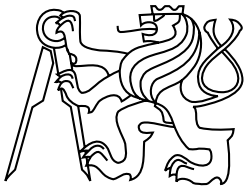
ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

Fakulta stavební
Katedra mapování a kartografie



Diplomová práce

Návrh modulu GRASSu pro import dat ve výměnném formátu ISKN

Květen 2005

Martin Landa

Poděkování

Bývá dobrým zvykem na začátku diplomové práce poděkovat svému nejbližšímu okolí za potřebnou podporu a to jak v osobní, tak v odborné rovině. V tomto ohledu chci poděkovat vedoucímu mé diplomové práce – Prof. Ing. Aleši Čepkovi, CSc. – a to především za jeho důvěru a neocenitelnou podporu nejen během vzniku této diplomové práce. Mé další profesní směřování tak dostává jasnější obrysy a to osobně považuji za velmi důležité. Taktéž děkuji Ing. Jiřímu Cajthamlovi, přál si to.

Nemohu také opomenout osobní podporu v rodinném prostředí, kterou jsem cítil během celého mého studia.

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně a uvedl jsem veškerou použitou literaturu a ostatní podklady.

Litoměřice, 31. května 2005

Podpis

Copyright © 2005 Martin Landa

Je dovoleno kopírovat, šířit a/nebo modifikovat tento dokument za podmínek licence GNU FDL, verze 1.2 nebo vyšších publikovaných nadací Free Software Foundation. Kopie licence je dostupná na adrese `http://www.gnu.org/licences/fdl.txt`. Toto dovolení je platné pouze pro státy, kde obsah díla daný jeho názvem není v rozporu se zákonem.

Předmluva

Svobodný software (*free software*) je bezesporu nepopiratelným fenoménem dnešních dnů. Přičemž dosahuje nejen pozoruhodné šíře a členitosti, ale také nesporné kvality a robustnosti, která k němu přivádí v houfech nové uživatele. Hlavním hybatelem jeho vývoje však není pouhá touha po zisku, ale daleko spíše ryzí tvořivost a také duch spolupráce, jakýsi *genius communitatis*. Stopa, kterou svobodný software či open source zanechal ve světě počítačových programů, je dnes už nepřehlédnutelná a stěží vymazatelná.

Fenomén svobodného softwaru neminul ani svět geoinformačních technologií. Důkazem toho snad mohou být takové projekty jako je GIS GRASS, QGIS a řada dalších. Navíc se objevuje mnoho nových projektů a stále více lidí se podílí na jejich vývoji. Velmi rychle také roste komunita uživatelů. Čím více lidí se do tohoto procesu zapojí (na jakékoliv úrovni – uživatelské a vývojářské), tím kvalitnější tyto produkty budou. Neváhejte tedy a zapojte se také!

Obsah

1 Úvod do problému **1**

2 GIS GRASS **3**

2.1 Základní charakteristika systému 3

2.2 Vývoj systému, poznámky k historii projektu 4

2.3 Základní pojmy 5

2.3.1 Databanka GRASSu 6

2.3.2 Struktura modulů 7

2.4 Import a export dat 8

2.5 Zpracování rastrových dat 9

2.6 Zpracování vektorových dat 9

2.6.1 Struktura databanky GRASSu v souvislosti s vektorovými daty 10

2.6.2 Správa popisných dat 10

2.7 Zpracování obrazových dat 13

2.7.1 Ortorektifikace leteckých snímků 13

2.8 Konverze dat 13

2.9 Vizualizace dat 13

2.9.1 3D vizualizace a animace 13

2.9.2 Mapový výstup 14

3 Informační systém katastru nemovitostí **16**

3.1 Vývoj ISKN 16

3.1.1 Výhled do budoucna 17

3.2 Základní charakteristiky ISKN 18

3.2.1 Hlavní přínosy projektu 19

3.3 Poskytování údajů z KN 19

3.3.1 Dálkový přístup k datům katastru nemovitostí 19

3.3.2 Výměnný formát ISKN 21

3.4 Na okraj: Aktuální stav digitální podoby KN 21

4 Výměnný formát ISKN **24**

4.1 Struktura nového výměnného formátu ISKN 25

4.1.1 Hlavička souboru 26

4.1.2 Datové bloky 28

4.1.3 Koncová značka 35

4.2 Další vývoj výměnného formátu 35

5 Modul GRASSu pro import dat ve formátu ISKN **37**

5.1 Vytvoření podkladových tabulek a načtení datových vět 38

5.1.1 Zpracování hlavičky 39

5.1.2 Zpracování jednotlivých datových bloků 41

5.2 Tvorba vektorové geometrie 44

5.2.1 Vektorové elementy 45

5.2.2 Připojení atributových tabulek 45

5.2.3 Zápis vektorových bodů, linií a centroidů 47

5.3 Závěrečné operace 51

5.4 Výstupní vektorová mapa 52

6 Zpracování importované katastrální mapy **54**

6.1 Založení lokace 54

6.2 Import dat ve formátu VKF 55

6.3 Další zpracování vektorové mapy 58

6.4 Vizualizace dat 61

6.4.1 Mapové značky katastrální mapy 61

6.5 Jednoduché příklady analýzy dat 63

6.5.1 Zastoupení jednotlivých druhů pozemků v katastrálním území Bylany . . 63

6.5.2 Výběr parcel podle oprávněných subjektů 65

6.5.3 Vyhledání parcel v okruhu 100 m 65

7 Závěr **67**

Literatura **67**

A Instalace modulu a dodatečné informace **72**

A.1 Proměnné prostředí 72

A.2 Instalace GRASSu a modulu `v.in.vfk` 72

A.2.1 Instalace celého systému . . . 73
A.2.2 Začlenění modulu do existující instalace GRASSu . . . 74
A.2.3 Lokalizace modulu do češtiny . . . 74
A.3 Manuálová stránka modulu `v.in.vfk` . . . 75
A.4 Modifikace vybraných modulů GRASSu . . . 75
A.4.1 Modul `d.vect` . . . 76
A.4.2 Modul `d.what.vect` . . . 76
A.4.3 Modul `v.category` . . . 77
A.4.4 Modul `v.extract` . . . 78
A.4.5 Vektorová knihovna – funkce `Vect_set_varray_from_db ()` . . . 78
A.4.6 Symboly – klíčové slovo `ARC` . . . 78
A.5 Typ prvku prostorových dat . . . 79
A.6 Seznam databázových tabulek . . . 85
A.7 Grafické znázornění vazeb mezi tabulkami . . . 87
A.8 Obsah přiloženého CD . . . 89

B Problematika mapového výstupu **90**
B.1 Testovací data . . . 90
B.2 Příklady mapových výstupů . . . 90
B.3 Vybrané mapové značky katastrální mapy . . . 95
B.3.1 Body bodových polí a hraniční znaky . . . 95
B.3.2 Druh pozemků a způsob jejich využití . . . 96
B.3.3 Stavební objekty . . . 100

Seznam obrázků

2.1 Mapa vývoje GRASSu 4
2.2 Analýza počtu instalací jednotlivých verzí GRASSu 5
2.3 GUI Tcl/Tk modulu `v.in.vfk` 6
2.4 Adresářová struktura databanky GRASSu – vektorová data 11
2.5 Připojení více atributových tabulek k jedné vektorové mapě 12
2.6 3D vizualizace a nástroj NVIZ 14

3.1 Proces digitalizace popisných dat katastru nemovitostí 22

5.1 Relační vztahy mezi tabulkami počínaje tabulkou souradnice_obrazu 46
5.2 Orientace os S-JTSK a GIS 48
5.3 Vektorová linie a její segmenty 51

6.1 Kompozice základních tematických vrstev katastrální mapy 61
6.2 Mapová značka druhu pozemku – „zahrada“ 62
6.3 Zastoupení jednotlivých druhů pozemků v k.u. „Bylany“ 64
6.4 Výběr parcel podle oprávněných subjektů 66
6.5 Výběr parcel v okruhu 100 m od výchozího bodu 66

A.1 Grafický symbol – oprava chyby u klíčového slova `ARC` 79
A.2 Grafické znázornění vazeb mezi tabulkami 87

B.1 Příklad mapového výstupu 93
B.2 Příklad mapového výstupu – rozlišení druhu pozemku 94

Seznam tabulek

2.1 Skupiny příkazů GRASSu 8

3.1 Stav digitalizace katastrálních map (rok 2004) 22

4.1 Hlavička výměnného formátu ISKN 26
4.2 Datové typy výměnného formátu ISKN 29
4.3 Skupina datových bloků „Nemovitosti“ – `NEMO` 30
4.4 Skupina datových bloků „Jednotky“ – `JEDN` 30
4.5 Skupina datových bloků „Bonitní díly parcely“ – `BDPA` 30
4.6 Skupina datových bloků „Vlastnictví“ – `VLST` 31
4.7 Skupina datových bloků „Jiné právní vztahy“ – `JPVZ` 31
4.8 Skupina datových bloků „Řízení“ – `RIZE` 32
4.9 Skupina datových bloků „Prvky katastrální mapy“ – `PKMP` 34
4.10 Skupina datových bloků „BPEJ“ – `BPEJ` 34
4.11 Skupina datových bloků „Geometrický plán“ – `GMPL` 35
4.12 Skupina datových bloků „Rezervovaná čísla“ – `REZE` 35

5.1 PostgreSQL: konverze datových typů 43
5.2 Podkladové tabulky obsahující grafickou informaci 45

6.1 Shrnutí tematických vrstev importované mapy 57

A.1 Vybrané proměnné prostředí GRASSu 72
A.2 Seznam podkladových databázových tabulek 85

Kapitola 1

Úvod do problému

Na katastr nemovitostí (KN) lze nahlížet jako na informační systém (IS), jehož úkolem je systematické shromažďování, uchovávání a zpřístupňování informací o nemovitostech, jejich geometrickém a polohovém určení, o vlastnických a jiných právech k nim.

Informační systém katastru nemovitostí (ISKN) patří svou rozsáhlou databází mezi největší informační systémy v České republice, přičemž svým rozsahem často daleko předstihuje i většinu bankovních či armádních systémů. Vždyť jen v současné době se v něm nachází řádově desítky milionů položek (vlastníků, uživatelů, parcel, apod.).

ISKN je moderní informační systém s vlastnostmi GIS, zahrnující plošné území celé republiky s rozlišením na jednotlivé parcely. V návaznosti na strukturu katastrálního operátu je ISKN členěn na systém popisných informací (SPI), který zabezpečuje správu stanovených popisných údajů o nemovitostech a systém geodetických informací (SGI), který v podobě digitální katastrální mapy (DKM) obsahuje údaje o geometrickém a polohovém určení nemovitostí.

Datové zdroje GIS jsou různorodé a odpovídají specializaci daného systému. Mezi nejčastější vstupní data patří právě data katastrální, inženýrské sítě, data ZABAGED (Základní báze geografických dat) či ortofotomapy.

GRASS, jako zástupce svobodného GISu, podporuje celou řadu externích datových formátů, což především souvisí s implementovanou knihovnou GDAL/OGR. Nicméně podpora specificky českých datových formátů je prakticky nulová. Tento stav alespoň na poli výměnného datového formátu ISKN se snaží změnit právě tato práce.

Nový modul `v.in.vfk` umožňuje na základě dat ve výměnném formátu ISKN vytvořit odpovídající vektorovou mapu umístěnou v databance GRASSu. Soubor geodetických informací, tj. vektorová geometrie, je uložen v nativním vektorovém formátu GRASSu (verze 6). Popisná data mapy (tj. soubor popisných informací) jsou skladována v odpovídající množině atributových tabulek připojených k mapě. Uživateli je tak umožněno tato data bez větších problémů do GRASSu importovat, dále zpracovat a využít při dalších analýzách. Není tak nutno provádět v externích programech ztrátovou konverzi do nějakého GRASSem podporovaného formátu (např. ESRI ShapeFile).

Nicméně je vhodné zdůraznit, že GRASS neobsahuje žádné speciální nástroje pro práci

s katastrálními daty. To však neznamená, že při troše snahy nelze s těmito specifickými daty v GRASSu pracovat. Snad je toho důkazem tato skromná práce.

Kapitola 2

GIS GRASS

Text této kapitoly vznikl především na základě následujících publikací:

- *Open source GIS: A GRASS GIS Approach* [NM02]
- český překlad *GRASS Handbuch* [Net03]
- a *GRASS GIS 6.0 Kursskript* [DHNR05][1]

Jejím cílem však není v žádném případě komplexní popis GRASSu jako geografického informačního systému (GIS) – o to se do značné míry snaží právě výše zmíněné publikace.

Kromě základní charakteristiky systému a jeho vlastností tak bude kladen důraz především na správu a zpracování vektorových dat, jejich vizualizaci včetně tvorby mapového výstupu.

Náplň kapitoly je tedy striktně orientována zejména na popis funkčních nástrojů, které uživatel využije při zpracování importované digitální katastrální mapy v prostředí GRASSu.

2.1 Základní charakteristika systému

GIS GRASS (*Geographical Resources Analysis Support System*) můžeme specifikovat jako geografický informační systém určený pro správu geoprostorových dat (rastrových a vektorových), obrazových záznamů (družicových i leteckých snímků), produkci vysoce kvalitní grafiky, prostorové modelování a vizualizaci dat.

Jedná se o svobodný software (**free software/open source** [25; 24]) publikovaný pod licencí *GNU General Public License* (GNU GPL) [21]. GRASS je napsán (téměř kompletně) v programovacím jazyce C a patří k nejrozsáhlejším open source projektům na světě.

[1] Vydání německého originálu „GRASS Handbuch“ a „GRASS GIS 6.0 Kursskript“ pod licencí GNU FDL [20] umožnilo bezproblémové vytvoření korespondujícího českého překladu (přirozeně opět pod licencí GNU FDL). České mutace těchto publikací jsou volně dostupné na webových stránkách „České sdružení uživatelů GISu GRASS“ [2].

Podporována je celá řada platforem od Intel x86 přes Sun SPARC až po HP PA-RISC – v tomto ohledu je prostor dostatečně široký. Podobná situace je u podporovaných operačních systémů zahrnující (samozřejmě) GNU/Linux (Intel, PowerPC), Solaris/SPARC, Solaris/i86, Mac OS X či Windows NT/Cygwin.

2.2 Vývoj systému, poznámky k historii projektu

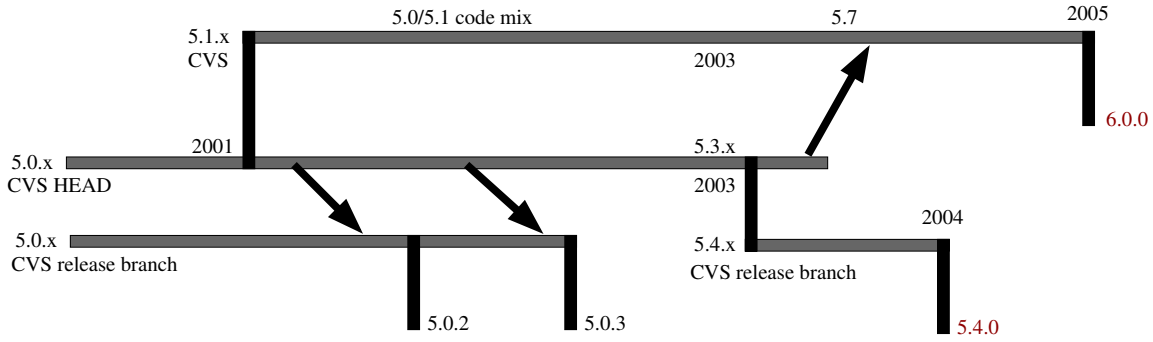
GRASS má za sebou velmi bohatou a svým způsobem i vzrušující historii, která sahá až do 80-tých let. Na jeho vývoji se zpočátku podílela celá řada státních organizací, univerzit, jednotlivců či soukromých společností, především z USA.

Projekt později získal oficiální záštitu *United States Army Construction Engineering Research Laboratories (USA-CERL)* a byl dále vyvíjen jako nástroj pro uzemní správu a plánování, primárně určený pro americkou armádu. USA-CERL publikoval roku 1992 verzi GRASS 4.1, v roce 1995 se vzdal jeho dalšího vývoje a nabídl jeho kompletní zdrojový kód široké komunitě.

Tento krok rozhodně neznamenal pro GRASS zánik, ba naopak, krátce poté, v roce 1997 byla na Baylor University připravena nová verze – GRASS 4.2. V následujících letech byla přidána řada nových modulů. Vše vyvrcholilo v roce 1998, kdy byl již pod dohledem Markuse Netelera vydán GRASS 4.2.1.

Zlomovým datem se stal 6. říjen 1999, kdy byl vydán GRASS 5.0 již nově pod licencí GNU GPL. To vše za úzké spolupráce Baylor University a koordinátora celého projektu Markuse Netelera. Během tohoto roku se pomalu zformoval *GRASS Development Team*, jak jinak než pod vedením Netelera. V současnosti jde přibližně o 20 – 30 osob s poměrně bohatou českou a slovenskou účastí.

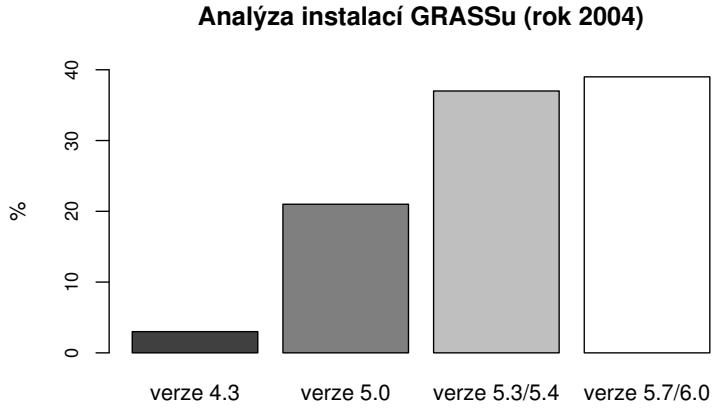
Mapa vývoje GRASSu (od řady 5 až po aktuální stav, tj. červen 2005) na první pohled může působit příliš komplikovaným dojmem (viz obr. 2.1).



Obrázek 2.1: Mapa vývoje GRASSu

V roce 2004 byla po poměrně dlouhé době vydána nová stabilní verze GRASSu 5.4, která vychází přímo ze svého předchůdce – verze 5.0. GRASS 5.4 měl tedy jednoznačný cíl – nahradit instalace GRASSu 5.0 či dokonce starší verze řady 4. To se zčásti povedlo, výsledky historicky první analýzy uživatelské základny GRASSu provedené na sklonku minulého roku

hovoří jasně (viz obr. 2.2). Kompletní výsledek této analýzy je součástí druhého čísla věstníku *GRASSNews* [22].



Obrázek 2.2: Analýza počtu instalací jednotlivých verzí GRASSu

V roce 2001 začal vývoj nové řady GRASSu 5.1 (později přečíslován na 5.7). Již od počátku byl striktně oddělen od hlavní větve projektu, která končí u verze 5.4. V budoucnu se tedy již neuvažuje o dalším vývoji v rámci řady 5 (opravovány jsou pouze případné chyby).

Veškerý potenciál tak byl přesunut do větve 5.7, která dala vzniknout nové generaci GRASSu 6. První stabilní zástupce této řady (GRASS 6.0.0) spatřil světlo světa v březnu tohoto roku.

V dalším textu budeme věnovat pozornost řadě 6, navíc modul `v.in.vfk` spolupracuje pouze s GRASSem 6.x!

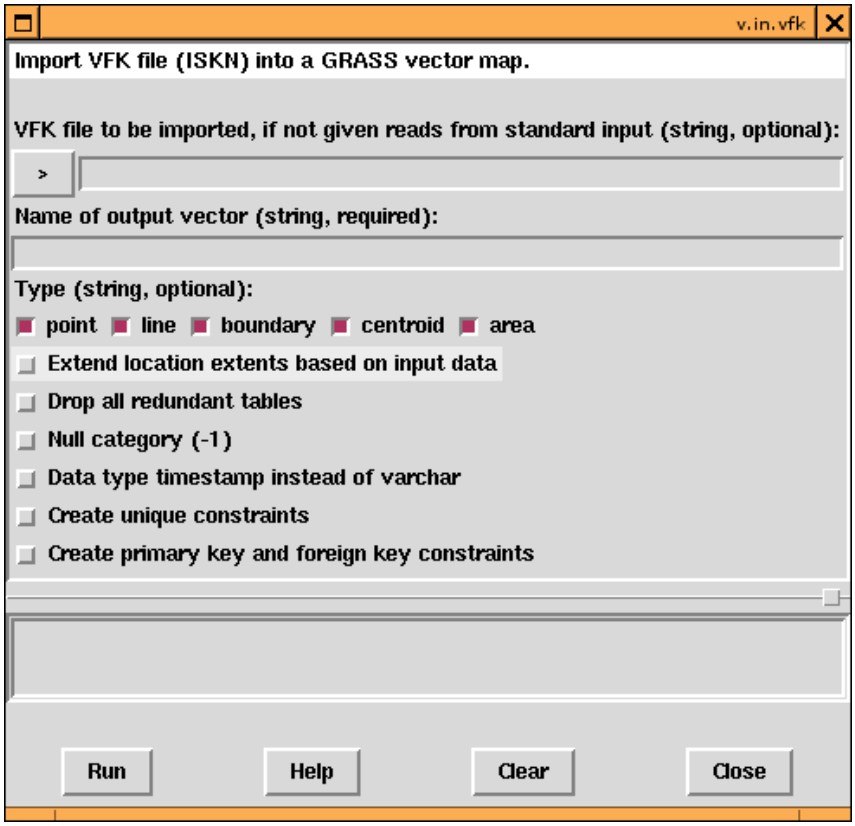
2.3 Základní pojmy

Všechny moduly GRASSu je možné ovládat jak z příkazové řádky, tak z grafického uživatelského rozhraní (GUI) postaveném na multiplatformní grafické knihovně Tcl/Tk (viz obrázek 2.3). Navíc je verze GRASSu 6 doplněna o tzv. „Display Manager“, který ulehčuje práci hlavně při vizualizaci dat v GRASS monitoru.

GRASS je velmi silný ve zpracování a analýze geoprostorových dat. Jakousi daní této široké funkcionality je relativně nízká úroveň uživatelského komfortu[2].

Právě v souvislosti s poměrně zastaralým GUI GRASSu, které bylo původně myšleno jako přechodné řešení (v budoucnu se vzhledem k okolnostem stalo „trvalým“), se uvažuje o integraci GRASSu a poměrně mladého, dynamicky se vyvíjejícího projektu QGIS [12].

