet obyvatel
- Obce
  - Název, počet obyvatel, area
- Silnice
  - Název, druh
- Zdravotnická zařízení
  - Název, druh

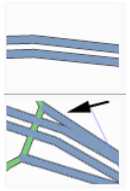


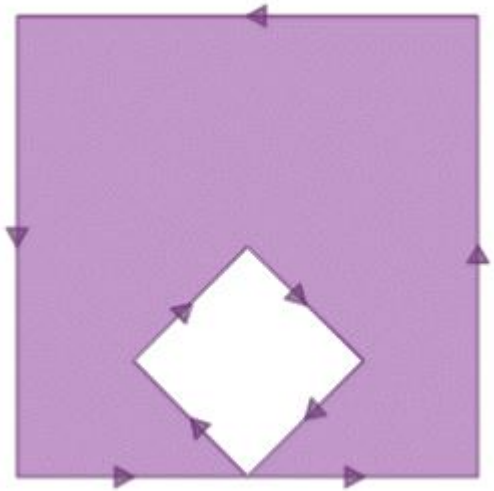
Vytvoření vrstev

```
CREATE TABLE silnice AS
SELECT osm_id, name AS nazev, highway AS druh, geom
FROM czech_line
WHERE highway IN ('motorway', 'trunk','primary',
'secondary', 'tertiary','motorway_link', 'trunk_link',
'primary_link', 'secondary_link', 'tertiary_link');
```

Úprava dat

- Validace vrstev
  - *ST_IsValid()*
  - *ST_MakeValid()*
  - *SELECT Populate_Geometry_Columns*





Funkce ST_MakeValid()

Úprava dat

- `SELECT gid FROM silnice WHERE not st_isvalid(geom);`

- `SELECT Populate_Geometry_Columns('silnice'::regclass);`

Úprava dat

- Odstranění duplicit
  - Jelikož se zdravotnická zařízení nachází jak v polygonové tak v bodové vrstvě OpenStreetMap dochází k duplikaci zdravotnických zařízení. To znamená, že jedno zdravotnické zařízení se nachází v obou vrstvách a proto je nutné tyto duplicity odstranit. Byly odstraněny ty body, které leží v polygonu a mají stejná název i druh.

```
DELETE FROM zdravb
WHERE zdravb.geom IN
(SELECT zdravb.geom
FROM zdravb JOIN zdravp
ON ST_WITHIN(zdravb.geom,zdravp.geom) AND
zdravb.nazev=zdravp.nazev AND zdravb.druh=zdravp.druh);
```

Úprava dat

- Sloučení vrstev
  - Nejdříve bylo nutné převést polygonovou vrstvu zdravotnických zařízení na vrstvu bodovou. To se provede pomocí funkce ***ST_Centroid***, která vrací těžiště polygonových prvků. Následně pomocí funkce *INSERT_INTO* byla takto vzniklá vrstva naimportována do vrstvy zdravb a následně tabulka přejmenována na zdravotnicka_zarizeni.

```
CREATE TABLE zdravp_b AS
SELECT osm_id,nazev,druh, ST_CENTROID(geom) as geom
FROM zdravp;
```

Příprava pro síťové analýzy

- Úprava dat pro PgRouting
  - Přidáni sloupce
    - Source
    - Target
    - Length
  - Vytvořeni topologie
  - Výpočet délky jednotlivých úseků

```
ALTER TABLE silnice ADD COLUMN source INTEGER;
CREATE INDEX source_index ON silnice(source);
SELECT assign_vertex_id('b13','silnice',1,'geom','gid');
UPDATE silnice SET length  = ST_Length(geom);
```

