

Západočeská univerzita v Plzni

Fakulta aplikovaných věd

Katedra matematiky

Bakalářská práce

**Automatizovaná validace
prostorových dat informačního
systému Heracleum**

Originál zadání práce je vložen do tištěné podoby bakalářské práce.

Prohlášení

Předkládám tímto k posouzení a následné obhajobě bakalářskou práci zpracovanou na závěr bakalářského studia na Fakultě aplikovaných věd Západočeské univerzity v Plzni. Prohlašuji, že jsem práci vypracovala samostatně pod vedením vedoucího bakalářské práce a uvedla jsem veškeré použité zdroje.

V Plzni dne 4.6.2014

Poděkování

Na tomto místě bych ráda poděkovala vedoucímu bakalářské práce panu Ing. Karlu Janečkovi, Ph.D. za téma práce, odborné vedení celé práce, inspirativní rady a připomínky. Dále bych chtěla poděkovat Agentuře projektového a dotačního managementu Karlovarského kraje a to především paní RNDr. Lence Pocové, díky níž mi byla půjčena data, s kterými pracuji v bakalářské práci.

Abstrakt

Tato práce je věnována problematice topologické čistoty prostorových dat informačního systému Heracleum. Nejdříve popisuje informační systém Heracleum a data použitá pro tuto práci. Dále vysvětluje pojem topologie a následně pomocí dvou open source programů kontroluje topologickou čistotu dat. Těžištěm práce jsou dva vytvořené manuály pro kontrolu topologie dat.

Klíčová slova

Topologie, validace dat, IS Heracleum, QGIS, PostGIS

Abstract

This thesis is devoted to the problematics of the topological cleanliness of data of the information system Heracleum. At first the thesis describes the information system Heracleum and the data that I have used for writing this thesis. Then, the thesis explains the concept of the topology and it analyses the topological cleanliness of the data by using two open source programs. The main content of this thesis are two created manuals for the control of the topology of the data.

Key words

Topology, validation of data, IS Heracleum, QGIS, PostGIS

Obsah

1. Úvod.....	10
2. Projekt omezení výskytu invazních rostlin v Karlovarském kraji	11
2.1 O agentuře.....	12
2.2 O projektu	12
2.3 Cíl projektu	12
2.4 Harmonogram projektu.....	12
2.5 Činnosti plánované po ukončení projektu	13
3. IS Heracleum.....	13
4. Prostorová data v IS Heracleum.....	16
4.1 Získání prostorových dat	16
4.2 Vymezení prostorových dat.....	17
5. Požadavky na topologickou kvalitu dat	17
5.1 Pojem topologie.....	17
5.2 Vybrané definice topologie	17
5.3 Práce zabývající se topologií a topologicky čistými daty.....	18
6. Topologické chyby v prostorových datech	19
7. QGIS.....	24
7.1 Topologická pravidla.....	24
7.2 Manuál pro kontrolu topologie	26
7.3 Výsledek ověření topologie	37
8. PostGIS.....	43
8.1 Kontrola topologie.....	43
8.2 Manuál pro kontrolu topologie	45
8.3 Výsledek ověření topologie	56
9. Porovnání	58
10. Vybrané pravidla a algoritmy pro čištění prostorových dat	60

10.1 QGIS	60
10.2 PostGIS	60
11. Závěr.....	63
12. Seznam použitých zdrojů	64

Seznam použitých zkratk

EU	Evropská unie
GIS	Geografický informační systém
GINA	Geografický informační asistent
GPS	Globální polohový systém
IS Heracleum	Informační systém Heracleum verze 1.5
QGIS	Quantum GIS verze 2.2 Valmiera

Seznam obrázků

- Obr. 2.1: Bolševník velkolepý
- Obr. 2.2: Netýkavka žláznatá
- Obr. 2.3: Křídlatka česká
- Obr. 3.1: Vzhled IS Heracleum
- Obr. 3.2: Ukázka soupisu lokalit a jejich podrobných informací o nich v IS Heracleum
- Obr. 3.3: Ukázka soupisu parcel a jejich podrobných informací v IS Heracleum
- Obr. 3.4: Mapový výstup v IS Heracleum; konkrétně zde je zobrazena topografická mapa a hranice mapovacích úseků
- Obr. 3.5: Mapový výstup v IS Heracleum; konkrétně zde je zobrazena topografická mapa a související projekty – zmapování úseku Kosí potok v letech 2009 a 2010
- Obr. 6.1: Překrývající se objekty – ukázka č. 1
- Obr. 6.2: Překrývající se objekty – ukázka č. 1 – zvýraznění
- Obr. 6.3: Překrývající se objekty – ukázka č. 2
- Obr. 6.4: Překrývající se objekty – ukázka č. 2 – zvýraznění
- Obr. 6.5: Překrývající se objekty – ukázka č. 3
- Obr. 6.6: Překrývající se objekty – ukázka č. 3 – zvýraznění
- Obr. 6.7: Objekt mající mezeru – ukázka č. 1
- Obr. 6.8: Objekt mající mezeru – ukázka č. 2
- Obr. 6.9: Objekt mající mezeru – ukázka č. 3
- Obr. 6.10: Objekt mající mezeru – ukázka č. 3 – detail
- Obr. 6.11: Duplicitní objekt – ukázka č. 1
- Obr. 6.12: Duplicitní objekt – ukázka č. 2
- Obr. 6.13: Duplicitní objekt – ukázka č. 3
- Obr. 7.2.1: Otevření filtru
- Obr. 7.2.2: Filtrace prvního druhu rostlin
- Obr. 7.2.3: Otevření nástroje pro kontrolu topologie
- Obr. 7.2.4: Nastavení topologických pravidel – výběr vrstvy
- Obr. 7.2.5: Nastavení topologických pravidel – výběr pravidla
- Obr. 7.2.6: Nastavení topologických pravidel – potvrzení nastavení topologických pravidel
- Obr. 7.2.7: Ověření topologických pravidel
- Obr. 7.2.8: Zobrazení topologické chyby
- Obr. 7.2.9: Filtrace následujícího druhu rostlin
- Obr. 7.2.10: Ověření topologických pravidel následujícího druhu rostlin
- Obr. 7.2.11: Filtrace posledního druhu rostlin
- Obr. 7.2.12: Ověření topologických pravidel posledního druhu rostlin
- Obr. 7.3.1: Překrývající se objekty – ukázka č. 1 - kontrola pomocí QGIS
- Obr. 7.3.2: Překrývající se objekty – ukázka č. 2 - kontrola pomocí QGIS
- Obr. 7.3.3: Překrývající se objekty – ukázka č. 3 - kontrola pomocí QGIS
- Obr. 7.3.4: Objekt mající mezeru – ukázka č. 1- kontrola pomocí QGIS
- Obr. 7.3.5: Objekt mající mezeru – ukázka č. 2- kontrola pomocí QGIS

Obr. 7.3.6: Objekt mající mezeru – ukázka č. 3 - detail- kontrola pomocí QGIS
Obr. 7.3.7: Duplicitní objekt – ukázka č. 1- kontrola pomocí QGIS
Obr. 7.3.8: Duplicitní objekt – ukázka č. 2- kontrola pomocí QGIS
Obr. 7.3.9: Duplicitní objekt – ukázka č. 3- kontrola pomocí QGIS
Obr. 7.3.10: Nalezená chyba „díry“
Obr. 8.2.1: Otevření importu shapefile
Obr. 8.2.2: Připojení k databázi
Obr. 8.2.3: Nastavení importu
Obr. 8.2.4: Vybrání shapefile
Obr. 8.2.5: Import dat do PostGIS
Obr. 8.2.6: Spuštění editoru SQL v pgAdmin III
Obr. 8.2.7: Spuštěný editor SQL
Obr. 8.2.8: Vložení funkce
Obr. 8.2.9: Vytvoření funkce
Obr. 8.2.10: Vložení a spuštění příkazu
Obr. 8.2.11: Nalezení chyb
Obr. 8.3.1: Datový výstup 1
Obr. 8.3.2: Datový výstup 2

Seznam tabulek

Tab. 7.3.1: Souhrn nalezených topologických chyb
Tab. 8.3.1: Souhrn nalezených topologických chyb
Tab. 8.3.2: Souhrn skutečného počtu topologických chyb
Tab. 9.1: Porovnání počtu topologických chyb

1. Úvod

V roce 2012 jsem se účastnila Projektu omezení výskytu invazních rostlin v Karlovarském kraji. V terénu jsem hledala tři druhy invazních rostlin. Zmapovala jsem část území Žluticko a některé pozemky patřící Sokolovské uhelné a.s. Jelikož byla tato data zmapována a digitalizována většinou biology, data obsahují chyby. Pro správnou analýzu dat je důležité mít topologicky čistá data. Proto se moje bakalářská práce se zabývá topologickou čistotou dat.

Cílem bakalářské práce je navrhnout postup validace prostorových dat pomocí dvou volně dostupných nástrojů, jimiž jsou program QGIS a nástroj PostGIS, což je rozšíření databázového systému PostgreSQL. Cílem je vytvořit manuál pro pracovníky IS Heracleum, aby si byli sami schopni ověřit topologickou čistotu dat. Validací prostorových dat je zde míněna topologická čistota dat. Dále chci touto prací přiblížit téma topologie a její důležitost v datech.

Na začátku bakalářské práce je popsán Projekt omezení výskytu invazních rostlin v Karlovarském kraji, jelikož data použitá v bakalářské práci byla získána během realizace toho projektu.

Ve 3. a 4. kapitole se zabývám IS Heracleum, který zajišťuje informační podporu již zmíněného projektu. Hlavním cílem těchto kapitol je seznámit se s IS Heracleum a prostorovými daty, které používá. IS Heracleum sice umí pracovat s body, liniemi a polygony, avšak vzhledem k tomu, že v předpisu zakázky bylo definováno, aby výsledný výstup obsahoval prostorové útvary, tak před odevzdáním zakázky byly body a linie převedeny na polygony, proto se v práci zabývám pouze topologickými pravidly pro polygony.

V kapitole 5 vymezuji pojem topologie a zmiňuji se o několika vědeckých článcích, které se touto problematikou zabývají.

Následně v další kapitole poukazuji na typy topologických chyb, které jsem našla v datech a u kterých poté kontroluji, zda program správně ověřil topologickou čistotu dat.

V 7. kapitole popisuji vybrané algoritmy pro čištění topologických dat.

V kapitole 8 se zabývám vymezením topologických chyb a kontrolou topologie v programu QGIS. V kapitole 9 řeším topologii pomocí nástroje PostGIS, napsala jsem funkci pro kontrolu daných chyb. V obou kapitolách je vytvořen manuál pro kontrolu dat a je vypsán počet nalezených chyb. V následující kapitole porovnávám výsledky dat a zdůvodňuji, proč je počet nalezených chyb odlišný.

2. Projekt omezení výskytu invazních rostlin v Karlovarském kraji

Tento projekt vznikl za účelem najít způsob, jak potlačit téměř kalamitní výskyt invazních rostlin, hlavně:

- bolševníku velkolepého (*Heracleum mantegazzianum*, viz obr. 2.1),
- netýkavky žláznaté (*Impatiens glandulifera*, viz obr. 2.2) a
- tři druhy křídlatky - křídlatka japonská (*Reynoutria japonica*), sachalinská (*Reynoutria sachalinensis*) a česká (*Reynoutria bohemica*, viz obr. 2.3).



Obr. 2.1: Bolševník velkolepý¹



Obr. 2.2: Netýkavka žláznatá²



Obr. 2.3: Křídlatka česká³

Mapování se provedlo na celém území Karlovarského kraje vyjma výcvikového prostoru Hradiště a ploch, kde již probíhaly projekty likvidace bolševníku (povodí Kosího potoka, říčky Tiché a Hutského potoka).

„Projekt je financován především díky dotaci z Operačního programu Životního prostředí⁴, osa 6.2 - Ochrana biodiverzity, dále financování zajistil i Karlovarský kraj⁵. Na mapování oblasti kraje přispělo finančně 14 obcí, významně přispěl i státní podnik Povodí Ohře s.p.⁶ „[1].“

¹ <http://botanika.wendys.cz/kytky/foto.php?719>:

² <http://botanika.wendys.cz/kytky/foto.php?152:2>

³ <http://botany.cz/cs/reynoutria-bohemica/>

⁴ <http://www.opzp.cz/>

⁵ <http://www.kr-karlovarsky.cz/Stranky/Default.aspx>

⁶ <http://www.poh.cz/>

a Sokolovská uhelná a.s.⁷

Příspěvkovou organizací zřízenou pro administraci a řízení projektů je Agentura projektového a dotačního managementu Karlovarského kraje.

2.1 O agentuře

Agentura projektového a dotačního managementu Karlovarského kraje, p.o.⁸ vznikla v roce 2006 jako příspěvková organizace kraje, jejíž hlavní činností je příprava, realizace a monitorování projektů Karlovarského kraje[1]. Tyto projekty byly především financovány ze zdrojů EU v naplánovaných letech 2007 až 2013. Projekt omezení výskytu invazních rostlin stále pokračuje a je nadále financován.

„Agentura se dále zabývá konzultační a poradenskou činností, tvorbou rozvojových dokumentů, analýz, strategií, koncepcí, programů, plánů a projektů a publikační a propagační činností.“[5]

2.2 O projektu

„Karlovarský kraj je, pokud jde o invazní rostliny, v rámci České republiky nejvíce zasaženým regionem“ [1]. Z invazních rostlin převažují především bolševník velkolepý, všechny tři druhy křídlatek a netýkavka žláznatá. „Tyto rostliny se v přírodě velice rychle rozrůstají a zároveň tím vytlačují původní druhy rostlin a snižují tak biodiverzitu, nemluvě o tom, že bolševník velkolepý představuje pro obyvatele a návštěvníky kraje zdravotní riziko. Křídlatky a netýkavka se rozrůstají hlavně na břehách vodních toků, způsobují tak neprůchodnost krajiny a erodují půdu. Kraj proto připravil ve spolupráci s dalšími subjekty tento projekt likvidace vybraných druhů invazních rostlin. Projekt vychází z již uskutečněného pilotního projektu Omezení výskytu invazních rostlin v povodí Kosího potoka, který probíhal v letech 2010 až 2013.“ [1]

2.3 Cíl projektu

Cílem projektu je omezení výskytu invazních rostlin - bolševníku velkolepého, křídlatek a netýkavky žláznaté. V době, kdy byl projekt na svém začátku, tedy v roce 2011, bylo zamořeno odhadem 3 až 5 % území Karlovarského kraje.

Řešená plocha: 2800 km²

Odhadované celkové náklady: cca 170 mil. Kč [1]

2.4 Harmonogram projektu

Harmonogram byl rozdělen na 3 následující fáze:

1) Přípravná fáze: říjen 2011 – září 2012

„Přípravná fáze zahrnovala kromě jiného podrobné mapování výskytu všech tří druhů rostlin, zjištění vlastnických vztahů k zasaženým pozemkům a získání souhlasů s realizací projektu. K řízení a kontrole projektu byl vytvořen geografický informační systém, který umožňuje mj. zjistit veškerá omezení daná ochranou vod, ochranou přírody, zařazením v režimu ekologického zemědělství a dalšími omezeními a zohledňuje je při výběru vhodných metodik likvidace. Na financování přípravné fáze přislíbily svojí účast také některé obce, města a firmy v Karlovarském kraji.“ [1]

2) Podání žádosti o dotaci: říjen – listopad 2012

„K úspěšnému podání žádosti bylo nutné dodat podrobně zpracovanou projektovou dokumentaci s metodikami likvidace a také získat souhlasy všech vlastníků a uživatelů

⁷ <http://www.suas.cz/>

⁸ <http://apdm.cz/>

zasažených pozemků a vyjádření všech dotčených orgánů.“ [1]

3) Vlastní realizace prací: jaro 2013 – listopad 2015

„Žádost o dotaci byla úspěšná a po obdržení akceptačního listu byla zahájena výběrová řízení na dodavatele prací s likvidací. Likvidace začala po získání Rozhodnutí o poskytnutí dotace a uzavření Smlouvy o dotaci. Likvidační práce budou pokračovat až do roku 2015. Během vlastní likvidace se také koná monitoring, kontrola a publikační činnost. Během projektu se uskutečnilo a ještě uskuteční několik workshopů o zkušenostech s likvidací invazních rostlin. Již na počátku projektu se scházela odborná konzultační skupina, která diskutovala o řadě otázek týkajících se projektu“ . [1]

2.5 Činnosti plánované po ukončení projektu

Až bude projekt ukončen, bude osud zasažených oblastí opět záležet na vlastnících pozemků, kteří mají ze zákona povinnost zabránit dalšímu šíření těchto rostlin. „Kontrolu budou v rámci udržitelnosti projektu provádět kromě zástupců žadatele také k tomu kompetentní úřady a prostřednictvím webových stránek i veřejnost“ [1].

3. IS Heracleum

IS Heracleum je webová aplikace⁹ (viz. obr. 3.1), která byla vyvinuta pro informační podporu všem, kteří jí potřebují během činnosti projektu. Aplikace není dostupná pro širokou veřejnost, je určena pouze pro veškeré pracovníky podílející se na jakékoli části projektu.

Lokality	
Celkem	13 837
Bolševník	7 769
Křídlatka	3 316
Netýkavka	2 752
Výměra	61 879 702 m ²
Bolševník	49 921 975 m ²
Křídlatka	4 436 162 m ²
Netýkavka	7 521 565 m ²

Parcely	
Celkem	24 210
Průnik	48 753
Metodika likvidace	
sečení nebo pastva	9 610
vyrývání	3 780
vytrhávání	1 789
plošný postřik	492
bodový postřik	18 596
nátěr	14 486
seseknutí a nátěr	0

Osoby a organizace	
Celkem	4 597
Právnícké osoby	641
Fyzické osoby	3 956
Vyjádření	
Souhlas	9 374
Částečný souhlas	3 778
Nesouhlas	0
Nevyjádřeno	675
Bez vlastníka	10
Souhlas	18 092 258 m ²
Částečný souhlas	42 953 009 m ²
Nesouhlas	0 m ²
Nevyjádřeno	833 877 m ²
Bez vlastníka	558 m ²

Obr. 3.1: Vzhled IS Heracleum k 20.4.2014 [3]

Součástí IS Heracleum je databáze, která obsahuje již zmapovaný výskyt invazních rostlin a zároveň další důležité informace, např. druh, pokryvnost nebo nebezpečnost rostliny (viz. obr. 3.2).

⁹ <http://gis.kr-karlovarsky.cz/heracleum/Login.aspx?ReturnUrl=%2fheracleum%2fLokality%2fLokality.aspx>

Číslo	Úsek	Část	Rostlina	Typ	Plocha	Habi...	Pokr...	Nebe...	Man...	Kont...	Map...	P	Poč...	Čelk...
35	V	14	Křídlatka	Polygon	32	Undefined	3	4	Ano	Nekontrol...	73400645... fot...	1	1	1
48	V	14	Křídlatka	Bod	5	Louka	3	2	Ne	Nekontrol...	73400645... lo...	1	1	1
54	V	14	Křídlatka	Linie	22	Undefined	2	3	Ne	Nekontrol...	73400645... po...	2	3	2
55	V	14	Křídlatka	Polygon	18	Undefined	4	3	Ne	Nekontrol...	73400645... na...	2	3	2
56	V	14	Křídlatka	Linie	15	Ruderal	3	3	Ano	Nekontrol...	73400645... ru...	1	2	1
58	V	14	Křídlatka	Polygon	94	Ruderal	3	3	Ne	Nekontrol...	73400645... ru...	2	2	2
59	V	14	Křídlatka	Bod	5	Ruderal	2	3	Ano	Nekontrol...	73400645... ru...	1	2	1
60	V	14	Křídlatka	Bod	5	Louka	2	2	Ne	Nekontrol...	73400645... lo...	1	1	1
61	V	14	Křídlatka	Polygon	57	Louka	4	4	Ne	Nekontrol...	73400645... lo...	1	1	1
62	V	14	Křídlatka	Linie	75	Louka	2	3	Ne	Nekontrol...	73400645... lo...	2	2	2

Obr. 3.2: Ukázka soupisu lokalit a jejich podrobných informací o nich v IS Heracleum [3]

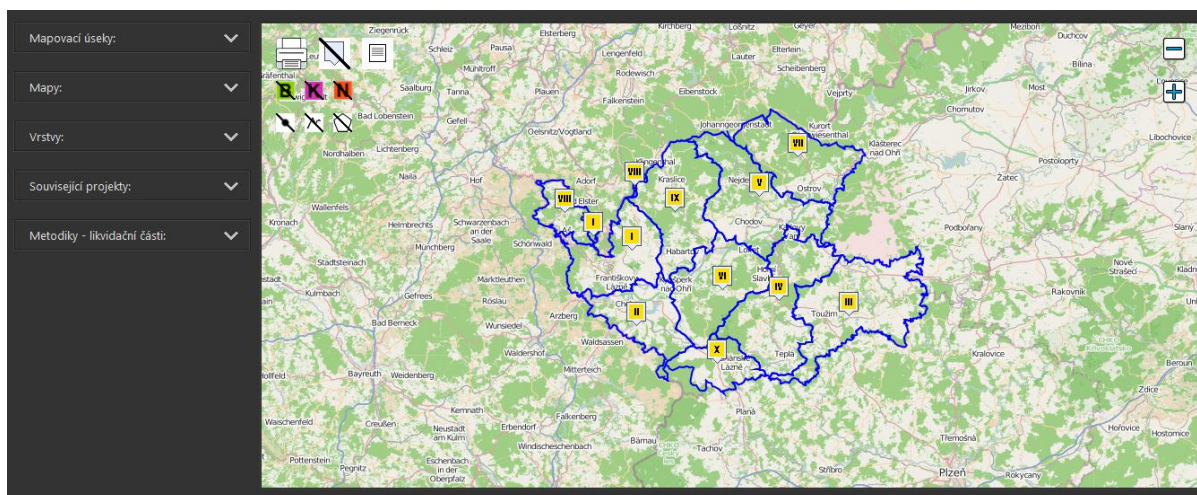
Aplikace poskytuje automatizaci celého procesu projektu, což znamená, že tato aplikace umí na základě zákresu výskytu rostliny zjistit údaje o zasaženém území (viz. obr. 3.3), např. rozměr a plochu lokality, příslušnost k ochranným pásmům a katastrální data zasažených parcel včetně adres vlastníků [15].

Číslo	Číslo...	K.ú.	Vým...	Průnik	Druh	CHKO	EKO	Evl	Pto	Meto...	Výji...	Poč...	Č.(n...	Úpln...	Id
.1	679635	Zlatá u K.	89	36	zastavěn...	0	Ne			nátěr	Ne	1	Ano	Ano	512781409
.1	672246	Dolína u ...	124	125	zastavěn...	0	Ne			nátěr	Ne	1	Ano	Ano	494331409
.1/1	676641	Nová Ves...	209	1	zastavěn...	0	Ne			nátěr	Ne	1	Ano	Ano	14441844...
.1/1	766402	Teplička	490	73	zastavěn...	3	Ne			bodový p...	Ne	1	Ano	Ano	858379403
.1/1	707696	Hřízn u N...	13 480	423	zastavěn...	0	Ne			bodový p...	Ne	1	Ne	Ne	13746424...
.1/1	707686	Hřízn u N...	13 480	10	zastavěn...	0	Ne			nátěr	Ne	1	Ne	Ne	13746424...
.1/1	707686	Hřízn u N...	13 480	41	zastavěn...	0	Ne			nátěr	Ne	1	Ne	Ne	13746424...
.1/1	728699	Popovice	619	4	zastavěn...	3	Ne			vyřívání	Ano	2	Ano	Ne	820343403
.1/1	631043	Cihelny	8 631	117	zastavěn...	3	Ne			bodový p...	Ne	1	Ano	Ano	726097403
.1/1	716642	Brť	4 825	719	zastavěn...	4	Ne			bodový p...	Ne	4	Ano	Ano	10575844...

Obr. 3.3: Ukázka soupisu parcel a jejich podrobných informací v IS Heracleum [3]

Dále po vytvoření informačních dopisů, které jsou zaslány vlastníkům zasažených území, IS Heracleum sleduje souhlas nebo nesouhlas s likvidací invazních rostlin. „Dále připravuje ekonomické a geografické podklady k likvidaci rostlin, navrhne určitou metodiku likvidace, předpoví odhad výše nákladů a sleduje fakturace prací. Umožňuje vyhodnotit libovolná statistická data a jejich vztahy, generuje výstupy pro zprávy o řešení projektu“. [15]

IS Heracleum má též i mapový výstup, kde lze interaktivně sledovat data. Na mapovém podkladě můžeme pozorovat jak zákresy výskytu invazních rostlin, tak i například hranice mapovacích úseků (viz. obr. 3.4), hranice jednotlivých území či hranice jednotlivých ochranných zón. Vše záleží na konkrétním výběru dostupných možností.



Obr. 3.4: Mapový výstup v IS Heracleum; konkrétně zde je zobrazena topografická mapa a hranice mapovacích úseků [3]

Na výběr je 10 mapových úseků, u kterých je možné vybrat více úseků naráz, po jejich zvolení se zobrazí všechny objekty, které byly ve vybraném úseku zmapovány. Kliknutím na zobrazený objekt lze rychle získat údaje o objektu, např. jeho fotografii, druh rostliny apod.