[2]Pojem „uživatelské přívětivosti“ je značně subjektivní a do jisté míry zavádějící. Představa uživatele UNIXu se může v tomto ohledu významně lišit od představy uživatele MS Windows ...



Obrázek 2.3: GUI Tcl/Tk modulu `v.in.vfk`

2.3.1 Databanka GRASSu

Všechna data (snad kromě popisných dat skladovaných v externím databázovém systému jako je např. PostgreSQL), ke kterým GRASS přistupuje, jsou uložena v jediném adresáři (proměnná prostředí `$GISDBASE`) – tzv. *databance GRASSu*. Tento adresář (většinou jde o adresář `$HOME/grassdata`) obsahuje další podadresáře – tzv. *lokace (location)*. *Lokace* zahrnuje data, která mají souvislost s daným projektem (v našem případě např. „iskn“). Je definována souřadnicovým systémem (kartografickým zobrazením, referenčním elipsoidem a geodetickým datem) a velikostí zájmového území (označovaného jako *výchozí region*).

Jednotlivé *lokace* obsahují tzv. *mapsety (mapsets)*, které mohou být spojeny s určitým podprojektem nebo uživatelem (to v případě, že na projektu pracuje více lidí). Hierarchická struktura *mapsetu* je následující:

`cats/`	hodnoty kategorií a atributů rastrových map
`cell/`	vlastní rastrové mapy (celočíselné)
`cellhd/`	hlavičkové soubory rastrových map
`cell_misc/`	statistická data rastrových map
`colr/`	tabulky barev rastrových map
`dbf/`	atributová data vektorových map ve formátu xBase
`fcell/`	rastrové mapy s plovoucí desetinnou čárkou („floating point maps“)

`hist/`	historie rastrových map
`vector/`	vlastní vektorové mapy (geometrie, topologie atd.)
`WIND`	údaje k aktuálnímu *regionu* a mapové projekci *mapsetu*

Poznámka: Jakási „roztroušenost“ souborů definujících rastrovou mapu v několika adresářích (`cats/`, `cell/`, `cellhd/` ...) má historické důvody. Naopak všechny soubory vektorové mapy (kromě popisných dat) jsou uloženy pouze v jediném adresáři a to `vector/mapa`. Tento přístup umožňuje snadnou práci s daty na úrovni interní struktury databanky GRASSu a v neposlední řadě dovoluje současné (v rámci jednoho *mapsetu*) uložení mapy jak v novém vektorovém formátu GRASSu 6, tak ve starším formátu GRASSu 5.x. Za poznámku stojí, že se plánuje během potenciálního přepsání nativního rastrového formátu GRASSu (ten se totiž od vydání GRASSu 5.0.0 v podstatě nezměnil) přepracování i této nevhodné interní adresářové struktury směrem k modelu použitého právě v případě nového vektorového formátu. Vzhledem k počtu aktivních vývojářů a naléhavosti této věci je to však hudba budoucnosti.

Základní *mapset*, který musí být součástí každé *lokace*, nese unikátní název „PERMANENT“. Tento *mapset* obsahuje informace o použité mapové projekci, výchozím regionu a rozlišení rastrových dat, navíc je určen pro skladování základních dat daného projektu.

Znalost vnitřní struktury databanky je velmi užitečná a výhodná při řešení celé řady úloh. Nicméně správu dat na této interní úrovni je nutno přenechat GRASSu. Všechny operace (kopírování, mazání, přejmenování) tak musí být provedeny patřičnými příkazy GRASSu (`g.copy`, `g.remove`, `g.rename`) tak, aby se promítly do všech částí mapy (geometrie, popisná a grafická data).

2.3.2 Struktura modulů

GRASS je *modulárně* postavený systém. To znamená, že nabízí celou řadu drobných programů určených pro specifický účel či úlohu. Tento přístup přináší řadu výhod. Především vede k minimalizaci systémových požadavků. Navíc lze sestavit úzce specializovaný GIS, který bude obsahovat pouze vybrané moduly s ohledem na řešené úlohy.

Názvy těchto programů (v terminologii GRASSu – *modulů*) mají pevně danou strukturu: jsou uvozeny identifikátorem skupiny modulů, tečkou a krátkým názvem jednoznačně popisujícím funkci modulu. Tak například modul `d.vect` patří do skupiny „display“ a je určen pro vizualizaci vektorových dat v monitoru GRASSu.

Seznam definovaných skupin modulů obsažených v oficiální distribuci GRASSu je uveden v tabulce 2.1.

Po spuštění GRASSu se skupina systémových příkazů operačního systému (v dalším textu budeme mlčky předpokládat, že se jedná o GNU/Linux) rozšíří o nové příkazy GRASSu. Lze tak výhodně kombinovat jak příkazy GRASSu, tak systémové příkazy.

Ke každému modulu GRASSu existuje odpovídající nápověda. Krátký popis modulu včetně jeho syntaxe (přepínačů a parametrů) lze získat pomocí parametru `help`. Rozsáhlejší

Tabulka 2.1: Skupiny příkazů GRASSu

předpona	třída	krátký popis
`db.*`	database	práce s externími databázovými systémy
`d.*`	display	grafické výstupy a vizuální dotazy
`g.*`	general	obecné příkazy pro manipulaci s daty
`r3.*`	grid 3D	zpracování 3D rastrových dat („volume“)
`i.*`	imagery	zpracování obrazových dat
`p.*`	paint	příkazy určené pro tvorbu mapových výstupů
`ps.*`	postscript	příkazy pro tvorbu map ve formátu PostScript
`r.*`	raster	zpracování rastrových dat
`v.*`	vector	zpracování vektorových dat

nápovědu lze zobrazit pomocí specializovaného modulu `g.manual`. Tak například nápověda k modulu `v.extract`:

```
GRASS:~ > g.manual v.extract
```

2.4 Import a export dat

Minula doba, kdy jednotlivé GIS byly „izolovanými ostrůvky“ různých datových formátů. V současné době GIS, který nepodporuje import a export ostatních datových formátů, nemá na trhu geoinformačních technologií budoucnost.

Možnost importu či exportu geoprostorových dat je tak velmi důležitou vlastností každého GISu. Data z externích zdrojů nejsou totiž téměř nikdy k dispozici v nativním formátu používaného GISu (vzhledem k pozici GRASSu na trhu se tato pravděpodobnost rovná téměř nule). Jindy naopak je potřeba prezentovat výsledek analýzy dat ve zvoleném externím (výměnném) formátu.

Na tomto poli GRASS v porovnání s jinými (v drtivé většině proprietárními) GISy v žádném případě nezaostává. A to díky integrované, masově používané knihovně *GDAL/OGR* [17] podporující opravdu velké množství rastrových a vektorových formátů (režim čtení a zápisu). V případě rastrových dat jde o modul `r.in/out.gdal`. Pro import, resp. export vektorových dat existuje korespondující modul `v.in/out.ogr`. Kromě těchto modulů je k dispozici několik dalších určených pro formáty jako DXF, ESRI E00 či ESRI ARCGRID.

Navíc lze na základě geodat obsahujících informace o souřadnicovém systému automaticky založit novou *lokaci*.

Za poznámku stojí důležitá vlastnost GRASSu. Při analýze (a také exportu) rastrových dat hraje důležitou roli nastavení aktivního regionu a rozlišení. Lze tak velmi snadno exportovat pouze výřez mapy ve zvoleném rozlišení. Tento postup však neplatí pro data vektorová. Tady

se promítne aktuální region pouze při vizualizaci dat.

2.5 Zpracování rastrových dat

V této části se vzhledem k rozsahu a komplexnosti tématu zpracování rastrových dat v GRASSu budeme pohybovat pouze v obecné rovině. Navíc na rozdíl od kapitoly popisující zpracování dat vektorových, zúžíme rozsah této části na naprosté minimum. Hlavním důvodem je orientace této práce pouze na data vektorová.

Zpracování rastrových dat je dlouhodobě silnou stránkou GRASSu. Dokonce lze tvrdit, že v této oblasti překonává řadu drahých proprietárních produktů. Z kvantitativního hlediska GRASS disponuje téměř 150 moduly určenými pro zpracování rastrových dat (skupina příkazů `r.*`). Kromě 2D dat jsou podporována i 3D rastrová data (*voxel*, skupina příkazů `r3.*`), včetně jejich vizualizace pomocí nástroje NVIZ (viz podkapitola 2.9.1).

Kromě jiného je k dispozici nástroj pro *mapovou algebru* – modul `r.mapcalc`, moduly pro interpolaci dat (algoritmy RST – „Regular Spline with Tension“, modul `v.surf.rst` a IDW – „Inverse Distance Weighted“, modul `r.surf.idw` a `v.surf.idw`).

Jako další můžeme například uvést možnost *hydrologických* analýz (tvorba mapy hustoty povrchového odtoku, povodí či rozvodí).

Za poznámku stojí fakt, že jednotlivé rastrové mapy umístěné v jednom *mapsetu* mohou mít individuální rozlišení. Před jejich vytvořením se jednoduše změní aktuální rozlišení pomocí modulu `g.region`. Podobně lze velmi snadno vytvářet i jednotlivé výřezy rastrové mapy pouhou změnou regionu a posléze aplikací `r.mapcalc`.

2.6 Zpracování vektorových dat

Zcela zásadní rozdíl mezi řadou GRASSu 5 a 6 leží právě v oblasti správy vektorových dat.

Vektorová architektura GRASSu $<= 5.4$ je značně zastaralá a v žádném případě neodpovídá současným potřebám uživatelů GIS. Snad největším či lépe řečeno nejviditelnějším omezením je možnost přiřadit v interní databance GRASSu pouze jeden atribut na vektorový prvek.

Nyní si tedy popíšeme základní vlastnosti nové vektorové architektury GRASSu především v souvislosti s návrhem modulu pro import dat ve výměnném formátu ISKN.

Tak především byly kompletně přepracovány základní složky – vektorová knihovna a nativní vektorový formát GRASSu. Nutno podotknout, že tyto změny znamenaly také přepsání všech modulů určených pro zpracovaní vektorových dat (skupina modulů `v.*`).

Geometrie: Nová vektorová knihovna podporuje *multiformát*, tzn. externí datové formáty bez nutnosti jejich konverze do nativního formátu GRASSu. V neposlední řadě byl implementován i *prostorový index*.

Atributová data: Popisná data jsou skladována v externím databázovém systému (DBMS) podporujícím strukturovaný dotazovací jazyk (SQL). Konkrétně lze atributová data uložit ve formátu dBase (výchozí nastavení), a nebo lépe v externím databázovém systému typu PostgreSQL.

Tady se dostáváme k **zásadní** vlastnosti nového vektorového formátu. A to možnosti připojit k vektorové mapě hned *několik* různých atributových tabulek. Tento prvek hrál při návrhu modulu `v.in.vfk` velmi důležitou roli – k importované mapě je totiž připojeno hned několik desítek těchto tabulek.

2D/3D vektorový formát: Pouze pro úplnost – nově jsou podporována i 3D vektorová data včetně jejich vizualizace nástrojem NVIZ.

Síťové analýzy: Novinkou jsou také moduly určené pro síťové analýzy. Konkrétně jde o moduly `v.net` – údržba vektorové sítě, `v.net.alloc` – tvorba podsítě korespondující s daným prvkem, resp. prvky, `v.net.iso` – určení dostupnosti prvku, resp. prvků v rámci vektorové sítě, `v.net.path` – nalezení nejkratší cesty s využitím váhy (linií či uzlů sítě), `v.net.salesman` – „problém obchodního cestujícího“, `v.net.steiner` – „minimální Steinerův strom“ a konečně i `d.path` – zobrazení nejkratší cesty přímo v monitoru GRASSu, bez nutnosti vytvářet novou vektorovou mapu.

2.6.1 Struktura databanky GRASSu v souvislosti s vektorovými daty

Všechny soubory vektorové mapy (vyjma popisných dat) jsou uloženy v jediném adresáři – `vector/mapa/`. Tento adresář obsahuje celkem šest souborů:

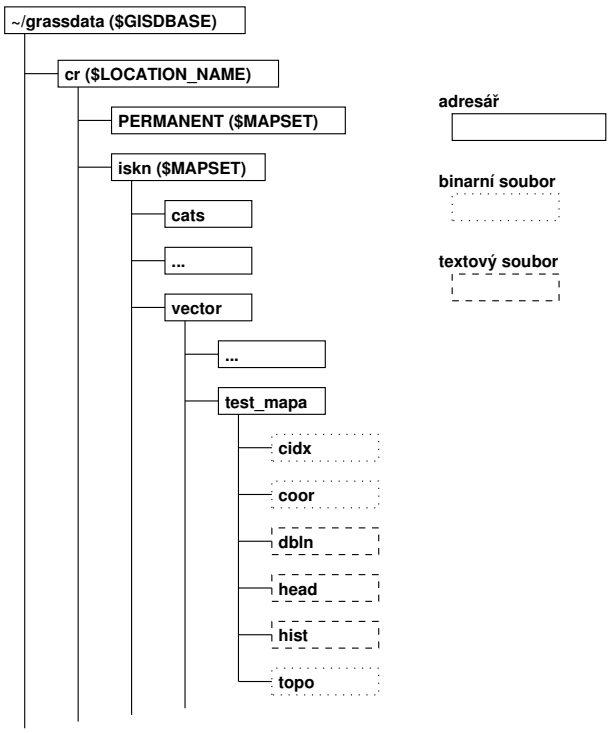
`cidx`	binární soubor obsahující informace nutné pro sestavení prostorového indexu
`coor`	binární soubor obsahující souřadnice vektorových elementů
`dbln`	textový soubor s odkazy na atributové tabulky (číslo vrstvy, název tabulky, sloupec kategorie, databáze a ovladač)
`head`	textový soubor s hlavičkou vektorové mapy
`hist`	textový soubor obsahující historii mapy
`topo`	binární soubor s topologií vektorové mapy

Daná adresářová struktura je názorně zobrazena na obrázku 2.4.

2.6.2 Správa popisných dat

Ve výchozím nastavení jsou popisná data ukládána do souborů ve formátu dBase (v adresáři *mapsetu* `dbf/`). Nicméně modul `v.in.vfk` podporuje pouze databázový systém typu PostgreSQL (ovladač `pg`, `mysql` či `odbc`).

Jak již bylo zmíněno dříve, nový vektorový formát GRASSu umožňuje připojení v podstatě libovolného počtu atributových tabulek. Seznam odkazů na tyto tabulky je umístěn v textovém



Obrázek 2.4: Adresářová struktura databanky GRASSu – vektorová data

souboru `dbln` v interní struktuře databanky GRASSu (viz obr. 2.4). Struktura tohoto souboru je následující:

```
číslo_vrstvy název_databázové_tabulky sloupec_kategorie databáze ovladač
```

Lze tak velmi jednoduše připojit k vektorové mapě atributové tabulky z různých zdrojů současně.

Jednoznačná identifikace datového záznamu je dána číslem vrstvy (*layer*) + hodnotou kategorie (*cat*) a korespondující databází (*database*) + ovladačem (*driver*).

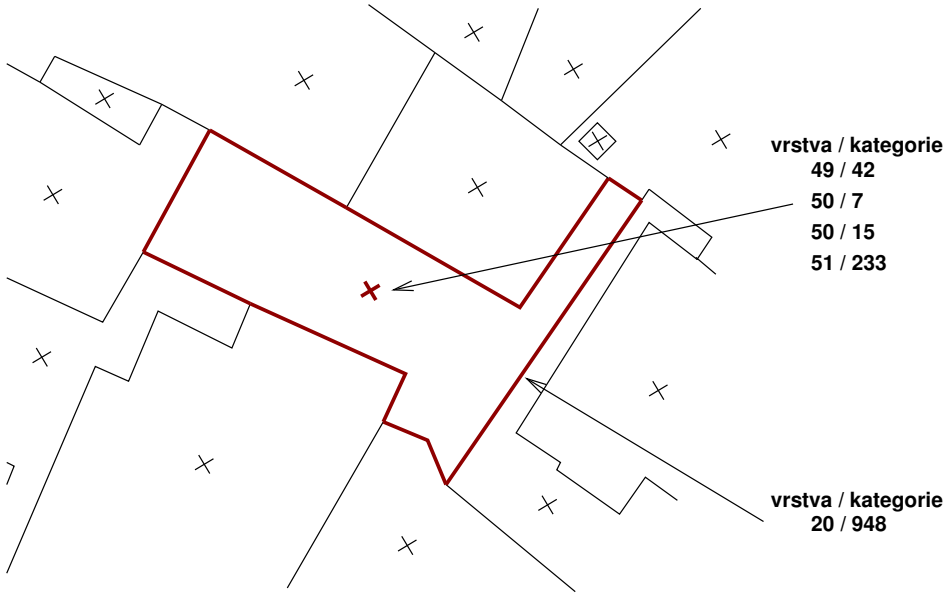
Pro názornou ilustraci si uvedeme příklad pro parcelu. Máme k dispozici mapu parcel, ke které je připojena celá řada atributových tabulek. Tyto tabulky obsahují tematicky členěná popisná data. Celá situace je zachycena na obrázku 2.5.

Z výše uvedeného vyplývá, že kromě libovolného počtu připojených atributových tabulek lze i v rámci jedné vrstvy (tj. jedné atributové tabulky) přiřadit vektorovému prvku více datových záznamů.

Pro dokreslení situace doplníme popis o názvy atributových tabulek a zkrácený výpis jednotlivých datových záznamů.

```
20 public.kml_hp_hranice_parcel cat iskn pg
```
[3]

[3] Číslo vrstvy, schéma.mapa_tabulka, sloupec kategorie, databáze, ovladač.



Obrázek 2.5: Připojení více atributových tabulek k jedné vektorové mapě

```
 cat |    id     | typppd_kod | ... | par_id_1 | par_id_2
-----+-----------+------------+-----+----------+----------
 948 | 153434708 |      21900 | ... | 92574708 | 92572708
```

```
49 public.kml_hp_obrazy_parcel cat iskn pg
```

```
 cat |    id    | ... | par_id   | ... | souradnice_x | souradnice_y
-----+----------+-----+----------+-----+--------------+--------------
  42 | 36306708 | ... | 92572708 | ... |   1069344.92 |     651368.2
```

```
50 public.kml_hp_t_prvku_p_dat cat iskn pg
```

```
 cat | kod | polohopis |              vyznam               | typ_prvku
-----+-----+-----------+-----------------------------------+-----------
   7 |  18 | n         | Číslo (def.bod) pozemkové parcely |         4
  15 | 304 | n         | Zahrada                           |         7
```

```
51 public.kml_hp_parcely cat iskn pg
```

```
 cat |    id    | katuze_kod | kmenove_cislo_par | vymera_parcely
-----+----------+------------+-------------------+----------------
 233 | 92572708 |     616567 |                16 |            862
```

2.7 Zpracování obrazových dat

GRASS není v žádném případě jednostranně orientovaný GIS. Zjednodušeně řečeno nabízí vše, co jeho uživatelé požadují a současně jsou schopni do GRASSu přidat.

Z historického hlediska je GRASS určen spíše pro zpracování rastrových dat (tento fakt velmi úspěšně mění právě nová generace GRASSu 6). S tím souvisí i existence celé řady modulů určených pro zpracování obrazových dat, např. dat dálkového průzkumu Země (DPZ). Tyto nástroje jsou akumulovány ve skupině příkazů `i.*`.

Navíc se pro skladování obrazových dat používá nativní rastrový formát (výjimkou jsou data vytvořená při Fourierově transformaci), což umožňuje kombinaci modulů z rodin `r.*` a `i.*`.

Tato problematika opět rámcově přesahuje tematické zaměření práce a je zde zmíněna pouze pro úplnost. GRASS lze považovat za poměrně mocný nástroj pro zpracování obrazových dat. Důkazem toho snad může být fakt, že je již dva semestry používán při výuce volitelného předmětu „Zpracování obrazových záznamů v DPZ“ vyučovaným na Katedře mapování a kartografie, ČVUT v Praze, Fakultě stavební.

2.7.1 Ortorektifikace leteckých snímků

Pouze pro doplnění – GRASS poskytuje potřebné nástroje pro *ortorektifikaci* (tj. diferenciální překreslení) leteckých snímků. Jedná se především o modul `i.ortho.photo`.

Právě ortofotomapa v kombinaci s katastrální mapou bývá často používána jako názorná prezentace určená pro širokou veřejnost.

2.8 Konverze dat

GRASS přirozeně obsahuje nástroje umožňující konverzi mezi rastrovým a vektorovým formátem.

Pro automatickou vektorizaci rastrových dat existuje modul `r.to.vect` (body, linie, plochy). V případě tvorby izolinií (například vrstevnic z digitálního výškového modelu terénu) je vhodné nasadit modul `r.contour`.

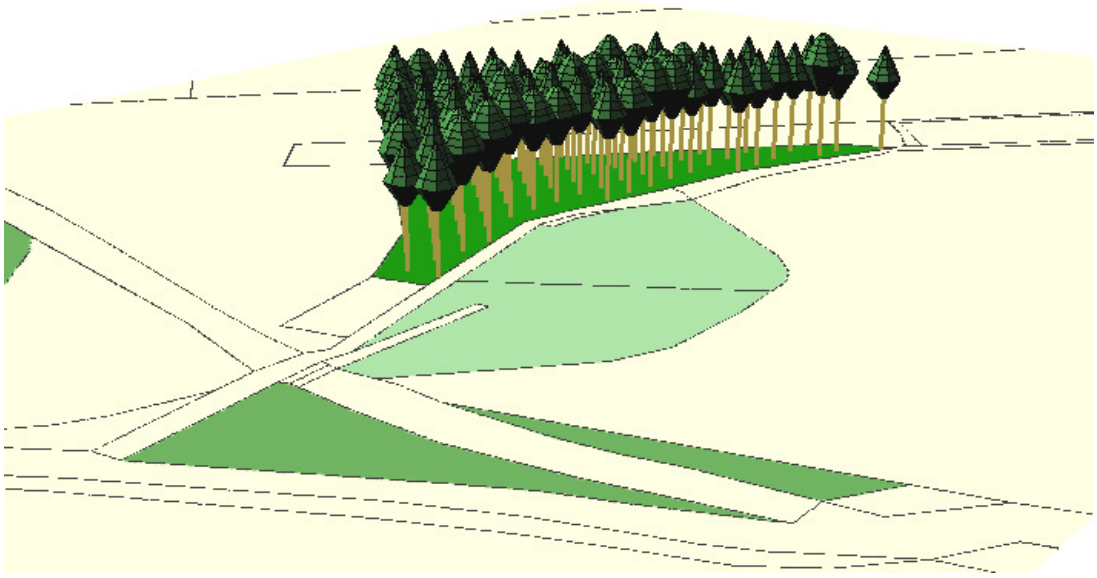
Opačný postup, tj. rasterizaci vektorových dat umožňuje komplexní modul `v.to.rast`.

