

Úvod do zpracování prostorových dat - Projekt

Vyhledání přerušení řeky v místě vodní plochy a následná interpolace chybějící části

Skupina b10

Petra Svobodová
Renata Duchnová
Miroslav Kopecký
Tereza Pantůčková

Původní zadání

- vytvoření tématických vrstev pomocí dat z databáze `pgis_osm`
- otestování datové integrity, odstranění nekonzistence v datech
- vytvoření sady atributových a prostorových dotazů

Úkol skupiny

- změna zadání pro neúplnost dat
- vyhledání přerušení řeky v místě vodní plochy a následná interpolace chybějící části
- propojení databáze s python-skriptem

Postup práce

- vytvoření vrstev pomocí dat z databáze `psql_osm`
- vyhledání přerušených řek a příslušných vodních ploch
- propojení s databází
- interpolace chybějící části řeky

Vytvoření potřebných vrstev

- `pgis_osm` → importovaná OpenStreetMap data ČR
- vrstva `b10.reky`
 - `CREATE TABLE b10.reky AS SELECT osm_id, name, way FROM czech_line WHERE waterway = 'river';`
 - `ALTER TABLE b10.reky ADD PRIMARY KEY (osm_id);`
 - `SELECT Populate_geometry_columns('b10.reky'::regclass);`
 - `GRANT SELECT ON b10.reky TO postgis;`

Vytvoření potřebných vrstev

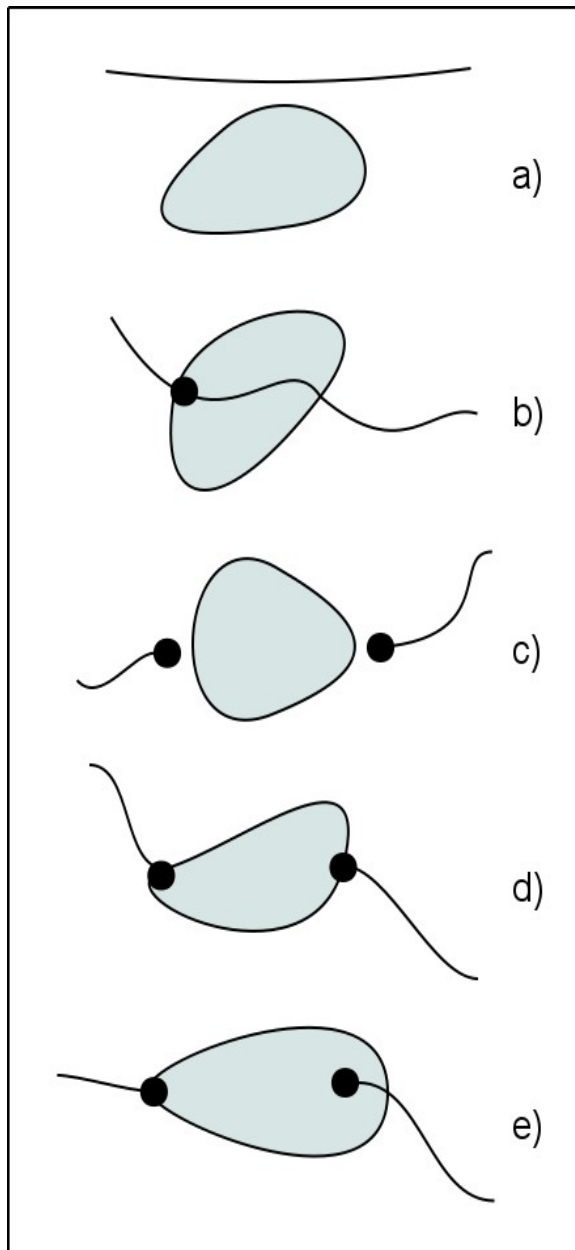
- vrstva b10.vod_plochy_naz
 - CREATE TABLE b10.vod_plochy_naz AS SELECT
osm_id, name, way FROM czech_polygon WHERE
landuse IN ('reservoir','basin') OR
czech_polygon.natural = 'water';
 - ALTER TABLE b10.vod_plochy_naz ADD PRIMARY
KEY (osm_id);
 - SELECT Populate_geometry_columns
('b10.vod_plochy_naz'::regclass);
 - GRANT SELECT ON b10.vod_plochy_naz TO
postgis;

Vyhledání přerušených řek a příslušných vodních ploch

Princip:

- hledání právě dvou koncových bodů řek ve vodní ploše v bufferu 175 m, 350 m a 500 m
- spojení nalezených výsledků do jedné tabulky
- (u více bodů by bylo složité připojení geometrie - např. soutok dvou řek)