Dostupné mapové úseky jsou:

- Chebsko - sever,
- Chebsko - jih,
- Žluticko,
- Teplá,
- Karlovarsko-nejdecko,
- Sokolovsko - jih,
- Ostravsko,
- Sála,
- Kraslicko a
- Mariánsko-lázeňsko

Dále je na výběr 5 typů podkladových map, můžeme využít i jejich kombinaci:

- topografická,
- turistická,
- ortofoto ČUZK,
- katastr kraje a
- katastr ČUZK

Následně můžeme vybírat různé vrstvy dat, které zobrazují hranice vybraného území:

- mapovací úseky,
- části likvidace,
- ochrana přírody
 - CHKO kraj
 - přírodní park
 - mokřady Ramsarské úmluvy
 - ...
- Ekozemědělství – LPIS
- ...

Další možností je zobrazení souvisejících projektů v letech 2009 nebo 2011 (viz. obr. 3.5):

- Kosí potok
- Sekersko



Obr. 3.5: Mapový výstup v IS Heracleum; konkrétně zde je zobrazena topografická mapa a související projekty – zmapování úseku Kosí potok v letech 2009 a 2010 [3]

Poslední možností k zobrazení jsou metodiky – likvidační části, což je zobrazené území, kde již byla zahájena likvidace invazních rostlin, je zde na výběr z 29 oblastí, mezi něž patří např.:

- Žluticko
- Chebsko – sever 1
- Chebsko – sever 2
- Chebsko – sever 3
- Chebsko – sever 4
- Chebsko – jih

Mapová komponenta je také dostupná pro širokou veřejnost, avšak s omezenou funkcí, není zde veškerá přístupnost k informacím o pozemcích, majitelích a podobně. Portál projektu pro veřejnost je dostupný na <http://gis.kr-karlovarsky.cz/heracleum-public/Web/Mapa.aspx>

4. Prostorová data v IS Heracleum

Prostorová data byla získána mapováním terénu. Já jsem zmapovala část území Žluticko a některé pozemky patřící Sokolovské uhelné a.s., při mapování jsem používala GPS technologii. Abych data netestovala pouze na několika desítkách polygonech, které jsem zmapovala, požádala jsem si o veškerá zmapovaná data, která mne byla následně propůjčena, za což děkuji Agentuře projektového a dotačního managementu Karlovarského kraje.

4.1 Získání prostorových dat

Mapování probíhalo pochůzkou přímo v terénu. Prostorová data byla získána dvěma způsoby:

- první způsob získání dat byl kvalifikovaný zákres např. do ortofotomapy, poté následovala jejich digitalizace,
- druhý způsob sběru dat byl pomocí GPS technologií, kde byl využit systém GINA¹⁰.

Následně se všechna data zpracovala v IS Heracleum, výsledky je možné si prohlédnout na <http://gis.kr-karlovarsky.cz/heracleum-public/Web/Mapa.aspx>

4.2 Vymezení prostorových dat

Objekty, které je možné nad mapovým podkladem IS Heracleum sledovat a vyhodnocovat, jsou body, linie a uzavřené polygony. K těmto objektům existuje řada dalších dat a metadat, např. vlastnictví parcel, charakter území apod. Tato data jsou buď přímo uložena v systému nebo jsou propojena s dalšími informačními systémy. Data, která jsou přímo uložena v systému, jsou taková, která umí systém sám vytvořit, např. umí spočítat plochy průniků zákresů s plochami parcel. Příkladem druhé možnosti je propojení s digitální katastrální mapou Karlovarského kraje a katastrem nemovitostí, odkud jsou získávána aktuální data o vlastnících dotčených parcel, jejich adresy, formy vlastnictví atd.

Vzhledem k tomu, že v předpisu zakázky bylo definováno, aby výsledný výstup obsahoval prostorové útvary, tak před odevzdáním zakázky byly body a linie převedeny na polygony. Bod je aproximací kruhu o ploše 5m² a linie má šířku 1m. Proto všechna data v shapefile, u kterých testují topologická pravidla, jsou polygony.

5. Požadavky na topologickou kvalitu dat

Během plnění IS Heracleum reálnými a naměřenými daty došlo k různým nesrovnalostem a chybám, které byly způsobeny nepřesností nebo nevědomostí pracovníků. Topologie je užitečný aspekt vektorových datových vrstev, protože tyto chyby, jako např. překrývání objektů nebo mezery mezi objekty, minimalizuje.

5.1 Pojem topologie

Topologie (slovo topologie pochází z řečtiny: topos = místo, logos = studie) - je vyjádření pro to, jak bod, linie a polygon sdílejí geometrii¹¹.

Topologie studuje sousednost, konektivitu a omezení objektů. Pomocí topologie můžeme prozkoumat strukturu a tvar objektu.

Topologické chyby představují stav, kdy vzájemná poloha prostorových prvků nevyhovuje požadavkům pro tvorbu modelu [11].

5.2 Vybrané definice topologie

„Topologie – obor zabývající se topologickými prostory, jejich zobrazením a charakteristikami, nauka o spojitých přiřazeních (nejvíce zobecnitelná geometrická disciplína a vzhledem k současným snahám o geometrizaci matematiky i nejobecnější disciplína matematiky).“ [6]

„Topologie je matematická disciplína zabývající se podáním precizních definic obecného pojmu prostorové struktury a pojmu spojitosti (vymezením obecného pojetí prostoru)

¹⁰ <http://www.ginasoftware.cz/cs/>

¹¹ http://webhelp.esri.com/arcgisserver/9.3/java/index.htm#geodatabases/topology_in_arcgis.htm

srovnávající rozličná pojetí prostorové struktury a jejich spojitých deformací a zkoumající vztahy mezi vlastnostmi, které lze uvažovat v topologických systémech – tj. množinách s prostorovými strukturami na nich definovaných. “ [7]

„Topologie je obor matematiky, který se zabývá popisem a analýzou prostorových vztahů mezi geometrickými objekty. Zkoumá geometrické vlastnosti, jež jsou pro určité druhy transformací invariantní, jako je třeba roztažení, nebo ohýbání. “ [8]

„Topologie je věda a matematické vztahy používané pro ověřování geometrie vektorových prvků, pro operace jako prohledávání sítí a testování sousednosti polygonů. “ [9]

5.3 Práce zabývající se topologií a topologicky čistými daty

„Rovinné objekty se často používají k zobrazení, mimo jiného, zemského povrchu, katastrálních parcel a správních hranic. V praxi jsou často tyto objekty uloženy jako soubor jednotlivých polygonů, které mají své atributy, což způsobuje různé chyby a nesrovnalosti (např. mezery a přesahy), které jsou vytvořeny během jejich vytvoření, manipulace či změny. Tyto chyby zásadně omezují používání rovinných objektů v dalších software (např. vzhledem k nesprávným předpokladům způsobuje chybné výpočty).“ Tento problém vyzdvihují ve své práci [14] pánové Ohoří, Ledoux a Meijers.

V dokumentu [14] navrhuje novou metodu pro validaci a automatickou opravu rovinných objektů. Základem jejich metody je použití vynucené triangulace polygonů. Jak validace, tak oprava funkcí se provádí s relativně jednoduchými operacemi. Jedná se o označení trojúhelníků a standardní graf průchodu algoritmů (např. prohledávání do šířky či hloubky). Jejich software vezme jako vstupní vrstvu polygony, které následně ověřuje, pokud obsahují chyby, tak je opravuje a vrátí novou sadu bezchybných polygonů. Po opravě dat je zachována topologická konzistence celé rovinné oblasti.

„Prostorové modely jsou často založeny na polygonech a to jak ve 2D tak i v 3D. Mnoho Geo - ICT produktů podporuje prostorové datové typy, jako je polygon, na základě OpenGIS " Simple Features Specification" . OpenGIS a ISO se rozhodli, že sjednotí jejich specifikace a normy.“ [16]

Takto začíná článek [16], který se zabývá výhradně polygony. V dokumentu se nejdříve diskutuje o standardech a poté porovnává několik implementací. Jsou zde rozebrány definice polygonu podle výpočetní geometrie, ISO a OpenGIS. Poslední definice, o které se zde zmiňují je jejich vlastní návrh. Autoři vytvořili svojí definici, kde používají toleranci v okolí bodů („žádný bod neleží v epsilonové toleranci jakéhokoli jiného bodu nebo hrany“ [16]).

V článku se zmiňují, že „během testů si všimli nepatrných, ale zásadních rozdílů ve způsobu, jakým jsou vymezeny polygony (dokonce i ve 2D při použití rovných čar). Následky mohou být velmi nepříjemné. Například je vybrán jiný počet objektů, když použijeme stejný dotaz pro stejná data, ale v různých prostředích. Dalším následkem je, že data mohou být ztracena při přenosu z jednoho systému do druhého, validní polygony platné v jednom prostředí, nemusí být přijaty v jiném prostředí.“ [16]

Výsledky testů ukazují značné rozdíly v implementacích, což způsobuje další problémy. Autoři článku zdůvodňují tyto rozdíly různými interpretacemi standardů OpenGIS a ISO, jelikož nemají stejnou definici pro polygon.

V další části řeší validaci polygonů a to jak testovali data s různými programy a funkcemi (Geo - DBMS , Oracle , Informix , PostGIS , ArcSDE binary a LaserScan Radius Topology).

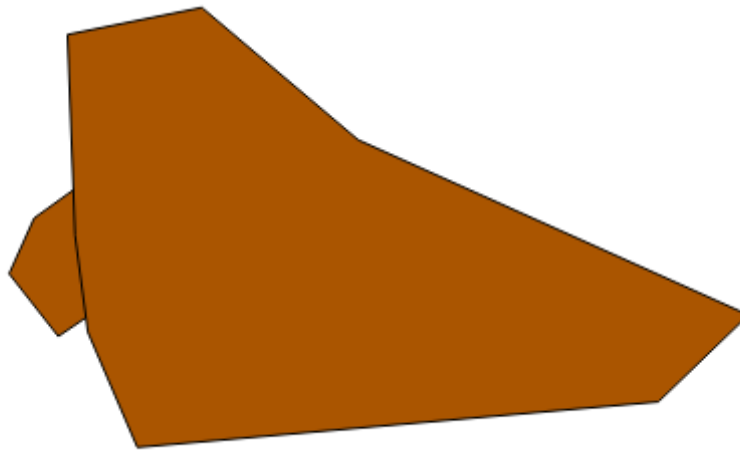
Zjistili, že i v praxi existují významné rozdíly ve validaci polygonů. V článku vysvětlují, že je to způsobené hlavně přenosem dat, během kterého se ztratí některá data. Dále zde naléhají na normalizační organizaci a Geo – ICT prodejce, aby řešily tento problém a zvažili navrhovanou definici.

6. Topologické chyby v prostorových datech

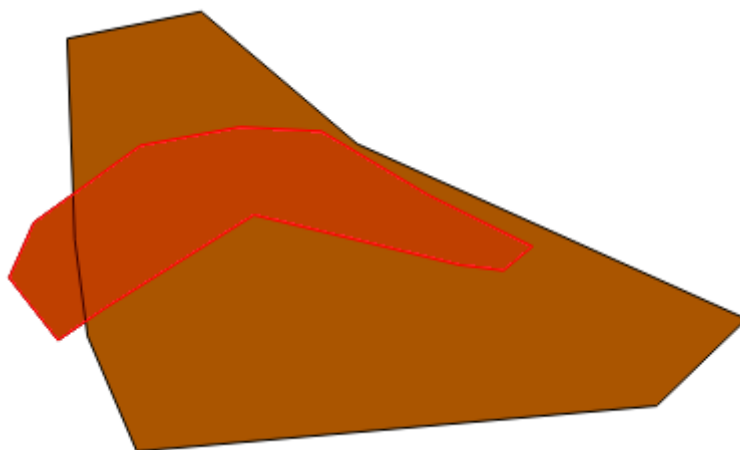
Data obsahují tři druhy rostlin: bolševník velkolepý, který je označen písmeny Hm, netýkavka žláznatá s písmeny Ig a křídlatky s označením Rsp. Pro názornou ukázkou jsem zbarvila polygony rostlin: Hm – zelená, Rsp – hnědá a Ig – růžová.

Níže jsem vybrala a popsala nalezené topologické chyby v datech IS Heracleum:

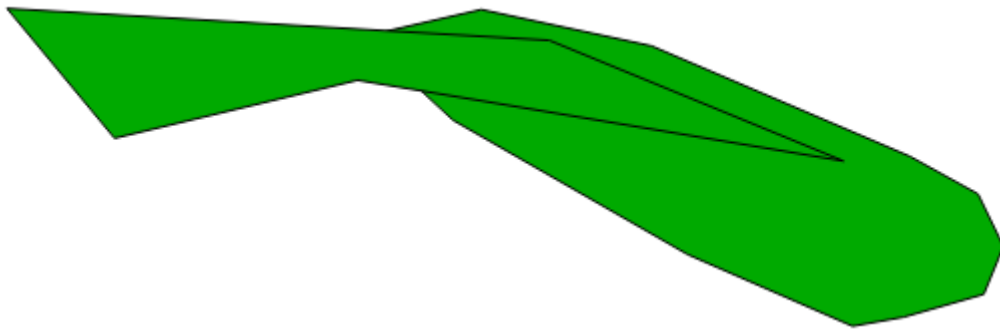
- 1) objekty se překrývají – polygony se navzájem úplně nebo částečně překrývají, což bylo s největší pravděpodobností způsobeno pracovníkem, který při mapování dat neřešil překrývání polygonů



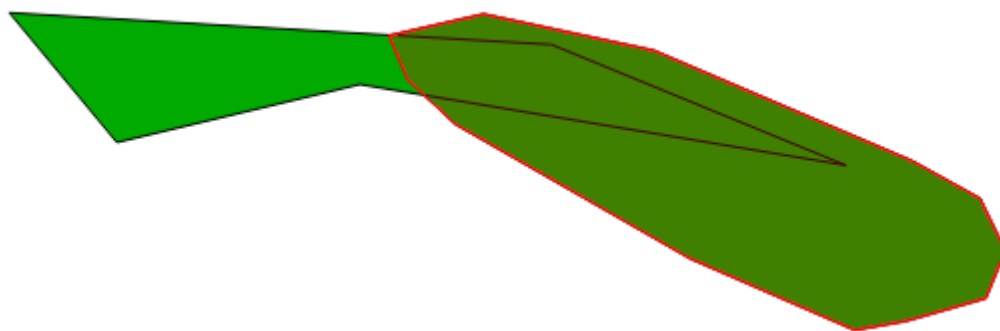
Obr. 6.1: Překrývající se objekty – ukáзка č. 1



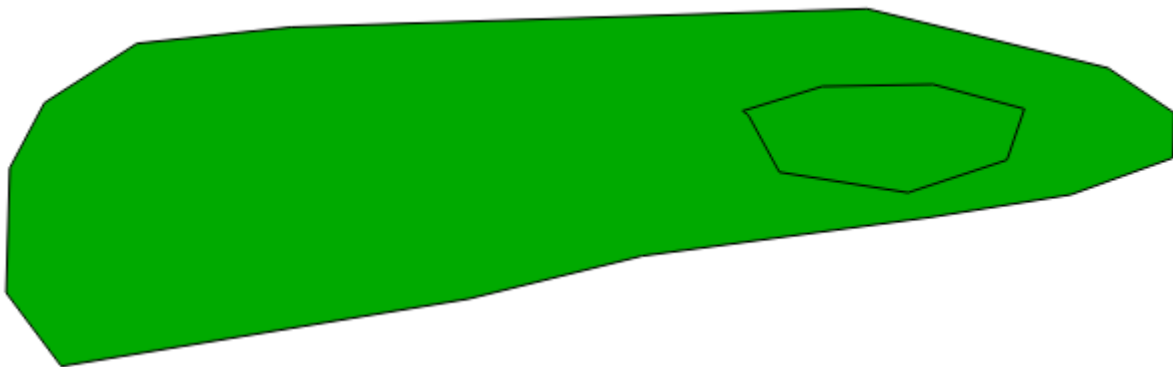
Obr. 6.2: Překrývající se objekty – ukáзка č. 1 – zvýraznění



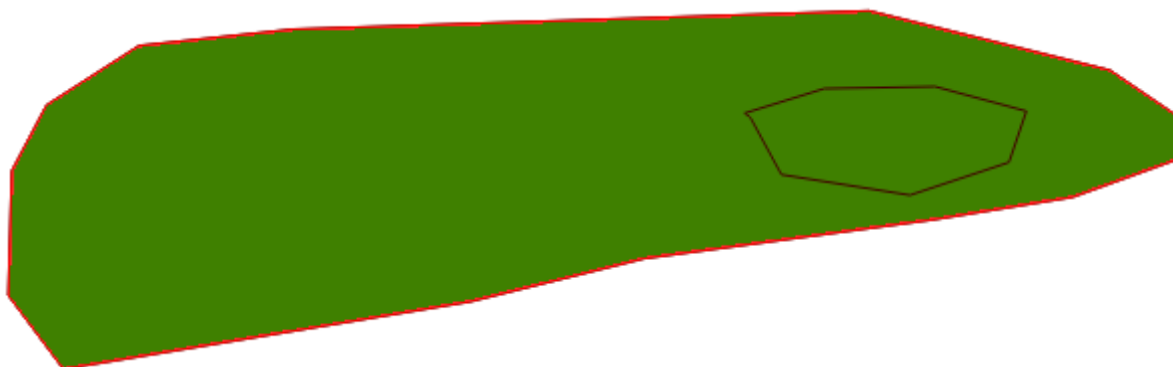
Obr. 6.3: Překrývající se objekty – ukázka č. 2



Obr. 6.4: Překrývající se objekty – ukázka č. 2 – zvýraznění

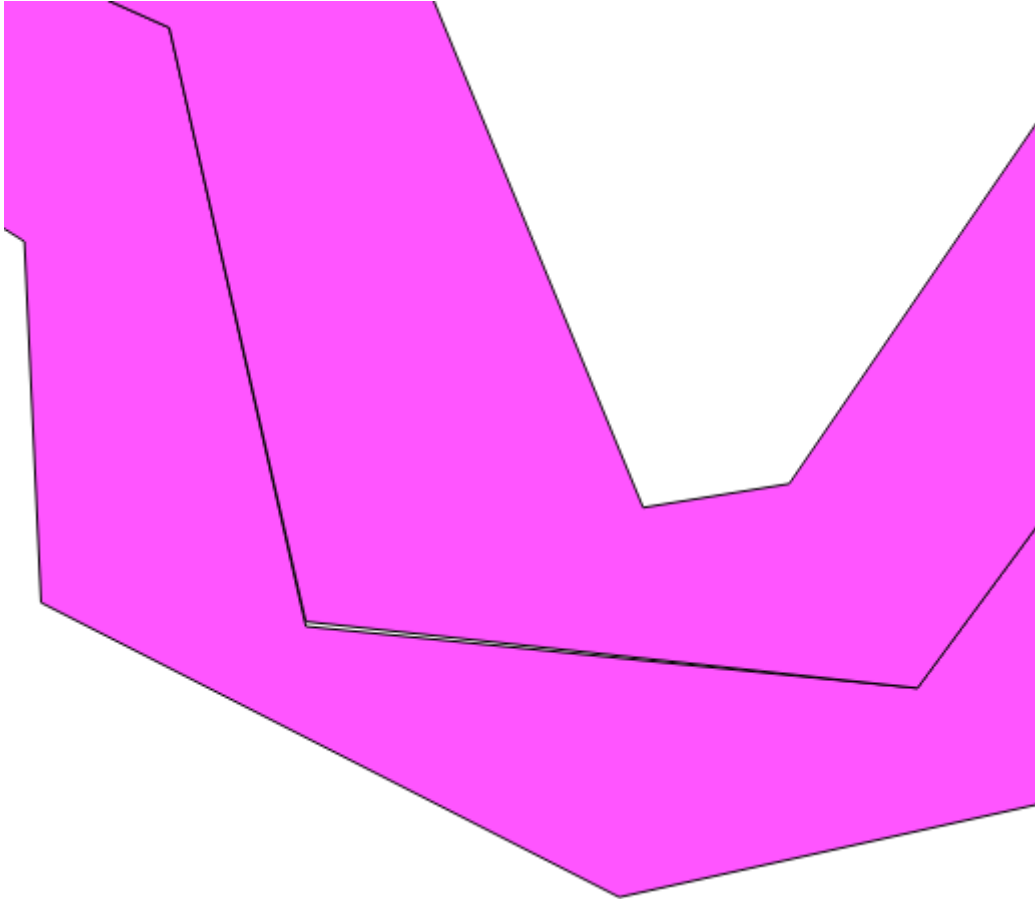


Obr. 6.5: Překrývající se objekty – ukázka č. 3

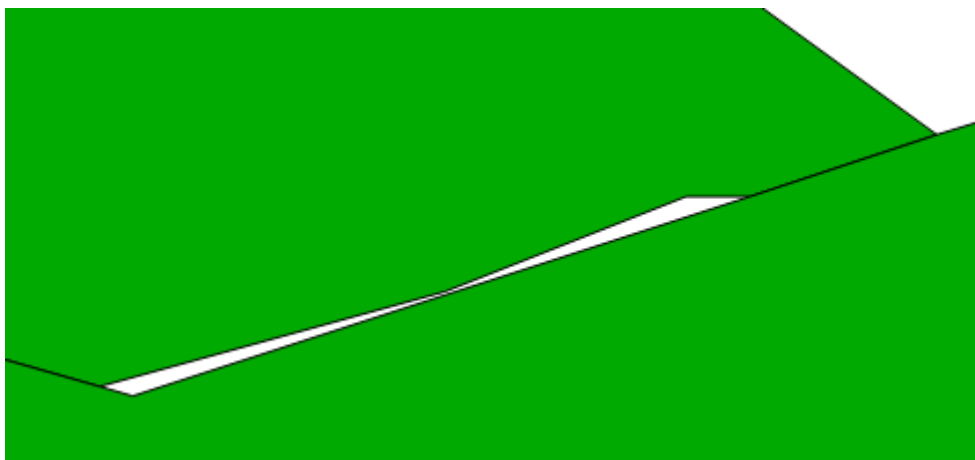


Obr. 6.6: Překrývající se objekty – ukázka č. 3 – zvýraznění

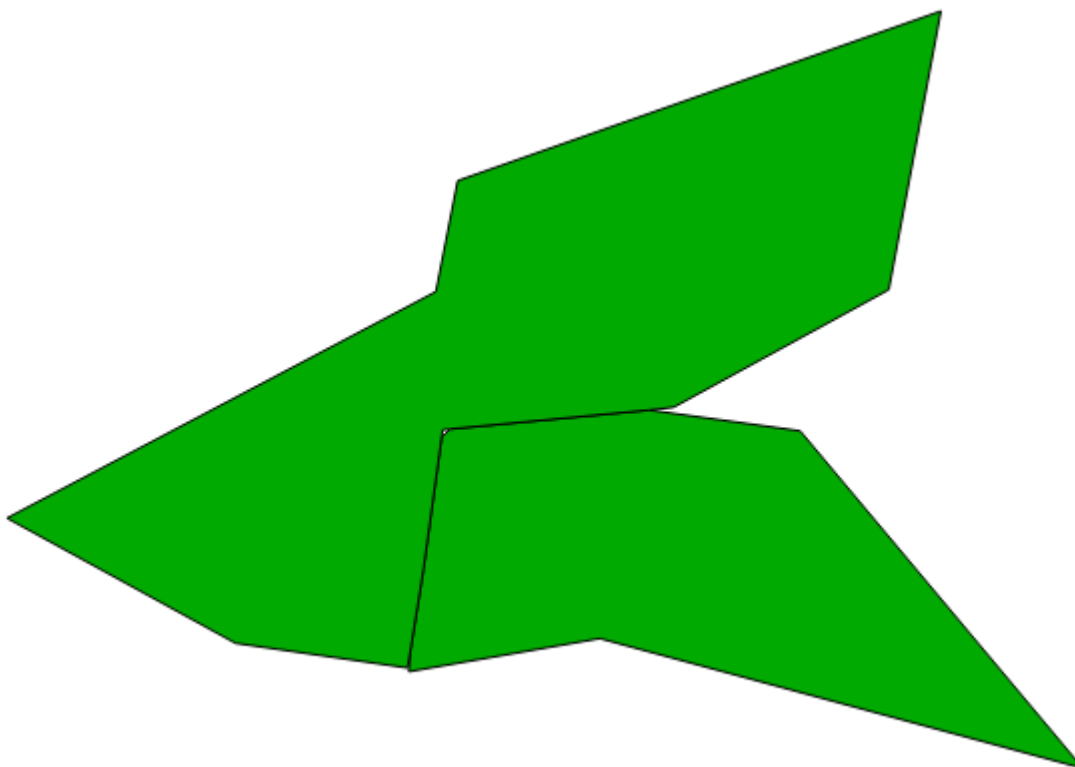
- 2) objekty obsahující mezeru – polygony, které společně sousedí, mají na hranici díru, což mohlo být způsobeno špatným zanesením dat do přístroje, který byl použit při mapování nebo špatným zdigitalizováním dat - data zmapována na papírovou mapu byla následně přepracována do digitální podoby. Další možný důvod je nepoužití snapování při digitalizaci dat.