2.9 Vizualizace dat

2.9.1 3D vizualizace a animace

Pro 3D vizualizaci je určen interní modul GRASSu s názvem NVIZ. Jde o komplexní vizualizační prostředí umožňující zobrazení různých datových podkladů (2D, 3D rastrová a vektorová data). Nabízí řadu funkčních nástrojů, od změny osvětlení až po vytváření animací. Rozhodně však není „všemocný“ a nelze ho tedy srovnávat se specializovanými (komerčními) produkty.

Příklad 3D vizualizace pomocí nástroje NVIZ je zobrazen na obrázku 2.6. Pravda, kompozice poněkud chudá - podkladová katastrální mapa s barevným rozlišením druhu pozemku a lesní porost (3D vektorová data) na parcele s druhem pozemku „Lesní půda bez rozlišení porostu“. Stromy byly vytvořeny pomocí skriptu `trees.pl` [8].



Obrázek 2.6: 3D vizualizace a nástroj NVIZ

2.9.2 Mapový výstup

Tomuto tématu se na rozdíl od 3D vizualizace budeme věnovat podrobněji – ono se totiž dotýká problematiky vizualizace katastrálních dat poněkud více.

Mapový výstup je důležitou funkcionalitou každého GISu. Nutno podotknout, že vizualizace dat (včetně mapového výstupu) nikdy nepatřila mezi silné stránky GRASSu. Tento stav se pomalu lepší, nelze ale GRASS jako GIS srovnávat se specializovanými (kartografickými) programy, které poskytují nesrovnatelně mnohotvárnější mapové výstupy. GRASS je možno použít pro tvorbu jednoduchých mapových děl.

Existují v podstatě dva přístupy pro tvorbu mapových výstupů (mapových děl). Prvním je modul `ps.map`, který na základě řídícího souboru (textový soubor obsahující klíčová slova definující podobu výsledného mapového díla) vytvoří vysoce kvalitní výstup ve formátu PostScript.

Jistou nevýhodou tohoto modulu je absence interaktivního ovládaní (v podobě GUI). Je tak nutno tento řídící soubor připravit manuálně, což při troše cviku jde poměrně rychle. Tuto nepříjemnou vlastnost v jisté míře překonává „Display Manager“ (modul `d.m`) umožňující přímý výstup do souboru ve formátu PostScript. V komplikovanějších případech je však nutno vytvořit řídící soubor manuálně.

Příklady mapových výstupů vytvořených pomocí modulu `ps.map` včetně ukázky řídícího souboru jsou prezentovány v dodatku B.2.

Druhým přístupem je export dat z GRASSu v podobě obrázku (např. pomocí modulu `d.out.png`, či komplexněji přes ovladač `PNG`). A následné zpracování tohoto polotovaru ve vhodném grafickém editoru jako je například Xfig [7] či Skencil [6]. Další možností je nasazení specializovaného balíčku GMT [30].

Kapitola 3

Informační systém katastru nemovitostí

Tato kapitola vznikla zejména na základě informací dostupných na webových stránkách Českého úřadu zeměměřického a katastrálního (ČÚZK) [34].

Informačním systémem katastru nemovitostí (ISKN) rozumíme integrovaný informační systém primárně určený pro podporu výkonu státní správy a uživatelských služeb katastru nemovitostí. ISKN tak obsahuje nástroje umožňující správu souboru popisných a geodetických informací, prostředky pro podporu správních a administrativních činností při samotném vedení katastru nemovitostí a pro správu dokumentačních fondů.

Katastrem nemovitostí ČR rozumíme technicko-právní instrument, zřízený zákonem o katastru nemovitostí České republiky, integrujícím geometrické zobrazení, soupis a popis veškerých pozemků na území státu a pozemkové knihy, ve kterých jsou zapsány důležité právní poměry týkající se těchto pozemků.

3.1 Vývoj ISKN

Na tvorbě a implementaci ISKN se v letech 1997 – 2001 podílela jako systémový integrátor a dodavatel aplikačního programového vybavení soukromá společnost APP Czech s.r.o. (nyní NESS Czech s.r.o.) [15]. Dalšími kooperujícími společnostmi byly Infinity a.s. [1], Compaq Computer s.r.o. (nyní HP) [28], Oracle Czech s.r.o. [29], Bentley Systems s.r.o. [27], BEA Systems s.r.o. [26].

ISKN byl uveden do ostrého provozu v září 2001 a to na všech katastrálních pracovištích (včetně centrály). Během následujícího roku 2002 probíhalo dolaďování systému, převzetí závěrečných etap a konečně i dokončení auditu ISKN.

Plná implementace systému se ve svém důsledku dotkla více jak 5 000 zaměstnanců na 112 pracovištích po celé republice. Toto nové řešení tak mohlo plně nahradit původní způsob vedení

KN. Došlo tak k integraci vedení a správy katastru nemovitostí do jediného informačního systému společného pro všechna pracoviště katastrálních úřadů a podpůrné centrum.

Jako skladiště dat ISKN je využíván Spatial Cartridge Option a databáze Oracle. Vzdálený přístup k datům prostřednictvím sítě Internet umožňuje technologie BEA WebLogic, více informací je dostupných na webových stránkách služby „Dálkový přístup k údajům KN“ [23]. Pro systémový management je využíváno nástrojů CA Unicenter.

Systém byl navržen tak, aby byla zvýšena jeho celková bezpečnost a bylo možno bezproblémově začlenit data z externích datových zdrojů. Navíc umožňuje velmi efektivní a v dnešním prostředí potřebné poskytování požadovaných dat z katastru nemovitostí prostřednictvím dálkového přístupu.

V rámci tohoto projektu bylo vyvinuto zcela nové aplikační programové vybavení, došlo k nezbytné obměně hardwarových prostředků pracovišť katastrálních úřadů a jejich detašovaných pracovišť, vybudování nového centra a propojení všech pracovišť resortu privátní sítí WAN, spojení všech komponent do funkčního celku, migraci dat z dosavadního systému a nutnému zaškolení zaměstnanců.

V červenci roku 2003 byla úspěšně ukončena atestace ISKN. Systém tak splňuje standard ISVS pro náležitosti životního cyklu (v. 02.01) a pro popis jednoduchých datových prvků (v. 02.01).

Dalším rozvojem a údržbou informačního systému katastru nemovitostí byla pověřena soukromá společnost – dané výběrové řízení pro období 2004 – 2006 vyhrála již dříve zmíněná společnost NESS Czech s.r.o.

3.1.1 Výhled do budoucna

Základní cíle dalšího vývoje ISKN do roku 2006 jsou následující:

- Primární je *zajištění provozu systému* a jeho údržby ve stávající podobě. Jsou tak průběžně řešeny a odstraňovány provozní problémy a to tak, aby nebyl v žádném okamžiku ohrožen řádný výkon státní správy katastru nemovitostí.

- *Modifikace částí existujícího systému* (zejména v důsledku legislativních změn, obecného vývoje v oblasti informačních systémů veřejné správy (ISVS), realizace státní informační politiky a rozvoje v oblasti katastru nemovitostí). Cílem je tedy rozšiřování o nové funkce, zkvalitňování stávajících za účelem poskytování lepších služeb interním i externím uživatelům systému. Nutné je také zapracovat úpravy systému související s připojením na komunikační infrastrukturu veřejné správy, nové či aktualizované standardy vydané Ministerstvem informatiky, které se týkají provozu informačních systémů veřejné správy.

- V současné době jsou nejakutnější potřebné *úpravy dálkového přístupu* k datům katastru nemovitostí a to v souvislosti s „nečekanou“ novelou katastrálního zákona s účinností od 1.4.2004. Podle této novely se údaje katastru poskytují územně samosprávným celkům k výkonu jejich působnosti bezúplatně. Úpravy jsou vedeny tak, aby dálkový přístup ze sa-

mosprávných celků neohrozil kvalitu služeb dálkového přístupu pro platící uživatele. Tyto modifikace tak budou prováděny ve dvou fázích.

Přechodné řešení (na bázi stávající technologie) spočívá v možnosti oddělené a v denním čase proměnlivé regulace současně pracujících platících uživatelů a uživatelů ze samosprávných celků. Cílové řešení, které vzhledem k očekávanému velkému zatížení dálkového přístupu uživateli ze samosprávných celků bude vyžadovat podstatnou změnu technologické infrastruktury části centra ISKN. Podmínky nutné pro jeho realizaci se projednávají.

3.2 Základní charakteristiky ISKN

Uložení dat: Pro optimální uložení dat byl zvolen jediný datový model a to jak pro popisná (včetně dat týkajících se správních řízení), tak prostorová data. Data jsou konkrétně skladována v databázovém prostředí společnosti Oracle. Tento model umožňuje současnou aktualizaci popisných a prostorových dat a ve svém důsledku tak udržení jejich vzájemné konzistence.

Dále byla přijata koncepce samostatné *evidence budov* a *bezešvé digitální katastrální mapy.*

Počínaje zářím 2001 jsou taktéž uchovávána veškerá historická data (popisná i prostorová). Tento přístup tak umožňuje sestavovat data do potřebných výstupů k historickému datu (tj. sledovat časový vývoj). ISKN tak ve svém důsledku změnilo i pohled na pojem „list vlastnictví“ – výpis z KN. V této rovině je výpis chápan jako nemovitost nebo soubor nemovitostí, které tvoří právní jednotku ve vlastnictví téže osoby. Nejedná se tedy o seskupení na straně vlastníků, ale na straně nemovitostí.

Nutno podotknout, že jsou data strukturována v souladu se standardem ISVS pro popis jednoduchých datových prvků (v. 02.01).

Optimalizace procesů při správě KN: Za jednu z hlavních předností ISKN lze považovat zavedení řady automatických kontrol do procesu zapsání změny do KN a možnost převzetí aktuálních dat z registru obyvatel. Tím je zajištěna velmi důkladná kontrola výsledného stavu katastru a celý proces realizace změny v prostředí ISKN je navíc zajištěn i technicko-organizačními opatřeními.

Nové procesy zpracování dat/návrhu změn dávají možnost částečného nabytí platnosti geometrického plánu s automatizovanou změnou návrhu změny v budoucím stavu. Umožňují také aktualizaci dat katastru nemovitostí způsobem, který nezamyká aktualizovaná data, ale pouze řeší konflikty při pokusu aktualizovat stejná data.

Jednotná centrální správa číselníků: Vnáší jednotnost do procesů zpracování změn na katastrálních úřadech a zvyšuje tak kvalitu a konzistenci datové základny. Některé z těchto číselníků jsou přebírány z externích zdrojů (např. číselníky územní identifikace, PSČ, registr ekonomických subjektů).

3.2.1 Hlavní přínosy projektu

Za hlavní pozitiva implementace ISKN můžeme považovat:

- okamžitá dostupnost informací z katastru nemovitostí odkudkoliv
- lepší zabezpečení dat
- zavedení datových a jiných standardů
- řešení snadno přizpůsobitelné legislativním změnám
- usnadnění fungování státní správy propojením katastru nemovitostí s dalšími veřejnými rejstříky
- usnadnění fungování trhu s nemovitostmi
- zefektivnění fungování soudního systému
- vyšší produktivita práce
- lepší služby veřejnosti
- nižší celkové náklady na vlastnictví

3.3 Poskytování údajů z KN

Jak již bylo dříve nastíněno technologické zázemí ISKN je řešeno tak, že všechna pracoviště jsou spojena s centrem prostřednictvím sítě WAN[1]. Data jsou tak z jednotlivých lokálních pracovišť replikována do centrální databáze v pravidelných dvouhodinových intervalech.

Toto řešení umožňuje poměrně snadné zavedení dálkového přístupu k veškerým digitálním datům KN prostřednictvím veřejné sítě Internet. Nutno poznamenat, že údaje poskytované dálkovým přístupem nemají charakter veřejných listin. Přístup k datům katastru nemovitostí je definován katastrálním zákonem a způsob poskytování je popsán ve vyhlášce 162/2001 Sb. ve znění pozdějších předpisů.

Katastrální pracoviště jsou tudíž schopna poskytovat požadovaná data katastru nemovitostí z celého území státu ve formě tzv. interního dálkového přístupu.

3.3.1 Dálkový přístup k datům katastru nemovitostí

Jak již bylo zmíněno výše „klasické“ způsoby (návštěva katastrálního pracoviště, písemné objednávky apod.) poskytování dat, byly v roce 2001 rozšířeny o možnost dálkového přístupu k datům KN.

[1]Wide Area Network – rozlehlá datová komunikační síť, která pokrývá geograficky rozlehlé území a využívá spojení veřejných poskytovatelů přenosových služeb.

V listopadu roku 2003 byly implementovány důležité bezpečnostní prvky jako zabezpečení hesla a šifrování přenosu dat. Tím se služba dálkového přístupu posunula na obdobnou bezpečnostní úroveň jako je např. on-line práce s bankovními účty.

Přihlédneme-li k velmi vysoké úrovni aktuálnosti poskytovaných dat (vzhledem k použité technologii stáří dat nemůže přesáhnout 2 hodiny), lze považovat kvalitu této služby za velmi vysokou a v plné míře využitelnou externími uživateli.

Za necelé tři roky poskytování této služby vzrostl počet uživatelů dálkového přístupu KN z původního počtu 1 540 (rok 2001) na 3 387. Za zmínku také stojí příjmy z této služby pro státní rozpočet, které mají vzrůstající tendenci.

Dálkový přístup byl do 31.3.2004, až na výjimky, *placenou* službou. Nutno podotknout, že se služba potýkala s malým zájmem ze strany samosprávy a státní správy. Nicméně změnou katastrálního zákona došlo od 1.4 2004 k tomu, že resort musí poskytovat veškerá data z katastru nemovitostí (nezávazně na množství) samosprávným celkům *zdarma*.

Další poskytované služby

Na tomto místě se také zmíníme o dalších službách poskytovaných resortem a to „Nahlížení do katastru nemovitostí“ a spolupráci na projektu „Kiosky EVA“.

Nahlížení do katastru nemovitostí

Jde o aplikaci, která umožňuje pomocí Internetu zdarma sledovat průběh vyřizování správních řízení na katastrálních úřadech. Poskytuje tak přístup k vybraným údajům o parcelách a budovách. Tato služba je dostupná na adrese [5].

Aplikace využívá centrální databázi a umožňuje přistupovat k vybraným digitálním údajům katastru nemovitostí a správním řízením. Výstupy jsou tedy zpožděné zhruba o 24 hodin. Služba není omezena lokalitou, každý uživatel má přístup k vybraným datům KN v rozsahu celé ČR. Poskytované údaje slouží pouze pro informativní účely a nemají charakter veřejné listiny.
Lze je rozdělit do následujících kategorií:

- informace o řízení
  - informace o řízení (detail průběhu řízení)
  - přehled řízení
  - seznam přijatých řízení
- informace z katastru nemovitostí
  - informace o parcele
  - informace o budově

Je tak široké veřejnosti poměrně nenáročnou formou umožněno sledovat průběh řízení a porovnávat stav daného řízení s ostatními řízeními.

Zájem veřejnosti o tuto službu je poměrně značný; za první tři měsíce provozu bylo vyhotoveno okolo 9 mil. sestav, tj. zhruba 100 000 sestav denně. Navíc získala na konferenci ISSS pořádané na konci března 2004 v Hradci Králové dvě ceny – Českého zavináče a Cenu ministra informatiky za elektronický přehled průběhu řízení.

Kiosky EVA

Tyto kiosky jsou koncipovány tak, aby poskytovaly občanům důležité informace potřebné pro styk s úřady. Služba je, až na výjimky, bezplatná a bez nutné registrace.

V rámci projektu „Ekonomicky vlídná administrativa“ (EVA) je dostupná i aplikace zjednodušeného dálkového přístupu (ZDP), která vznikla v úzké spolupráci s ČÚZK. Na rozdíl od ostatních služeb je ZDP službou placenou. Výstupy ZDP mají stejný vzhled, obsah, kvalitu a aktuálnost jako výstupy získávané dálkovým přístupem a mají taktéž pouze informativní charakter (tj. nemají váhu veřejné listiny).

Kiosky jsou fyzicky umístěny v počtu několika desítek především ve středních Čechách.

3.3.2 Výměnný formát ISKN

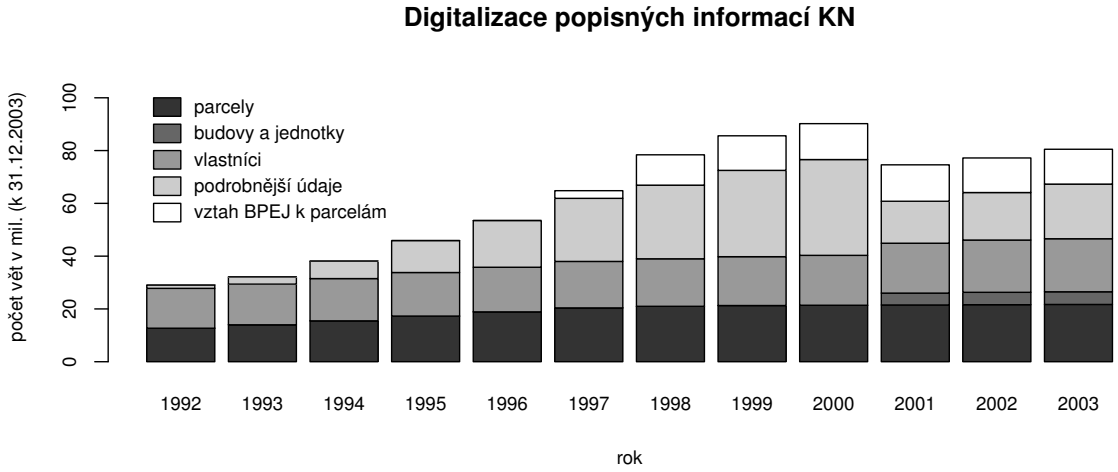
Data KN jsou poskytována veřejnosti přirozeně také ve formě souborů s definovaným obsahem (tzv. *nový výměnný formát* – NVF). Popis tohoto formátu je volně dostupný na stránkách ČÚZK. Zatím poslední verze dokumentu „Struktura výměnného formátu informačního systému katastru nemovitostí České republiky“ byla vydána dne 26.11.2004 [31]. Tento dokument vychází z původního znění (ze dne 6.12.2002) a dodatků 1 až 4.

Tématu výměnného formátu ISKN se blíže dotýká kapitola 4.

3.4 Na okraj: Aktuální stav digitální podoby katastru nemovitostí

Na úvod si připomeňme, že digitalizace katastru nemovitostí byla zahájena, v souladu s usnesením vlády č. 492 ze dne 8. září 1993 a s koncepcí digitalizace katastru nemovitostí vydanou Českým úřadem zeměměřickým a katastrálním na základě usnesení vlády ČR ze dne 16. 6. 1993 č. 312, již v roce 1993.

V roce 1998 byla dokončena digitalizace popisných dat katastru nemovitostí (jedná se o téměř 80 mil. datových záznamů). Proces digitalizace včetně aktuálního stavu je graficky zachycen na obr. 3.1. Tento graf si ovšem zaslouží jednu malou poznámku – a to pokles v roce 2001. Příčinou je přechod na ISKN a s tím související změna struktury databáze katastru nemovitostí. Některé údaje dosud vedené v samostatných větách byly sloučeny do jedné věty. To se tedy projevilo snížením počtu vět o podrobnějších údajích katastru i celkového počtu vět.



Obrázek 3.1: Proces digitalizace popisných dat katastru nemovitostí

Je vcelku přirozené, že souběžně s digitalizací popisné části probíhala, resp. probíhá digitalizace (vektorizace) katastrálních map. Tato činnost měla být dokončena do roku 2006. Předpokládaný termín však nebude s největší pravděpodobností dodržen. Aktuální stav (rok 2004) je prezentován v tabulce 3.1.

Tabulka 3.1: Stav digitalizace katastrálních map (rok 2004)

typ KM	počet katastrál. území	plocha území republiky [%]
digitální katastrální mapa	2 199	16,0
katastrální mapa digitalizovaná	1 225	8,3
analogová mapa	9 603	75,7

Stávající tempo digitalizace ovlivňuje řada faktorů. Ve většině případů se jedná o objektivní problémy, které jsou resortem řešeny. Mezi hlavní aspekty patří:

- odstranění následků nedokončeného přídělového a scelovacího řízení
- nedostatečná kapacita katastrálních úřadů
- rozmanitost a různá kvalita využitelných mapových podkladů v jednotlivých katastrálních územích
- nutnost doplnění parcel zaniklých v průběhu kolektivizace a evidovaných zjednodušeným způsobem
- hledání optimální metody digitalizace/obnovy analogových map přepracováním při zachování požadované kvality a přesnosti výsledné digitální mapy

V současné době byla dokončena analýza využitelnosti mapových podkladů v jednotlivých katastrálních územích. Byl tak stanoven diferencovaný přístup k jejich digitalizaci. Hlavním aspektem je rozdílná časová náročnost digitalizace vytypovaných skupin map. Vlastní postup digitalizace tak byl rozdělen na dva následující bloky:

1. První blok zahrnuje téměř 8 000 katastrálních území o celkové výměře okolo 6 000 ha. Zde se předpokládá, že bude možno digitalizaci provést z dostupných grafických podkladů s omezenou potřebou ověřovacích prací v terénu.

2. Druhý blok zahrnující asi 2 300 katastrálních území o celkové výměře téměř 1 700 ha s ohledem na nedokončené přídělové a scelovací řízení předpokládá dokončení pozemkové úpravy (komplexní nebo jednoduché). Při realizaci je tak vyžadována součinnost s pozemkovými úřady, rozsah je přímo podmíněn možnostmi financování těchto pozemkových úprav.

Kapitola 4

Výměnný formát ISKN

Obsah této kapitoly čerpá především z dokumentu „Struktura výměnného formátu informačního systému katastru nemovitostí České republiky“ ze dne 26.11.2004 v plném znění dostupného na [31].

Vývoj výměnného datového formátu byl ovlivněn přirozeně implementací ISKN v roce 2001. V ISKN jsou totiž data popisná a geodetická uložena ve společném datovém modelu, což ve svém důsledku vedlo k vytvoření nového výměnného formátu (NVF). Tento nový formát tak pomalu nahradil starý výměnný formát (SVF), jehož oficiální podpora byla zajištěna ještě tři roky po nasazení ISKN (tedy do září 2004).

V současné době tak lze narazit na dva typy výměnných formátů:

1. výměnný formát ISKN v textovém tvaru (*nový výměnný formát* – **NVF**)
   Tento formát obsahuje podle zadané kombinace *datových bloků* popisné i grafické informace včetně dat o řízení.