Případy vodních ploch a řek



a) řeka plochou neprotéká

b) řeka je přerušena v jednom bodě, ale protéká celou plochou

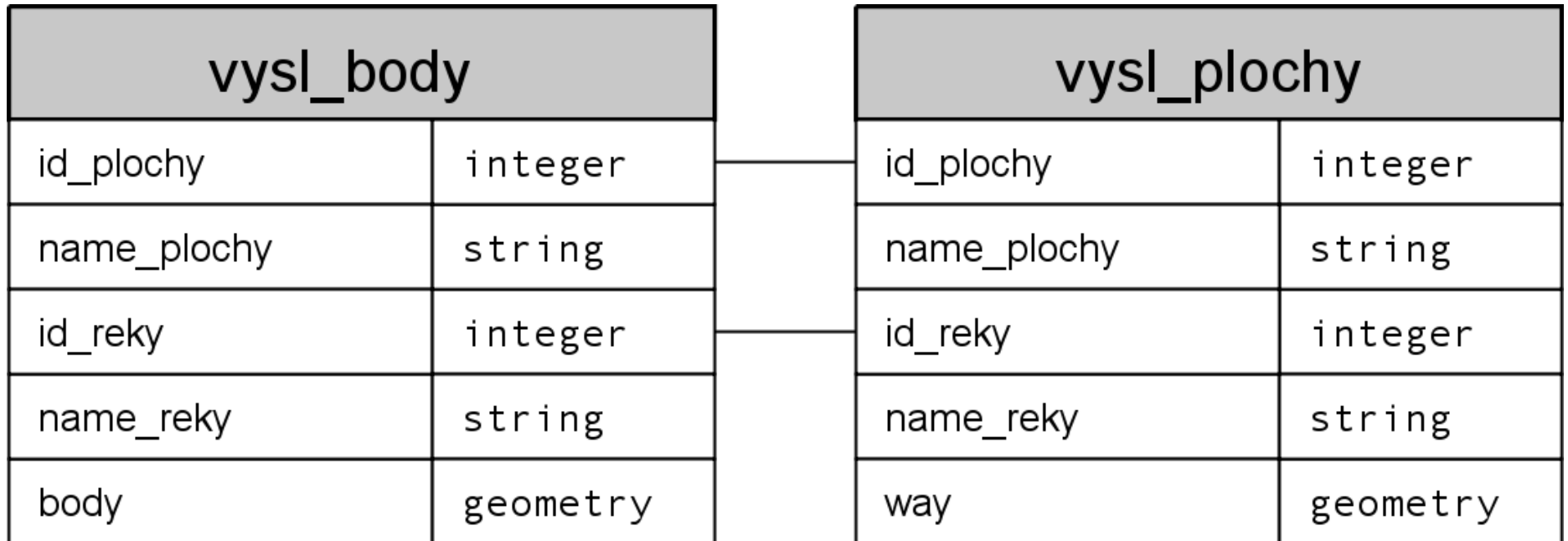
c) řeka je přerušena, žádný její konec neleží na hranici plochy

d) řeka je přerušena, konce řeky leží na hranici plochy

e) řeka je přerušena, první konec leží na hranici plochy, druhý uvnitř plochy

c), d), e) jsou hledané případy

Tabulky



- použité tabulky spojené přes sloupce `id_plochy` a `id_reky`

Propojení s databází

- provádí skript `prehrady.py`
- připojení pomocí modulu `psycopg2`
- zjištění z tabulek hodnoty pro řeku a plochu
- zavolání funkce `najdi_cestu` z modulu `uloha.py`

Interpolace chybějící části řeky

- provádí skript `uloha.py`
 - funkce `najdi_cestu`
 - dva parametry:
 - dva koncové body rozdělené řeky
 - body reprezentující polygon vodních ploch
 - rozdělení polygonu na trojúhelníky
 - algoritmus na triangulaci byl použit z <http://www.siggraph.org>
 - pro každý trojúhelník vypočteno těžiště
 - nalezení nejkratší cesty přes sousední těžiště mezi jednotlivými konci řeky
 - klasický Dijkstruv algoritmus implementovaný v modulu `dijkstra.py` (převzetí <http://aspn.activestate.com>)
 - výsledná cesta aproximuje průchod řeky skrz vodní plochu

Co je ještě potřeba ...

- vyhledané body začlenit do správného místa přerušené řeky v rámci databáze
- propojení těchto bodů linií

Děkuji za pozornost

Prostor pro dotazy