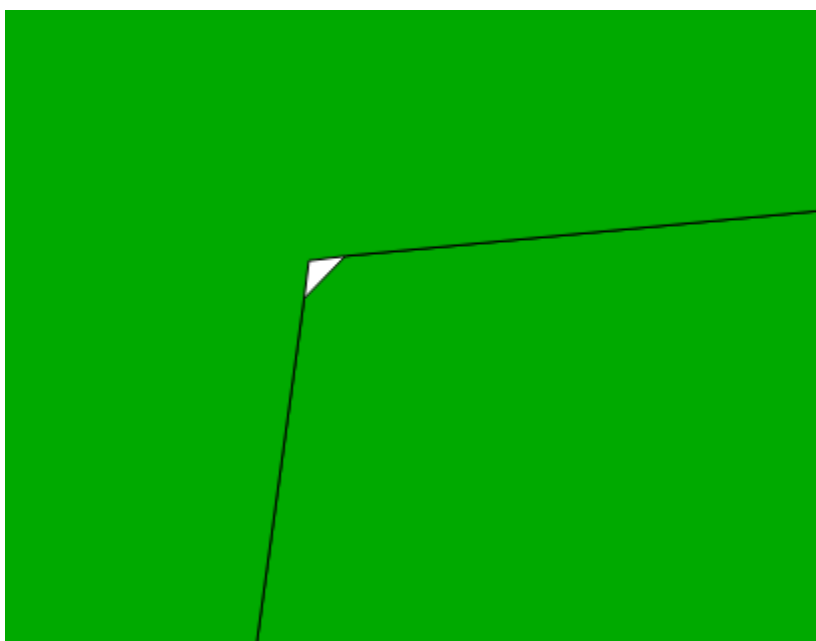
Obr. 6.7: Objekt mající mezeru – ukázka č. 1



Obr. 6.8: Objekt mající mezeru – ukázka č. 2

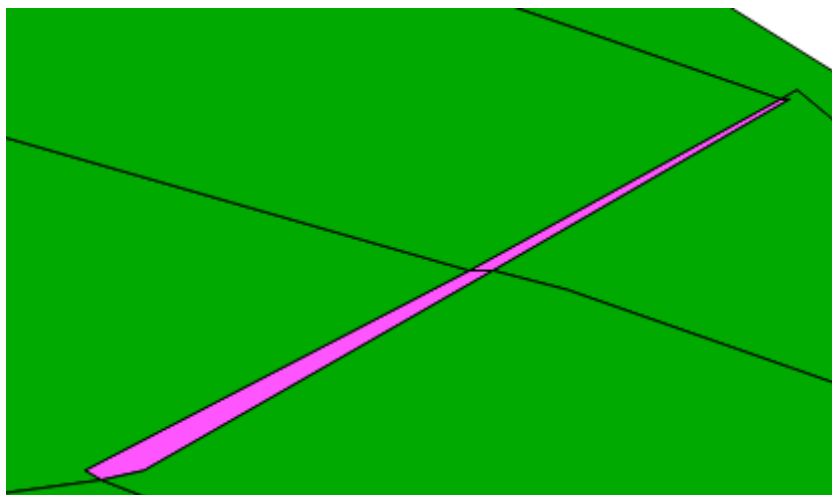


Obr. 6.9: Objekt mající mezeru – ukázka č. 3

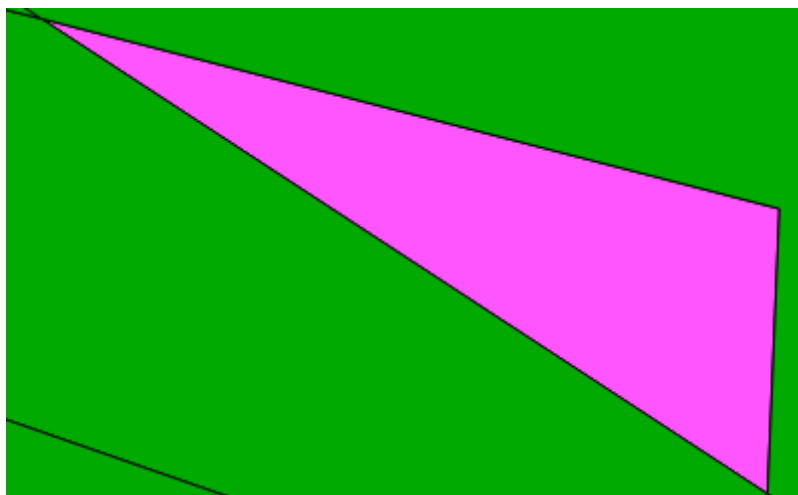


Obr. 6.10: Objekt mající mezeru – ukázka č. 3 – detail

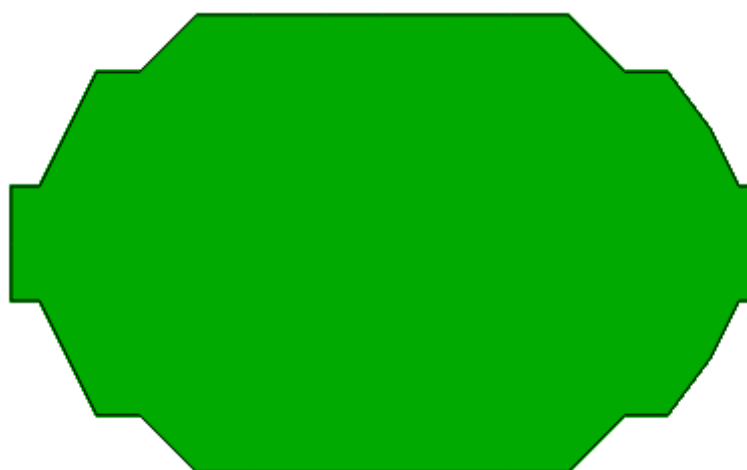
- 3) duplicitní objekty – polygony jsou zobrazeny jednou či vícekrát, mají totožné hranice polygonů, což mohlo být způsobeno buď pracovníkem, který zanášel data do systému nebo samotným systémem



Obr. 6.11: Duplicitní objekt – ukázka č. 1



Obr. 6.12: Duplicitní objekt – ukázka č. 2



Obr. 6.13: Duplicitní objekt – ukázka č. 3

7. QGIS

QGIS představuje jeden z open source GIS. Open source znamená, že softwarový kód může být bezplatně kýmkoliv prohlížen a upravován, pouze toto omezuje GPL licence tak, že veškeré úpravy se musí pro Quantum GIS projekt zveřejnit a nesmí se vydat nová verze Quantum GISu pod uzavřenou licenci. Vývoj začal v roce 2002, poslední vydaná verze 2.2 (22.2.2014) se jmenuje Valmiera. QGIS lze využívat v operačních systémech Linux, Windows, OSX, BSD a beta Android. QGIS podporuje vektorové, rastrové a databázové formáty. Program slouží ke zobrazení, vytváření, zpracování nebo analýze vektorových i rastrových dat. Výborná je celkem široká paleta zásuvných modulů (pro mne je důležitý modul Kontrola topologie), tyto moduly jsou psané v jazycích C++ nebo Python. [17] Další výhodou např. oproti PostGIS je relativně intuitivní ovládání, lokalizace do českého jazyka a nulová pořizovací cena.

7.1 Topologická pravidla

Topologie může být použita k detekci chyb v datech. U některých nástrojů, jako je síťová analýza, jsou topologicky čisté údaje nezbytné[2]. QGIS má zásuvný modul Kontrola topologie, který definuje topologii jako pravidla vztahů a nechává uživatele vybrat pravidla, které mají být provedena ve vektorové vrstvě.

S kontrolou topologie je možné se podívat na vektorové objekty a zkontrolovat topologii s několika topologickými pravidly. Tato pravidla si poradí s prostorovými vztahy, avšak vše záleží na konkrétních okolnostech, které použijete v topologických pravidlech na vektorová data.

QGIS má vestavěné topologické úpravy objektů, které jsou určeny pro vytváření nových objektů bez chyb [2]. Existující chyby v datech a uživatelsky vyvolané chyby jsou obtížně zjistitelné.

Seznam a popis topologických pravidel níže napsaných vychází z [4].

QGIS disponuje následujícími topologickými pravidly pro polygonové vrstvy:

- *Musí obsahovat* (must contain): Polygonová vrstva musí obsahovat alespoň jeden bod geometrie z druhé vrstvy.
- *Nesmí mít duplikáty* (must not have duplicates): Polygony ze stejné vrstvy nesmí mít identické geometrie. Kdykoliv jsou polygony zastoupeny dvakrát nebo vícekrát, objeví se v poli " Chyba ".
- *Nesmí mít mezery* (must not have gaps): Sousední polygony by mezi sebou neměli tvořit mezery.
- *Nesmí být neplatná geometrie* (must not have invalid geometries): Kontroluje, zda je geometrie platná. Některé z pravidel, které definují platnou geometrii, jsou:
 - Polygony musí být uzavřené.
 - Polygony, které definují otvory, by měly být uvnitř polygonu, který definuje vnější hranice.
 - Polygony nesmí protínat sami sebe (nesmějí se dotýkat, ani křížit).
 - Polygony se nesmí dotýkat jiného polygonu, akceptován je pouze hraniční bod.
- *Nesmí být vícedílné geometrie* (must not have multi-part geometries): Někdy je geometrie vlastně sbírka jednoduchých útvarů.

Tato geometrie se nazývá geometrie vícedílná. Pokud obsahuje pouze jeden typ jednoduché geometrie, říkáme ji vícebodová, víceliniová nebo vícepolygonová.

- *Nesmí se překrývat* (must not overlap): Sousední polygony by neměly sdílet společný prostor.
- *Nesmí se překrývat s* (must not overlap with): Sousední polygony z jedné vrstvy by neměly sdílet společný prostor s polygony z jiné vrstvy.

7.2 Manuál pro kontrolu topologie

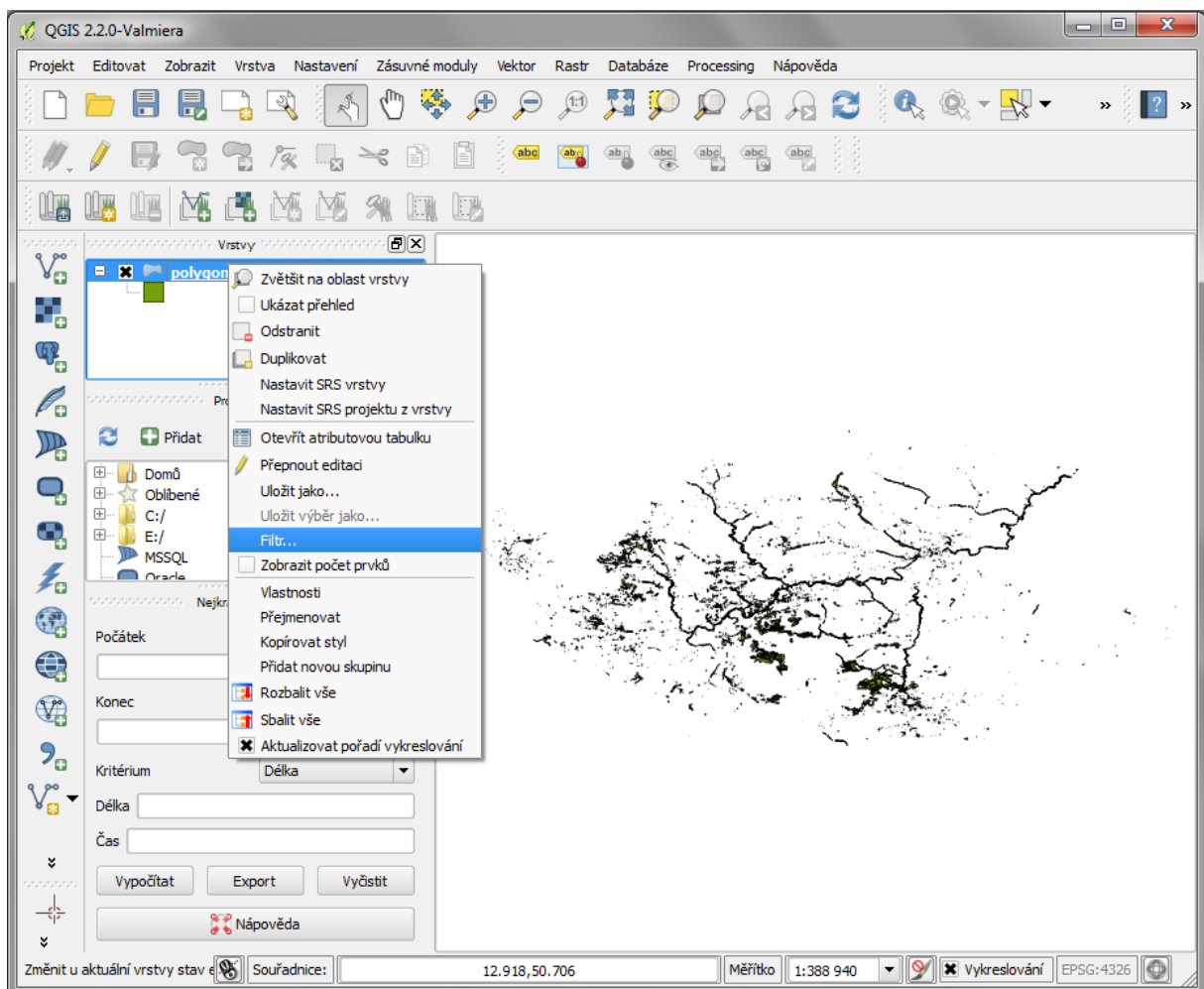
Jak jsem již psala v kapitole 4.2, moje data jsou pouze polygony, proto dále používám pouze pravidla pro kontrolu topologie polygonové vrstvy.

Data obsahují tři druhy rostlin: bolševník velkolepý, který je označen jako Hm, netýkavka žláznatá s písmeny Ig a křídlatky s označením Rsp. Tyto rostliny lze teoreticky v terénu nalézt na stejném místě, což znamená, že polygony různých druhů rostlin se mohou navzájem překrývat, proto nejdříve vyberu data pouze pro jeden druh rostliny, na kterých testuji pravidla a následně postup opakuji pro další druhy rostlin.

Podle následujících obrázků s popisem mohou pracovníci zkontrolovat svá data, zda jsou topologicky čistá.

Níže je manuál, podle kterého lze postupovat:

Po přidání vektorové vrstvy „polygony“ do programu QGIS budu kontrolovat topologickou čistotu dat zvlášť pro každý druh rostliny. Nejdříve pomocí pravého tlačítka myši klepnu na vrstvu „polygony“ vyberu „Filtr...“ viz. obr. 7.2.1.



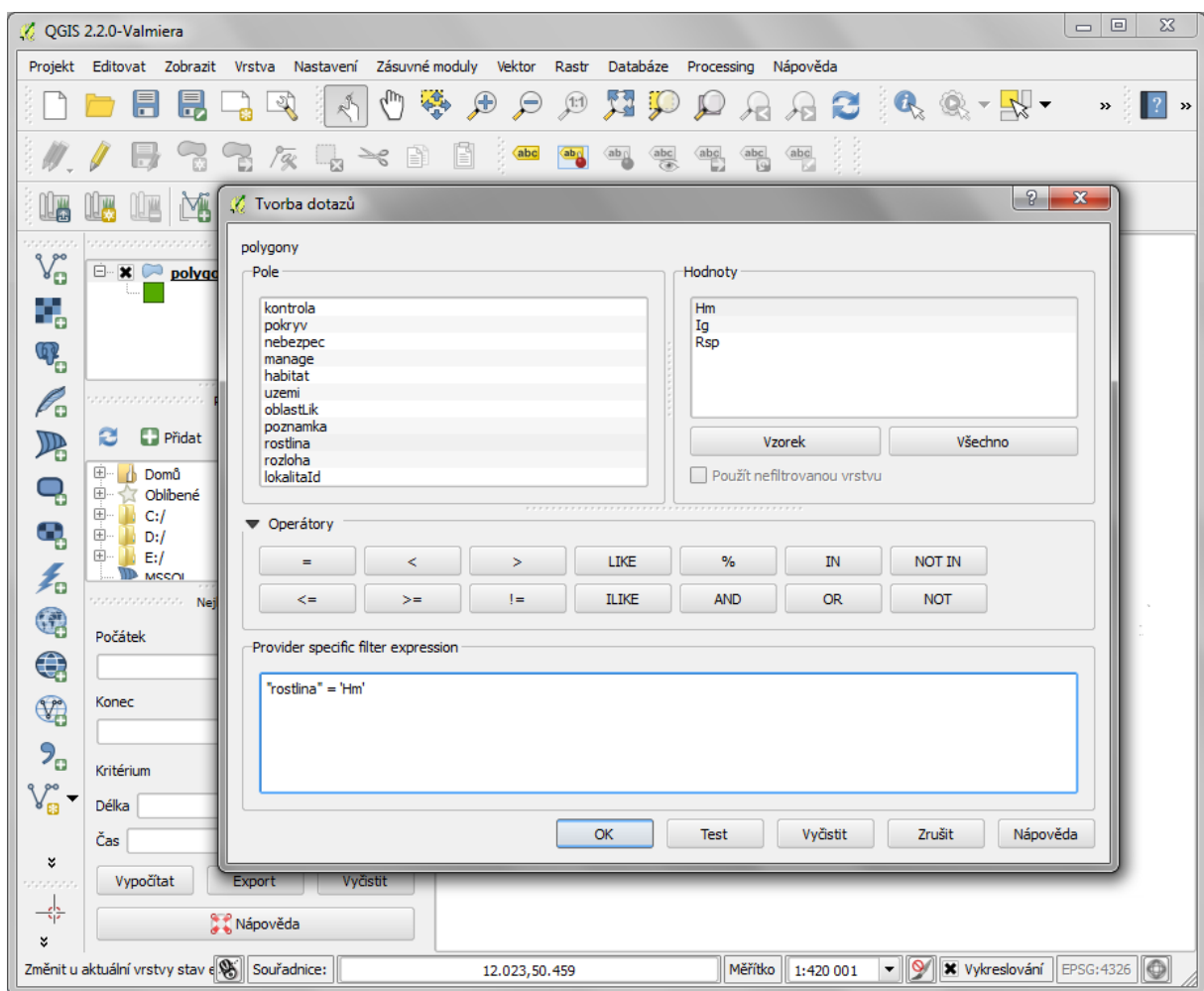
Obr. 7.2.1: Otevření filtru

Následně vytvořím dotaz "rostlina"='Hm', kterým vyberu všechny polygony bolševníku viz. obr. 7.2.2.

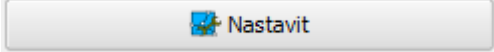
Mám dvě možnosti, jak vytvořit filtraci:

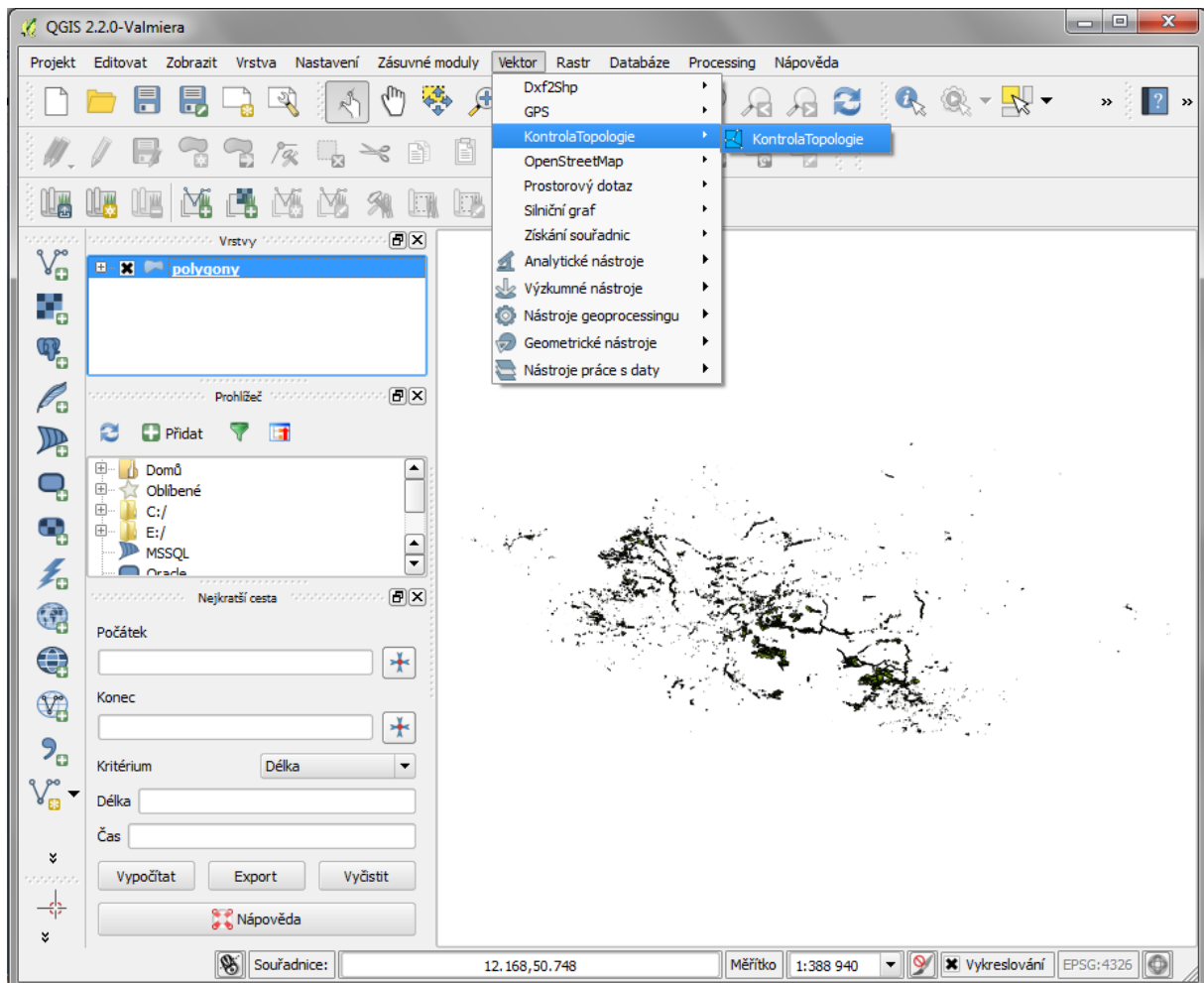
- 1) dotaz "rostlina"='Hm' napíši do pole „Provider specific filter expression“ nebo
- 2) v poli „Pole“ poklepu na „rostlina“, čímž se mi v poli „Provider specific filter expression“ zobrazí text "rostlina". Chci aby byly rostliny pouze bolševníky, proto následně klepnu na operátor v poli „Operátory“, čímž mám část dotazu: "rostlina" =. Dále klepnu na a v poli „Hodnoty“ se zobrazí všechny typy prvků, které jsou uloženy v tabulce ve sloupci rostlina. Zobrazí se Hm, Ig a Rsp. Poklepem na „Hm“ získám celý dotaz "rostlina" = 'Hm'.

Následně zvolím a proběhne filtrace prvků.