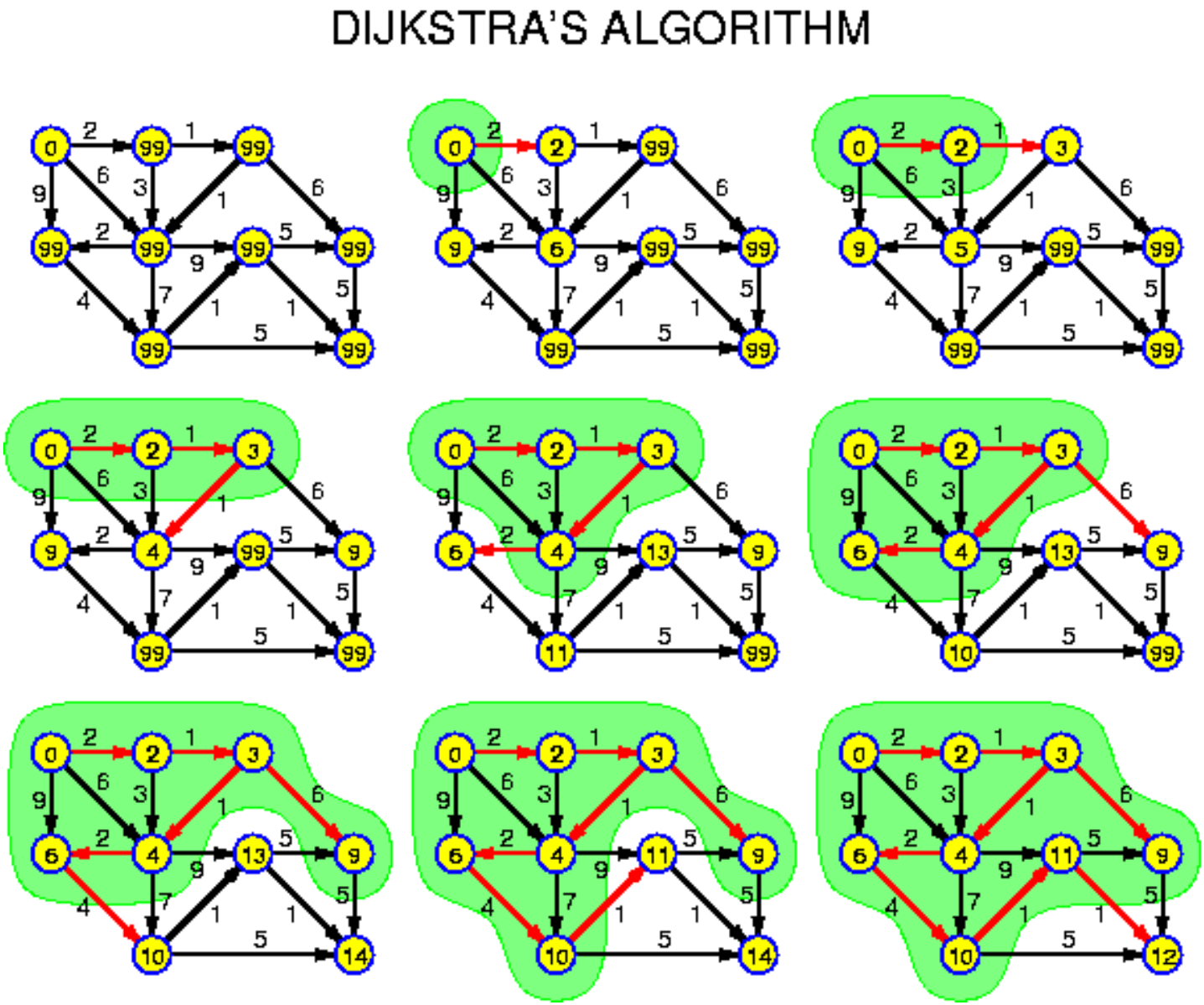
Příprava pro síťové analýzy

- Vyhledání nejkratší cesty
- Dijkstrův algoritmus
  - Jeden z prvních pro PgRouting

```
CREATE OR REPLACE FUNCTION shortest_path(
sql text,
source_id integer,
target_id integer,
directed boolean,
has_reverse_cost boolean                )

      RETURNS SETOF path_result
```

Příprava dat

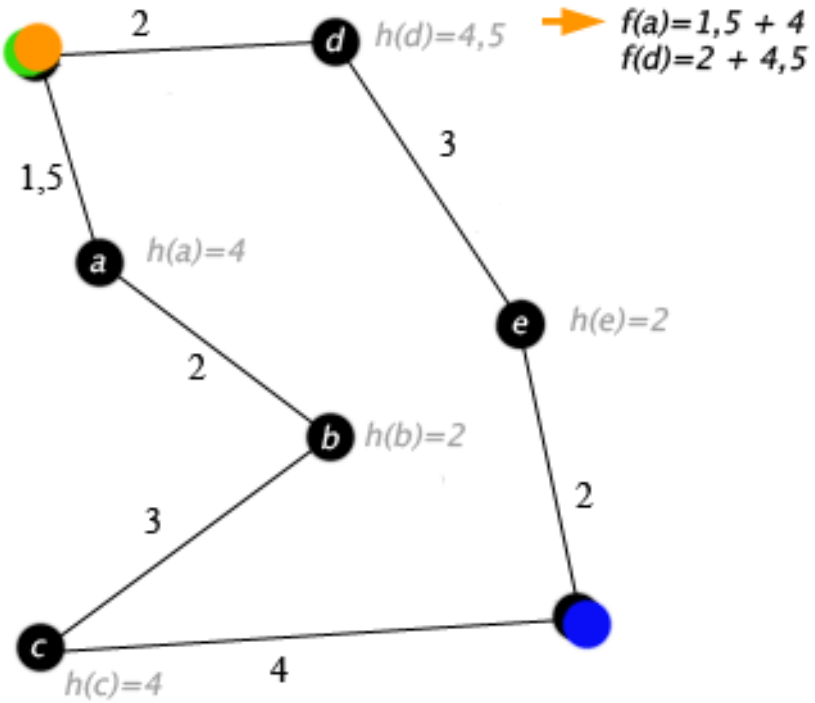


Příprava pro síťové analýzy

- algoritmus A-STAR
  - Pro větší datasety
  - Přidání sloupců pro souřadnice uzlových bodů
  - Výpočet souřadnic uzlových bodů

```
ALTER TABLE silnice ADD COLUMN x1 DOUBLE PRECISION;
ALTER TABLE silnice ADD COLUMN y1 DOUBLE PRECISION;
UPDATE silnice SET x1=ST_x(ST_Startpoint(geom));
UPDATE silnice SET y1=ST_y(ST_Startpoint(geom));
```