   Data se mohou vztahovat na následující *časové režimy*:

   - prvotní data (z libovolného časového okamžiku po zavedení ISKN)
   - změny za zadané období (datum od – do včetně časového otisku)

   Dále lze definovat *rozsah* poskytovaných dat:

   - územní jednotka (katastrální území, obec, okres, ČR)
   - oprávněný subjekt
   - výběr parcel
   - výběr parcel polygonem v mapě

   Struktura tohoto formátu je kompletně popsána v dokumentu [31].

2. výměnný formát KN před zavedením ISKN (*starý výměnný formát* – **SVF**)

Tento formát obsahuje dvě samostatné části:

- SPI (soubor popisných informací KN) – informace o parcelách, vlastnících, nabývacích titulech
  - soubory ve formátu DBF – a to ve 2 variantách:
    - SPI bez jiných právních vztahů
    - SPI s jinými právními vztahy
  - soubory ve formátu TXT

  Rozsah poskytovaných dat:
  - územní jednotka (katastrální území, obec, okres, ČR)
  - výběr parcel
  - oprávněný subjekt (pouze soubory ve formátu TXT)

  Rozdíly mezi daty uloženými ve formátech DBF a TXT jsou způsobené převážně neexistencí některých položek v novém datovém modelu ISKN, nebo jejich jinou interpretací.
- SGI (soubor geodetických informací) – informace o poloze předmětné nemovitosti
  DKM (digitální katastrální mapa) – soubory ve formátu VKM
  Rozsah poskytovaných dat:
  - katastrální území, kde již byla provedena digitalizace

V dalším textu se budeme zabývat pouze *novým výměnným formátem*. To ve svém důsledku znamená, že návrh modulu `v.in.vfk` se dotýká pouze tohoto formátu!

4.1 Struktura nového výměnného formátu ISKN

Cílem této kapitoly není ani v nejmenším vyčerpávající popis struktury nového výměnného formátu ISKN – úplný popis tohoto formátu je volně dostupný na Internetu [31]. Spíše si pouze zdůrazníme ty prvky, které jsou zásadní pro návrh modulu `v.in.vfk`.

Datový soubor výměnného formátu je prostý textový[1] (ASCII) soubor, který obsahuje tři základní části:

- *hlavička* `&H`
- *datové bloky* `&B` obsahující informace o atributech a jejich formátu
- vlastní *datové řádky* `&D`
- *koncový znak* `&K`

Každá věta je zakončena znaky `<CR><LF>`, přičemž znak „¤“ signalizuje, že následující řádek je pokračováním předchozího řádku a tvoří jedinou větu.

[1] Kódování češtiny dle ČSN ISO 8859-2 (ISO Latin2), pouze ve výjimečných případech může být použito kódovaní WIN1250. Bohužel není zmíněno kódování Unicode/UTF-8.

4.1.1 Hlavička souboru

Hlavička je uvozena znakem `&H`, následuje označení položky a příslušné hodnoty oddělené středníkem, např.:

```
&HVERZE;"2.8"
```

Seznam položek hlavičky je uveden v tab. 4.1.

Tabulka 4.1: Hlavička výměnného formátu ISKN

`VERZE`	označení verze VF
`VYTVORENO`	datum a čas vytvoření souboru
`PUVOD`	původ dat
`CODEPAGE`	označení kódové stránky
`SKUPINA`	seznam skupin datových bloků souboru
`JMENO`	jméno osoby, která soubor vytvořila
`PLATNOST`	časová podmínka použita pro vytvoření souboru
`KATUZE`	omezující podmínka – katastrální území
`OPSUB`	omezující podmínka – oprávněné subjekty
`PAR`	omezující podmínka – parcely
`POLYG`	omezující podmínka – polygon

VERZE

Označuje verzi výměnného formátu.

VYTVOŘENO

Specifikuje datum a čas vytvoření souboru.

PŮVOD

Určuje původ dat, standardní hodnotou je „ISKN“.

CODEPAGE

Povolené hodnoty jsou „WE8ISO8859P2“ (ČSN ISO 8859-2) a „EE8MSWIN1250“ (MS EE WINDOWS-1250).

PLATNOST

Data odpovídají stavu v určitém čase (např. ”03.06.1999 09:56:42”;”03.06.1999 09:56:42”) nebo v období (např. ”05.07.1998 09:56:42”;”03.06.1999 09:56:42”).

Související položky: `ZMENY`, která může nabývat hodnoty 0 (věty platné v určitém okamžiku požadovaného období) nebo 1 (změnové věty za požadované období), a `NAVRHY`: 0 (nejsou obsaženy potvrzené GP) nebo 1 (jsou obsaženy potvrzené GP).

KATUZE

První řádek (uvozen `&B`) definující hlavičku omezující podmínky je následován řádkem s daty této omezující podmínky. Počet řádků tak představuje počet katastrálních území definujících omezující podmínku. Např.:

```
&HKATUZE;KOD N6;OBCE_KOD N6;NAZEV T48;PLATNOST_OD D;PLATNOST_DO D
&DKATUZE;675008;588296;"Kotojedy";"06.01.1999 12:00:00";""
```

OPSUB

Definiční řádek (`&B`) omezující podmínky může být následován řádky s daty tvořící tuto omezující podmínku. Např. (v jednom řádku):

```
&HOPSUB;ID N30;STAV_DAT N2;DATUM_VZNIKU D;DATUM_ZANIKU D;PRIZNAK_KONTEXTU N1;
RIZENI_ID_VZNIKU N30;RIZENI_ID_ZANIKU N30;ID_JE_1_PARTNER_BSM N30;
ID_JE_2_PARTNER_BSM N30;ID_ZDROJ N30;OPSUB_TYPE T10;CHAROS_KOD N2;ICO N8;
DOPLNEK_ICO N3;NAZEV T255;NAZEV_U T255;RODNE_CISLO T10;TITUL_PRED_JMENEM T35;
JMENO T24;JMENO_U T24;PRIJMENI T35;PRIJMENI_U T35;TITUL_ZA_JMENEM T10;
CISLO_DOMOVNI N4;CISLO_ORIENTACNI T4;NAZEV_ULICE T32;CAST_OBCE T48;OBEC T48;
OKRES T32;STAT T23;PSC N5

&DOPSUB;603800708;0;"11.11.1996 12:00:00";"";2;910483708;;;;;"OPO";4;621;6;
"MO ČR-VUSS BRNO";"MO CR-VUSS BRNO";"";"";"";"";"";"";"";;"";"";"";
"BRNO SVATOPLUKOVA 84";"";"";
```

PAR

Princip je podobný – definiční řádek (`&B`) podmínky je doplněn množinou datových řádků (`&D`) popisující omezující podmínku. Např. (v jednom řádku):

```
&HPAR;ID N30;STAV_DAT N2;DATUM_VZNIKU D;DATUM_ZANIKU D;PRIZNAK_KONTEXTU N1;
RIZENI_ID_VZNIKU N30;RIZENI_ID_ZANIKU N30;PKN_ID N30;PAR_TYPE T10;
KATUZE_KOD N6;KATUZE_KOD_PUV N6;DRUH_CISLOVANI_PAR N1;KMENOVE_CISLO_PAR N5;
ZDPAZE_KOD N1;PODDELENI_CISLA_PAR N3;DIL_PARCELY N1;MAPLIS_KOD N30;
ZPURVY_KOD N1;DRUPOZ_KOD N2;ZPVYPA_KOD N4;TYP_PARCELY N1;VYMERA_PARCELY N9;
CENA_NEMOVITOSTI N14.2;DEFINICNI_BOD_PAR T100;TEL_ID N30;PAR_ID N30;
BUD_ID N30;IDENT_BUD T1

&DPAR;487180708;-1;"22.10.1998 12:00:00";"14.02.2002 04:35:25";
```

```
1;910775708;915717708;;"PKN";675008;;1;3;;2;;6780;2;13;;;40;;"";
265115708;;95239708;"a"
```

POLYG

Může nabývat pouze hodnot 0 nebo 1. Pokud je uvedena hodnota 1 (tj. obsah souboru je odvozen z omezujícího polygonu), musí být daný polygon definován položkami `&HPOLYGDATA` (v požadovaném pořadí). Např.:

```
&HPOLYGDATA;675124.12;1024587.24
&HPOLYGDATA;675224.12;1024687.24
&HPOLYGDATA;675184.12;1024537.24
```

4.1.2 Datové bloky

Datové bloky obsahují dva typy řádků:

- uvozující řádek (`&B`) – seznam atributů včetně datových typů (viz tab. 4.2). V podstatě obsahuje definici databázové tabulky, název a datový typ jednotlivých sloupců. Jejich pořadí je určující pro další datové záznamy.
- řádek s vlastními daty (`&D`), jednotlivé hodnoty jsou odděleny středníkem

Příklad pro datový blok „parcela“ (vše v jednom řádku):

```
&BPAR;ID N30;STAV_DAT N2;DATUM_VZNIKU D;DATUM_ZANIKU D;PRIZNAK_KONTEXTU N1;
RIZENI_ID_VZNIKU N30;RIZENI_ID_ZANIKU N30;PKN_ID N30;PAR_TYPE T10;
KATUZE_KOD N6;KATUZE_KOD_PUV N6;DRUH_CISLOVANI_PAR N1;KMENOVE_CISLO_PAR N5;
ZDPAZE_KOD N1;PODDELENI_CISLA_PAR N3;DIL_PARCELY N1;MAPLIS_KOD N30;
ZPURVY_KOD N1;DRUPOZ_KOD N2;ZPVYPA_KOD N4;TYP_PARCELY N1;VYMERA_PARCELY N9;
CENA_NEMOVITOSTI N14.2;DEFINICNI_BOD_PAR T100;TEL_ID N30;PAR_ID N30;
BUD_ID N30;IDENT_BUD T1

&DPAR;28256708;0;"12.12.199612:00:00";"";2;96835708;;;"PKN";675008;;1;100;
;;;6780;2;13;;;113;;"";13913708;;4282708;"a"
```

Dále následuje popis jednotlivých skupin datových bloků, v uvedených tabulkách je obsažen identifikátor skupiny *is* a jednotlivých datových bloků v rámci skupiny *ib*. Tyto identifikátory nejsou součástí oficiální dokumentace výměnného formátu ISKN a jsou užity pouze pro interní potřebu modulu `v.in.vfk`. Pro zachování kontinuity jsou uvedeny již nyní, více v kapitole 5.1.2.

Skupina datových bloků „Nemovitosti“

První a zároveň největší skupina datových bloků může obsahovat celkem až 17 různých typů datových bloků.

Tabulka 4.2: Datové typy výměnného formátu ISKN

kód	datový typ	číslo za kódem
N	číselný	maximální délka položky
	10.2 → maximálně 10 číslic a z toho 2 za desetinnou čárkou bez nevýznamných nul – 0.2 → .2 kladná čísla bez znaménka	
T	textový	maximální délka textové položky
D	datumový	tvar `DD.MM.YYYY HH:MI:SS`
	např. „09.09.1999 08:24:05“	

Jsou zde obsaženy informace o parcelách[2] jako územním prvku, např. výměra parcely (včetně způsobu určení), druh pozemku (zemědělská kultura, stavební parcela), využití pozemku v rámci druhu pozemku, zdroj evidence parcel v půdních celcích.

Další položkou jsou budovy – typ stavby/budovy, cena nemovitosti, číslo popisné včetně popisu částí budov. Zakomponovány jsou informace dotýkající se způsobu ochrany nemovitostí z oblasti ochrany přírody, kultury či zdravotnictví atd.

Dále obsahuje číselník mapových listů katastrálních map, dotčených katastrálních území, obcí a jejich částí (existujících a zaniklých), okresů a krajů.

Seznam všech definovaných datových bloků v rámci této skupiny je uveden v tab. 4.3, identifikátor skupiny je $is = 10$.

Pozn: Hvězdička uvedená za kódem datového bloku označuje číselníky centrálně spravované.

Skupina datových bloků „Jednotky“

Jednotkou se v tomto smyslu rozumí byt či nebytový prostor vyčleněný na základě příslušné listiny jako jednotka. Popis jednotky obsahuje jednoznačnou identifikaci v rámci budovy, typ a popis způsobu využití jednotky (byt, ateliér, garáž atd.). Součástí zápisu vlastnictví je taktéž spoluvlastnický podíl (vyjádřený zlomkem ve vztahu k poměru velikosti podlahové plochy a celkové plochy všech jednotek v domě).

Seznam datových bloků v rámci této skupiny je uveden v tab. 4.4, identifikátor skupiny je $is = 11$.

[2]Parcelou se rozumí pozemek, který je geometricky a polohově určen, zobrazen v katastrální mapě a označen parcelním číslem.

Tabulka 4.3: Skupina datových bloků „Nemovitosti“ – NEMO

$is = 10$			
ib	kód	databázová tabulka	popis
01	PAR	parcely	parcely
02	BUD	budovy	budovy
03	CABU	casti_budov	části budov
04	ZPOCHN*	zp_ochrany_nem	číselník způsobů ochrany nemovitosti
05	DRUPOZ*	d_pozemku	číselník druhů pozemku
06	ZPVYPO*	zp_vyuziti_poz	číselník způsobů využití pozemku
07	ZDPAZE*	zdroje_parcel_ze	číselník zdrojů parcel ZE
08	ZPURVY*	zp_urceni_vymery	číselník způsobů určení výměry
09	TYPBUD*	t_budov	číselník typů budov
10	MAPLIS*	mapove_listy	číselník mapových listů
11	KATUZE*	katastr_uzemi	číselník katastrálních území
12	OBCE*	obce	číselník obcí
13	CASOBC*	casti_obci	číselník částí obce
14	OKRESY*	okresy	číselník okresů
15	KRAJE*	kraje	číselník krajů
16	RZO	r_zpochr	přiřazení způsobu ochrany k nemovitostem
17	ZPVYBU*	zp_vyuziti_bud	způsob využití budov

Tabulka 4.4: Skupina datových bloků „Jednotky“ – JEDN

$is = 11$			
ib	kód	databázová tabulka	popis
01	JED	jednotky	jednotky
02	TYPJED*	t_jednotek	číselník typů jednotek
03	ZPVYJE*	zp_vyuziti_jed	způsob využití jednotek

Skupina datových bloků „Bonitní díly parcely“

Tato skupina může obsahovat pouze jeden datový blok, který nese informace o bonitních dílech parcely – popisuje tak vztah mezi BPEJ (bonitovaná půdně-ekologická jednotka) a parcelou.

Daná skupina je popsána v tabulce 4.5, identifikátor skupiny je $is = 12$.

Tabulka 4.5: Skupina datových bloků „Bonitní díly parcely“ – BDPA

$is = 12$			
ib	kód	databázová tabulka	popis
01	BDP	bonit_dily_parc	bonitní díly parcel

Skupina datových bloků „Vlastnictví“

Jak již název této skupiny napovídá, jsou zde uloženy informace o vlastnictví (vlastnický vztah jako typ právního vztahu).

Vlastník je v tomto kontextu chápan jako fyzická, právnická osoba nebo jiný oprávněný (manželé v bezpodílovém spoluvlastnictví) uživatel.

Seznam všech definovaných datových bloků v rámci této skupiny je uveden v tab. 4.6, identifikátor skupiny je $is = 13$.

Tabulka 4.6: Skupina datových bloků „Vlastnictví“ – `VLST`

$is = 13$			
ib	kód	databázová tabulka	popis
01	`OPSUB`	oprav_subjekty	oprávněné subjekty
02	`VLA`	vlastnictvi	vlastnictví
03	`CHAROS`*	char_os	číselník charakteristik oprávněných subjektů
04	`TEL`	telesa	katastrální tělesa

Skupina datových bloků „Jiné právní vztahy“

Jiný právní vztah je chápan jako jiný než vlastnický vztah jednoho subjektu (oprávněného subjektu nebo nemovitosti) k jednomu předmětu (k jedné nemovitosti, k vlastnictví nebo k dalšímu jinému právnímu vztahu). Číselník typu právního vztahu (vlastnictví, věcné břemeno, právo trvalého užívání, atd.) je spravován centrálně a má hierarchickou stavbu, tj. vyšší úroveň v hierarchii číselníku je označena jako předmět řízení při přípravě aktualizace.

Seznam těchto datových bloků obsahuje tab. 4.7, identifikátor skupiny je $is = 14$.

Tabulka 4.7: Skupina datových bloků „Jiné právní vztahy“ – `JPVZ`

$is = 14$			
ib	kód	databázová tabulka	popis
01	`JPV`	jine_prav_vztahy	jiné právní vztahy
02	`TYPRAV`*	t_pravnich_vzt	číselník typů právních vztahů

Skupina datových bloků „Řízení“

Druhou největší skupinu datových bloků „Řízení“ je možné do NVF zařadit při exportu změnových vět (tj. zadáno datum `datum_od` a `datum_do`). Potom pro všechny entity tohoto bloku platí, že jsou předávány stejně jako změnové věty ostatních číselníků (tj. obsahují pouze záznamy, které byly v zadaném časovém intervalu změněné).

První tabulka rizeni definuje hlavičku řízení, která obsahuje základní atributy každého řízení. V další tabulce je vyjádřen vztah řízení ke katastrálnímu území. Přirozeně zde existuje vazba na objekt řízení – nemovitost (tj. parcela, jednotka nebo budova). V objektu řízení však nejsou zapsány nově vznikající nemovitosti. Kromě objektu je uveden taktéž předmět (a jeho centrálně spravovaný typ) řízení, který podrobněji specifikuje náplň daného řízení. Uveďme si příklad: pro řízení typu vklad je typem předmětu řízení charakteristika práva zapisovaného do katastru daným řízením (může se jednat např. o vlastnické právo, věcné břemeno atd.).

Skupina zahrnuje také informace o účastnících řízení. Účastníkem se rozumí fyzická či právnická osoba dotčená příslušným řízením. Jeden účastník může mít více typů adres (viz tabulka adresy). Typ účastníka je uveden v centrálně spravovaném číselníku – typy_ucastniku.

Dalším faktorem jsou listiny (základní údaje listiny sloužící jako podklad pro rozhodování a zápis do KN v rámci řízení). Listina je potom zařazena podle jejího typu do kategorií dle použití, významu a právních úkonů spojených s listinami (např. „Smlouva o převodu vlastnictví nemovitosti“). Navíc je uchovávána informace o stavu řízení i ve formě obeslání v rámci definovaných operací (doručenka přijata, storno korespondence, atd.).

Seznam všech definovaných datových bloků v rámci této skupiny je uveden v tab. 4.8, identifikátor skupiny je $is = 15$.

Tabulka 4.8: Skupina datových bloků „Řízení“ – RIZE

$is = 15$			
ib	kód	databázová tabulka	popis
01	RIZENI*	rizeni	řízení (vklad, záznam)
02	RIZKU*	rizeni_ku	vazba řízení – katastrální území
03	OBJRIZ*	objekty_rizeni	objekty řízení (parcely, budovy,..)
04	PRERIZ*	predmety_rizeni	předměty řízení
05	UCAST*	ucastnici	účastníci řízení
06	ADRUC*	adresy	adresy účastníků řízení
07	LISTIN*	listiny	listiny
08	DUL*	dalsi_udaje_listiny	další údaje listin
09	LDU*	listiny_dalsi_udaje	vazba listiny – další údaje listin
10	TYPLIS*	t_listin	číselník typů listin
11	TYPPRE*	t_predmetu_r	číselník typů předmětu řízení
12	TYPRIZ*	typy_rizeni	typy řízení
13	TYPUCA*	typy_ucastniku	typy účastníků řízení
14	UCTYP*	ucastnici_typ	vazba mezi účastníky a typy účastníků řízení
15	RL	r_list	přiřazení listin k nemovitostem, vlastnictví a jiným právním vztahům
16	OBESMF*	obeslani_mf	obeslání účastníků řízení

Skupina datových bloků „Prvky katastrální mapy“

Tato zcela zásadní skupina datových bloků (alespoň pro modul `v.in.vfk`) obsahuje kromě popisných i *geodetické* informace. Tedy datový podklad na základě něhož je vytvořena geometrie vektorové mapy, více v kapitole 5.

Hned první dva uvedené datové bloky v podstatě tvoří kostru vektorové mapy a jsou tak zcela nepostradatelné. Jde o tabulku souradnice_obrazu, která obsahuje body polohopisu (čísla bodů, souřadnice obrazu v mapě a odkaz na centrálně spravovaný číselník přesnosti určení souřadnic daného bodu). Přirozeně je definován taktéž použitý souřadnicový systém (entita t_sourad_sys). Druhá tabulka v pořadí (tj. spojeni_b_poloh) naopak reprezentuje vazbu mezi těmito body, jejichž spojením vzniká liniový polohopisný prvek katastrální mapy – tj. linie či hranice plochy. Např. v případě hraniční linie parcely (sloupec `hp_id`) existuje vazba na tabulku hranice_parcel.

Dále je definována vazba mezi podrobnými body, jejichž spojením vzniká liniový nepolohopisný prvek katastrální mapy (viz spojeni_b_mapy).

Tabulky obrazy_parcel a obrazy_budov obsahují entity zobrazující parcelu, resp. budovu nově zapsanou v KN (a další objekty zjištěné při obnově katastrálního operátu novým mapováním). Budovy jsou v katastru nemovitostí geometricky a polohově určeny obdobně jako parcely. Obvod budovy je liniovým prvkem mapy reprezentující vnější obvod zastavěné plochy. V terminologii GRASSu jsou tyto tabulky podkladem pro tzv. centroidy (definiční body ploch). Další prvky mapy (polohopisné a popisné prvky) jsou uloženy v tabulce dalsi_prvky_mapy. Bližší informace jsou uvedeny v kapitole 5.2.3.

Předposlední tabulka obrazy_bodu_bp obsahuje body polohových polí (ty jsou součástí katastrálních map dekadických měřítek a DKM, zobrazení tohoto bodového pole v KM-D je problémem vymezení obsahu KM-D).

Obsah této skupiny zachycuje tab. 4.9, identifikátor skupiny je $is = 16$.

Skupina datových bloků „BPEJ“

Tato skupina datových bloků nese informaci o bonitovaných půdně ekologických jednotkách (BPEJ), které jsou základní mapovací a oceňovací jednotkou zemědělských půd. Vyjadřuje rozdílné produkční a ekonomické efekty zemědělského území.

Hranice BPEJ je liniový prvek, který tvoří rozhraní mezi dvěma různými BPEJ, není však polohopisným prvkem katastrální mapy.

Seznam těchto datových bloků je uveden v tab. 4.10, identifikátor skupiny je $is = 17$.