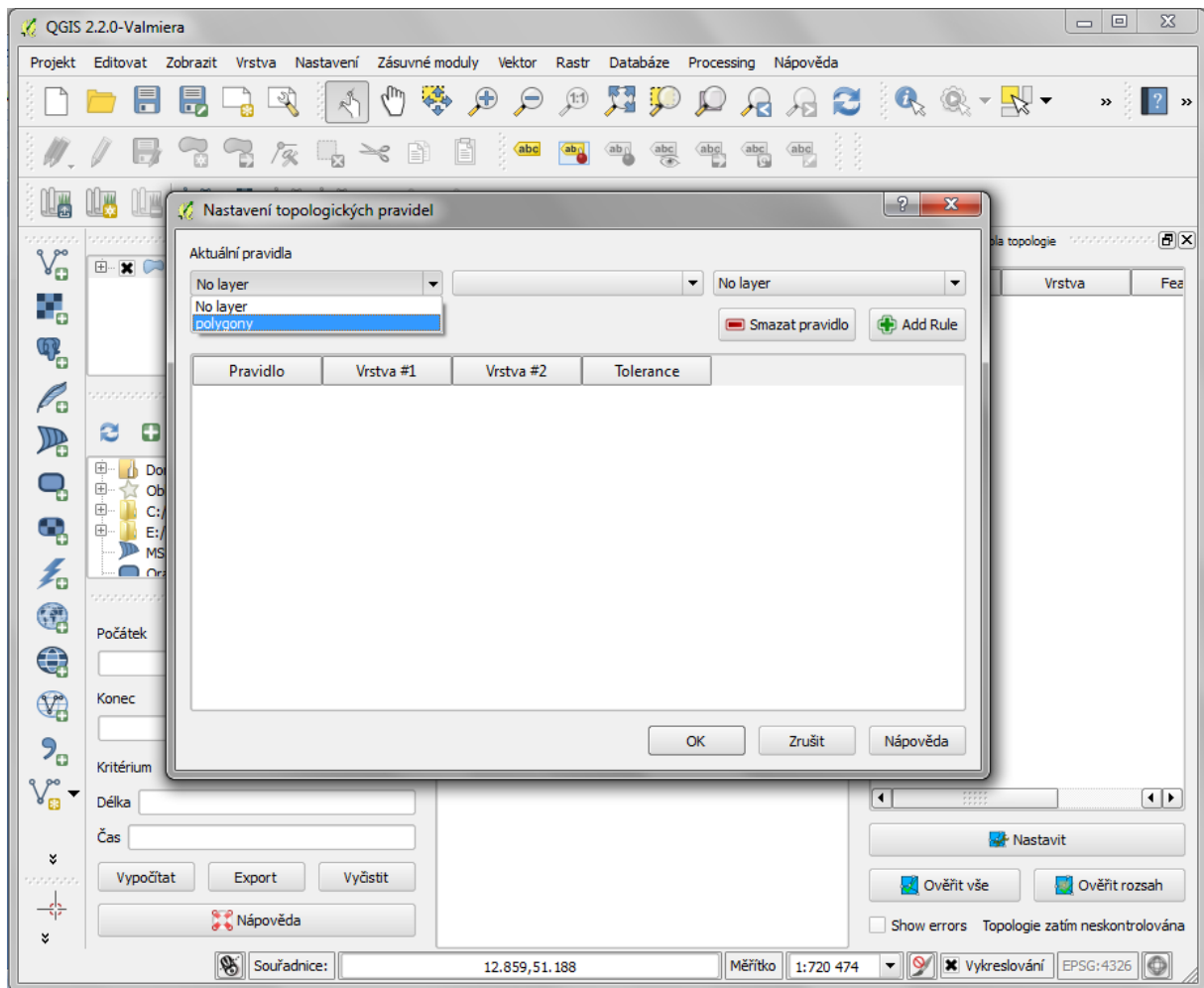
Obr. 7.2.2: Filtrace prvního druhu rostlin

Pro kontrolu topologie musím otevřít nástroj „KontrolaTopologie“, což lze udělat podle obr. 7.2.3, „Vektor“ – „KontrolaTopologie“ – „KontrolaTopologie“. Následně se zobrazí okno „Kontrola topologie“, zde klepnu na  pro nastavení kontroly topologie.



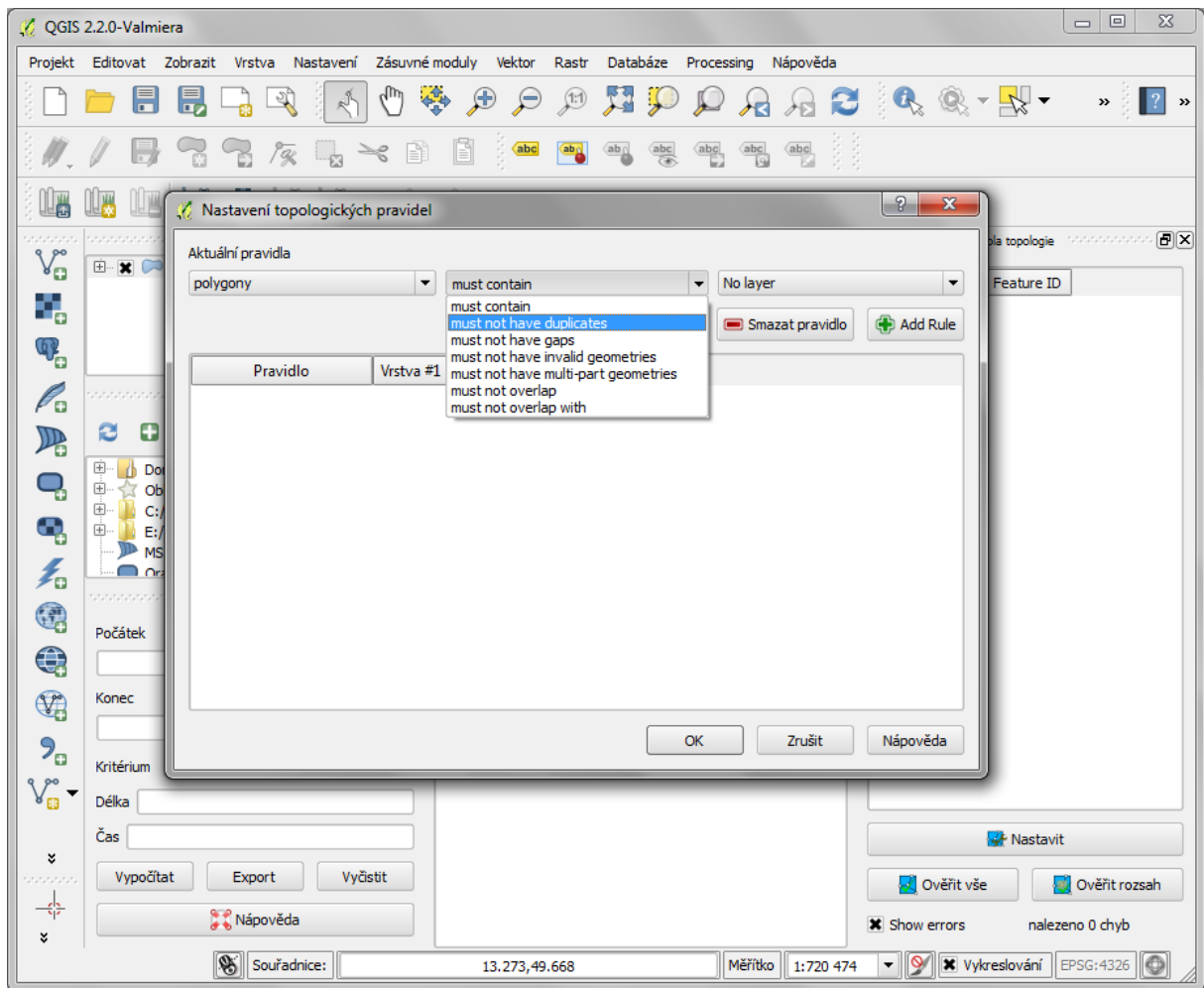
Obr. 7.2.3: Otevření nástroje pro kontrolu topologie

Dále vyberu svoji vrstvu „polygony“ viz. obr. 7.2.4. Zde lze vybrat vektorové vrstvy, které jsou již nahané v programu QGIS.



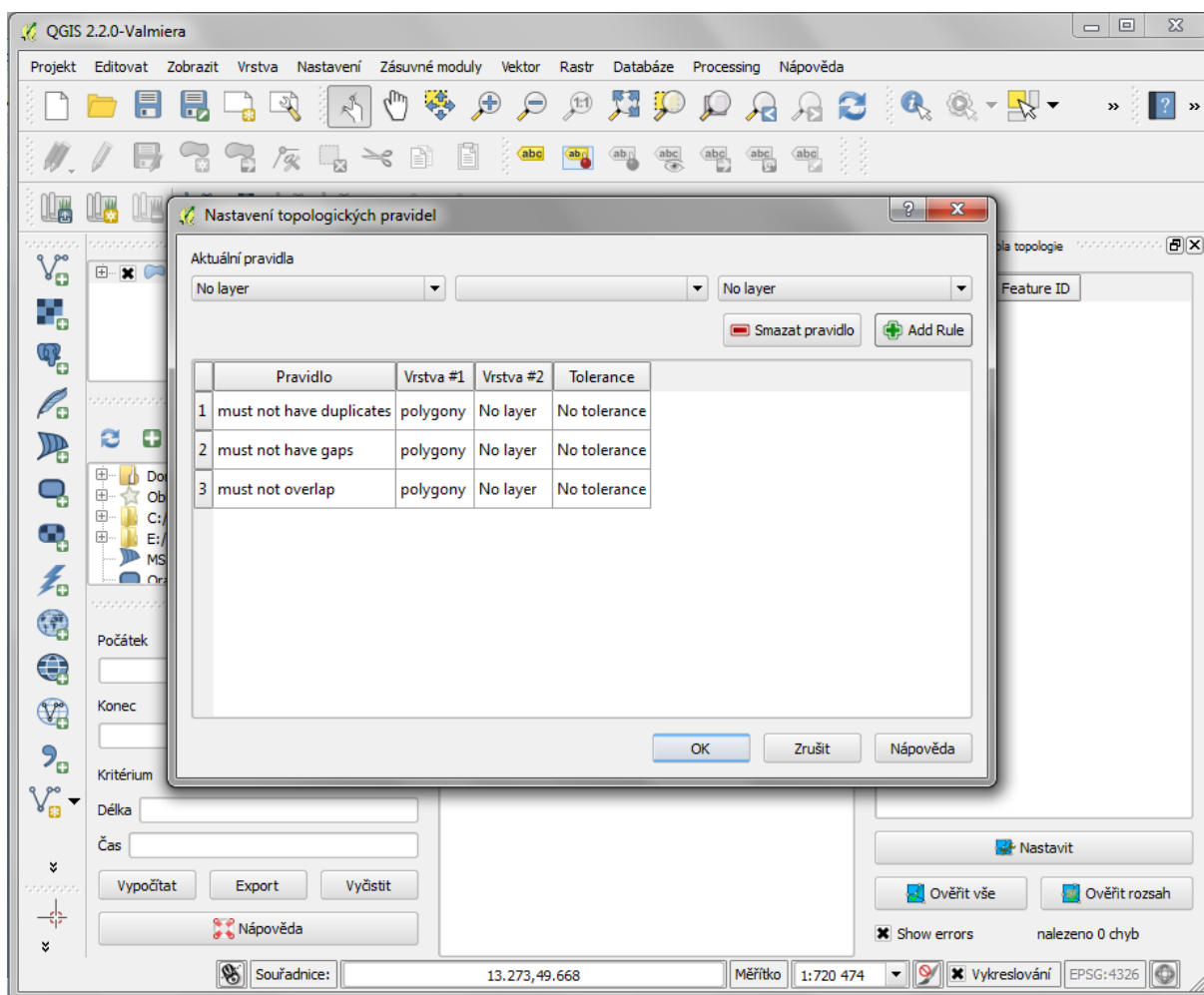
Obr. 7.2.4: Nastavení topologických pravidel – výběr vrstvy

Následně vyberu topologické pravidlo *Nesmí mít duplikáty* (must not have duplicates) viz. obr. 7.2.5 a přidám pravidlo klepnutím na .




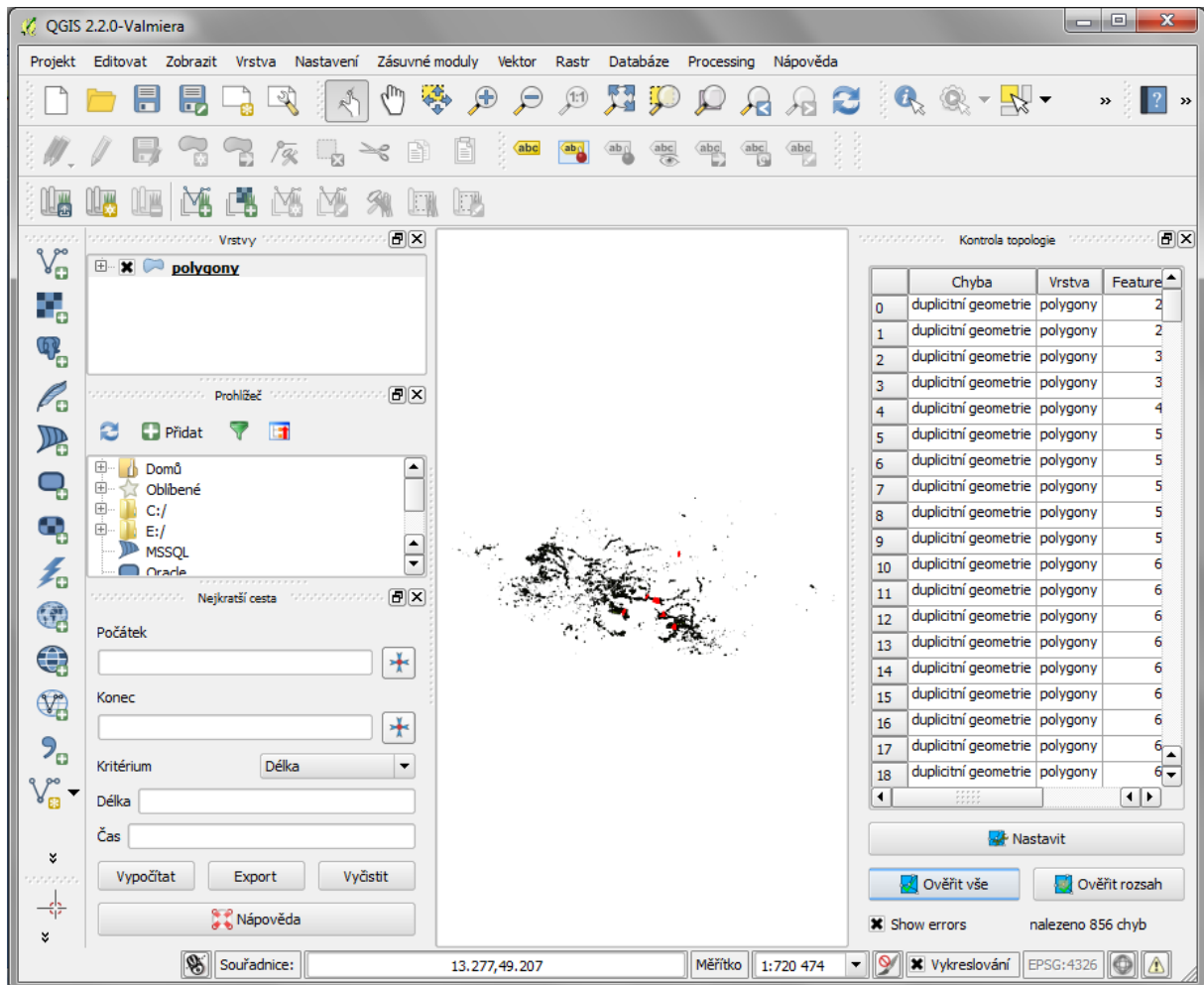
Obr. 7.2.5: Nastavení topologických pravidel – výběr pravidla

Postup s přidáním vrstvy, výběru a přidání pravidla opakuji dvakrát – pro pravidla *Nesmí mít mezery* (must not have gaps) a *Nesmí se překrývat* (must not overlap). Jakmile mám vybrána pravidla, potvrdím jej klepnutím na viz. obr. 7.2.6.



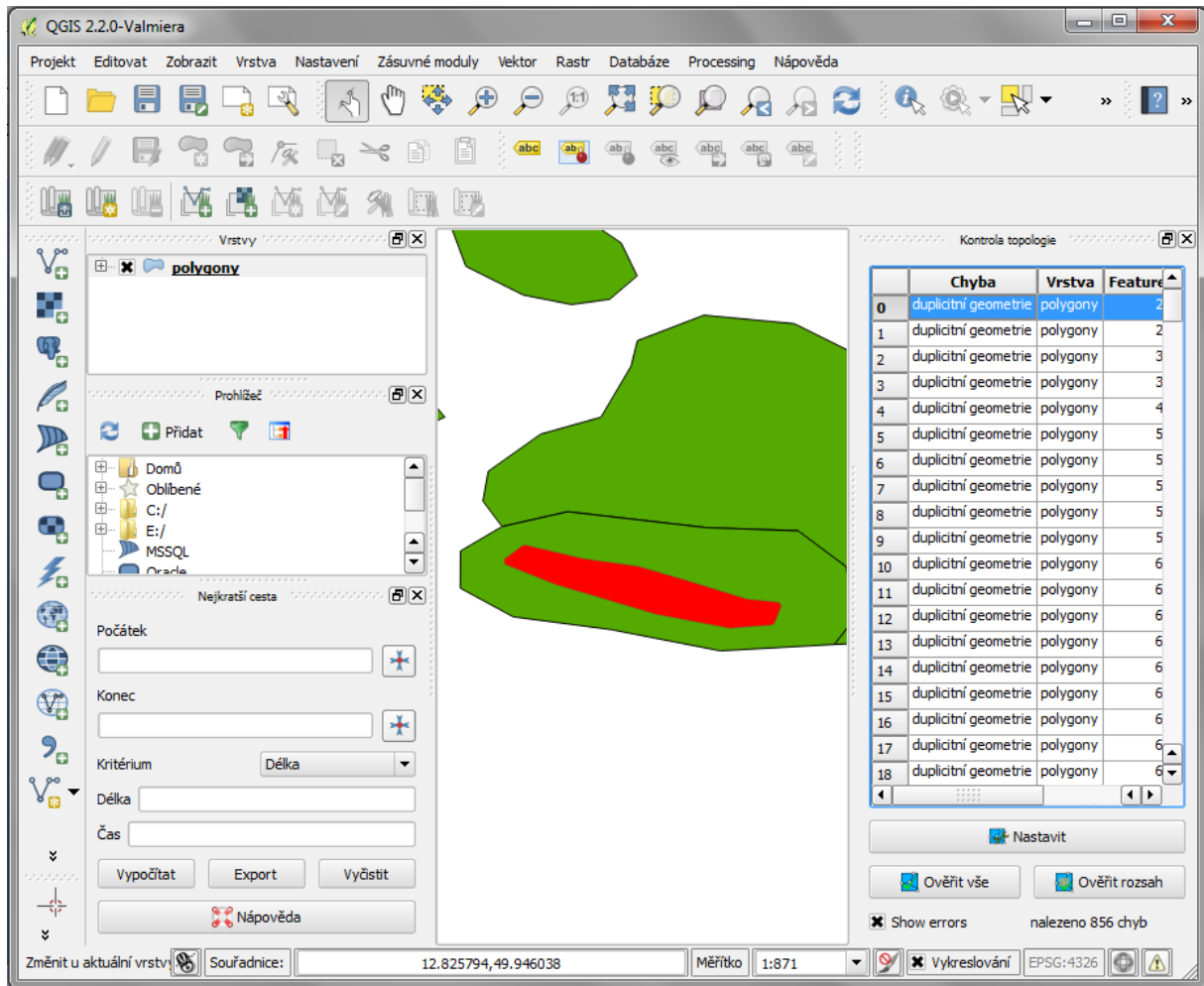
Obr. 7.2.6: Nastavení topologických pravidel – potvrzení nastavení topologických pravidel

Nakonec spustím kontrolu topologických pravidel klepnutím na . V okně „Kontrola topologie“ se zobrazí všechny nalezené chyby viz. obr. 7.2.7. Chyba „duplicitní geometrie“ je chyba vycházející z pravidla *Nesmí mít duplikáty* (must not have duplicates), chyba „díry“ vychází z pravidla *Nesmí mít mezery* (must not have gaps) a chyba „překryty“ z pravidla *Nesmí se překrývat* (must not overlap).



Obr. 7.2.7: Ověření topologických pravidel

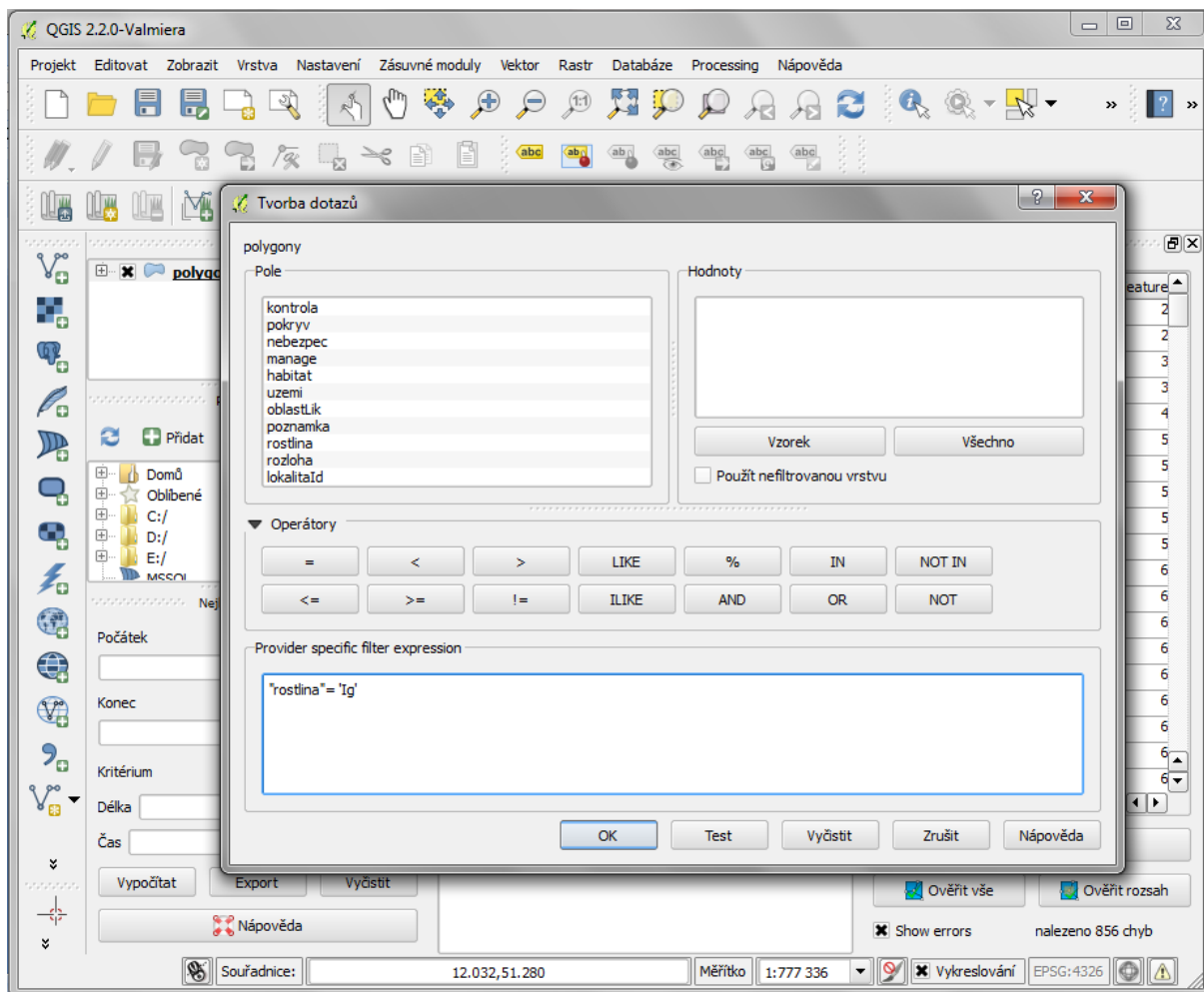
Pro zobrazení topologické chyby stačí klepnout na danou chybu v okně „Kontrola topologie“ ve sloupci „Chyba“ viz. obr. 7.2.8. V tomto okně naleznou všechna topologicky chybná data.



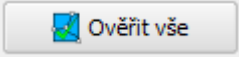
Obr.7.2.8: Zobrazení topologické chyby

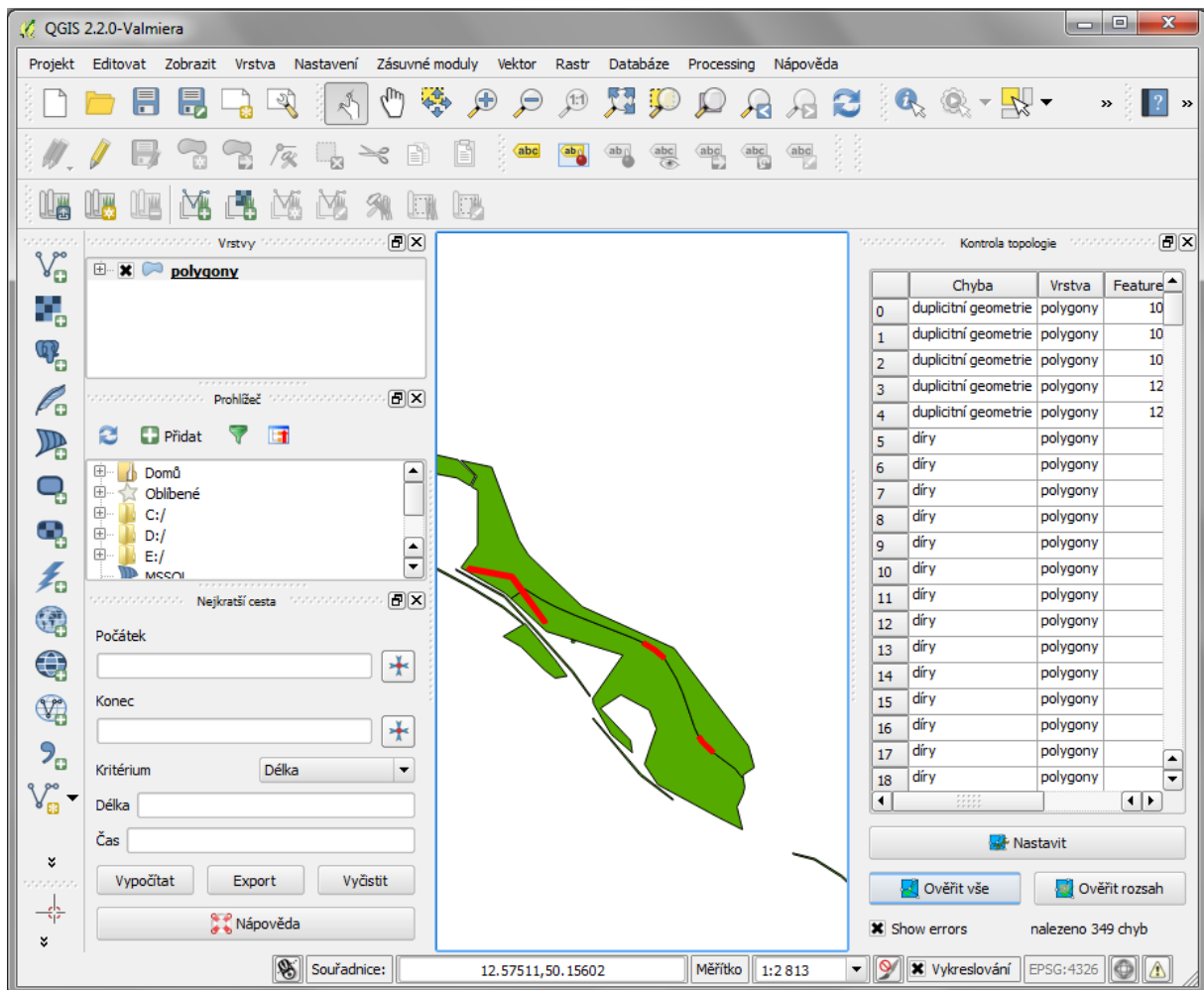
Tímto postupem jsem našla topologické chyby pro polygony znázorňující bolševník velkolepý. Obdobným postupem najdu topologické chyby i pro další druhy rostlin.