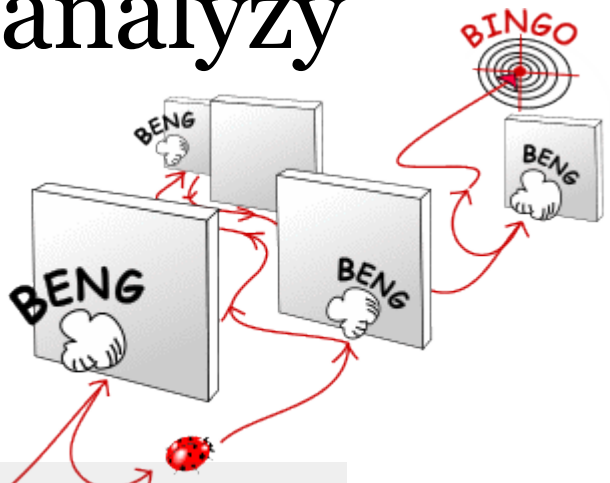
A-STAR algoritmus



Zdroj: wikipedia
Key:
green: start
blue: goal
orange: visited

Příprava pro síťové analýzy

- algoritmus Shooting Star
  - Využití hran a uzlů
  - Heuristická metoda
  - Přidání sloupců rule, to_cost




```
SELECT id, source, target, cost, x1, y1, x2, y2, rule,
to_cost FROM edges
ALTER TABLE silnice ADD COLUMN rule TEXT;
ALTER TABLE silnice ADD COLUMN to_cost DOUBLE PRECISION;
```

Příprava pro síťové analýzy

- Vyhledání uzlů
  - ***find_node_by-nearest_link_within_distance***
  - ***find_nearest_link_within_distance***

```
SELECT id, source, target, cost, x1, y1, x2, y2, rule,
to_cost FROM edges
ALTER TABLE silnice ADD COLUMN rule TEXT;
ALTER TABLE silnice ADD COLUMN to_cost DOUBLE PRECISION;
```

Atributové dotazy

- Kolik zdravotnických zařízení máme v databázi

```
select count(*) from zdravotnicka_zarizeni;
```

87.8307

- Vypište všechny benzínky, jejichž název začíná na "A" a obsahuje písmeno p

```
select nazev from benzinky where nazev like 'A%p%';
```

"Agip","Autosprint Veletiny","Autogas Opava"

- Kolik km silnic máme v databázi

```
select SUM(length)/1000 AS delka FROM silnice;
```

96270.3

- Který kraj má největší rozlohu

```
select * from kraje order by area limit 1;
```

HK

Prostorové dotazy

- Kolik nemocnic má vzdálenou lékárnu do 300 metru

```
select nem.osm_id from (select * from
zdravotnicka_zarizeni where druh = 'lekarna') as lek

join (select * from zdravotnicka_zarizeni where druh =
'nemocnice') as nem

on st_DWithin(nem.geom,lek.geom, 300)group by
nem.osm_id;
```

157 nemocnic

Prostorové dotazy

- Kolik benzinek má v okruhu 50m další benzinku
- `select count(b2.osm_id) from benzinky as b1`
- `join benzinky as b2 on (ST_Dwithin(b1.geom,b2.geom,50))`
- `and ST_Distance(b1.geom,b2.geom)>0;`

110 benzinek

Síťové analýzy

- Kolik zdravotních zařízeni je v okruhu 500m od cesty z prahy do Dolních Jirčan

```
SELECT DISTINCT zz.gid, zz.nazev, zz.druh
FROM (SELECT ST_buffer(geom,500) AS buffer,silnice.gid
FROM shortest_path('SELECT gid AS id,source,target,length AS cost
FROM b13.silnice',
(select id from find_node_by_nearest_link_within_distance((SELECT
ST_AsText(geom) FROM mesta WHERE nazev = 'Praha'),5000,'silnice')),
(select id from find_node_by_nearest_link_within_distance((SELECT
ST_AsText(geom) FROM mesta WHERE nazev = 'Dolní
Jirčany'),5000,'silnice'))
,false,false) AS path
JOIN silnice ON path.edge_id=silnice.gid) AS A
JOIN zdravotnicka_zarizeni as zz ON ST_intersects(A.buffer,zz.geom);
```

výsledek 37 řádků

Síťové analýzy

- Cesta z prahy do Liberce

```
SELECT sum(cost)/1000 FROM shortest_path(
'SELECT gid AS id,source,target,length AS cost FROM
b13.silnice',
(SELECT id FROM
find_node_by_nearest_link_within_distance(
(SELECT ST_AsText(geom) FROM mesta WHERE nazev =
'Praha'),1000,'silnice')),
(SELECT id FROM
find_node_by_nearest_link_within_distance(
(SELECT ST_AsText(geom) FROM mesta WHERE nazev =
'Liberec'),1000,'silnice')),false,false);
```

- *výsledek 168,9*