Tabulka 4.9: Skupina datových bloků „Prvky katastrální mapy“ – PKMP

$is = 16$			
ib	kód	databázová tabulka	popis
01	SOBR*	souradnice_obrazu	souřadnice obrazu bodů polohopisu v mapě
02	SBP	spojeni_b_poloh	spojení bodů polohopisu – definuje polohopisné liniové prvky
03	SBM	spojeni_b_mapy	spojení bodů mapy – definuje nepolohopisné liniové prvky
04	KODCHB*	kody_char_q_bodu	číselník kódů charakteristiky kvality bodu
05	TYPSOS*	t_sourad_sys	číselník typů souřadnicových systémů
06	HP	hranice_parcel	hranice parcel
07	OP	obrazy_parcel	obrazy parcel (parcelní číslo, značka druhu pozemku,..)
08	OB	obrazy_budov	obrazy budov (obvod budovy, značka druhu budovy)
09	DPM	dalsi_prvky_mapy	další prvky mapy
10	OBBP	obrazy_bodu_bp	obrazy bodů bodových polí
11	TYPPPD*	t_prvku_p_dat	číselník typů prvku prostorových dat

Tabulka 4.10: Skupina datových bloků „BPEJ“ – BPEJ

$is = 17$			
ib	kód	databázová tabulka	popis
01	HBPEJ	hranice_bpej	hranice BPEJ
02	OBPEJ	oznaceni_bpej	označení BPEJ

Skupina datových bloků „Geometrický plán“

Předposlední skupina datových bloků popisuje geometrický plán (hlavičku dalších změn v KN, které nejsou prováděny geometrickým plánem).

Jsou zde skladovány taktéž záznamy podrobného měření změn (ZPMZ) jako doklad, do něhož se zaznamenávají výsledky zaměřování změn v terénu. ZPMZ se vyhotovuje i k vyznačení takových změn, které nejsou spojené s měřením v terénu (např. slučování parcel, demolice budov, atd.).

Poslední uvedená tabulka souradnice_polohy obsahuje geodetické informace, její obsah tak může ovlivnit podobu výsledné vektorové mapy. Jde o body polohopisu (čísla bodů a měřené souřadnice) včetně odkazu na dané ZPMZ.

Seznam těchto datových bloků je uveden v tab. 4.11, identifikátor skupiny je $is = 18$.

Poznámka: Pokud je nastaveno &HNAVRHY;1, pak datové bloky NZ a NZZP nejsou součástí importu.

Tabulka 4.11: Skupina datových bloků „Geometrický plán“ – GMPL

$is = 18$			
ib	kód	databázová tabulka	popis
01	NZ	navrhy_zmen_km	hlavičky geometrických plánů a ostatních změn KM
02	ZPMZ	zpmz	hlavičky ZPMZ
03	NZZP	nz_zpmz	vazební tabulka návrhy změn KM – ZPMZ
04	SPOL	souradnice_polohy	souřadnice polohy bodů polohopisu (měřené)

Skupina datových bloků „Rezervovaná čísla“

Poslední skupina datových bloků reprezentuje tzv. „rezervovaná čísla“. Konkrétně jde o parcelní čísla, která byla rezervována pro účely vyhotovení geometrického plánu (při potvrzování geometrického plánu se kontroluje, zda byla použita přidělená rezervovaná parcelní čísla). To ve svém důsledku znamená, že při zápisu nové parcely do katastru nemovitostí je její číslo z této tabulky vymazáno.

Tabulka dotcena_par_cisla obsahuje všechna parcelní čísla, která kdy byla použita za dobu elektronického vedení katastru nemovitostí v informačním systému ISKN (od roku 2001). Blok tak slouží ke kontrole již jednou použitého parcelního čísla při obnově katastrálního operátu bez přečíslování. Obdobně jsou skladována všechna parcelní čísla, která byla použita za dobu elektronického vedení katastru nemovitostí v informačním systému KN_2000 (1993 – 2001).

Tyto datové bloky jsou zachyceny v tab. 4.12, identifikátor skupiny je $is = 19$.

Tabulka 4.12: Skupina datových bloků „Rezervovaná čísla“ – REZE

$is = 19$			
ib	kód	databázová tabulka	popis
01	RECI	rez_parcelni_cisla	rezervovaná parcelní čísla
02	DOCI	dotcena_par_cisla	dotčená parcelní čísla
03	DOHICI	dot_hist_par_cisla	dotčená historická parcelní čísla

4.1.3 Koncová značka

Import dat je ukončen ve chvíli, kdy je načten řádek vstupního souboru uvozený značkou &K (nemusí se nutně jednat o řádku poslední), a nebo poslední řádka tohoto souboru.

4.2 Další vývoj výměnného formátu

V tomto roce (2005) se připravuje doplnění výměnného formátu na platformě metajazyka XML. Převod NVF do XML bude integrován přímo v ISKN tak, že bude možné při exportu/importu

zvolit přímo formát XML. Předpokládá se, že oba formáty, NVF a XML budou používány paralelně.

Oba výměnné formáty budou mít samozřejmě stejný obsah a totožné interní zpracování bude zaručovat 100% konzistenci mezi údaji obsaženými v obou formátech.

Poznámka: Navržený modul `v.in.vfk`, vzhledem k fázi implementace výměnného formátu postaveného na bázi XML, pracuje pouze s formátem NVF.

Kapitola 5

Návrh modulu GRASSu pro import dat ve výměnném formátu ISKN

V rámci této diplomové práce je nastíněn návrh „plnohodnotného“ modulu GRASSu, který je schopen importovat data ve výměnném formátu ISKN (VFK) a vytvořit odpovídající vektorovou mapu v databance GRASSu.

Ve své podstatě to znamená, že na základě souboru geodetických informací (SGI) vznikne geometrie vektorové mapy (v nativním vektorovém formátu GRASSu řady 6), a soubor popisných informací (SPI) je uložen v souvisejících databázových (atributových) tabulkách. V současnosti je vzhledem k možnostem ovladače `dbf` a konstrukci modulu `v.in.vfk` podporováno skladovaní popisných dat pouze v externím databázovém systému PostgreSQL [3] (tj. ovladač `pg`). Teoreticky lze využít i MySQL či jiné databázové systémy přes rozhraní ODBC. Důvody jsou blíže rozebrány v podkapitole 5.2.2.

Jak již bylo zmíněno v kap. 2, GRASS je především *topologickým* GISem. Výše zmíněný modul však vytvoří mapu, kterou nelze v žádném případě považovat z hlediska topologie dat za „korektní“. To je dáno především charakterem vstupních dat, která dle své definice mohou obsahovat duplicitní linie či centroidy (definiční body ploch), např. podzemní vedení se může krýt s hranicí parcely. Navíc vstupní data obsahují minimálně tři tematické vrstvy – „obrazy parcel“, „obrazy budov“ a „další prvky mapy“.

Importovanou vektorovou mapu je tedy nutné dále zpracovat tak, aby tyto podmínky do jisté míry splňovala. Proto vznikla v rámci této práce také sada jednoduchých podpůrných skriptů (jde především o `v.vfk.extract`). Cílem však nebylo vytvořit komplexní sadu nástrojů, které by postihovaly většinu potřeb uživatelů při práci s digitální katastrální mapou v GRASSu. Jde spíše o ukázku automatizace často rozsáhlých a náročných úloh, bližší informace na toto téma jsou uvedeny v kapitole 6.3.

Náš ostře sledovaný modul nese název `v.in.vfk`. Je napsán v programovacím jazyce C (podobně jako drtivá většina zdrojového kódu GRASSu). Chová se stejně jako ostatní moduly GRASSu – pokud tedy uvedete parametr „help“, zobrazí se krátká charakteristika modulu:

```
GRASS:~ > v.in.vfk help

Description:
 Import VFK file (ISKN) into a GRASS vector map.

Usage:
 v.in.vfk [-ernduf] [input=name] output=name [type=name[,name,...]]

Flags:
  -e   Extend location extents based on input data
  -r   Drop all redundant tables
  -n   Null category (-1)
  -d   Data type timestamp instead of varchar
  -u   Create unique constraints
  -f   Create primary key and foreign key constraints

Parameters:
   input   VFK file to be imported, if not given reads from standard input
  output   Name of output vector
    type   Type
           options: point,line,boundary,centroid,area
           default: point,line,boundary,centroid,area
```

Povinným parametrem je tedy pouze název výstupní mapy (`output`). Vstupní data lze určit pomocí parametru `input` či je jednoduše do modulu přesměrovat. Pro názornost si uvedeme praktickou demonstraci obou přístupů:

```
GRASS:~ > v.in.vfk in=Exportvse.vfk out=bylany
```

```
GRASS:~ > cat Exportvse.vfk | v.in.vfk out=bylany
```

```
GRASS:~ > v.in.vfk out=bylany < Exportvse.vfk
```

Modul navíc disponuje příslušnou manuálovou stránkou (viz dodatek A.3), kterou lze zobrazit pomocí modulu `g.manual`:

```
GRASS:~ > g.manual v.in.vfk
```

Grafické uživatelské rozhraní modulu (které je automaticky generováno funkcí `G_parser ()`) je zobrazeno na obr. 2.3.

Nyní si tedy popíšeme jakým způsobem navržený modul pracuje.

5.1 Vytvoření podkladových tabulek a načtení datových vět

V první fázi modul načte vstupní data a vytvoří příslušné databázové tabulky, které jsou k vektorové mapě připojeny pod předefinovaným číslem vrstvy. Tyto tabulky budeme označovat jako „podkladové“, jelikož v další fázi poslouží pro vytvoření samotné vektorové geometrie a jako základ atributových tabulek. Toto pevné, resp. předdefinované číslo vrstvy se skládá ze dvou

částí:

$$is\,ib \tag{5.1}$$

kde:

is ... identifikátor skupiny datových bloků	$is = [10, 19]$
ib ... identifikátor datového bloku v rámci skupiny datových bloků	$ib = [01, 17]$

Tak například datový blok `SOBR` (tj. tabulka souradnice_obrazu) je součástí skupiny „Prvky katastrální mapy“ (`PKMP`), jejíž pořadové číslo je $is = 16$. Pořadové číslo tohoto bloku v rámci skupiny je $ib = 01$. Podkladová tabulka souradnice_obrazu tak bude k mapě připojena pod číslem vrstvy 1601.

Na konci této fáze je k dispozici prázdná vektorová mapa (neobsahuje tedy žádné vektorové elementy) a řada připojených podkladových tabulek (číslo vrstvy se pohybuje v intervalu $[999, 1903]$). Na základě datových záznamů uložených v těchto tabulkách vznikne později samotná vektorová geometrie mapy.

Formát vstupního souboru může být DOS (řádek je ukončen znaky `<CR><LF>`) nebo UNIX (znak `<LF>`). V případě formátu DOS vypíše modul pouze upozornění:

```
WARNING: DOS text format found, attempting import anyway
```

Import dat tím není nijak ovlivněn (dle definice výměnného formátu jsou jednotlivé věty zakončeny právě kombinací znaků `<CR><LF>`).

5.1.1 Zpracování hlavičky

Data uvedená v hlavičce souboru (tj. řádky uvozené značkou `&H`) můžeme považovat do jisté míry za metadata mapy (tj. data popisující samotná data).

Bohužel právě správa metadat je jednou z řady slabin GRASSu. V současné době (verze 6.0) jsou metadata uložena v prostém textovém (ASCII) souboru. Jeho struktura se navíc liší podle toho, zda jde o data rastrová či vektorová. Doufejme tedy, že bude brzo vybudován jednotný robustní systém správy metadat, který bude respektovat současné standardy a trendy.

Velmi krátce si tedy popíšeme aktuální stav. Metadata rastrové mapy jsou uložena v adresáři `hist/` a částečně i v `cat/` (v rámci dané *lokace* a *mapsetu*, viz kap. 2.3).

Naopak všechna vektorová data obsažena v *mapsetu* jsou uložena pouze v jednom adresáři – `vector/`. Ten obsahuje podadresáře odpovídající jednotlivým vektorovým mapám. Tuto strukturu ukazuje obrázek 2.4. Metadata jsou uložena v souborech `head` (hlavička) a částečně i `hist` (historie). Struktura těchto souborů je podobná – obsahující řádky: „položka: hodnota“.

Položka souboru `hist` obsahuje tři řádky, které jsou zapsány do souboru pomocí knihovní funkce `Vect_hist_command ()`. Význam jednotlivých řádků je natolik zřejmý, že podrobný popis není třeba. Např.:

```
COMMAND: v.in.vfk -r input="Exportvse.vfk" output="kml" type="point,line,area"
GISDBASE: /home/martin/grassdata/
LOCATION: iskn MAPSET: PERMANENT USER: martin DATE: Tue Apr 19 21:39:40 2005
```

Soubor `head` obsahující hlavičku mapy je o něco bohatší, disponuje následujícími položkami:

- *organization* – klíčové slovo `PUVOD`; knihovní funkce:

  `Vect_set_organization ()`

- *digit date* – datum a čas importu dat

  `Vect_set_date ()`

- *digit name* – jméno osoby, která importovaný soubor vytvořila, klíčové slovo `JMENO`

  `Vect_set_person ()`

- *map name* – titulek vektorové mapy, nastaveno na „DKM“

  `Vect_set_map_name ()`

- *map date* – datum exportu z ISKN, klíčové slovo `VYTVORENO`

  `Vect_set_map_date ()`

- *map scale* – měřítko mapy, výchozí hodnota „1“ je nahrazena hodnotou „1000“; modul `v.in.vfk` tedy pracuje s výchozím měřítkem pro vektorovou mapu 1 : 1 000

  `Vect_set_scale ()`

- *other info* – obsahuje zbylé řádky hlavičky. Délka tohoto řádku je omezena pouze na 65 znaků!

  `Vect_set_comment ()`

- *zone* – UTM zóna, v případě S-JTSK je uvedena hodnota „0“

- *map thresh* – prahová hodnota mapy, ponechána je výchozí hodnota („0.000000“)

Kódování

Pokud není nastavena položka `CODEPAGE` na hodnotu „WE8ISO8859P2“, vypíše modul upozornění[1], samotný import však přerušen není. Pokud je tedy kódování vstupního souboru jiné než ISO-8859-2 (např. WIN1250), je vhodné soubor překódovat a posléze nastavit hodnotu položky `CODEPAGE` na „WE8ISO8859P2“. Předpokládáme totiž, že výchozí kódování databázového systému PostgreSQL je právě ISO-8859-2. Obecně je vhodné, aby se kódování vstupního souboru shodovalo s výchozím kódováním databázového systému.

[1] `WARNING: Reading file [test.vfk], codepage is not "ISO-8859-2".`

Jak bylo uvedeno výše, položka „Other info“ může obsahovat maximálně 65 znaků. Většinou však délka množiny řádků uvozených značkou `&H` je delší než tento limit. V tomto případě vypíše modul upozornění

```
WARNING: Reading file [Exportmapa.vfk], header is too long, only
first 65 chars used.
```

Řešení vzniklé situace je následující: Do řádku „Other Info“ je vloženo prvních 65 znaků z množiny datových vět hlavičky (značka `&H`). Zároveň je vytvořena pomocná databázová tabulka [mapa]_hlavicka a připojena k mapě pod číslem vrstvy 999. Tabulka obsahuje dva sloupce: `cat` (integer) a `polozka` (varchar), jehož délka je omezena na 1 024 znaků. Položka hlavičky (tj. řádek uvozený značkou `&H`) je do tabulky vložena jako datový záznam s pořadovým číslem `cat`. Obsah této tabulky může vypadat například takto:

```
 cat |              polozka
-----+-----------------------------------
   1 | &HVERZE;"2.8"
   2 | &HVYTVORENO;"13.10.2004 09:00:26"
   3 | &HPUVOD;"ISKN"
   4 | &HCODEPAGE;"WE8ISO8859P2"
   5 | &HSKUPINA;"PKMP"
```

Tato tabulka neobsahuje popisná data, která mají návaznost k vektorovým elementům mapy – proto přízvisko „pomocná“. Hlavním důvodem jejího připojení k mapě je zachování konzistence dat skladovaných v databázovém systému. Při manipulaci s mapou (kopírování, odstranění) zasáhne daná operace i tuto tabulku. Podobná definice se vztahuje i na „podkladové“ tabulky, jejichž funkce bude vysvětlena v dalším textu. Na rozdíl od této tabulky však slouží k vytvoření atributových tabulek, potažmo vektorové geometrie.

Nutno dodat, že zpracování hlavičky vstupních dat je provizorní a bude jistě v budoucnu přepracováno.

5.1.2 Zpracování jednotlivých datových bloků

Jak je zmíněno v kap. 2.6, nativní vektorový formát GRASSu (od verze 5.1) umožňuje připojení více atributových tabulek k jedné vektorové mapě. Přístup k jednotlivým tabulkám je řízen jednoznačným identifikátorem – číslem vrstvy. Záznam v atributové tabulce je tedy jednoznačně identifikovatelný na základě čísla vrstvy (*layer*) a hodnoty kategorie (*cat*). Výše popsaný princip je názorně zobrazen na obr. 2.5, str. 12.

Tato technologie přináší řadu výhod a možností uplatnění. Toho je důkazem i popisovaný návrh modulu `v.in.vfk`. Struktura dat ve výměnném formátu ISKN (VFK) předurčuje jejich rozložení do více databázových tabulek. Navíc tu vyvstává nutnost přiřadit k jednomu vektorovému elementu (tedy grafické informaci) hned několik sad popisných informací umístěných

v různých atributových tabulkách.

Proměnná *layer* je celočíselná, proto je jednotlivým datovým blokům navíc přiřazen i číselný identifikátor *isib* (viz tab. A.2).

Vytvoření podkladových tabulek

Řádek uvozený značkou `&B` (označující začátek datového bloku) ve své podstatě obsahuje definici příslušné databázové tabulky. Například na základě (v jednom řádku)

```
&BSBP;ID N30;STAV_DAT N2;DATUM_VZNIKU D;DATUM_ZANIKU D;PRIZNAK_KONTEXTU N1;
RIZENI_ID_VZNIKU N30;RIZENI_ID_ZANIKU N30;BP_ID N30;PORADOVE_CISLO_BODU N38;
OB_ID N30;HP_ID N30;DPM_ID N30;PARAMETRY_SPOJENI T100
```

je vytvořena tabulka [mapa]_spojeni_b_poloh, která je připojena k mapě pod číslem vrstvy 1602 (název mapy `bylany`):

```
CREATE TABLE bylany_spojeni_b_poloh (
  cat integer,
  id numeric(30,0),
  stav_dat numeric(2,0),
  datum_vzniku character varying(19),
  datum_zaniku character varying(19),
  priznak_kontextu numeric(1,0),
  rizeni_id_vzniku numeric(30,0),
  rizeni_id_zaniku numeric(30,0),
  bp_id numeric(30,0),
  poradove_cislo_bodu numeric(38,0),
  ob_id numeric(30,0),
  hp_id numeric(30,0),
  dpm_id numeric(30,0),
  parametry_spojeni character varying(100)
);
```

Další řádky vstupního souboru obsahují jednotlivé datové záznamy, které jsou postupně načítány a vkládány do vytvořené tabulky. Některé sloupce tabulek podle popisu struktury výměnného formátu ISKN [31] nesmí obsahovat tzv. nulové hodnoty (*NULL values*). Modul během vkládání jednotlivých datových záznamů do databázové tabulky kontroluje výskyt těchto hodnot a v případě, že narazí na výskyt hodnoty NULL ve sloupci, který podle definice formátu VFK tuto hodnotu obsahovat nesmí, vypíše upozornění

```
WARNING: Reading file [test.vfk], line 16: found NULL value in column
         <kodchb_kod>!
```

Uživatel je pouze upozorněn na nekonzistenci vstupních dat, samotný import pozastaven není. Výstupní mapa však může obsahovat řadu chyb či chybějících prvků.

Na tomto místě je vhodné upozornit na existenci přepínače `-u`, který má za následek vytvoření *unikátního klíče* na základě struktury formátu VFK (viz [31]).

Pokud je navíc použit přepínač `-n` (prvky mapy bez atributů získají hodnotu kategorie „-1“), je do tabulky přidán tzv.„nulový záznam“.

Takto se postupuje až do řádku se značkou `&B` (nová databázová tabulka, celý proces se opakuje) nebo `&K` (ukončení vstupu, nemusí se nutně jednat o poslední řádek vstupního souboru).

Jako příklad uvedeme import souboru `Exportmapa.vfk` obsahující skupinu datových bloků číslo 16, tj. „PKMP“ – výpis z modulu `v.db.connect` (číslo vrstvy, název tabulky, primární klíč, databáze, ovladač):

```
999 test_hlavicka cat iskn pg
1601 test_souradnice_obrazu cat iskn pg
1602 test_spojeni_b_poloh cat iskn pg
1603 test_spojeni_b_mapy cat iskn pg
1604 test_kody_char_q_bodu cat iskn pg
1605 test_t_sourad_sys cat iskn pg
1606 test_hranice_parcel cat iskn pg
1607 test_obrazy_parcel cat iskn pg
1608 test_obrazy_budov cat iskn pg
1609 test_dalsi_prvky_mapy cat iskn pg
1610 test_obrazy_bodu_bp cat iskn pg
1611 test_t_prvku_p_dat cat iskn pg
```

V souvislosti s volbou datového typu položky (viz tabulka 5.1) vyvstává jedno z bolavých míst tohoto návrhu. A to volba relevantního datového typu pro číselné položky (N). V případě hodnot s plovoucí desetinnou čárkou (v drtivé většině se jedná o souřadnice) je použit datový typ `double`, který dostal přednost před typem `numeric (x,y)`[2].

Tabulka 5.1: PostgreSQL: konverze datových typů

datový typ		
VFK	PostgreSQL	
N1 – N9	`integer`	`numeric(x,0)`
N10 a vyšší	`bigint`	
Nx.y	`double`	`numeric (x,y)`
T[délka]	`varchar (délka)`	
D	`varchar(19)` nebo `timestamp` (viz přepínač `-d`)	

Komplikovanější situace nastává v případě celočíselných položek. Nabízejí se celkem dva přístupy: zvolit datový typ `numeric` s uživatelsky definovanou délkou, nebo `int`/`bigint`. Proti datovému typu `numeric` hovoří fakt, že jeho zpracování je v systému PostgreSQL v porovnání

[2]Proměnná x určuje maximální počet číslic a y udává počet desetinných míst.

s datovým typem `int`, resp. `bigint` znatelně pomalejší (v případě testovacích dat `Exportvse.vfk` trval import při volbě datového typu `numeric` téměř o 40 % delší dobu než u typu `int/bigint`).

Navíc datové typy `numeric` a `bigint` nejsou v rámci DBMI (DataBase Management Interface) GRASSu podporovány! To znamená, že `numeric` je přetypován bez ohledu na měřítko na `double` a `bigint` na `int`! Ani jeden z přístupů se tedy ve svém důsledku neobejde bez modifikace DBMI.