Opět podle obr. 7.2.1 viz výše, vyberu „Filtr...“ pro výběr určitého druhu rostliny a do pole „Provider specific filter expression“ vložím dotaz "rostlina"= 'Ig', který potvrdím stisknutím „OK“ viz obr. 7.2.9.

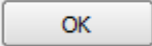


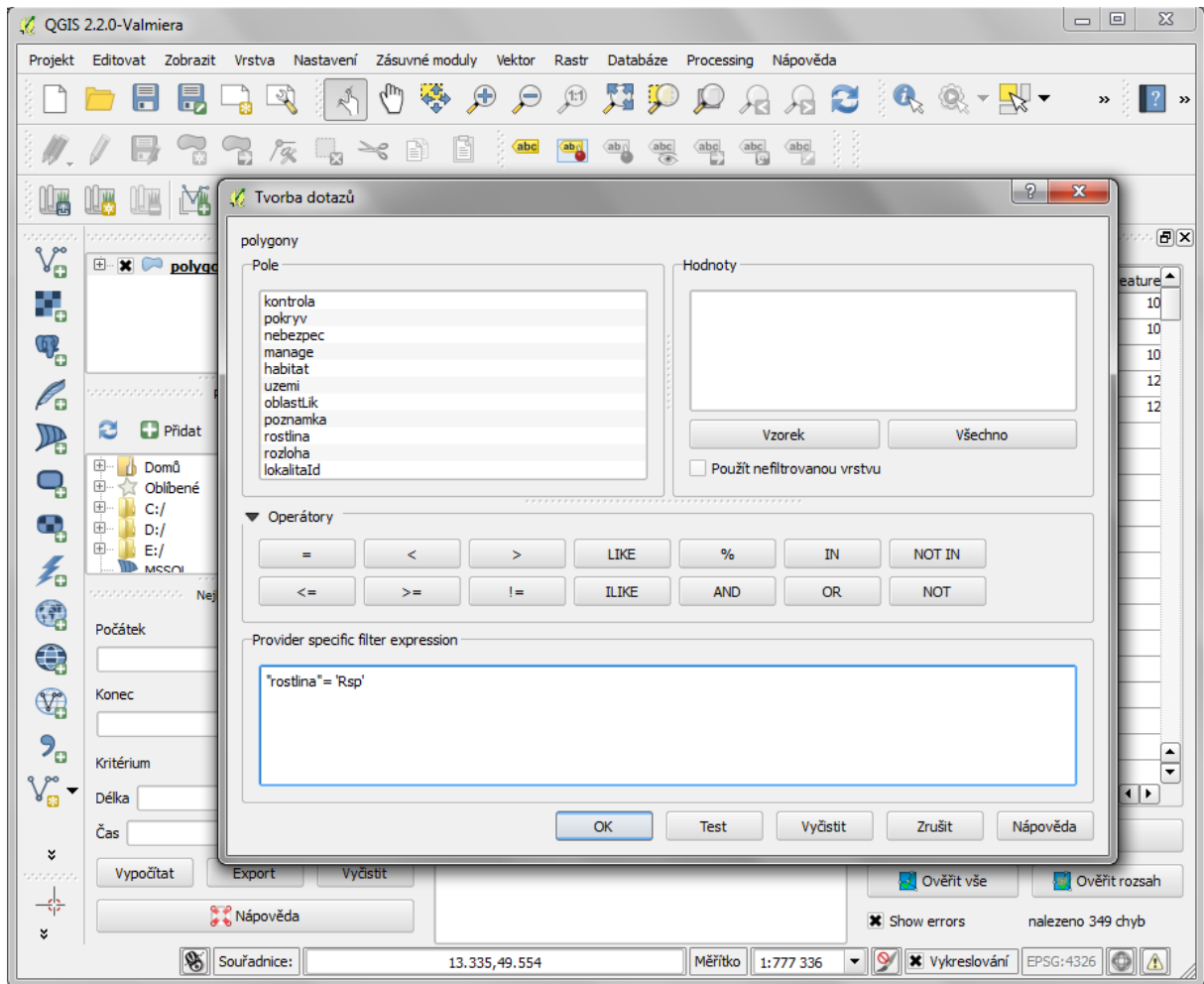
Obr. 7.2.9: Filtrace následujícího druhu rostlin

Následně už stačí pouze spustit kontrolu topologie dat klepnutím na  viz. obr. 7.2.10, nemusíme opět nastavovat vrstvu a topologická pravidla, jelikož výběr vrstvy a pravidel zůstává stejný a program tento výběr nesmazal. Nastavení kontroly topologie zůstává stejný, pouze jsem změnila data, na kterých budu daná pravidla používat. Opět v okně „Kontrola topologie“ vidím seznam všech topologických chyb.



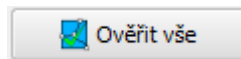
Obr. 7.2.10: Ověření topologických pravidel následujícího druhu rostlin

Opět podle obr. 7.2.1 viz výše, vyberu „Filtr...“ pro výběr určitého druhu rostliny a do pole „Provider specific filter expression“ vložíím dotaz "rostlina"= 'Rsp', který potvrdím stisknutím  viz obr. 7.2.11.

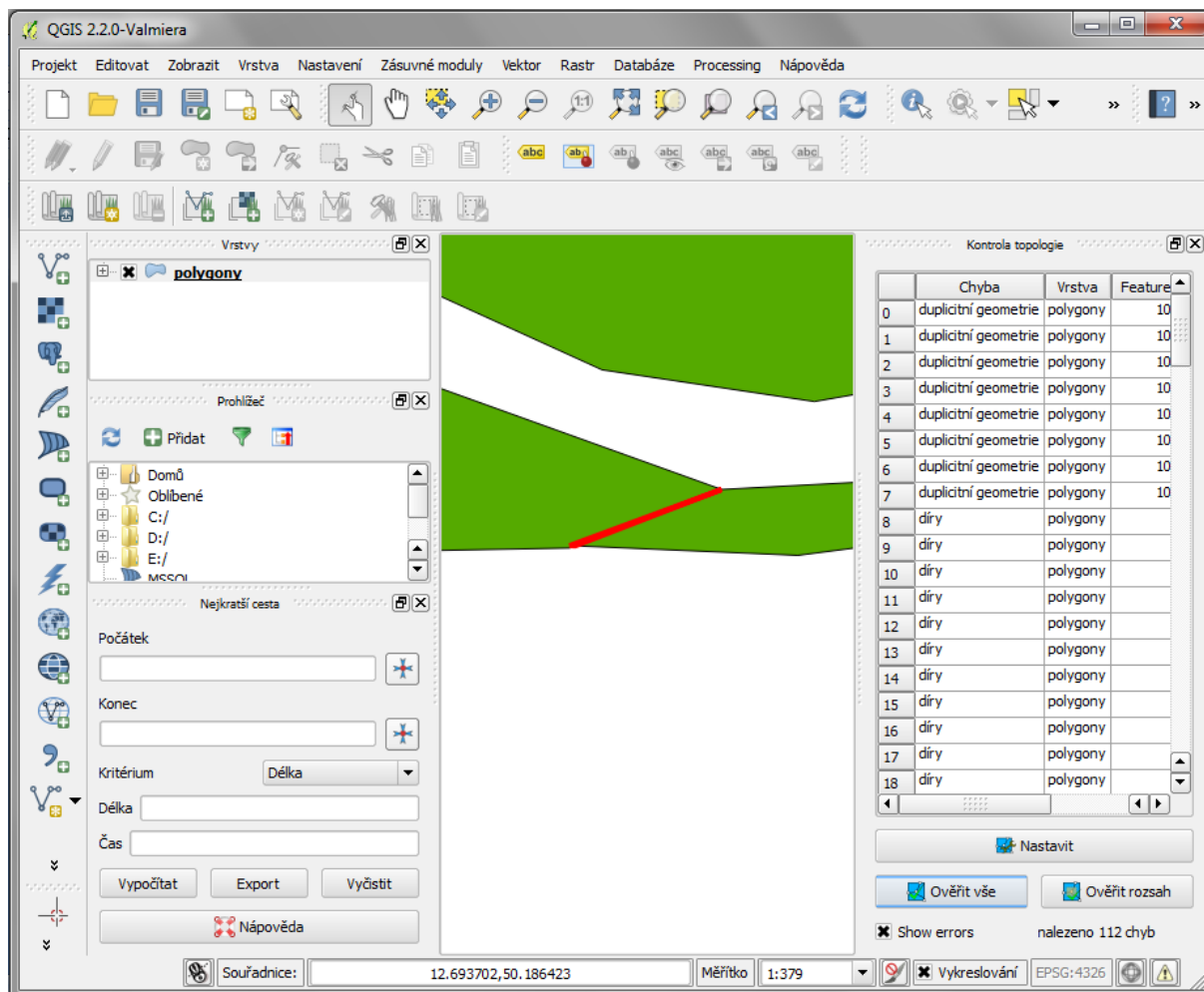


Obr. 7.2.11: Filtrace posledního druhu rostlin

Nakonec stačí pouze spustit kontrolu topologie dat klepnutím na



a opět



Obr. 7.2.12: Ověření topologických pravidel posledního druhu rostlin

7.3 Výsledek ověření topologie

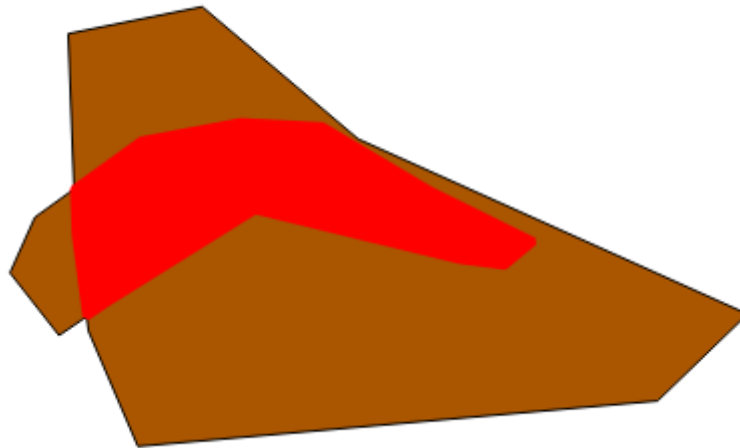
Celkem je v souboru 13 140 objektů, nalezeno bylo 1047 topologických chyb, podrobnější informace o počtu detekovaných topologických chyb lze najít v tab. 7.3.1. Již v kapitole 7.2 jsem vysvětlovala označení rostlin, avšak pro lepší přehlednost opět zkratky vysvětlím. Bolševník velkolepý je označen písmeny Hm, netýkavka žláznatá Ig a křídlatky Rsp.

Tab. 7.3.1: Souhrn nalezených topologických chyb

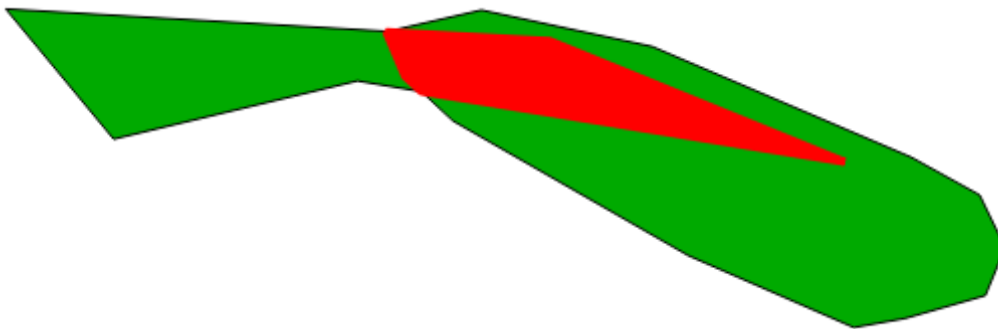
	duplicita	díry	překryt	celkem chyb	celkem objektů
Hm	47	483	326	856	7610
Ig	5	100	244	349	2636
Rsp	8	36	68	112	2894
Celkem	60	619	638	1047	13140

Existuje i takový objekt, který má více naráz více chyb. Např. existuje duplicitní objekt, který se překrývá s jiným polygonem, to znamená, že zde máme hned 2 chyby u jednoho polygonu – duplicitní objekt je první chyba, překrývání polygonu s jedním ze dvou totožných prvků je druhá chyba a překrývání polygonu s druhým prvkem, který je totožný s prvním, je třetí chyba.

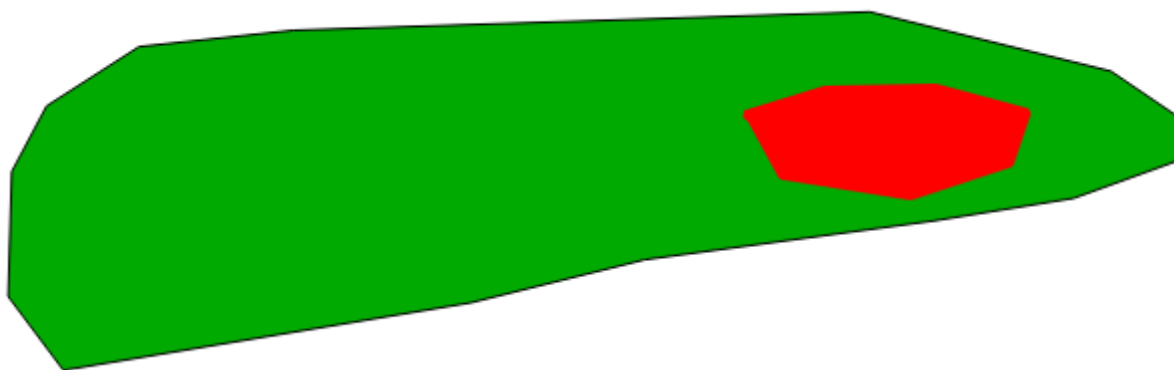
Níže jsou zobrazeny nalezené topologické chyby, které jsem sama našla a které jsem již použila v kapitole 6. Červenou barvou je vyznačena chyba. Duplicitní prvek se tedy zobrazí jako celý červený objekt, mezera se vybarví celá dočervena a při překrytí dvou objektů je červená buď část polygonu, která překrývá druhý nebo celý polygon, pokud je obsažen v jiném polygonu.



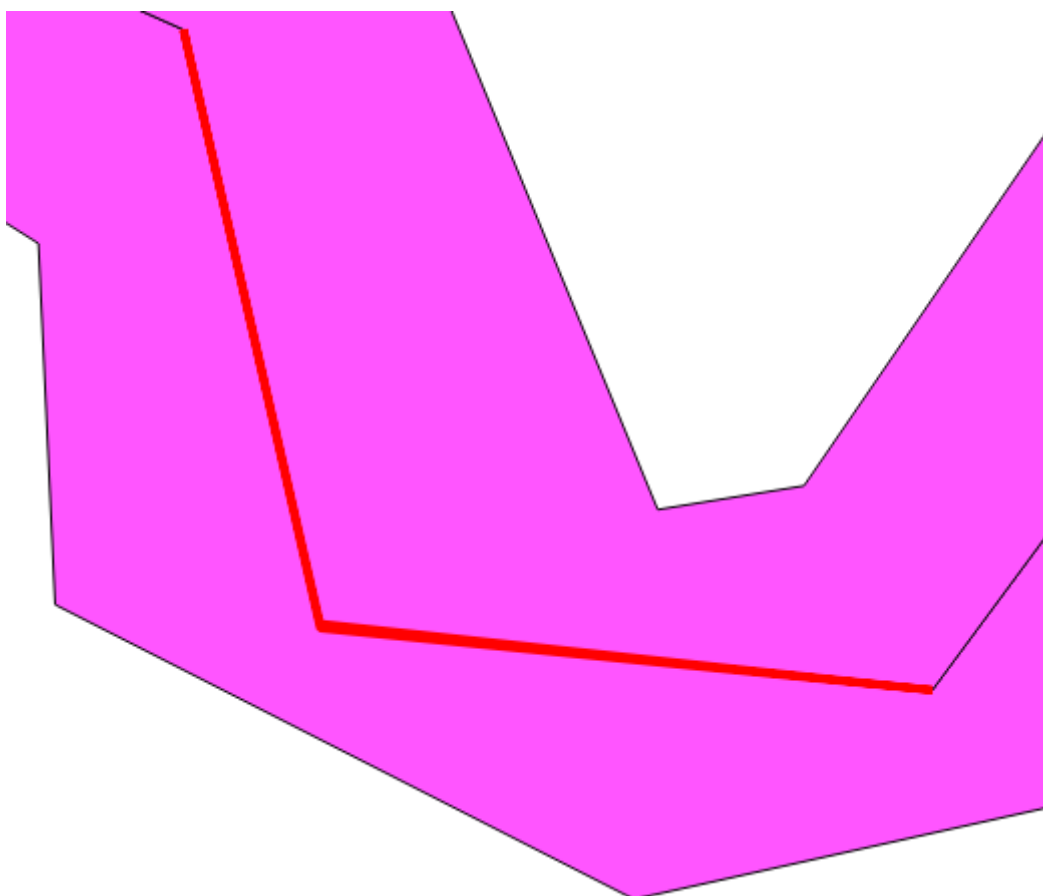
Obr. 7.3.1: Překrývající se objekty – ukázka č. 1 - kontrola pomocí QGIS



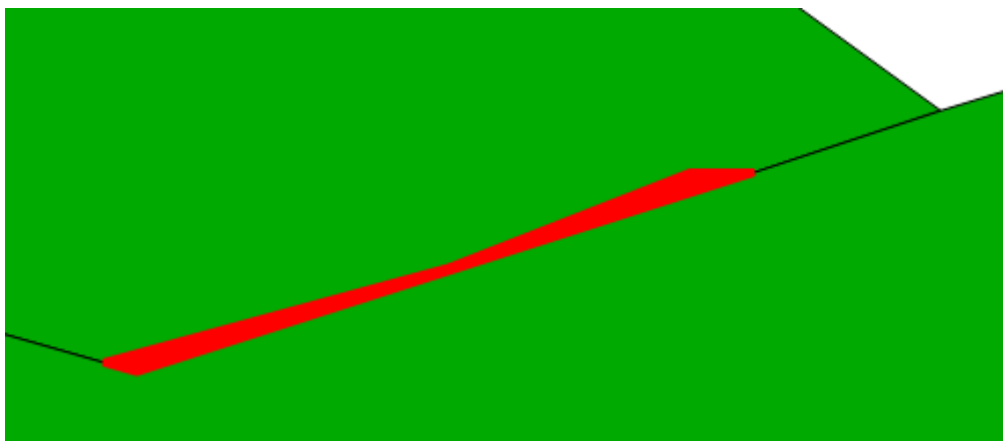
Obr. 7.3.2: Překrývající se objekty – ukázka č. 2 - kontrola pomocí QGIS



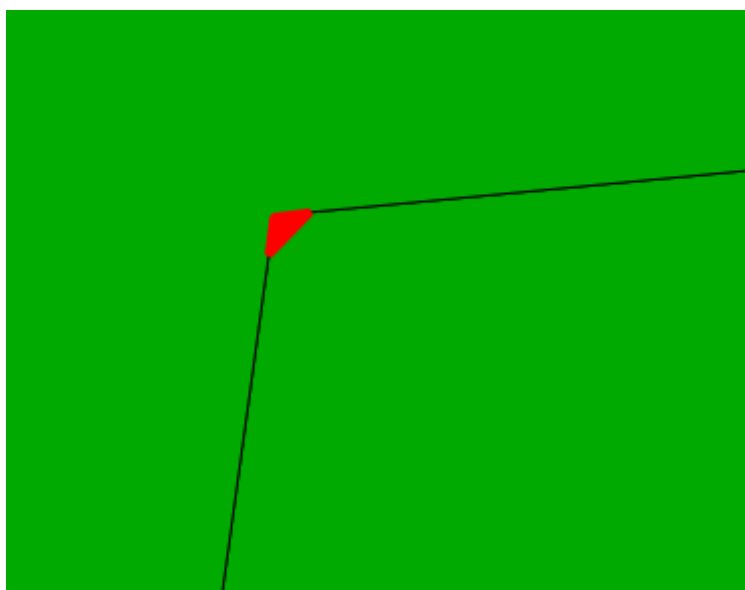
Obr. 7.3.3: Překrývající se objekty – ukázka č. 3 - kontrola pomocí QGIS



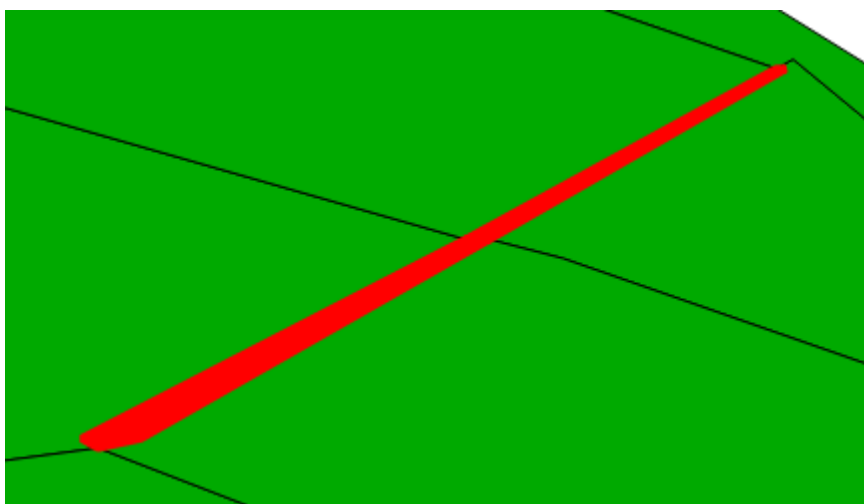
Obr. 7.3.4: Objekt mající mezeru – ukázka č. 1- kontrola pomocí QGIS



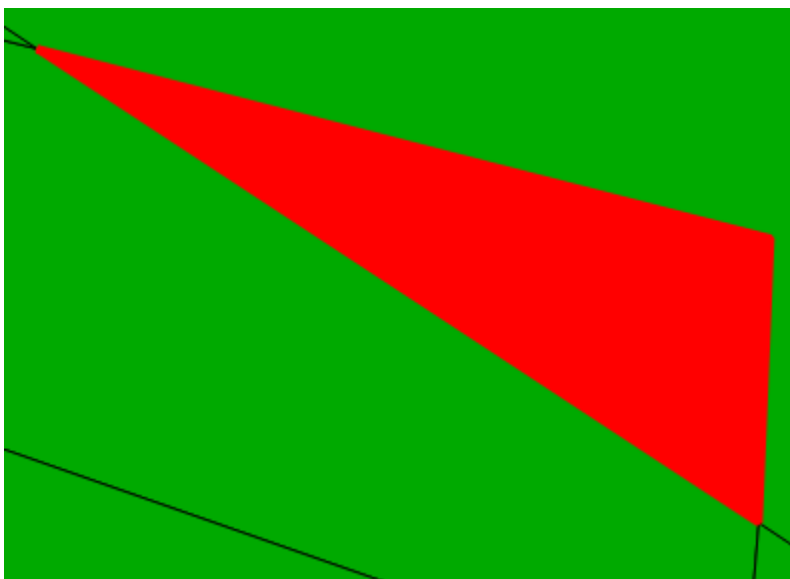
Obr. 7.3.5: Objekt mající mezeru – ukázka č. 2- kontrola pomocí QGIS



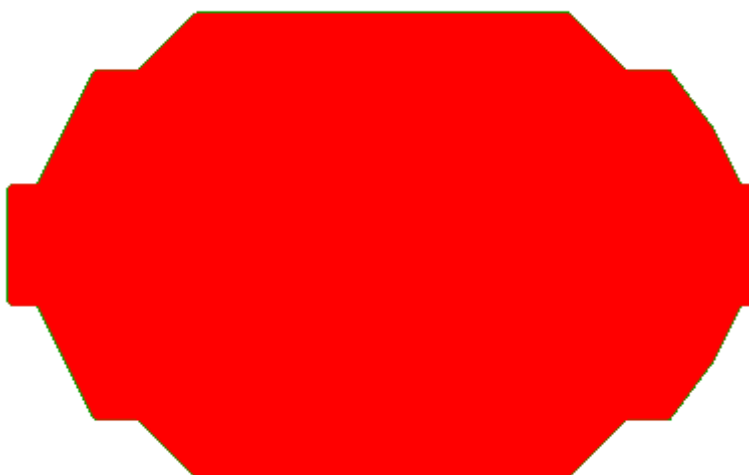
Obr. 7.3.6: Objekt mající mezeru – ukázka č. 3 - detail- kontrola pomocí QGIS



Obr. 7.3.7: Duplicitní objekt – ukázka č. 1- kontrola pomocí QGIS

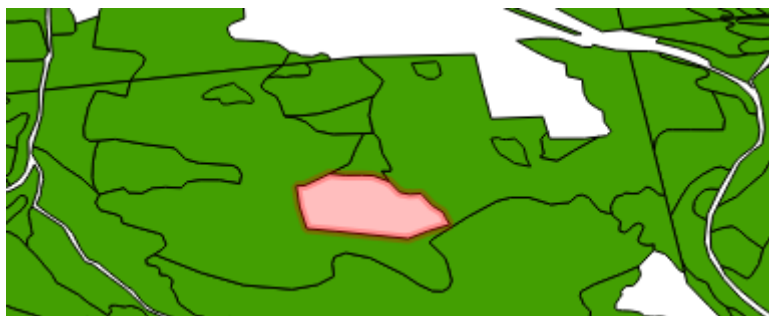


Obr. 7.3.8: Duplicitní objekt – ukázka č. 2- kontrola pomocí QGIS



Obr. 7.3.9: Duplicitní objekt – ukázka č. 3- kontrola pomocí QGIS

Na závěr bych chtěla říci, že tato topologická pravidla nejsou vhodná pro všechna data. Záleží na konkrétních datech. Zde jsem vybrala správně, i když jsem zde po kontrole topologie v datech našla opravu „díry“, kterou bych za chybu nepovažovala, viz. obr. 7.3.10.



Obr. 7.3.10: Nalezená chyba „díry“

Obr. 7.3.10 je v měřítku 1:10 000, je zde znázorněna chyba „díry“, která je ale s největší pravděpodobností v datech opravdu záměrně. Tak velká díra může znamenat například rybník. Ve zkontrolovaných datech jsem našla ještě 4 obdobné díry. Zde to považuji za chybnou kontrolu topologie, avšak k velkému množství děr, které jsou opravdu chybové, považuji toto topologické pravidlo za velice důležité.

8. PostGIS

PostGIS (verze 2.1.2, 17.8.2013) je open source knihovna [10], která umožňuje rozšíření objektově-relačního databázového systému PostgreSQL (verze 9.3.4, 20.3.2014), který tak umožňuje ukládání geografických objektů [12], díky tomu může být použit jako prostorová databáze pro GIS. Vývoj PostGIS byl zahájen v roce 2001 společností Refractions Research jako projekt open source technologie prostorové databáze. PostGIS je zveřejněn pod licencí GNU General Public License. [13]

PostgreSQL je vydáván pod licencí typu MIT, lze ho používat v operačních systémech Linux, UNIX a Windows, má vestavěné tři jazyky SQL, PL/PGSQL a C/C++, avšak když si uživatel nainstaluje další prostředí, jako např. Perl, Python nebo Java, může pracovat v dalších jazycích, např. PL / Perl, PL / Python nebo PL / Java [10]. Rozhraní pro správu databáze PostgreSQL jsem použila pgAdmin III (verze 1.18.1, 11.10.2013).

PgAdmin III je open source grafické administrační rozhraní pro PostgreSQL, je volně dostupný pod licencí The PostgreSQL Licence a je napsán v jazyce C++.

Program QGIS použiji pro nahrání dat do databáze.

8.1 Kontrola topologie

V programu PostGIS je možné si vytvořit svůj vlastní program s využitím funkcí, které program nabízí. Což jsem udělala, abych zajistila topologicky čistá data. Ve výsledku mám dva příkazy, pomocí kterých lze najít topologické chyby. **Modře** je dopsán dodatečný popis funkce. **Zeleně** je označeno, co musí uživatel změnit pro svoje konkrétní data.

marketa je tabulka, kterou jsem si vytvořila a nahrála do ní data. Každý uživatel si může vytvořit svoji tabulku se svým názvem, proto je následně potřeba přepsat zeleně označenou tabulku na svůj název.

Příkaz

Spustí funkci pro nalezení duplicitních polygonů, překrývajících se polygonů a polygonů, které mají od sebe vzdálené hranice do 2 metrů:

Pro spuštění funkce stačí vložit:

```
select * from marketa_kontrola_topologie()
```

funkce:

```
CREATE OR REPLACE FUNCTION marketa_kontrola_topologie()  
  RETURNS table(error_type character varying, id1 integer, id2  
integer) AS --hlavička funkce  
$BODY$  
  
DECLARE  
  points record;  
  totcount bigint;  
  rowcount bigint; --počítání, kolik už bylo zpracováno objektů  
  sql text;  
  
BEGIN  
  
  rowcount := 0;  
  FOR points IN EXECUTE 'SELECT * FROM marketa'  
--hlavní cyklus, projíždí celou tabulku marketa  
  LOOP
```

```

        IF rowcount % 1000 = 0 THEN
            RAISE NOTICE '% out of % processed', rowcount, totcount;
--počítání zpracovaných objektů
        END IF;

        rowcount := rowcount + 1;
    BEGIN
        return query SELECT cast('equal' as character
varying(255)) as error_type, m.id as id1, points.id as id2 from
marketa as m where ST_Equals(points.geometry,geometry) and points.id
!= id and rostlina = points.rostlina;
-- zjišťuje, zda existují duplicitní polygony
        return query SELECT cast('overlap' as character
varying(255)) as error_type, m.id as id1, points.id as id2 from
marketa as m where ST_Overlaps(points.geometry,geometry) and
points.id != id and rostlina = points.rostlina;
-- zjišťuje, zda se polygon protíná s jinými polygony
        return query SELECT cast('distance' as character
varying(255)) as error_type, m.id as id1, points.id as id2 from
(SELECT *, ST_Distance_Spheroid(points.geometry,geometry,
'SPHEROID["WGS 84",6378137,298.257223563]') as distance from marketa
where ST_Dwithin(points.geometry,geometry,0.001)) as m
        where points.id != m.id and m.rostlina = points.rostlina
and m.distance < 2 and m.distance != 0;
--kontroluje vzdálenost od polygonu (musí být větší než 2 metry nebo
musí být 0), zde je také použita funkce ST_DWithin(), která vybere
polygony, které by mohly splňovat danou podmínku a přesně se spočítá
až přes ST_Distance_Spheroid()

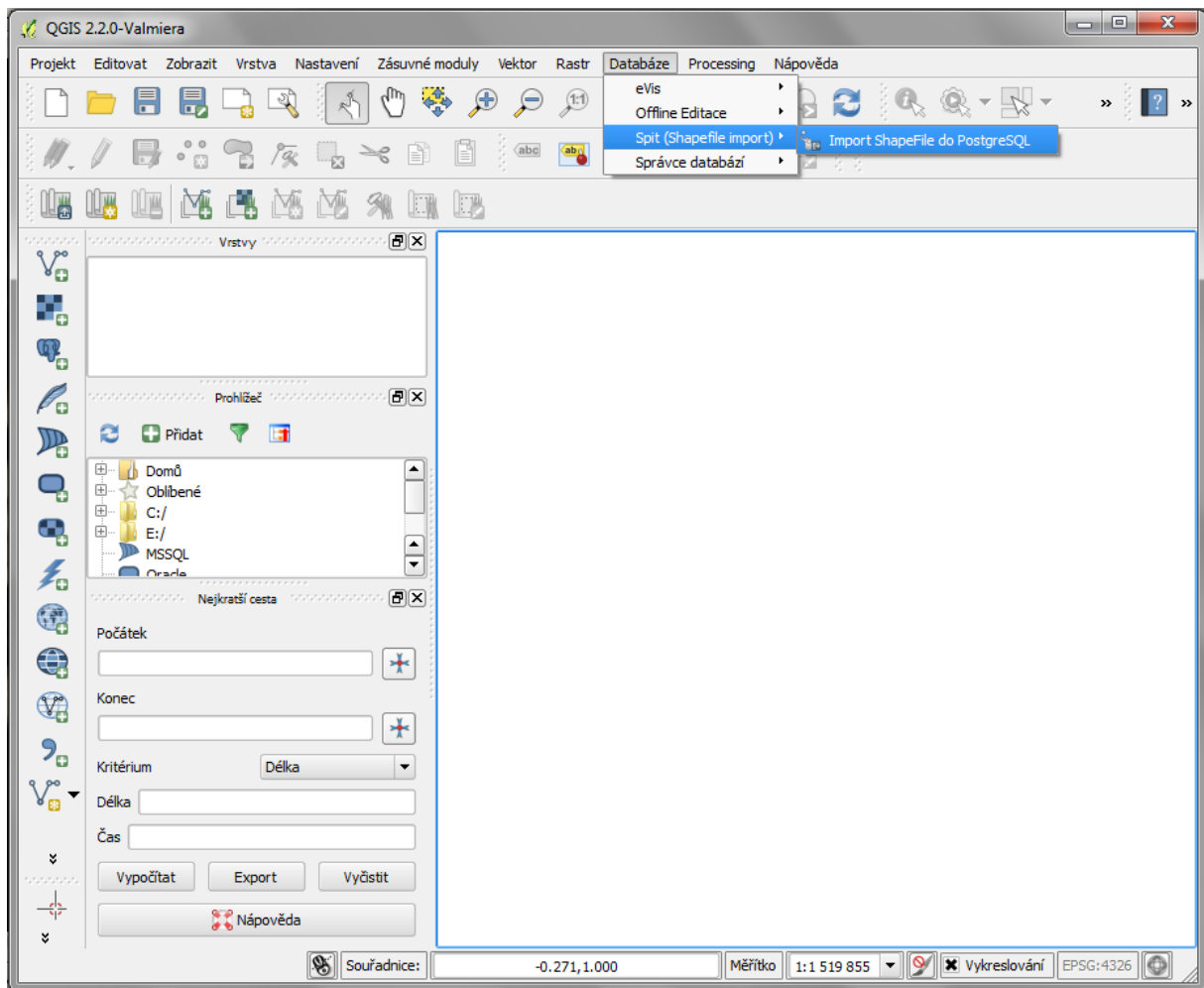
    EXCEPTION when others then
        RAISE NOTICE 'Chyba GEOS';
--zachycuje případné chyby hlavně v knihovně GEOS
    END;
    END LOOP;
    --vrací 'OK';

END;
$BODY$
LANGUAGE plpgsql VOLATILE STRICT
COST 100;

```

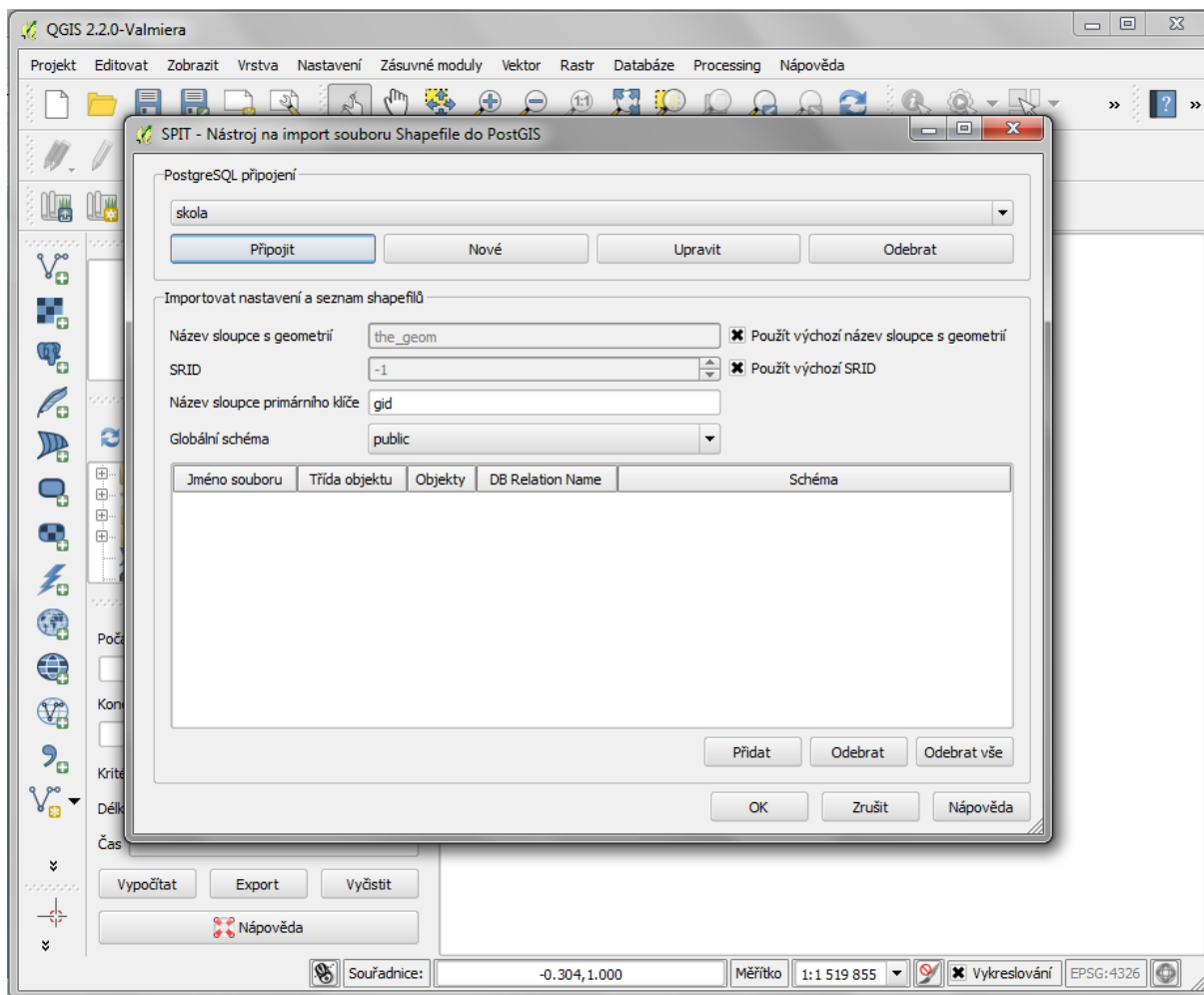
8.2 Manuál pro kontrolu topologie

Nejdříve pomocí programu QGIS importují data do databáze. Podle obr. 8.2.1 lze provést import. „Databáze“ – „Spit(Shapefile import)“ – „Import Shapefile do PostgreSQL“




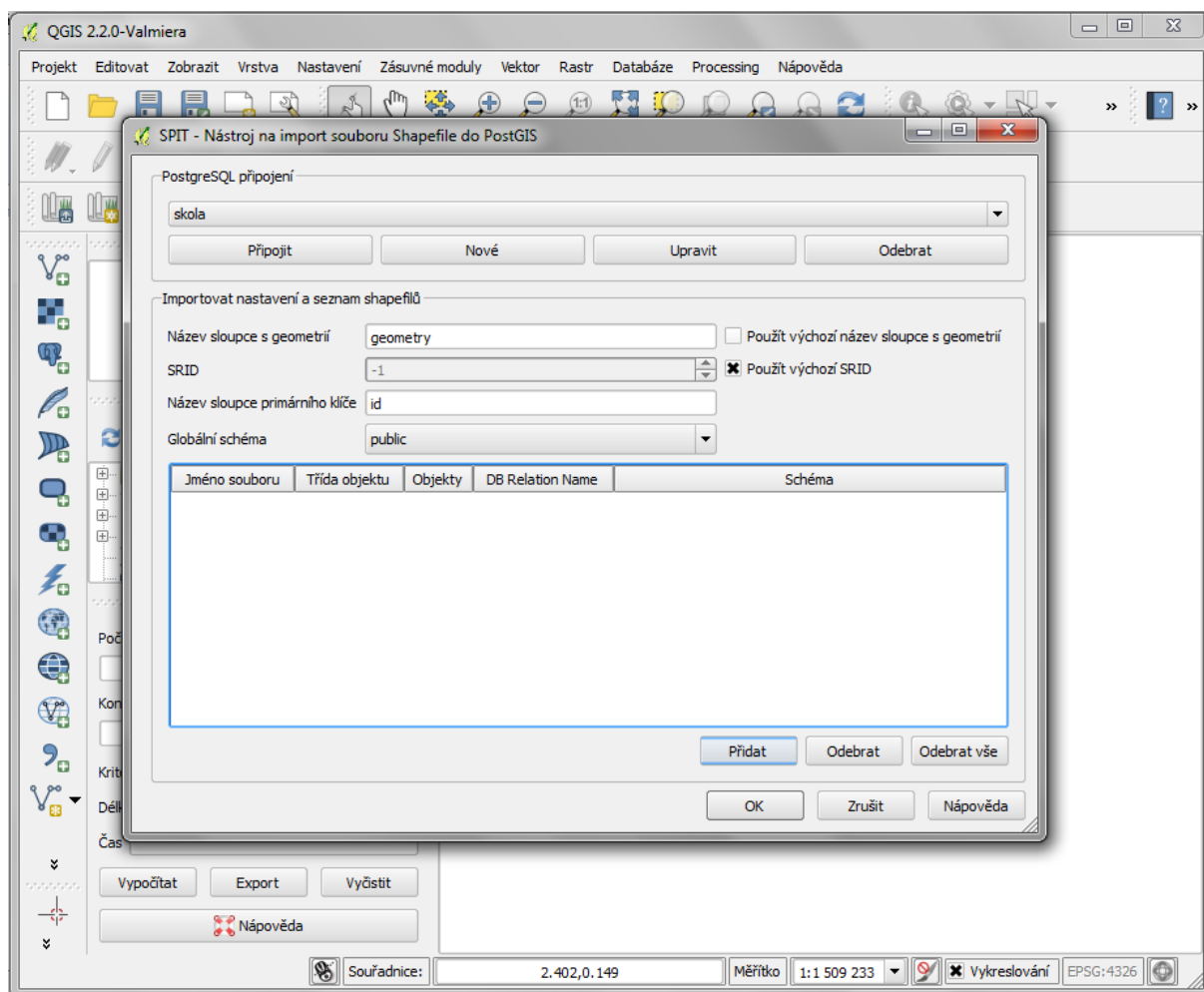
Obr. 8.2.1: Otevření importu shapefile

Poté zvolím nastavení připojení, viz. obr. 8.2.2.

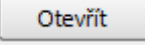


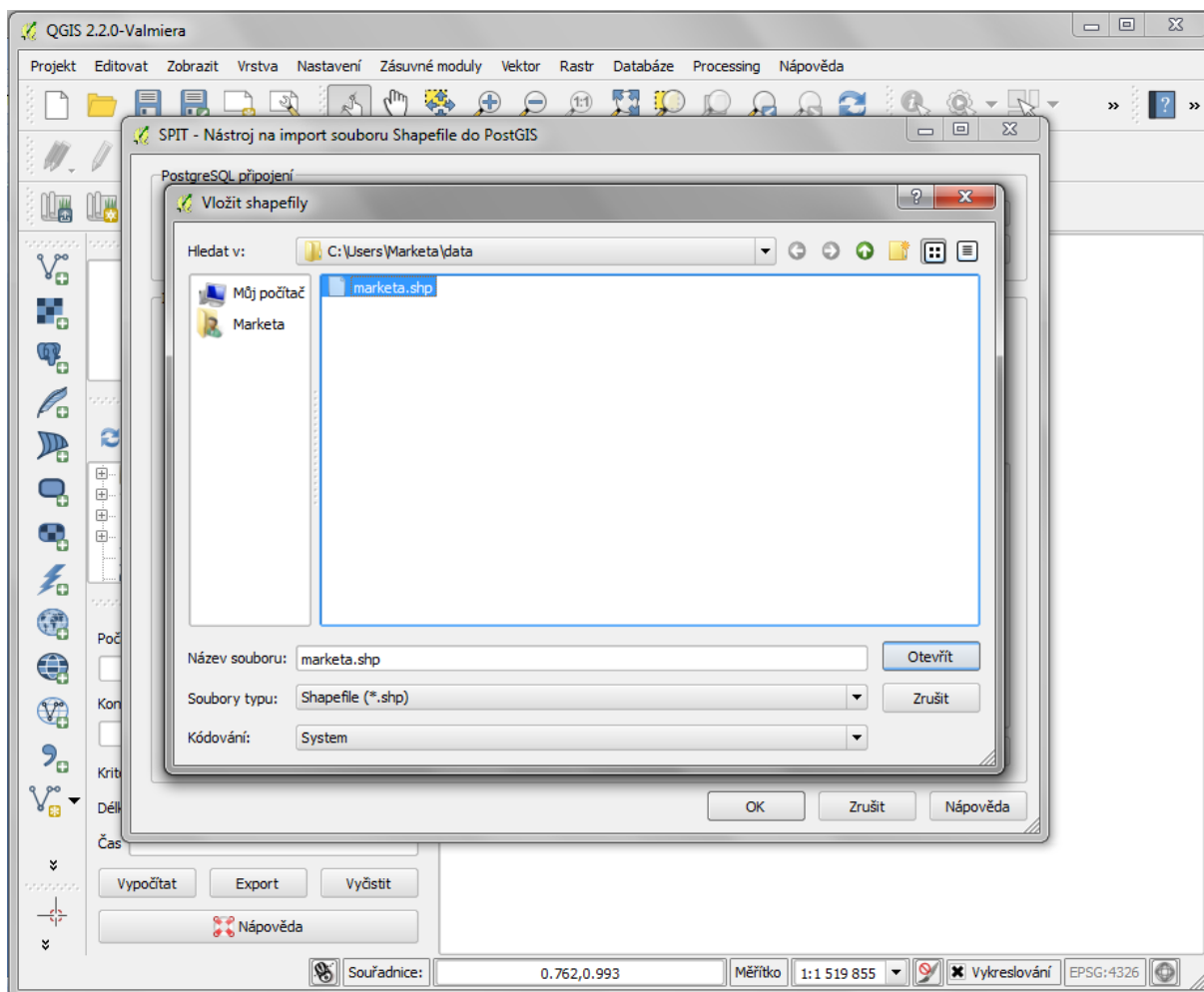
Obr. 8.2.2: Připojení k databázi

Dále si zvolím název sloupce s geometrií – geometry, název sloupce primárního klíče – id a globální schéma – public. Přehledně je nastavení možné vidět na obr. 8.2.3. Následně pro přidání shapefile klepnu na .