V případě, kdy dojde skutečně k přetypování `bigint` na `int`, vypíše modul korespondující upozornění:

```
WARNING: column 'id' : type int8 (bigint) is stored as integer (4 bytes)
         some data may be damaged
```

Toto přetypování zejména v souvislosti s primárními klíči může vést až k nekorektnímu chování modulu. Proto se jeví jako vhodné přechodné řešení daného problému nasazení datového typu `numeric`.

Na okraj zmíníme ještě přepínač `-f`, který v rámci databázového systému vytvoří strukturu cizích klíčů. To je však ve většině případů zbytečné a naopak to způsobuje problémy, například při snaze odstranit danou vektorovou mapu z databanky GRASSu. V tomto případě je nutné atributové tabulky odstranit manuálně[3].

Poznamenejme ještě možnost použití datového typu `timestamp` pro uložení položek obsahující datum a časový otisk (např. „2001-06-21 17:30:41“). Výchozí nastavení odpovídá typu `varchar` a to především proto, že datový typ `timestamp` není podporován ovladačem `ogr`. Což má za následek nemožnost exportu takové mapy (resp. přidružené atributové tabulky) pomocí modulu `v.out.ogr`.

5.2 Tvorba vektorové geometrie

Druhá fáze importu představuje samotné „naplnění“ vektorové mapy. V první části je vektorová mapa sice vytvořena, ale je prázdná – neobsahuje žádné vektorové elementy.

Dle struktury výměnného formátu ISKN je pevně stanoveno, které databázové tabulky obsahují sledované geoprostorové informace. Tyto tabulky jsou v seznamu připojených podkladových tabulek vyhledány a dále zpracovány. Náplň vektorové mapy je tak přímo závislá na existenci a obsahu těchto „primárních“ podkladových tabulek. Jejich seznam včetně typu vektorových objektů, které na jejich základě vznikají, je uveden v tabulce 5.2.

[3] `DROP TABLE <tabulka> CASCADE`.

Tabulka 5.2: Podkladové tabulky obsahující grafickou informaci

číslo vrstvy	databázová tabulka	vektorový element
1601	souradnice_obrazu	bod
1610	obrazy_bodu_bp	bod
1804	souradnice_polohy	bod
1602	spojeni_b_poloh	linie (hranice)
1603	spojeni_b_mapy	linie (hranice)
1607	obrazy_parcel	centroid
1608	obrazy_budov	centroid
1702	oznaceni_bpej	centroid

5.2.1 Vektorové elementy

Pokud se v seznamu podkladových tabulek nachází alespoň jedna z uvedených tabulek (tab. 5.2), je volána funkce

- pro body a centroidy,

  ```
  int create_points ()
  ```

- pro (hraniční) linie.

  ```
  int create_lines ()
  ```

Návratová hodnota výše uvedených funkcí odpovídá počtu zapsaných vektorových elementů (bodů, linií a centroidů). Jejich struktura je téměř stejná, jedním z parametrů je odkaz na podkladovou tabulku obsahující danou grafickou informaci. Navíc je vytvořen systém relací, kdy dochází k připojení již atributových tabulek k vektorové mapě. Datový záznam (popisná data) je tak jednoznačně přiřazen k danému vektorovému elementu. Tento proces je blíže popsán v dalším textu.

5.2.2 Připojení atributových tabulek

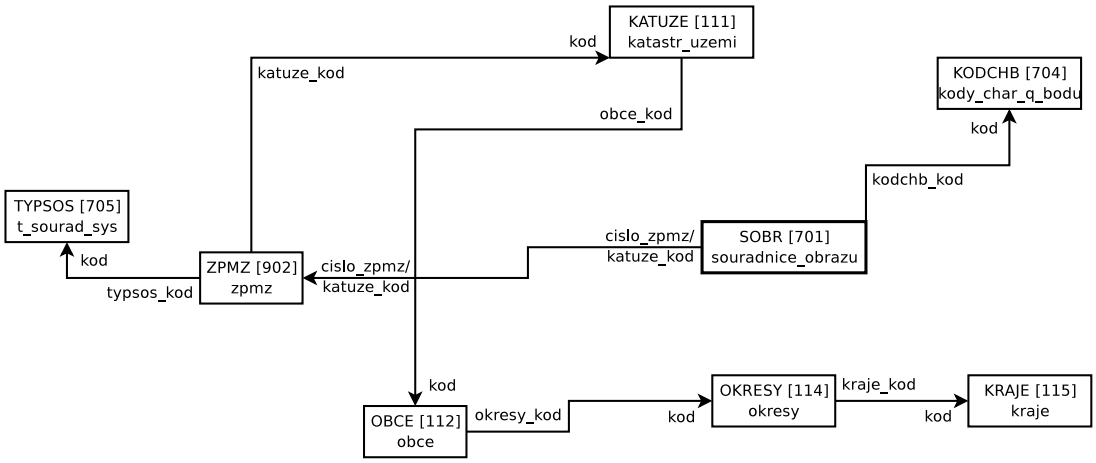
Po zavolání funkce `create_points ()`, resp. `create_lines ()` je před vlastním zapsáním vektorových elementů vytvořena struktura atributových tabulek a kategorií, které jsou posléze připojeny k vektorovým elementům na základě jednoznačné identifikace

$$\text{číslo vrstvy} + \text{kategorie} = \text{databáze} + \text{ovladač}$$
$$\textit{layer} + \textit{cat} = \textit{database} + \textit{driver}$$

V této fázi jsou tedy zkoumány vztahy mezi existujícími podkladovými tabulkami. Ve svém důsledku může být na konci importu k mapě připojeno několik desítek atributových tabulek (v případě souboru `Exportvse.vfk` jich je více než 100)! Některé tabulky jsou totiž připojeny k mapě vícenásobně.

Při této činnosti nedochází k žádné manipulaci v rámci databázového systému. Podkladové tabulky nejsou odstraňovány či kopírovány. Dochází pouze k přidávání odkazů na tyto tabulky do souboru `dbln` (viz obr. 2.4). V rámci použité terminologie se mění statut tabulek, přestávají být pouze „podkladovými“ a stávají se atributovými.

Pro názornou ilustraci jeden příklad: budou vytvořeny vektorové body na základě podkladové tabulky souradnice_obrazu, která je připojena pod číslem vrstvy 1601. Na obr. 5.1 jsou znázorněny vztahy mezi tabulkami, počínaje výše zmíněnou tabulkou. Ucelený pohled na souvislosti mezi všemi tabulkami poskytuje obrázek A.2 na straně 87.



Obrázek 5.1: Relační vztahy mezi tabulkami počínaje tabulkou souradnice_obrazu

V našem případě tak budou k vektorové mapě připojeny následující atributové tabulky (`v.db.connect`; číslo vrstvy, databázová tabulka, kategorie, databáze a ovladač):

```
1 kml_souradnice_obrazu cat iskn pg
2 kml_zpmz cat iskn pg
3 kml_katastr_uzemi cat iskn pg
4 kml_obce cat iskn pg
5 kml_okresy cat iskn pg
6 kml_kraje cat iskn pg
7 kml_t_sourad_sys cat iskn pg
9 kml_kody_char_q_bodu cat iskn pg
```

Souběžně s tím je naplněna datová struktura obsahující čísla příslušných vrstev a seznam hodnot kategorií, které budou posléze přidány k vytvořeným vektorovým elementům.

V tento okamžik tedy máme k mapě připojeny všechny dostupné atributové tabulky a uložené hodnoty kategorií tak, aby jednotlivé datové záznamy mohly být později korektně přiřazeny k vytvářejícím se elementům vektorové mapy. Vše je tedy připraveno k poslední – zásadní fázi a to vytvoření samotné geometrie vektorové mapy.

5.2.3 Zápis vektorových bodů, linií a centroidů

Obsah vytvořené mapy je dán existencí potřebných podkladových tabulek a parametrem `type`, který může obsahovat následující hodnoty:

- *point* – mapa bude obsahovat pouze body (podkladové tabulky 1601, 1610, a 1804)
- *line* – zápis pouze linií (podkladová tabulka 1602 a 1603)
- *boundary* – zápis pouze hranic ploch (podkladová tabulka 1602 a 1603)
- *centroid* – pouze centroidy (definiční body ploch; podkladové tabulky 1607, 1608 a 1702)
- *area* – kombinace *boundary* + *centroid*

Dané parametry lze libovolně kombinovat – jak demonstruje následující příklad. Výběr elementů přirozeně ovlivní i počet připojených atributových tabulek.

- body
  ```
  GRASS:~ > v.in.vfk -r in=Exportvse.vfk out=kml_body type=point
  ```
  Mapa bude obsahovat 6 146 bodů a připojeno bude celkem 19 atributových tabulek (kromě pomocné tabulky hlavicka).
- linie, plochy (hraniční linie a centroidy)
  ```
  GRASS:~ > v.in.vfk -r in=Exportvse.vfk out=kml_la type=line,area
  ```
  Mapa bude obsahovat 8 445 hraničních linií, 78 linií a 1 201 centroidů, počet připojených atributových tabulek tak vzroste na úctyhodných 82.
- body, linie a plochy
  ```
  GRASS:~ > v.in.vfk -r in=Exportvse.vfk out=kml
  ```
  Parametr `type` se nemusí při této kombinaci prvků použít, jde totiž o výchozí nastavení modulu. Počet připojených atributových tabulek bude 122.
- prázdná mapa
  ```
  GRASS:~ > v.in.vfk in=Exportvse.vfk out=km_prazda type=
  ```
  Jde o zvláštní případ, kdy výstupní mapa neobsahuje žádné prvky – je prázdná a jsou k ní připojeny všechny podkladové tabulky (proto nebyl záměrně použit přepínač `-r`, jinak by totiž byly nemilosrdně odstraněny). Takto jednoduše a do jisté míry i elegantně lze vytvořit odpovídající strukturu tabulek v prostředí databázového systému.

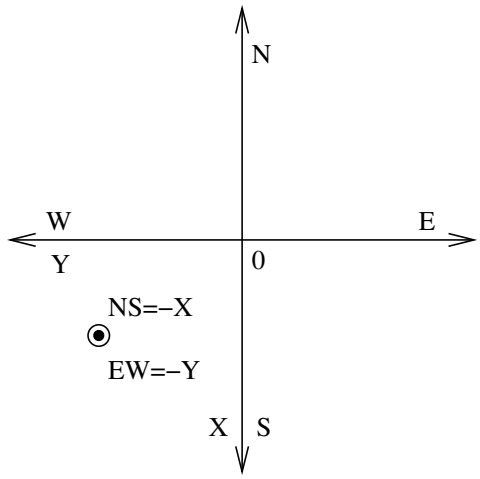
Nyní si tedy popíšeme proces tvorby jednotlivých elementů mapy. Jak již bylo zmíněno, v rámci funkce `create_points ()`, resp. `create_lines ()` je testována existence podkladové tabulky obsahující relevantní geoprostorové informace. Pokud tato tabulka existuje, nic nebrání dané vektorové elementy vytvořit.

Body a centroidy

V případě bodů a centroidů je datový záznam ekvivalentní vektorovému prvku. Zásadní jsou následující sloupce podkladové tabulky:

`cat` – jednoznačný identifikátor záznamu, resp. bodu/centroidu, tzv. *kategorie*

`souradnice_x` a `souřadnice_y` – souřadnice bodu nebo centroidu. Modul podporuje pouze souřadnicový systém S-JTSK. Vzhledem k orientaci os S-JTSK, která je nekompatibilní s požadavky kladenými GISem (hodnota $N > S$ a $E > W$) je nutné provést záměnu souřadnic a jejich vynásobení konstantou -1.0. Tuto situaci názorně ukazuje obrázek 5.2.



Obrázek 5.2: Orientace os S-JTSK a GIS

K vytvořenému vektorového elementu je přiřazena související množina kategorií. Jednoznačnou identifikací je hodnota kategorie (sloupce `cat`) podkladové tabulky obsahující zkoumanou informaci. Na základě této kategorie jsou vybrány hodnoty kategorií v jednotlivých vrstvách a připojeny k aktuálnímu elementu mapy.

Opět si uvedeme krátký demonstrativní příklad a to pro vektorové body vytvořené z podkladové tabulky souradnice_obrazu. Tak například na základě datového záznamu

```
 cat |    id     | ... | souradnice_y | souradnice_x | kodchb_kod
-----+-----------+-----+--------------+--------------+------------
  19 | 311058708 |     |    650715.08 |   1069707.98 |          2
```

je zapsán vektorový bod o kategorii 19 a souřadnicích

$$EW = -650\,715,08; \qquad NS = -1\,069\,707,98$$

Současně jsou tomuto bodu přiřazeny následující dvojice „číslo vrstvy/kategorie“ (viz výpis modulu `v.category`).

číslo vrstvy	1	2	3	4	5	6	7	8
hodnota kategorie	19	1	1	1	1	1	1	2

Seznam atributových tabulek a korespondující číslo vrstvy (opět modul `v.category`):

```
LAYER/TABLE 1/bylany_souradnice_obrazu:
type       count        min        max
point       6114          1       6114
line           0          0          0
boundary       0          0          0
centroid       0          0          0
area           0          0          0
all         6114          1       6114
LAYER/TABLE 2/bylany_zpmz:
type       count        min        max
point       5881          1         80
...
LAYER/TABLE 3/bylany_katastr_uzemi:
type       count        min        max
point       5881          1          1
...
LAYER/TABLE 4/bylany_obce:
type       count        min        max
point       5881          1          1
...
LAYER/TABLE 5/bylany_okresy:
type       count        min        max
point       5881          1          1
...
LAYER/TABLE 6/bylany_kraje:
type       count        min        max
point       5881          1          1
...
LAYER/TABLE 7/bylany_t_sourad_sys:
type       count        min        max
point       5881          1          1
...
LAYER/TABLE 8/bylany_kody_char_q_bodu:
type       count        min        max
point       6114          2          8
...
```

Při doplnění přepínače `-n` získají vektorové elementy, ke kterým nebyla nalezena odpovídající popisná data, hodnotu kategorie „-1“. Tabulky jsou navíc obohaceny o tzv. „nulový“ záznam (viz podkapitola 5.1.2).

Poznámka k centroidům

V případě centroidů nemusí nutně datový záznam reprezentovat vektorový element (podrobnější informace jsou uvedeny v tabulce A.5). Vstupní data mohou (a často tomu tak je) obsahovat hned

několik centroidů na jednu plochu. Typickým případem je situace, kdy jeden centroid definuje typ parcely (pozemková/stavební) a druhý naopak druh pozemku (např. zahrada). Taková plocha by však byla z topologického hlediska závadná – obsahovala by duplicitní definiční bod.

Nejdříve jsou tedy zpracovány ty záznamy, které odpovídají definičnímu bodu stavební, resp. pozemkové parcely (`typppd_kod` = 18 nebo 28). Posléze přicházejí na řadu zbývající datové záznamy dané tabulky. Testuje se návaznost na vektorovou geometrii. Na základě typu prvku prostorových dat (`typppd_kod`) je rozhodnuto, zda se skutečně jedná o centroid. Pokud ano, následuje hledání odpovídající plochy. Zde mohou nastat dvě možnosti:

(a) plocha již centroid obsahuje – v tomto případě jsou k tomuto existujícímu centroidu pouze přiřazeny nové kategorie,

(b) centroid této plochy nebyl ještě definován – může tak být do mapy doplněn (včetně souvisejících kategorií).

Linie

Odlišná situace nastává v případě vektorových linií. Tam je totiž jeden vektorový element, tj. linie, definován minimálně dvěma datovými záznamy v podkladové tabulce spojeni_b_poloh. Počet liniových tahů (segmentů) tvořící linii se řídí podle sloupce `poradove_cislo_bodu`.

V nejjednodušším případě linii tvoří pouze jeden segment. Tento případ je zobrazen na obr. 5.3 pod bodem a). Datové záznamy definující tuto linii mohou vypadat např. takto:

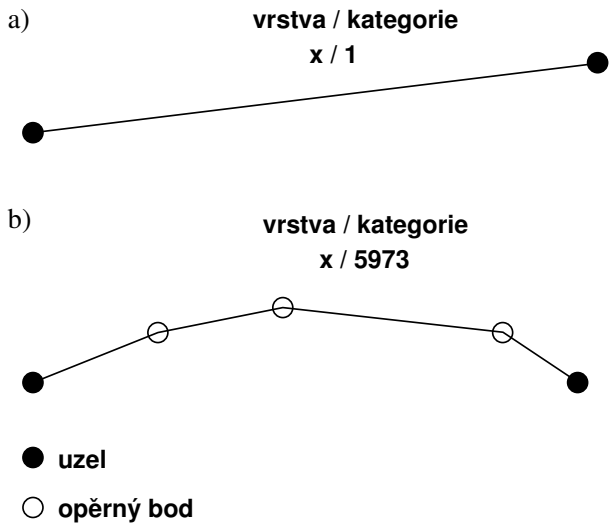
cat	id	...	poradove_cislo_bodu	...	bp_id	...
1	286100708		1		311127708	
2	286101708		2		311130708	

Jak již bylo zmíněno, linie může být tvořena v podstatě neomezeným počtem liniových segmentů. Přičemž daný segment odpovídá vždy páru datových záznamů. Tento případ je prezentován na obr. 5.3 jako bod b). Ukázka podkladových záznamů pro linii tvořenou čtyřmi liniovými segmenty:

cat	id	...	poradove_cislo_bodu	...	bp_id	...
5973	292072708		1		313005708	
5974	292073708		2		313006708	
5975	292074708		3		313007708	
5976	292075708		4		313008708	
5977	292076708		5		313009708	

Kategorie linie je odvozena od hodnoty ve sloupci `cat` prvního záznamu definující linii (počáteční bod prvního liniového segmentu). V prvním případě tak bude vytvořena linie s kategorií 1, v případě druhém pak 5 973.

Vztah mezi tabulkou spojnice_b_poloh a souradnice_obrazu (souřadnice bodů liniových segmentů) je realizován sloupcem `bp_id`, který obsahuje odkaz na `id` daného bodu.



Obrázek 5.3: Vektorová linie a její segmenty

Určení typu vektorové linie

Při určování, zda se jedná o linii, či hranici plochy hraje nezastupitelnou roli typ prvku prostorových dat, jehož kód je uložen ve sloupci `typppd_kod` příslušné tabulky, tj. obrazy_budov, hranice_parcel či dalsi_prvky_mapy.

Nelze však tento rozhodovací proces (linie versus hranice plochy) ztotožnit s určováním centroidů. Zde se rozhoduje pouze o tom, zda se jedná o linii (např. osa železniční koleje norm. rozchodu) či o hranici plochy (např. hranice parcely), nikoliv o existenci prvku jako takového.

Seznam typů prvku prostorových dat včetně jejich kódu a relevantního typu vektorového elementu je uveden v dodatku A.5.

5.3 Závěrečné operace

Po úspěšném načtení dat a vytvoření korespondující vektorové mapy je nutno provést ještě několik závěrečných kroků.

V první řadě jsou ze seznamu odkazů na podkladové tabulky odstraněny ty, které změnily statut a staly se tabulkami atributovými, tj. obsahují popisná data vektorové mapy. Tyto odkazy jsou tak již nadbytečné. Poznamenejme, že nedochází k odstraňování těchto tabulek, ale pouze odkazů na ně. Tuto činnost zajišťuje funkce `clean_dblinks ()`.

Volitelně je možné odstranit všechny zbylé podkladové tabulky (v tomto případě máme na mysli jejich fyzickou likvidaci). K výsledné mapě tak budou připojeny pouze atributové tabulky a tabulka obsahující hlavičku dat (číslo vrstvy 1 až 999). Jedná se o přepínač `-r`.

Nakonec je sestavena topologie vektorových dat – jde o knihovní funkci `Vect_build ()`. Výsledek této činnosti je vypsán na standardní výstup. Tak např. v případě testovacího souboru `Exportvse.vfk` se uživatel dozví asi toto:

```
Building topology ...
15870 primitives registered
Building areas:  100%
894 areas built
16 isles built
Attaching islands:  100%
Attaching centroids:  100%
Topology was built.
Number of nodes     :   7347
Number of primitives:   15870
Number of points    :   6146
Number of lines     :   78
Number of boundaries:   8445
Number of centroids :   1201
Number of areas     :   894
Number of isles     :   16
Number of incorrect boundaries   :   4174
Number of centroids outside area :   373
Number of duplicate centroids    :   106
Number of areas without centroid :   172
```

Kromě počtu zapsaných bodů, linií, hraničních linií, centroidů, sestavených ploch a ostrovů je dobré ověřit i počet nekorektních linií, duplicitních či chybějících centroidů. Tyto charakteristiky ukazují na stupeň „topologické čistoty“ vektorové mapy. Z uvedených hodnot v podstatě vyplývá nutnost dalšího zpracování mapy. A to rozdělení do několika tematických vrstev tak, aby v maximální míře splňovaly přízvisko „topologická data“. Postup dalšího zpracování importované mapy je rozebrán v kapitole 6.

5.4 Výstupní vektorová mapa

Jak již bylo zmíněno výše, výsledná mapa vytvořená modulem `v.in.vfk` obsahuje řadu duplicitních prvků (linií, centroidů). Tato nepříjemná vlastnost je přirozeným důsledkem charakteru vstupních dat.

Je tedy nanejvýš vhodné tuto mapu rozložit do jednotlivých tematických vrstev a to tak, aby tyto mapy v co možná největší míře splňovaly podmínky kladené na vektorová data v topologickém GISu.

Zejména proto vznikl doplňkový skript `v.vfk.extract`, který z mapy extrahuje zadanou tematickou vrstvu. V žádném případě se však nejedná o komplexní nástroj, jde spíše o jakousi ukázku automatizace často rozsáhlých a komplikovaných úloh. Do budoucna se počítá s rozšiřováním sady podobných skriptů s důrazem na potřeby uživatelů pracujících s katastrálními mapami v prostředí GRASSu.

Prvotní činnost – rozložení mapy do několika tematických vrstev (map) by mohla být teoreticky součástí i samotného modulu `v.in.vfk`. V tomto případě by se uživatel mohl rozhodnout, zda chce vytvořit jednu mapu či více tematicky diferencovaných map. Tato myšlenka však byla v době vytváření modulu jako součásti této diplomové práce zavržena. Hlavní důvodem bylo přesvědčení, že by měl mít každý modul na starost pouze jednu činnost – v tomto případě pouze import dat. Komplikovanost modulu jde ruku v ruce s počtem potencionálních chyb.

Postup dalšího zpracování importované mapy a využití dostupných skriptů je popsáno v další kapitole.