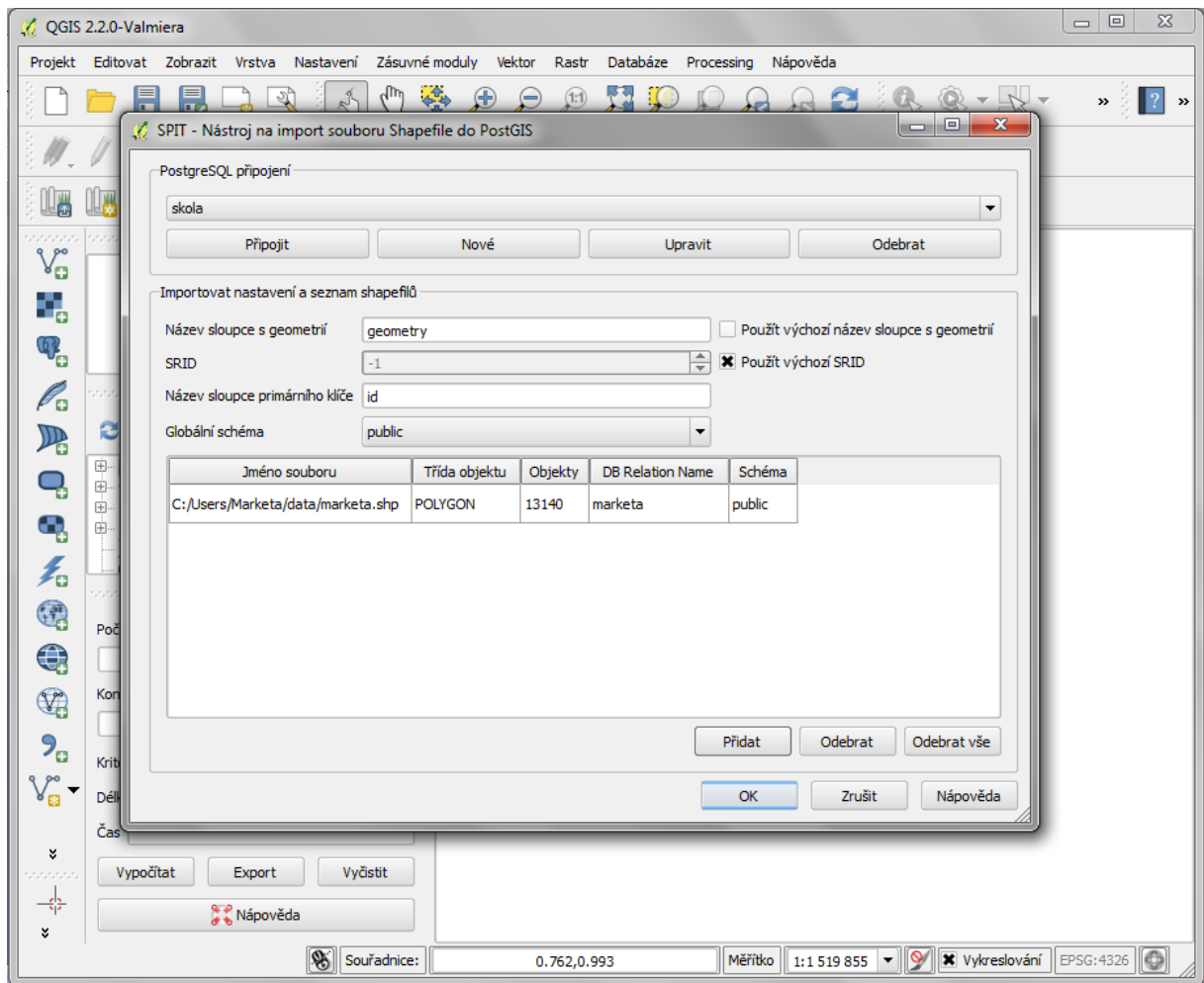
Obr. 8.2.3: Nastavení importu

Po otevření okna najdu složku, ve které mám uložený shapefile. Vyberu soubor a pomocí  soubor nahraju do nástroje pro import, viz. obr. 8.2.4.




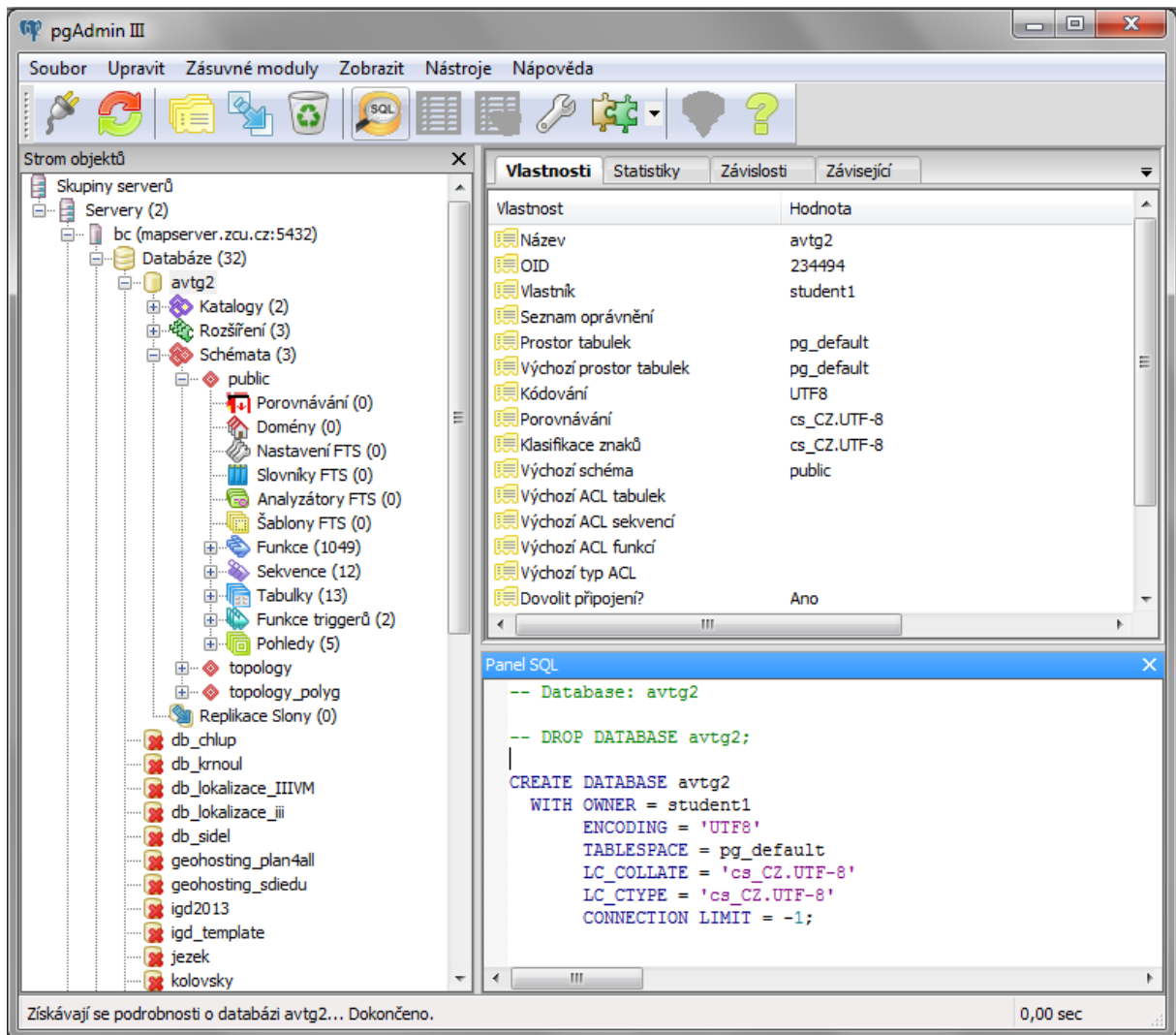
Obr. 8.2.4: Vybrání shapefile

Teď mám vše nastaveno pro import dat do PostGIS. Stisknutím , viz. obr. 8.2.5, importuji data.



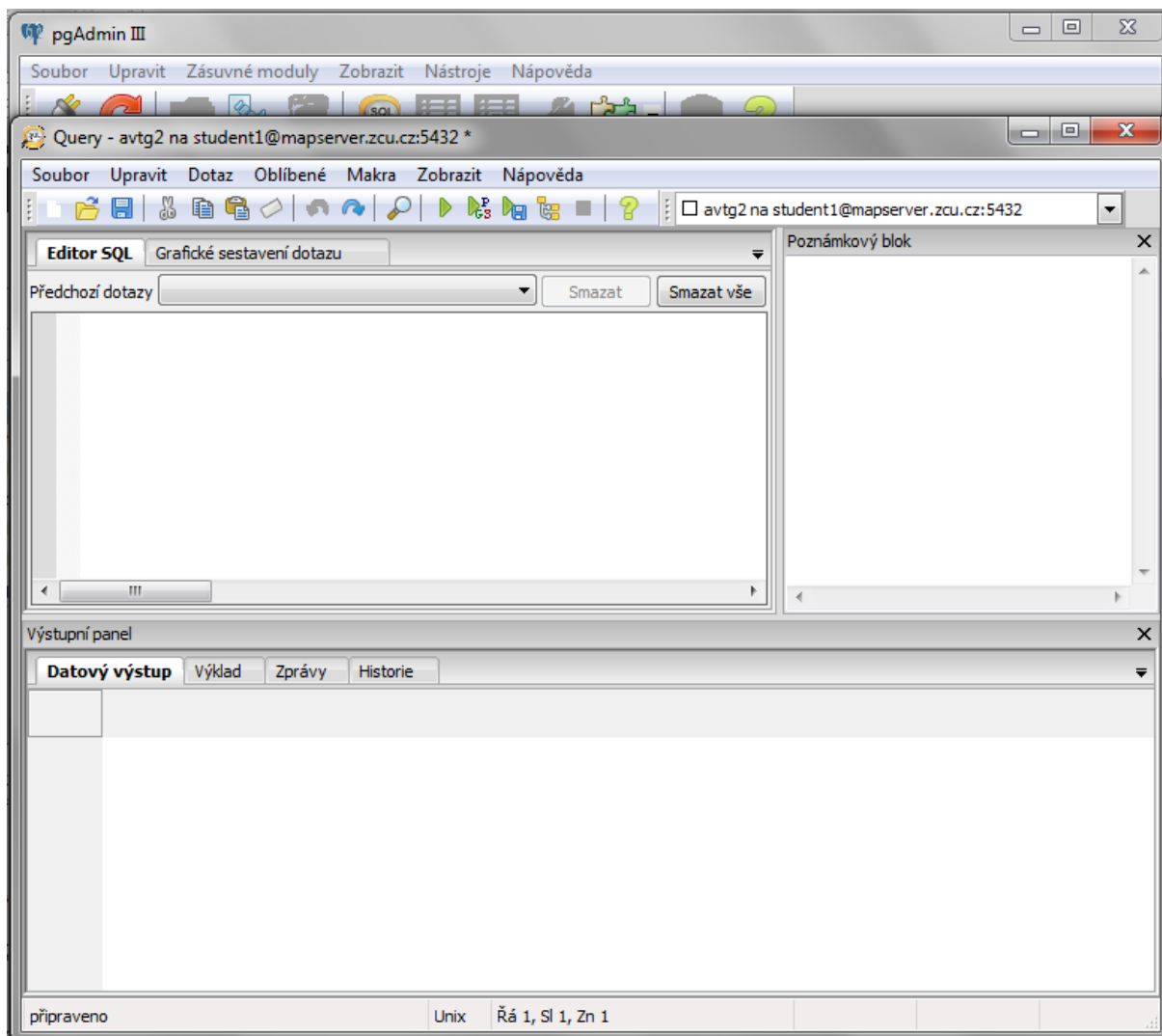
Obr. 8.2.5: Import dat do PostGIS

Spustím pgAdmin III a připojím se k databázi, viz. obr. 8.2.6. Pro spuštění okna, kde budu zadávat příkazy, klepnu na  .



Obr. 8.2.6: Spuštění editoru SQL v pgAdmin III

Dostala jsem se do editoru SQL, viz. obr. 8.2.7.



Obr. 8.2.7: Spuštěný editor SQL

Pro zjištění dalších topologických chyb napíši (zkopíruji), viz. obr. 8.2.13, do okna „Editor SQL“ funkci:

```
CREATE OR REPLACE FUNCTION marketa_kontrola_topologie()
  RETURNS table(error_type character varying, id1 integer, id2 integer) AS
  $BODY$

DECLARE
  points record;
  totcount bigint;
  rowcount bigint;
  sql text;

BEGIN

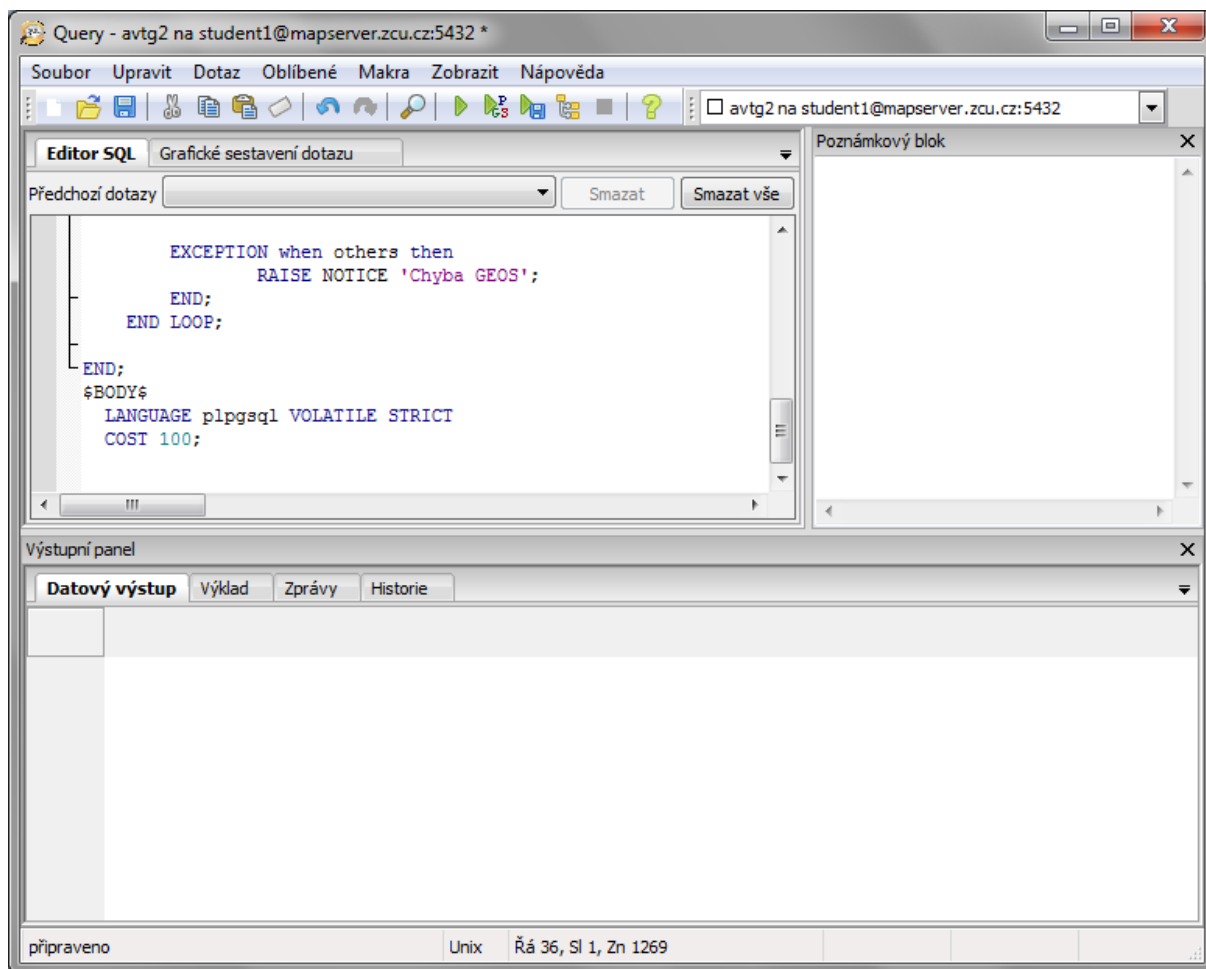
  rowcount := 0;
  FOR points IN EXECUTE 'SELECT * FROM marketa'
  LOOP

    IF rowcount % 1000 = 0 THEN
      RAISE NOTICE '% out of % processed', rowcount, totcount;
    END IF;


    rowcount := rowcount + 1;
    BEGIN
      return query SELECT cast('equal' as character varying(255)) as error_type, m.id
as id1, points.id as id2 from marketa as m where ST_Equals(points.geometry,geometry) and
points.id != id and rostlina = points.rostlina;
      return query SELECT cast('overlap' as character varying(255)) as error_type,
m.id as id1, points.id as id2 from marketa as m where ST_Overlaps (points.geometry,geometry)
and points.id != id and rostlina = points.rostlina;
      return query SELECT cast('distance' as character varying(255)) as error_type,
m.id as id1, points.id as id2 from (SELECT *, ST_Distance_Spheroid(points.geometry,geometry,
'SPHEROID["WGS 84",6378137,298.257223563]') as distance from marketa where
ST_Dwithin(points.geometry,geometry,0.001)) as m
      where points.id != m.id and m.rostlina = points.rostlina and m.distance < 2 and
m.distance != 0;

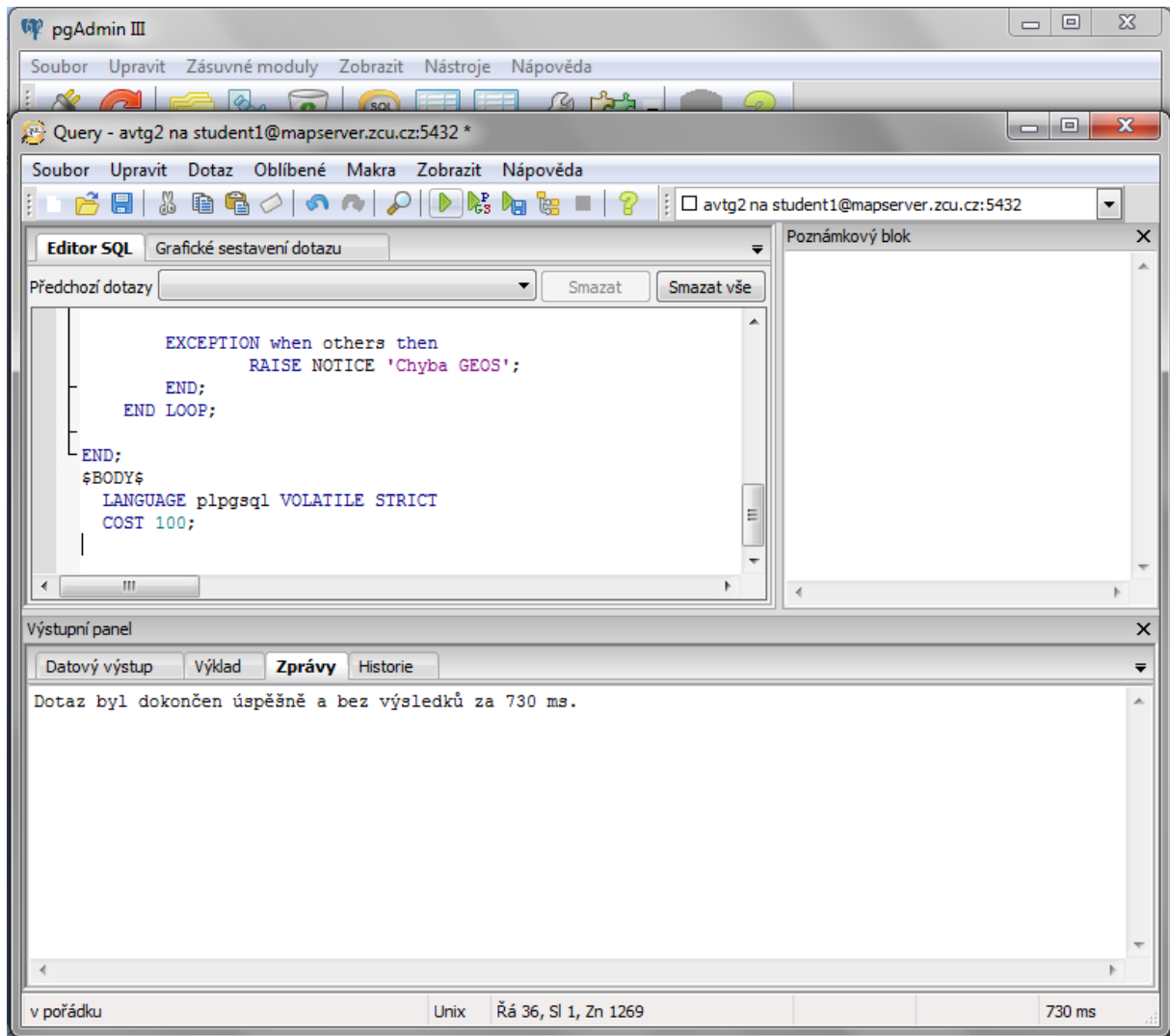
      EXCEPTION when others then
        RAISE NOTICE 'Chyba GEOS';
    END;
  END LOOP;

END;
$BODY$
LANGUAGE plpgsql VOLATILE STRICT
COST 100;
```



Obr. 8.2.8: Vložení funkce

Pomocí  spustím dotaz, čímž se mi vytvoří funkce, viz. obr. 8.2.9.

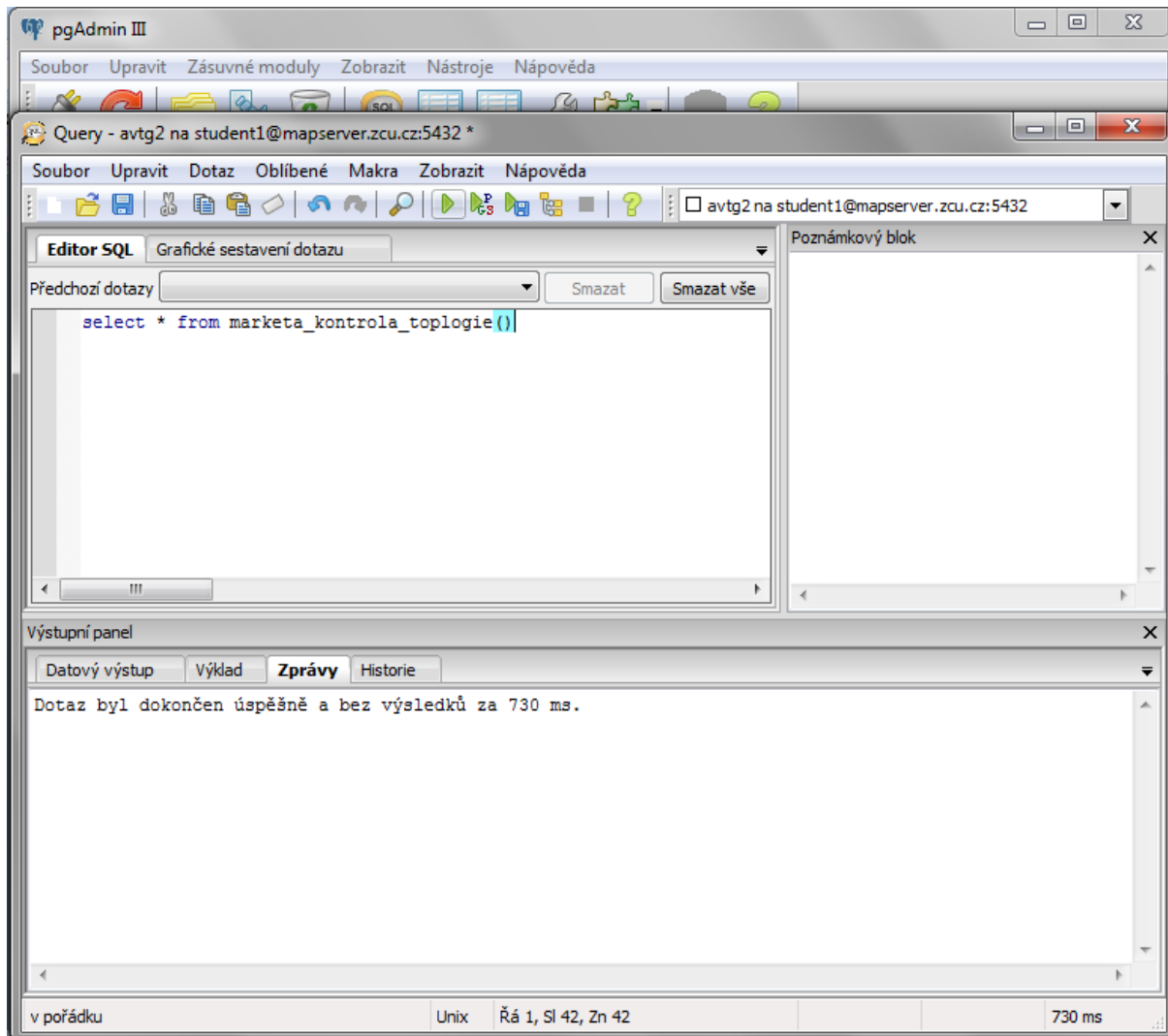


Obr. 8.2.9: Vytvoření funkce

Po vytvoření funkce text v poli „Editor SQL“ smažu a nahradím jej příkazem:

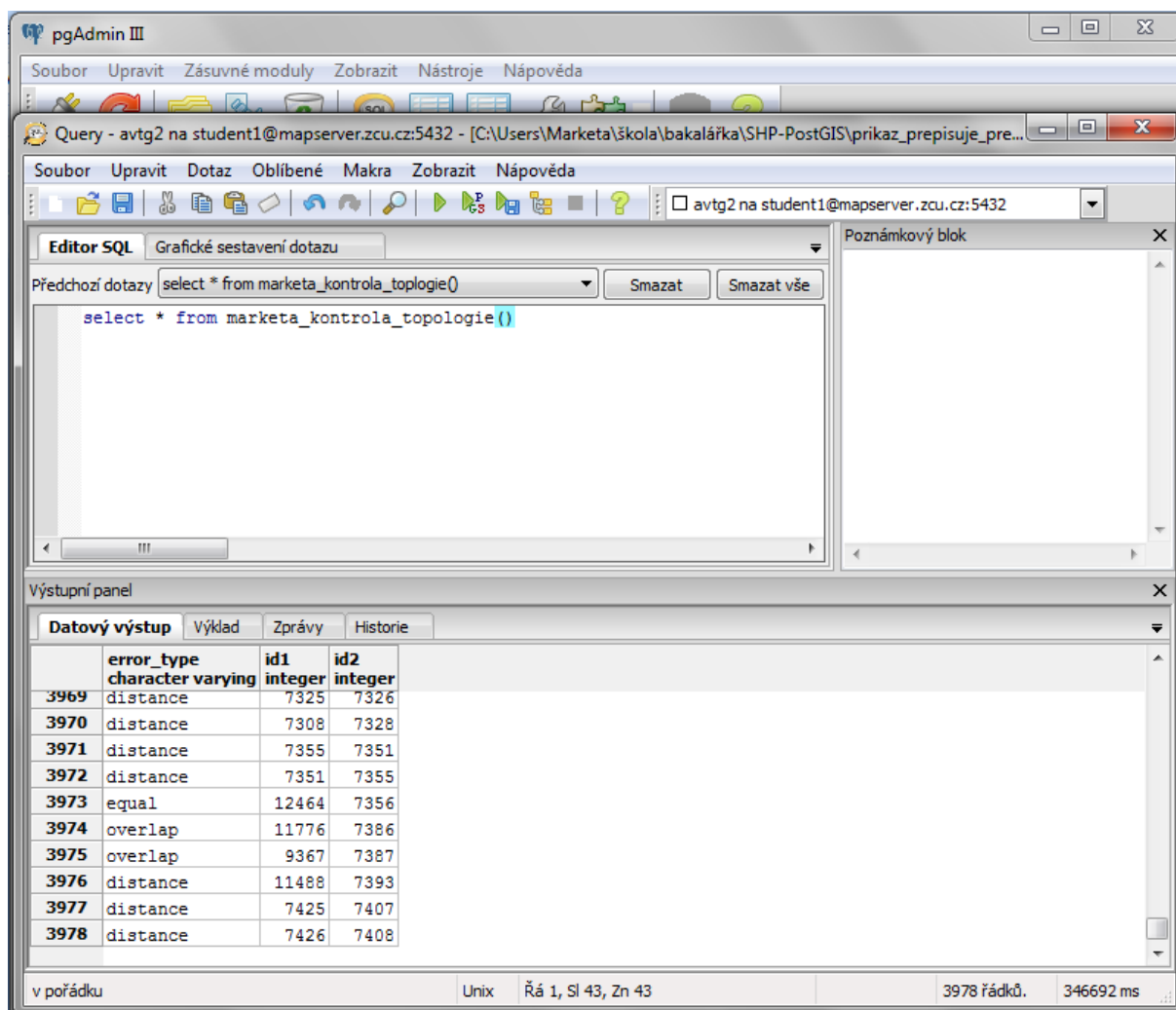
```
select * from marketa_kontrola_toplogie()
```

Poté opět kliknutím na  spustím příkaz, viz. obr. 8.2.10.