Kapitola 6

Zpracování importované katastrální mapy v systému GRASS

Po teoretické části dotýkající se GISu GRASS, informačního systému katastru nemovitostí (ISKN), struktury výměnného formátu ISKN (VFK) a konečně i návrhu modulu `v.in.vfk` určeného pro import dat ve formátu VFK se dostáváme do praktické roviny.

V této kapitole si tedy ukážeme praktický postup při importu dat ve výměnném formátu ISKN, poukážeme na další možnosti zpracování, vizualizace dat a upozorníme na případné problémy.

Jako testovací data byl použit volně dostupný dataset umístěný na webových stránkách ČÚZK [34]. Bližší informace jsou uvedeny v dodatku B.1.

6.1 Založení lokace

Nejdříve musíme založit odpovídající *lokaci* (location), která bude později obsahovat importovaná data. Předpokládejme, že máme k dispozici vstupní data v S-JTSK. Z adresáře, který tato data obsahuje, spustíme GRASS[1]. Zvolíme název naší *lokace* (např. „iskn“), *mapset* můžeme ponechat „PERMANENT“.

```
$ grass60
```

Danou lokaci můžeme založit automaticky na základě kódu EPSG [13]. Na tomto místě však bude popsán manuální přístup: Definujeme tedy mapovou projekci (`D Other Projection`), poté kartografické zobrazení (`krovak`). Geodetické datum doplníme později manuálně (to pro případ potřeby transformace dat do jiných souřadnicových systémů). Jako referenční elipsoid určíme Besselův elipsoid (`bessel`) a jako mapové jednotky ponecháme `meters`. Nakonec zvolíme výchozí region (můžeme ponechat i přednastavené hodnoty). Tento region bude později modifikován na základě vstupních dat (manuálně či přepínačem `-e` modulu `v.in.vfk`). Po potvrzení těchto údajů je založena nová *lokace* a vše je tak připraveno pro další práci. Nyní můžeme do-

[1] Číslovka odpovídá použité verzi, pro aktuální vývojovou větev tak platí `grass61`.

plnit slibované geodetické datum[2]. Nastavení mapové projekce zkontrolujeme pomocí `g.proj -p`.

```
-PROJ_INFO-------------------------------------------------
name       : Krovak
proj       : krovak
ellps      : bessel
towgs84    : 570.8,85.7,462.8,4.998,1.587,5.261,3.56
-PROJ_UNITS-----------------------------------------------
unit       : meter
units      : meters
meters     : 1.0
```

Je nutné taktéž změnit nastavení skladiště atributových dat (jak již bylo zmíněno v kapitole 2, výchozím formátem je dBase, ovladač `dbf`). Jelikož modul `v.in.vfk` aktuálně podporuje pouze externí relační databázový systém typu PostgreSQL, je nutno toto nastavení změnit.

Předpokládejme, že máme již tento databázový systém nainstalován a vytvořenu příslušnou databázi (v našem případě můžeme zvolit stejné jméno jako pro *lokaci*, tedy „iskn“). Přístup k databázi nastavíme pomocí modulu `db.connect`.

```
GRASS:~ > db.connect driver=pg database=iskn
```

Pro kontrolu můžeme toto nastavení vypsat do okna příkazového interpretu (`db.connect -p`)[3].

```
driver:pg
database:iskn
schema:(null)
group:(null)
```

Přístup k databázi lze řešit v rámci nastavení systému PostgreSQL, nebo pomocí modulu `db.login`.

6.2 Import dat ve formátu VKF

Nyní se tedy můžeme vesele pustit do samotného importu dat. Nejjednodušší případ zahrnuje název vstupního souboru a výstupní vektorové mapy umístěné v databance GRASSu.

```
GRASS:~ > v.in.vfk in=Exportvse.vfk out=bylany
```

či

```
GRASS:~ > cat Exportvse.vfk | v.in.vfk out=bylany
```

Třetím parametrem je typ vektorových elementů obsažených ve výsledné mapě. Povolené hodnoty jsou „point“, „line“, „boundary“, „centroid“ a „area“. Bližší informace jsou uvedeny v kapitole 5.2.3. Asi nejčastější kombinací je „point“ a „line,area“.

[2]Do souboru `$GISDBASE/$LOCATION_NAME/PERMANENT/PROJ_INFO` přidáme např. řádek `towgs84: 570.8,85.7,462.8,4.998,1.587,5.261,3.56`.

[3]Problematika schémat a skupin je nad rámec tohoto textu, bude ji tedy ignorovat.

Dále je k dispozici poměrně rozsáhlá množina přepínačů, které se většinou dotýkají popisných dat a jejich podoby.

přepínač	význam
`-e`	rozšířit výchozí region podle importované mapy
`-r`	odstranit všechny podkladové tabulky
`-n`	použít hodnotu kategorie „-1“ pro elementy bez atributů
`-d`	datový typ `timestamp` místo `varchar`
`-u`	vytvořit unikátní klíče
`-f`	vytvořit primární a cizí klíče

Po importu mapy vypíšeme seznam připojených atributových tabulek:

```
GRASS:~ > v.db.connect bylany -g

999 bylany_hlavicka cat iskn pg
1007 bylany_zdroje_parcel_ze cat iskn pg
1003 bylany_casti_budov cat iskn pg
1101 bylany_jednotky cat iskn pg
1102 bylany_t_jednotek cat iskn pg
1103 bylany_zp_vyuziti_jed cat iskn pg
1302 bylany_vlastnictvi cat iskn pg
1401 bylany_jine_prav_vztahy cat iskn pg
1402 bylany_t_pravnich_vzt cat iskn pg
1503 bylany_objekty_rizeni cat iskn pg
1505 bylany_ucastnici cat iskn pg
1506 bylany_adresy cat iskn pg
1509 bylany_listiny_dalsi_udaje cat iskn pg
1511 bylany_t_predmetu_r cat iskn pg
1513 bylany_typy_ucastniku cat iskn pg
1514 bylany_ucastnici_typ cat iskn pg
1516 bylany_obeslani_mf cat iskn pg
1508 bylany_dalsi_udaje_listiny cat iskn pg
1701 bylany_hranice_bpej cat iskn pg
1702 bylany_oznaceni_bpej cat iskn pg
1801 bylany_navrhy_zmen_km cat iskn pg
1803 bylany_nz_zpmz cat iskn pg
1 bylany_souradnice_obrazu cat iskn pg
2 bylany_zpmz cat iskn pg
...
8 bylany_kody_char_q_bodu cat iskn pg
10 bylany_spojeni_b_poloh cat iskn pg
11 bylany_souradnice_obrazu cat iskn pg
...
18 bylany_kody_char_q_bodu cat iskn pg
20 bylany_hranice_parcel cat iskn pg
21 bylany_parcely cat iskn pg
...
32 bylany_zp_vyuziti_bud cat iskn pg
34 bylany_obrazy_budov cat iskn pg
```

```
35 bylany_t_prvku_p_dat cat iskn pg
...
42 bylany_telesa cat iskn pg
44 bylany_dalsi_prvky_mapy cat iskn pg
45 bylany_t_prvku_p_dat cat iskn pg
47 bylany_spojeni_b_mapy cat iskn pg
49 bylany_obrazy_parcel cat iskn pg
50 bylany_t_prvku_p_dat cat iskn pg
...
76 bylany_predmety_rizeni cat iskn pg
78 bylany_obrazy_budov cat iskn pg
79 bylany_t_prvku_p_dat cat iskn pg
...
98 bylany_predmety_rizeni cat iskn pg
100 bylany_obrazy_bodu_bp cat iskn pg
101 bylany_souradnice_obrazu cat iskn pg
...
108 bylany_kody_char_q_bodu cat iskn pg
110 bylany_oznaceni_bpej cat iskn pg
112 bylany_souradnice_polohy cat iskn pg
```

Je zřejmé, že u poměrně značné části tabulek (téměř 30%) nebyla nalezena jejich souvislost s obsahem mapy. To může mít celkem dvě příčiny: buď tabulka neobsahuje žádné záznamy (12 z 20 tabulek), nebo nebyl nalezen její vztah k tabulkám obsahující geodetickou informaci (viz tab. 5.2). Druhý případ může ukazovat na jistou nekonzistenci struktury vztahů mezi tabulkami aplikovanou v tomto modulu.

Další tabulky (tj. číslo vrstvy 1 až 112) dle použité terminologie považujeme za atributové. Mezi jednotlivými tematickými vrstvami (viz tab. 6.1) je vynecháno pro lepší orientaci jedno číslo vrstvy.

Tabulka 6.1: Shrnutí tematických vrstev importované mapy

číslo vrstvy	informace	typ elementu
1 → 8	body podrobného bodového pole	vektorové body
10 → 18	polohopisné prvky mapy	linie, hranice
20 → 32	hranice parcel	hranice
34 → 42	obrazy budov	hranice
44 → 45	další prvky mapy	body, linie, hranice a centroidy
47	nepolohopisné liniové prvky	linie, hranice
49 → 76	obrazy parcel	centroidy
78 → 98	obrazy budov	centroidy
100 → 108	body bodových polí	vektorové body
110	BPEJ	centroidy
112	body polohopisu	vektorové body

Nutno poznamenat, že jsou k mapě připojeny všechny podkladové tabulky obsahující gra-

fickou informaci bez ohledu na to, zda obsahují či neobsahují datové záznamy. Toho je příkladem např. tabulka spojeni_b_mapy, která přestože neobsahuje žádná data, je k mapě připojena jako platná atributová tabulka.

Pokud použijeme při importu dat přepínač `-r`, tak budou všechny podkladové tabulky (tj. 1003 až 1801) nemilosrdně odstraněny (fyzicky, nejen odkazy na tabulky).

Na tomto místě zmíníme další dva přepínače – `-u` a `-f`. První z nich vytvoří dle struktury výměnného formátu ISKN *unikátní klíče*. Druhý přepínač doplní tabulky o *primární* a *cizí klíče*. Unikátní klíče zamezí tomu, aby daná tabulka obsahovala duplicitní záznamy. To by však mělo být ošetřeno již při exportu dat z ISKN. Nutno upozornit, že vytvoření systému cizích klíčů komplikuje manipulaci s daty – např. odstranit databázovou tabulku je možné pouze manuálně: `DROP TABLE <tabulka> CASCADE`!

Jako příklad uvedeme atributovou tabulku souradnice_obrazu:

```
Indexes:
    "bylany_souradnice_obrazu_cat" unique, btree (cat)
    "bylany_souradnice_obrazu_id" unique, btree (id)

--> -u
    "bylany_souradnice_obrazu_katuze_kod_key" unique, btree
    (katuze_kod, cislo_zpmz, cislo_tl, cislo_bodu)

--> -f
Foreign-key constraints:
    "f1" FOREIGN KEY (kodchb_kod) REFERENCES bylany_kody_char_q_bodu(kod)
    ON DELETE CASCADE
```

6.3 Další zpracování vektorové mapy

Vytvořená mapa mírně řečeno nesplňuje podmínky kladené *topologickým* GISem. Jak je vidět z výstupu modulu `v.build` obsahuje řadu nekorektních hranic, duplicitních centroidů či ploch bez centroidů.

```
  Number of nodes     :   7347
  Number of primitives:   15870
  Number of points    :   6146
  Number of lines     :   78
  Number of boundaries:   8445
  Number of centroids :   1201
  Number of areas     :   894
  Number of isles     :   16
  Number of incorrect boundaries   :   4174
  Number of centroids outside area :   373
  Number of duplicate centroids    :   106
  Number of areas without centroid :   172
```

Jak již bylo zmíněno v kapitole 5, je nanejvýš vhodné tuto mapu rozdělit do tematicky diferencovaných vektorových map. Pro tento účel vznikl pomocný skript `v.vfk.extract`.

Tento skript se vizuálně neliší od ostatních modulů GRASSu, disponuje GUI (Tcl/Tk) či krátkou nápovědou (parametr `help`).

Poznámka: Na rozdíl od modulu `v.in.vfk` je tento skript přímo v češtině.

```
Description:
 Extrahování jednotlivých tematických vrstev katastrální mapy.

Usage:
 v.vfk.extract [-tnc] mapa=name [volba=name]

Flags:
  -t   nekopírovat atributové tabulky
  -n   kopírovat atributové tabulky pomocí modulu v.extract
  -c   vyčistit výstupní vektorovou mapu (vhodné pro obrazy budov)

Parameters:
   mapa   vektorová mapa
  volba   tematická vrstva (body polohopisu, obrazy parcel,
          obrazy budov, další prvky mapy, BPEJ)
          options: bp,op,ob,dpm,bpej
          default: op
```

Z dané vstupní mapy (parametr `mapa`) tedy může být extrahováno celkem pět tematických vrstev (tj. vektorové mapy):

- `bp` – body polohopisu mapy a bodových polí (výchozí tabulky: souradnice_obrazu, obrazy_bodu_bp, souradnice_polohy)
- `op` – hranice a definiční body parcel (výchozí tabulky: hranice_parcel a obrazy_parcel)
- `ob` – hranice a definiční body budov (výchozí tabulka: obrazy_budov)
- `dpm` – další prvky mapy (výchozí tabulka: dalsi_prvky_mapy)
- `bpej` – BPEJ (výchozí tabulky: hranice_bpej a oznaceni_bpej)

Jméno výstupní mapy je složeno z jména vstupní mapy a dané volby, např. `bylany_op`. Měly by tedy vzniknout mapy (výjimku může tvořit tematická vrstva „další prvky mapy“ či „BPEJ“), které beze zbytku splňují podmínky kladené na topologická data.

```
GRASS:~ > v.vfk.extract bylany

GRASS:~ > v.build bylany_op

Building topology ...
7044 primitives registered
Building areas:  100%
```

```
1007 areas built
55 isles built
Attaching islands:  100%
Attaching centroids:  100%
Topology was built.
Number of nodes     :   6092
Number of primitives:   7044
Number of points    :   0
Number of lines     :   0
Number of boundaries:   6037
Number of centroids :   1007
Number of areas     :   1007
Number of isles     :   55
```

Skript pracuje v následujících krocích:

1. Vytvoření kopií množiny atributových tabulek (modul `db.copy`, pouze pokud není uveden přepínač `-t` nebo `-e`) – např. bylany_souradnice_obrazu → bylany_bp_souradnice_obrazu.

2. Extrahování jednotlivých vrstev (aplikace modulu `v.extract`, připomeňme, že je bezpodmínečně nutné použít modifikovanou verzi tohoto modulu! – viz dodatek A.4.4), v případě přepínače `-e` jsou v této fázi kopírovány i jednotlivé atributové tabulky. Jistou daní za rychlost této cesty je nevhodné pojmenovaní tabulek na základě čísla vrstvy, tak např. tabulka bylany_souradnice_obrazu dostane název bylany_bp_1.

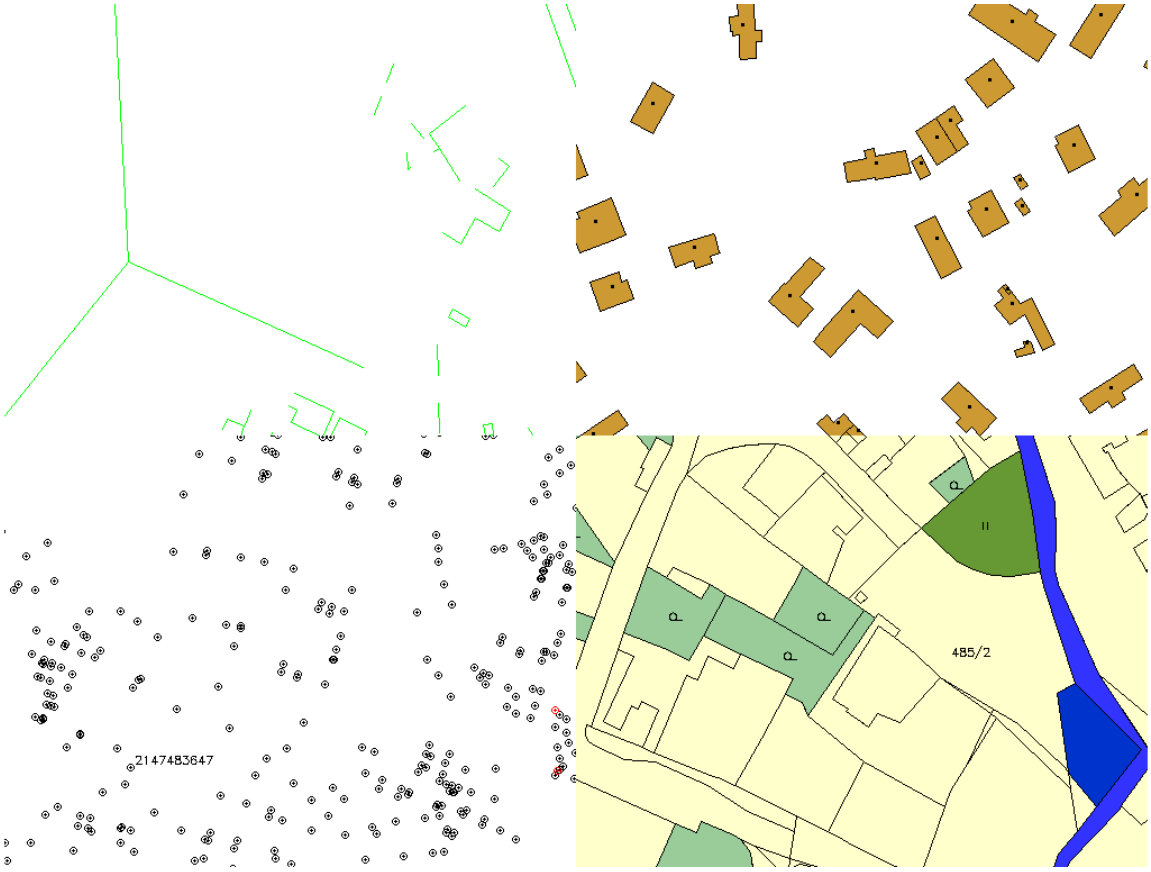
3. Korekce vektorové geometrie výstupní mapy (modul `v.clean`). Lze doporučit v případě tematické vrstvy „obrazy budov“, která může potenciálně obsahovat duplicitní hraniční linie (dvě budovy mají společnou zeď), v ostatních případech jde o zbytečnou operaci.

4. Připojení atributových tabulek k výstupní mapě (modul `v.db.connect`, pouze pokud není uveden přepínač `-t` nebo `-e`).

Vyjdeme-li z tabulky 6.1, tak budou jednotlivé mapy obsahovat tyto vrstvy:

- *body polohopisu*: $1-8;\ 11-18;\ 100-108;\ 112$
- *obrazy parcel*: $20-32;\ 49-76$
- *obrazy budov*: $34-42;\ 78-98$
- *další prvky mapy*: $44-45$
- *bpej*: daná data tuto tematickou vrstvu mapy neobsahují

Kompozice prvních čtyř tematických vrstev je zobrazena na obrázku 6.1.



Obrázek 6.1: Kompozice základních tematických vrstev katastrální mapy

6.4 Vizualizace dat

Kromě zobrazení vektorových dat v grafickém okně, tzv. GRASS monitoru (modul `d.vect`) se dotkneme i problematiky tvorby jednoduchých mapových výstupů s využitím modulu `ps.map`.

Příklad vizualizace dat pomocí modulu `d.vect` je uveden v podkapitole 6.5.2. Ukázky mapových výstupů s využitím modulu `ps.map` jsou naopak prezentovány v dodatku B.2.

6.4.1 Mapové značky katastrální mapy

V GRASSu je implementován poměrně primitivní systém mapových značek. V rámci této práce byla základní skupina značek doplněna o vybrané mapové značky katastrální mapy pokrývající body a centroidy (definiční body ploch). Jejich seznam je uveden v dodatku B.3 na straně 95.

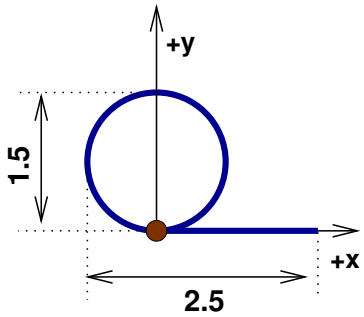
V této části si popíšeme princip tvorby symbolů (tj. mapových značek) v GRASSu, přičemž podrobný popis tohoto formátu je dostupný na GRASS Wiki [32] v sekci „The Grass Programming Howto“→„Display Symbols“.

Všechny značky (symboly) určené pro zobrazení bodů a centroidů se nacházejí v adresáři `$GISBASE/etc/symbol/` nebo lokálně `$GISDBASE/$LOCATION_NAME/$MAPSET/symbol/`. Tyto ad-

resáře obsahují další podadresáře, které odpovídají tematickému členění značek. V současnosti (k vydání verze 6.0) je dostupná tematická skupina „basic“ a „demo“. První uvedená skupina obsahuje, jak název napovídá, základní značky, druhá naopak spíše prezentuje možnosti při vytváření vlastních mapových značek (`smrk` a `muchomurka`).

My se tedy s chutí pustíme do nové tematické skupiny mapových značek `dkm`, vytvoříme adresář `$GISBASE/etc/symbol/dkm/`, kam tyto mapové značky posléze umístíme.

Jako příklad si ukážeme mapovou značku „zahrada“. Její tvar (viz obr. 6.2) je odvozen z vyhlášky č. 190/1996 Sb. [knz03].



Obrázek 6.2: Mapová značka druhu pozemku – „zahrada“

Soubor `zahrada` (v adresáři `$GISBASE/etc/symbol/dkm/`) má následující podobu:

```
VERSION 1.0
BOX -1 -1 1 1
POLYGON
  RING
    FCOLOR NONE
    ARC 0 0.75 0.75 360 0 C
  END
END
STRING
  LINE
     0    0
     1.75 0
  END
END
```

Klíčové slovo `BOX` definuje měřítko mapové značky. Proto je vhodné tyto hodnoty volit pro všechny mapové značky z dané skupiny stejné tak, aby měly totožné měřítko.

Dále je definován kruh (klíčové slovo `ARC`; střed $[0, 0]$; poloměr 0,75) bez vnitřní výplně `FCOLOR NONE`. Nakonec je přidána spodní linie (klíčové slovo `LINE`).

Tím je tato mapová značka hotova a lze ji bez meškání použít např. v souvislosti s modulem `d.vect` nebo `ps.map`.

6.5 Jednoduché příklady analýzy dat

Uvedené analýzy jsou spíše demonstracemi jednotlivých nástrojů GRASSu. Jejich pozice je tedy pouze ilustrativní a tomu odpovídá i zjednodušená forma jejich prezentace.

6.5.1 Zastoupení jednotlivých druhů pozemků v katastrálním území Bylany

V tomto případě se omezíme pouze na tematickou vrstvu katastrální mapy „obrazy parcel“ (odpovídající mapu lze vytvořit aplikací skriptu `v.vfk.extract` s volbou `op`, výsledná mapa může být např. `bylany_op`).