Obr. 8.2.10: Vložení a spuštění příkazu

Následně ve výstupním panelu v okně „Datový výstup“ vidím nalezené chyby, viz. obr. 8.2.11.



Obr. 9.2.11: Nalezení chyb

8.3 Výsledek ověření topologie

Celkem bylo nalezeno 3978 topologických chyb, podrobnější informace o počtu a typu detekovaných topologických chyb lze najít v tab. 8.3.1.

„Distance“ znamená chyba ve vzdálenosti od hranic polygonů, jinak řečeno nalezení mezery mezi polygony do 2 metrů. „overlap“ naopak znamená chybu v překrývání polygonů a „equal“ znamená duplicitní prvky. Ve sloupcích „id1 integer“ a „id2 integer“ jsou zapsány čísla, která označují polygony. Při importu dat si PostGIS označil každý polygony číslem. Ve výstupu jsou tedy zapsány vždy dvě čísla – dva polygony, které mají mezi sebou konflikt. Např. na obr. 8.3.1 vidím, že v první řádce je chyba „distance“, tedy mezera a tato mezera je mezi polygony číslo 180 a 175.

	error_type	id1	id2
1	distance	180	175

Obr. 8.3.1: Datový výstup 1

Jelikož se porovnává každý polygon s každým polygonem, ve výsledku dojde k tomu, že chyby jsou nalezené dvakrát, což je dobře viditelné na obr. 8.3.2. Poprvé je nalezena chyba č. 1 912, tato chyba je nalezena u polygonu číslo 12 659 – má stejnou geometrii jako polygon č. 12 657. Poté je nalezena chyba č. 1 921, tato chyba je nalezena u polygonu č. 12 657 – má stejnou geometrii jako polygon č. 12 659. Mohu říci, že tato chyba je jedna a to duplicitní prvek k původnímu prvku.

Datový výstup				Výklad	Zprávy	Historie
	error_type character varying	id1 integer	id2 integer			
1912	equal	12659	12657			
1913	distance	11498	12657			
1914	distance	12472	12657			
1915	distance	12656	12657			
1916	distance	12658	12657			
1917	distance	11498	12658			
1918	distance	12473	12658			
1919	distance	12657	12658			
1920	distance	12659	12658			
1921	equal	12657	12659			

Obr. 8.3.2: Datový výstup 2

V tabulce 8.3.1 je napsáno, kolik a jaký typ chyby byl nalezen.

Tab. 8.3.1: Souhrn nalezených topologických chyb

	duplicita	mezery	překryt	chyby	objekty
Celkem	120	2030	1828	3978	13140

Do tabulky 8.3.2 jsem vložila poloviční počet chyb. Jak jsem již vysvětlila, každá chyba byla nalezena dvakrát a pro porovnání s programem QGIS by byla tabulka 8.3.1 nevhodná.

Tab. 8.3.2: Souhrn skutečného počtu topologických chyb

	duplicita	mezery	překryt	chyby	objekty
Celkem	60	1015	914	1989	13140

Bohužel prakticky neexistuje žádný všestranný nástroj pro vizualizaci daných dotazů. Avšak mohu říci, že duplicitní prvky a překryty byly nalezeny všechny. Dále mohu jen tvrdit, že všechny mezery a díry, které jsou důležité pro topologii, byly také nalezeny. Zvolila jsem mezeru do velikosti 2 metrů, avšak každý uživatel může tuto hranici změnit. Touto hranicí jsem si zajistila, že nebyly nalezeny jako chyby velké díry, které mohou znázorňovat např. rybník.

9. Porovnání

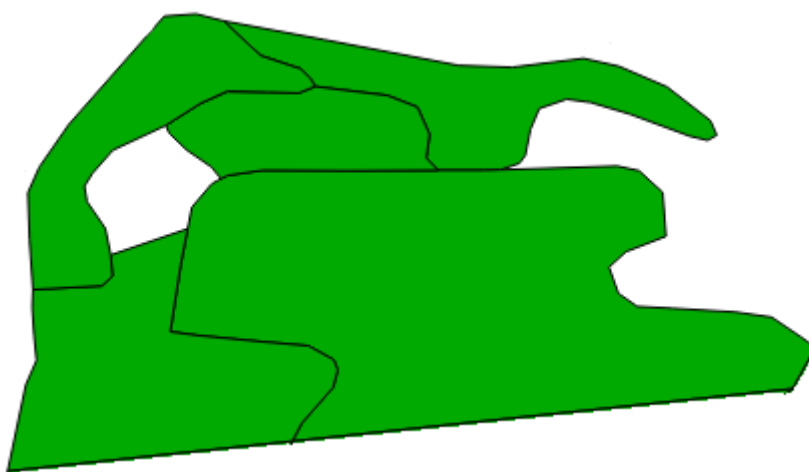
V tabulce 9.1 je souhrn nalezených chyb. Program PostGIS našel více chyb, důvod je ten, že každý program pracuje částečně odlišně. Pouze počet duplicitních prvků je stejný.

Tab.9.1: Porovnání počtu topologických chyb

	duplicita	mezery/díry	překryt	chyby
Celkem QGIS	60	619	638	1047
Celkem PostGIS	60	1015	914	1989

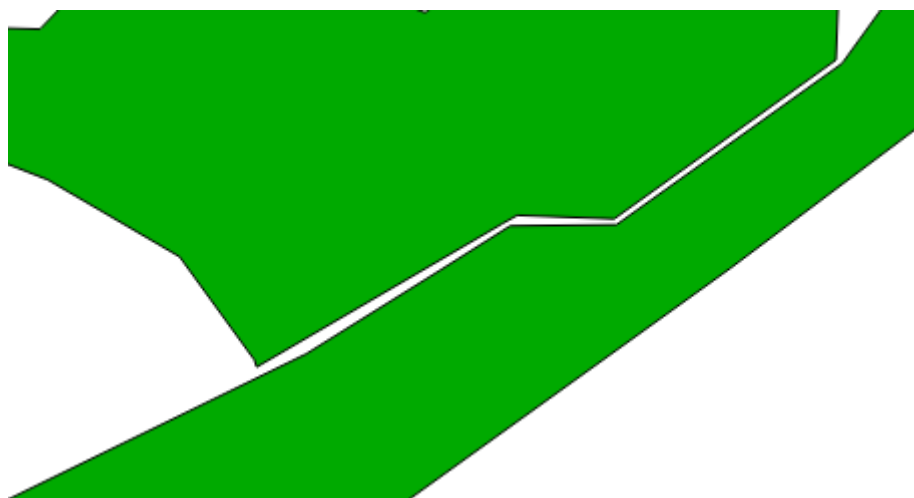
Duplicitní prvky oba programy vnímají jako objekty, které mají stejnou geometrii, proto musí být výsledek stejný.

Program QGIS vymezuje pojem díry jako prázdnou plochu mezi dvěma hranicemi polygonů, to znamená, že kolem díry jsou zanesená data. Díra je mezi polygony, viz. obr. 9.1.



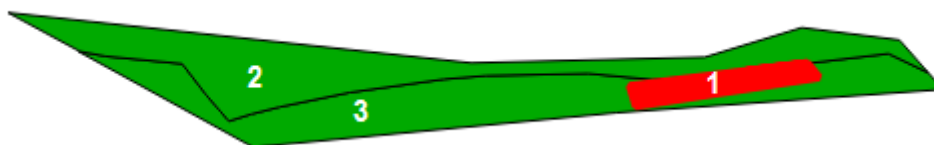
Obr. 9.1: Příklad díry

Avšak pomocí PostGIS zjišťují mezery (zároveň i díry) s maximální velikostí 2 metry. Tzn. že pokud mám mezeru nebo díru do dané vzdálenosti, vyhodnotí se jako chyba. Např. na obr. 9.2, který je v měřítku 1:1 000, je mezera, která se vyhodnotí jako chyba.



Obr. 9.2: Příklad mezery

Vymezení překrytu dvou polygonů je téměř totožné. Polygony se překrývají, jestliže jeden polygon obsahuje část plochy nebo celou plochu druhého polygonu, ale zároveň nemají stejnou geometrii. Avšak je zde rozdíl interpretaci chyb, tuto odlišnost popíši na obr. 9.3. Program QGIS vypíše jednu chybu a to červeně znázorněný polygon č.1, který překrývá další dva zelené polygony. Ovšem PostGIS kontroluje všechny polygony se všemi, což znamená, že zde najde dvě chyby: první chyba je překrytí polygonu č.2 polygonem č.1 a druhá chyba je překrytí polygonu č.3 polygonem č.1.



Obr. 9.3: Příklad překrytu

Kvůli těmto rozdílům jsem pomocí PostGIS našla více chyb.

Na závěr bych chtěla dodat, že PostGIS se jeví jako mnohem vhodnější způsob hledání chyb než pomocí programu QGIS. Sice je zbytečné, že pomocí PostGIS zjistím chyby dvakrát, ale hledání mezer je zde mnohem účinnější pro další analýzy s daty než pouhé nalezení děr.

10. Vybrané pravidla a algoritmy pro čištění prostorových dat

Níže jsou stručně popsána pravidla a algoritmy, které jsem využila při kontrole topologie.

10.1 QGIS

Pravidlo nesmí mít duplikáty (must not have duplicates)

„Dvě geometrie (polygony) jsou topologicky stejné, právě když se jejich vnitřky protínají a hranice jsou stejné a žádná část vnitřku nebo hranice jedné geometrie neprotíná vnějšek druhé.“ [19]

Úplný algoritmus je možné vidět na:

<https://github.com/qgis/QGIS/search?q=must+not+have+duplicates&type=Code>

Pravidlo nesmí mít mezery (must not have gaps)

„Sousední polygony by neměli tvořit hranici.“[4] Mezi společnou hranicí dvou geometrií by neměla být mezera. Pokud existuje společná linie pro dvě geometrie, neměl by být mezi koncovým a počátečním bodem společné linie bod nebo více bodů, které patří pouze jedné geometrii.

Úplný algoritmus je možné vidět na:

<https://github.com/qgis/QGIS/search?q=must+not+have+gaps&type=Code>

Pravidlo nesmí se překrývat (must not overlap)

„Objekty se překrývají, pokud mají některé, ale ne všechny body, společné. Zároveň mají stejný rozměr a průsečík vnitřku dvou geometrií má stejný rozměr jako samotné geometrie.“ [20]

Úplný algoritmus je možné vidět na:

<https://github.com/qgis/QGIS/search?q=must+not+overlap&type=Code>

10.2 PostGIS

Funkce ST_Equals

„Dvě geometrie (polygony) jsou topologicky stejné, právě když se jejich vnitřky protínají a hranice jsou stejné a žádná část vnitřku nebo hranice jedné geometrie neprotíná vnějšek druhé.“ [19]

Funkce:

```
„CREATE METHOD ST_Equals  
  (ageometry ST_Geometry)  
  RETURNS INTEGER  
  FOR ST_Geometry  
  RETURN SELF.ST_SymDifference(ageometry).ST_IsEmpty()“[21]
```

Popis metody:

„Metoda testuje, zda je hodnota ST_Geometry prostorově rovna s jinou hodnotu ST_Geometry.

Metoda ST_Equals (ST_Geometry) má jako vstupní parametr hodnotu geometrie ST_Geometry.

Po zvolení metody ST_Equals (ST_Geometry) se vrátí výsledek jako výraz hodnoty SELF.ST_SymDifference (ageometry). ST_IsEmpty ().“ [21]

Další informace o algoritmu je možné zjistit na:

<http://tsusiatsoftware.net/jts/javadoc/com/vividsolutions/jts/geom/Geometry.html>

Funkce ST_Overlaps

„Objekty se překrývají, pokud mají některé, ale ne všechny body, společné. Zároveň mají stejný rozměr a průsečík vnitřku dvou geometrií má stejný rozměr jako samotné geometrie.“ [20]

Funkce:

```
„CREATE METHOD ST_Overlaps
  (ageometry ST_Geometry)
  RETURNS INTEGER
  FOR ST_Geometry
  RETURN
    CASE
      WHEN (SELF.ST_Dimension() = 0 AND
            ageometry.ST_Dimension() = 0) THEN
        SELF.ST_Relate(ageometry, 'T*T***T**')
      WHEN (SELF.ST_Dimension() = 1 AND
            ageometry.ST_Dimension() = 1) THEN
        SELF.ST_Relate(ageometry, '1*T***T**')
      WHEN (SELF.ST_Dimension() = 2 AND
            ageometry.ST_Dimension() = 2) THEN
        SELF.ST_Relate(ageometry, 'T*T***T**')
      ELSE
        NULL
    END“ [21]
```

Popis metody:

„Metoda otestuje, zda se jedna geometrie překrývá s jinou geometrií.

Metoda ST_Overlaps (ST_Geometry) má jako vstupní parametr hodnotu geometrie ST_Geometry.

Po zvolení metody se vrátí:

- Je-li rozměr SELF roven 0 a rozměr *ageometry* se rovná 0 (nula), pak vrátí výsledek hodnotu výrazu: SELF.ST_Relate (*ageometry*, "T * T *** T **").
- Je-li rozměr SELF roven 1 a rozměr *ageometry* je rovna 1, pak vrátí výsledek hodnotu výrazu: SELF.ST_Relate (*ageometry*, '1 * T *** T ** ').
- Je-li rozměr SELF roven 2 a rozměr *ageometry* je roven 2, pak vrátí výsledek hodnotu výrazu: SELF.ST_Relate (*ageometry*, "T * T *** T **").
- V jiném případě vrátí hodnotu null.“ [21]

Další informace o algoritmu je možné zjistit na:

<http://tsusiatsoftware.net/jts/javadoc/com/vividsolutions/jts/geom/Geometry.html>

Funkce ST_Distance_Spheroid

Najde minimální vzdálenost na elipsoidu mezi všemi dvojicemi bodů, kde jeden bod je z jedné geometrie a druhý bod v páru je z druhé geometrie. [21]

Funkce:

```
„CREATE METHOD ST_Distance_Spheroid
  (ageometry ST_Geometry,
  aunit SPHEROID(<NAME>,<SEMI-MAJOR AXIS>,<INVERSE FLATTENING>))
```

```

RETURNS DOUBLE PRECISION
FOR ST_Geometry
BEGIN
    --
    -- See Description
    --
END“ [21]

```

Popis metody:

„Metoda `ST_Distance_Spheroid` (`ST_Geometry`) má jako vstupní parametr hodnotu geometrie `ST_Geometry`.

Po zvolení metody se vrátí:

- a) Je-li `SELF` prázdná množina, pak se vrátí hodnota null.
- b) Je-li `ageometry` prázdná množina, pak se vrátí hodnota null.
- c) Pokud se `SELF` a `ageometry` prolínají, pak se vrátí 0 (nula).
- d) V jiném případě vrátí vzdálenost mezi dvěma geometriemi, `SELF` a `ageometry`, je vypočtena v referenčním systému. Vzdálenost mezi dvěma body se vypočte s použitím elipsoidu.“ [21]

Celý algoritmus je možné vidět na:

https://svn.osgeo.org/postgis/tags/2.1.3/postgis/lwgeom_spheroid.c

Funkce `ST_Dwithin`

„Vrací true, pokud jsou geometrie od sebe v určité vzdálenosti.

Pro měření se použije `use_spheroid = true`, pro rychlejší kontrolu měření na sféroidu se použije `use_spheroid = false`“ [22]

Funkce:

```

„CREATE METHOD ST_DWithin
    (ageometry ST_Geometry)
    RETURNS INTEGER
    FOR ST_Geometry
    RETURN SELF.ST_Relate(ageometry)“ [21]

```

Popis metody:

„Metoda `ST_DWithin` (`ST_Geometry`) má jako vstupní parametr hodnotu geometrie `ST_Geometry`.

Po zvolení metody `ST_DWithin` (`ST_Geometry`) se vrátí výsledek jako výraz hodnoty `SELF.ST_Relate(ageometry)`.“ [21]

Algoritmus je možné najít na:

<https://svn.osgeo.org/postgis/tags/2.1.3/liblwgeom/measures.c>

11. Závěr

Cílem práce bylo vytvoření manuálů pro kontrolu topologie dat. Manuály jsou pro použití programu QGIS nebo knihovny PostGIS (v databázovém systému PostgreSQL). Hlavním faktorem při výběru programů k validaci topologie byla nulová pořizovací cena. Program QGIS i knihovna PostGIS (i databázový systém PostgreSQL) jsou volně dostupné open source. Vytvořené manuály jsou vytvořené především pro pracovníky IS Heracleum, kteří si chtějí zkontrolovat topologickou čistotu svých dat.

Avšak nejdříve před vytvořením manuálu jsem musela získat data. Data mi byla propůjčena od Agentury projektového a dotačního managementu Karlovarského kraje, jelikož jsem se účastnila projektu omezení výskytu invazních rostlin v Karlovarském kraji a zmapovala jsem část území. Data měla již na pohled (na <http://gis.kr-karlovarsky.cz/heracleum-public/Web/Mapa.aspx>) mnoho topologických chyb, proto jsem použila veškerá data z projektu, zkontroloval jejich topologickou čistotu a vytvořila manuál, díky kterému si můžou pracovníci zkontrolovat svoje data.

V první části práce jsem se zabývala již zmíněným projektem a IS Heracleum. Následně jsem popsala data, která byla zmapována. V datech jsem hledala chyby, které jsem následně po použití topologických pravidel programem QGIS zkontrolovala, ujišťovala jsem se, že program opravdu umí nalézt chyby. Pomocí knihovny PostGIS a databázového systému PostgreSQL jsem validovala data trochu jiným způsobem. Napsala jsem krátký program, díky kterému jsem našla chyby. V závěru jsem shrnula výsledky z obou metod validace dat. Vysvětlila jsem základní rozdíly v hledání chyb pomocí různých programů a popsala využití algoritmy.

Závěrem lze jen říci, že vytvořené manuály nejsou pouze pro pracovníky IS Heracleum, ale jsou vhodné pro každého, kdo si chce zkontrolovat svá data, zda obsahují topologické chyby.

12. Seznam použitých zdrojů

- [1] Agentura projektového a dotačního managementu Karlovarského kraje: Invazní rostliny [online]. [cit. 2014-02-10].
Dostupné z:
<<http://apdm.cz/projekty/Invazni-rostliny/>>
- [2] Documentation for QGIS 2.0: Topology 2014 [online]. [cit. 2014-02-10].
Dostupné z:
<http://docs.qgis.org/2.0/en/docs/gentle_gis_introduction/topology.html>
- [3] Heracleum. IS pro omezení výskytu invazních rostlin [online]. [cit. 2014-02-20].
Dostupné z :
<<http://gis.kr-karlovarsky.cz/heracleum/Default.aspx?stamp=635349146989383288>>
- [4] Topology Checker Plugin [online]. [cit. 2014-03-10].
Dostupné z:
<http://qgis.org/en/docs/user_manual/plugins/plugins_topology_checker.html>
- [5] Agentura projektového a dotačního managementu Karlovarského kraje: O nás [online]. [cit. 2014-03-10].
Dostupné z:
<<http://apdm.cz/o-nas/>>
- [6] KLIMEŠ, L. Slovník cizích slov. Praha: SPN, 1998, 862 s. ISBN 80-04-26710-6.
- [7] CHVALINA, J. Obecná topologie. Brno: Rektorát UJEP, 1984, 193 s.
- [8] KOLÁŘ, J. Geografické informační systémy 10. Praha: Vydavatelství ČVUT, 1997, 149 s.
- [9] LONGLEY, P. A. a spol. Geographic Informatic Systems And Science. Wiley, 2007, 517 s. ISBN 978-0470-87001-3.
- [10] OBE, Regina O., HSU, Leo s.: PostGIS in Action. Stamford: Manning Publications Co., ©2011. ISBN: 9781935182269
- [11] FUKS, P., T. PEŇÁZ, M. ŠEDĚNKOVÁ. Síťové analýzy v prostředí GIS [online]. 2006. [cit. 2014-04-30].
Dostupné z:
<http://gisak.vsb.cz/~pen63/Systemy_GIS_v_PO/Navod_ke_cvicenim.pdf>
- [12] NGUYEN, T. T. Indexing PostGIS Databases and Spatial Query Performance Evaluations. *International Journal of Geoinformatics*. 2009, roč. 5, č. 3, s. 1-9. ISSN 1686-6576
- [13] PostGIS, Home. What is PostGIS? [online]. [cit. 2014-05-3].
Dostupné z:
<<http://postgis.refractor.net/>>

[14] Ohori, K., A.; Ledoux, H.; Meijers, M. Validation and Automatic Repair of Planar Partitions Using a Constrained Triangulation. *the Journal of Photogrammetry. Remote Sensing and Geoinformation Processing*, 1-15, 2012.

[15] T-SOFT a.s., IS HERACLEUM [online]. [cit. 2014-04-15].

Dostupné z:

<http://www.tsoft.cz/sites/default/files/download/IS_HERACLEUM_case_study.pdf>

[16] van Oosterom P., Quak W., Tijssen T. About invalid, valid and clean polygons. *11th International Symposium on Spatial Data Handling. Developments in Spatial Data Handling*. 1-16, 2005, ISBN:978-3-540-22610-9.

[17] Welcome to the QGIS project: Discover QGIS [online]. 2014 [cit. 2014-05-6].

Dostupné z:

<<http://qgis.org/en/site/about/index.html>>

[18] HUANG, Fang, Dingsheng LIU, Xicheng TAN, Jian WANG, Yunping CHEN a Binbin HE. Explorations of the implementation of a parallel IDW interpolation algorithm in a Linux cluster-based parallel GIS. *Computers & Geosciences*. 2011, roč. 37, č. 4, s. 426-434. ISSN 0098-3004

[19] Davis M., Aquino J. Technical Specifications, *JTS Topology Suite*. 1-35, 2003. [cit. 2014-05-26].

Dostupné z:

<<http://www.vividsolutions.com/JTS/bin/JTS%20Technical%20Specs.pdf>>

[20] Application Programming Interface, Geometry. [online] 2006 [cit. 2014-05-30]

Dostupné z:

<<http://www.vividsolutions.com/jts/javadoc/com/vividsolutions/jts/geom/Geometry.html#overlaps%28com.vividsolutions.jts.geom.Geometry%29>>

[21] Ashworth M.: ISO/IEC WD 13249-3 Information technology - SQL Multimedia and Application Packages - Part 3: Spatial 3rd Edition. [online] 2004 [cit. 2014-06-01]

Dostupné z:

<<http://jtc1sc32.org/doc/N1101-1150/32N1107-WD13249-3--spatial.pdf>>

[22] PostGIS 2.1 Cheatsheet [online] 2013 [2014-06-01]

Dostupné z:

<http://www.postgis.us/downloads/postgis21_cheatsheet.pdf>