Pozornost bude zaměřena především na atributovou tabulku t_prvku_p_dat, která obsahuje kromě jiného i informace o druhu pozemku parcely (více v dodatku A.5). Budeme předpokládat, že je k uvedené vektorové mapě připojena pod číslem vrstvy 50.

Pro účely vizualizace mapy definujeme barevnou výplň parcel s ohledem na druh pozemku, a to hodnotou (RGB kódem) ve sloupci s unikátním názvem `grassrgb`. Nejprve tedy do příslušné databázové tabulky (tj. t_prvku_p_dat) přidáme nový sloupec:

```
GRASS:~ > echo "ALTER TABLE bylany_op_t_prvku_p_dat ADD COLUMN grassrgb \
varchar(11)" | db.execute
```

a poté definujeme standardní barvy vybraných druhů pozemků[4]:

```
GRASS:~ > echo "UPDATE bylany_op_t_prvku_p_dat SET grassrgb='156:210:150' \
WHERE kod=304" | db.execute
```

Nyní je vše připraveno pro vizualizaci mapy. Nejprve zobrazíme podkladovou mapu s rozlišením pozemkové a stavební parcely a poté již mapu s rozlišením druhu pozemku. Výsledek je uveden v dodatku B.2.

```
GRASS:~ > d.vect -a bylany_op layer=50 where="kod=18 or kod=28"

GRASS:~ > d.vect -a bylany_op layer=50 where="kod!=18 and kod!=28"
```

Kromě toho lze prezentovat výsledky v numerické formě či jako graf. Tabulku t_prvku_p_dat doplníme ještě o sloupce počet (`pocet`) a výměra (`vymera`) parcel. Hodnoty v těchto sloupcích nastavíme na „0“.

```
GRASS:~ > echo "ALTER TABLE bylany_op_t_prvku_p_dat add column pocet int;" | \
db.execute

GRASS:~ > echo "ALTER TABLE bylany_op_t_prvku_p_dat add column vymera int;" | \
db.execute

GRASS:~ > echo "UPDATE bylany_op_t_prvku_p_dat set pocet=0;" | db.execute

GRASS:~ > echo "UPDATE bylany_op_t_prvku_p_dat set vymera=0;" | db.execute
```

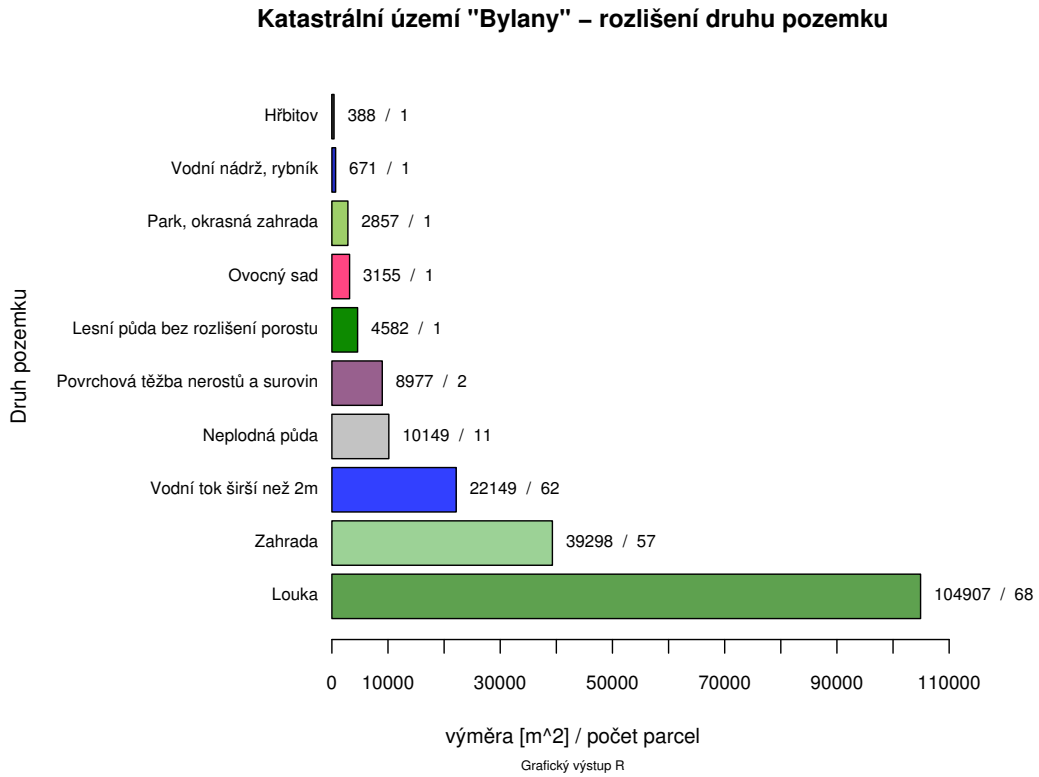
Počet ploch (tj. parcel) a jejich celkovou výměru $[m^2]$ určenou na základě typu prvku prostorových dat určíme pomocí modulu `v.to.db`.

[4] Zdrojový kód diplomové práce obsahuje pro tento účel dávkový soubor – `typppd_barvy.sql`.

```
GRASS:~ > v.to.db bylany_op type=centroid layer=50 opt=count column=pocet

GRASS:~ > v.to.db bylany_op type=centroid layer=50 opt=area column=vymera \
units=me
```

Výsledek bude v našem případě vypadat takto (viz obr. 6.3):

kod	vyznam	pocet	vymera
306	Louka	68	104907
304	Zahrada	57	39298
802	Vodní tok širší než 2m	62	22149
316	Neplodná půda	11	10149
701	Povrchová těžba nerostů a surovin	2	8977
308	Lesní půda bez rozlišení porostu	1	4582
305	Ovocný sad	1	3155
314	Park, okrasná zahrada	1	2857
803	Vodní nádrž, rybník	1	671
315	Hřbitov	1	388



Obrázek 6.3: Zastoupení jednotlivých druhů pozemků v k.u. „Bylany“

6.5.2 Výběr parcel podle oprávněných subjektů

V této části určíme pět oprávněných subjektů s největší výměrou parcel. Podobně jako v minulém odstavci nejprve přidáme do příslušné tabulky (tj. oprav_subjekty) sloupce `pocet` a `vymera`. Současně nastavíme hodnoty v těchto sloupcích na „0“. Počet a výměru parcel určíme opět pomocí modulu `v.to.db` (atributová tabulka oprav_subjekty je připojena k mapě pod číslem vrstvy 72):

```
GRASS:~ > v.to.db map=bylany_op layer=72 opt=count col=pocet
GRASS:~ > v.to.db map=bylany_op layer=72 opt=area col=vymera units=me
```

Prvních pět oprávněných subjektů seřazených podle celkové výměry parcel získáme:

```
GRASS:~ > echo "SELECT nazev,pocet,vymera from bylany_op_oprav_subjekty \
order by vymera desc limit 5;" | db.select
```

```
       nazev        |      nazev_u       | pocet | vymera
--------------------+--------------------+-------+--------
 CHMELOVÁ DITA      | CHMELOVA DITA      |    48 | 489811
 STRAČKA EMIL       | STRACKA EMIL       |    56 | 238332
 TULIPÁN JOSEF ING. | TULIPAN JOSEF ING. |    15 | 236429
 SRDCOVKA BEDŘICH   | SRDCOVKA BEDRICH   |    13 | 228394
 MERUŇKOVÁ JIŘINA   | MERUNKOVA JIRINA   |    24 | 220280
```

Nakonec zobrazíme příslušnou tematickou mapu (obr. 6.4):

```
GRASS:~ > d.vect bylany_op layer=72 where="nazev_u='CHMELOVA DITA'" fcol=red
GRASS:~ > d.vect bylany_op layer=72 where="nazev_u='STRACKA EMIL'" fcol=yellow
GRASS:~ > d.vect bylany_op layer=72 where="nazev_u='TULIPAN JOSEF ING.'" \
fcol=blue
GRASS:~ > d.vect bylany_op layer=72 where="nazev_u='SRDCOVKA BEDRICH'" \
fcol=magenta
GRASS:~ > d.vect bylany_op layer=72 where="nazev_u='MERUNKOVA JIRINA'" \
fcol=green
```

6.5.3 Vyhledání parcel v okruhu 100 m

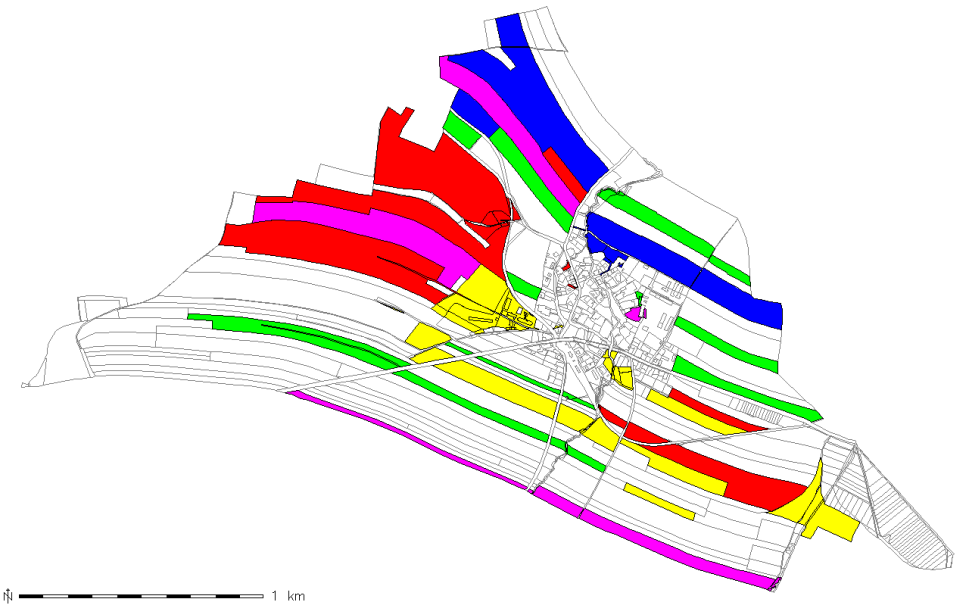
Vyhledáme všechny parcely, které leží v okruhu 100 m od zadaného bodu. Jako první tedy určíme tento výchozí bod, např. $Y = 651391$; $X = 1069366$ a vytvoříme příslušnou bodovou mapu.

```
GRASS:~ > echo "-651391|-1069366" | v.in.ascii out=vbod
```

Dále vytvoříme vyrovnávací kruhovou zónu kolem tohoto bodu o poloměru 100 m.

```
GRASS:~ > v.buffer in=vbod out=zona buffer=100
```

A nakonec vytvoříme vektorovou mapu obsahující pouze parcely zasahující do této zóny – tj. ty, jejichž vzdálenost od výchozího bodu nepřesahuje 100 m (viz levá část obrázku 6.5).



Obrázek 6.4: Výběr parcel podle oprávněných subjektů

```
GRASS:~ > v.overlay ain=bylany_op alayer=49 bin=zona ope=and out=parcely100
```

Kategorie této mapy využijeme pro výběr parcel z mapy `bylany_ob` – pravá část obrázku 6.5.

```
GRASS:~ > d.vect bylany_op layer=49 cats=‘echo "select a_cat from parcely100" \
| db.select -c | gawk ’printf $1","’‘-1
```

Obrázek 6.5: Výběr parcel v okruhu 100 m od výchozího bodu

Kapitola 7

Závěr

Cílem této práce bylo navrhnout modul GRASSu pro import dat ve výměnném formátu ISKN. To bylo, doufejme, do značné míry splněno – vznikl modul `v.in.vfk`. Byl tak ve svém důsledku rozšířen seznam GRASSem podporovaných datových formátů o nový vektorový formát – VFK.

Otázkou je však kvalita tohoto návrhu a použitelnost modulu běžnými uživateli GISu GRASS. Další směřování tohoto projektu v podstatě *plně závisí na odezvě uživatelů* a jejich potřebě pracovat s katastrálními daty dostupnými ve formátu VFK. Modul `v.in.vfk` tak může velmi snadno skončit v propadlišti dějin či naopak se dále rozvíjet.

Modul by měl na základě vstupních dat vytvořit odpovídající vektorovou mapu a doplnit ji příslušnou množinou atributových tabulek. Teoreticky by tak výsledná vektorová mapa měla obsahovat stejné geodetické a popisné informace jako vstupní soubor.

Pro testování modulu byl využit volně dostupný dataset umístěný na webových stránkách ČÚZK [34]. ČÚZK nicméně později poskytl pro tento účel další data (pokrývající část okresu Litoměřice). Tímto tedy děkuji pracovníkům tohoto úřadu za velmi vstřícný postoj, který, doufejme, přispěje k dalšímu zlepšení funkčnosti programu. Svůj dík si v tomto ohledu zaslouží především Ing. Tomáš Chotěnovský za zprostředkování těchto dat a Ing. Petr Kokeš za zodpovězení řady otázek týkajících se problematiky VFK.

Obsah této práce můžeme rozdělit na dvě základní části – *teoretickou* (základní charakteristika GISu GRASS, ISKN, výměnného formátu VFK, koncepce modulu `v.in.vfk`) a *praktickou* (zpracování digitální katastrální mapy v prostředí GRASSu, instalace modulu a jeho lokalizace).

Jako doplňkový produkt vedle modulu `v.in.vfk` vznikl podpůrný skript `v.vfk.extract` a sada vybraných mapových značek katastrální mapy.

Text diplomové práce je dostupný v elektronické podobě na adrese

`http://gama.fsv.cvut.cz/~landa/dp/`

Literatura

[BNM02] Radim Blažek, Marcus Neteler, Roberto Micarelli. The new GRASS 5.1 vector architecture. *Příspěvek na konferenci Open source GIS – GRASS users conference 2002*, 2002.

[DHNR05] Otto Dassau, Stephan Holl, Marcus Neteler, Manfred Redslob. *GRASS GIS 6.0 Kursskript*. 2005.
Český překlad: `http://gama.fsv.cvut.cz/~grass/docs/grass6/`.

[Jed03] Jiří Jedlička. Problematika zpracování vektorových dat v prostředí GIS GRASS, 2003.

[Kad02] Václav Kadlec. *Učíme se programovat v jazyce C*. Computer Press, 2002. ISBN 80-7226-715-9.

[knz03] *Edice „Úplné znění č. 398" – Katastr nemovitostí, zeměměřičství*. 2003. ISBN 80-7208-387-2.

[Kol97] Jan Kolář. *Geografické informační systémy*. Vydavatelství ČVUT, 1997.

[Lan05] Martin Landa. GIS GRASS jako pomůcka při výuce GIS a DPZ. *Příspěvek na konferenci GIS Ostrava 05*, 2005.
`http://gama.fsv.cvut.cz/~landa/grass/gis_ostrava05/ref_grass_go05.pdf`.

[MS01] Neil Matthew, Richard Stones. *Linux – Programujeme profesionálně*. Computer Press, 2001. ISBN 80-7226-532-6.

[Net03] Marcus Neteler. *GRASS Handbuch*. 2003.
Český překlad: `http://gama.fsv.cvut.cz/~grass/docs/handbuch/`.

[NM02] Markus Neteler, Helena Mitášová. *Open source GIS: A GRASS GIS Approach*. Kluwer Academic Publishers Group, 2002. ISBN 1-4020-7088-8.

[Ryb03] Jiří Rybička. *LaTeX pro začátečníky*. Konvoj Brno, 2003. ISBN 80-7302-049-1.

[SM00] Richard Stones, Neil Matthew. *Linux – Začínáme programovat*. Computer Press, 2000. ISBN 80-7226-307-2.

[Vok03] Lucie Vokonouvá. Návrh struktury datového modelu pro správu elektrických distribučních sítí ZČE v GIS analýzou mezinárodního datového modelu ArcFM, 2003.

[Ším03] Jiří Šíma. *Geoinformační terminologie pro geodety a kartografy.* VÚGTK, 2003. ISBN 80-85881-20-9.

[Čep02] Aleš Čepek. *Jemný úvod do C++.* Vydavatelství ČVUT, 2002. ISBN 80-01-01700-1.

Webové odkazy

[1] Společnost Infinity a.s.
`http://www.infinity.cz`.

[2] České sdružení uživatelů GISu GRASS.
`http://gama.fsv.cvut.cz/~grass`.

[3] Relační databázový systém PostgreSQL.
`http://www.postgresql.org`.

[4] Text diplomové práce v elektronické podobě.
`http://gama.fsv.cvut.cz/~landa/dp/`.

[5] Nahlížení do katastru nemovitostí.
`http://nahlizenidokn.cuzk.cz`.

[6] Grafický editor Skencil.
`http://www.skencil.org`.

[7] Grafický editor Xfig.
`http://www.xfig.org`.

[8] Trees for GRASS.
`http://www.fle.czu.cz/~jachym/programs/trees.html`.

[9] Projekt FreeGIS.
`http://www.freegis.org`.

[10] GRASS GIS.
`http://grass.itc.it`.

[11] Open Source GIS.
`http://opensourcegis.org`.

[12] Quantum GIS.
`http://www.qgis.com`.

[13] European Petroleum Survey Group.
`http://www.epsg.org`.

[14] GRASS: import dat ve výměnném formátu ISKN.
`http://gama.fsv.cvut.cz/~landa/grass/vfk/`.

[15] Společnost NESS Technologies Inc.
http://www.ness.com.

[16] Open GIS konsorcium.
http://www.opengis.org.

[17] GDAL Geospatial Data Abstraction Library.
http://www.remotesensing.org/gdal.

[18] OGR Simple Features Library.
http://gdal.velocet.ca/projects/opengis.

[19] PROJ.4 Cartographic Projections Library.
http://www.remotesensing.org/proj.

[20] GNU Free Documentation License.
http://www.gnu.org/licenses/fdl.html.

[21] GNU General Public License.
http://www.gnu.org/licenses/gpl.html.

[22] GRASS Newsletter.
http://grass.itc.it/newsletter/.

[23] Dálkový přístup k údajům katastru nemovitostí ČR.
http://katastr.cuzk.cz.

[24] Free software.
http://www.gnu.org/philosophy/free-software-for-freedom.html.

[25] Open Source.
http://www.opensource.org.

[26] Společnost BEA Systems s.r.o.
http://cz.bea.com.

[27] Společnost Bentley Systems s.r.o.
http://www.bentley.cz.

[28] Společnost HP s.r.o.
http://www.hp.cz.

[29] Společnost Oracle Czech s.r.o.
http://www.oracle.cz.

[30] The Generic Mapping Tools.
http://gmt.soest.hawaii.edu.

[31] Struktura výměnného formátu informačního systému katastru nemovitostí České republiky ze dne 26.11.2004 v plném znění.
http://www.cuzk.cz/GenerujSoubor.ashx?NAZEV=20-STR_VF_PLNE_ZNENI_PDF.

[32] GRASS GIS Wiki.
http://grass.gdf-hannover.de/twiki/bin/view/GRASS/WebHome.

[33] Časopis GeoInformace.
http://www.geoinformace.cz.

[34] Český úřad zeměměřický a katastrální.
http://www.cuzk.cz.

Příloha A

Instalace modulu a dodatečné informace

A.1 Proměnné prostředí

GRASS používá celou řadu proměnných prostředí (viz `g.manual variables`). Jejich znalost není pro uživatele bezpodmínečně nutná, ale rozhodně není k zahození (důkazem může být např. proměnná „OVERWRITE“).

V následující tabulce (tab. A.1) je uveden reprezentativní výběr proměnných, které byly zmíněny v textu této práce.

Tabulka A.1: Vybrané proměnné prostředí GRASSu

`$GISBASE`	...	adresář, kde je GRASS nainstalován, např. `/usr/local/grass-6.1.cvs/`
`$GISDBASE`	...	cesta k databance GRASSu např. `/home/martin/grassdata`
`$LOCATION_NAME`	...	název aktuální *lokace*
`$MAPSET`	...	název aktuálního *mapsetu* v rámci *lokace*

A.2 Instalace GRASSu a modulu `v.in.vfk`

Na oficiálních stránkách GRASSu [10] jsou v sekci „Download“ dostupné binární balíčky pro řadu distribucí GNU/Linuxu, Mac OSX či MS Windows/Cygwin.

Jak již bylo dříve zmíněno GRASS je zástupcem *svobodného softwaru*. Máte tak přirozeně možnost stáhnout jeho kompletní zdrojový kód (garantováno licencí GNU GPL) a zkompilovat jej vlastními silami. Tato cesta je přirozeně výhodnější – GRASS je ušit „na míru“ vašeho hardwaru a navíc máte šanci ovlivnit, které moduly bude a nebude instalace obsahovat. Lze tak postavit systém úzce specializovaný pouze na vybranou činnost. Samotná kompilace je relativně jedno-

duchá. Je však nutno nainstalovat všechny povinné (či volitelné) závislosti (např. knihovny), které některé moduly GRASSu pro svůj běh niterně vyžadují.

Nyní tedy obrátíme svoji pozornost na samotnou instalaci „ostře sledovaného“ modulu `v.in.vfk`. Nutno poznamenat, že další vývoj tohoto programu přímo závisí na případné odezvě českých uživatelů GISu GRASS. Pokud tedy narazíte na jakýkoliv problém, bez okolků kontaktuje autora[1].

Aktuální informace o vývoji tohoto programu včetně zdrojových kódů jsou dostupné na stránce `http://gama.fsv.cvut.cz/~landa/grass/vfk/`. Program je v tuto chvíli distribuován pouze ve formě zdrojového kódu a není tak součástí oficiální distribuce GRASSu.

Pro zkompilování modulu `v.in.vfk` je potřeba kompletní zdrojový kód GRASSu 6.x. Ten lze stáhnout z oficiálních stránek GRASSu, kde jsou dostupné vedle zdrojových kódů jednotlivých verzí (např. GRASS 6.0.0) také týdenní CVS[2] snapshoty. Pokud chcete získat opravdu zcela aktuální podobu zdrojového kódu, využijte služeb systému CVS. Bližší informace naleznete na GRASS GIS Homepage [10].

Předpokládejme tedy, že máme k dispozici zdrojový kód GRASSu a modulu `v.in.vfk`. Nejprve dekomprimujeme soubor obsahující zdrojový kód modulu a vzniklý adresář zkopírujeme do adresářového stromu zdrojového kódu GRASSu. Např.:

```
$ tar xvzf v.in.vfk.tar.gz
```

```
$ cp -r v.in.vfk /usr/src/grass6/vector/
```

V následujícím textu rozlišíme dva případy – konfiguraci, kompilaci (a instalaci) celého systému a pouhou kompilaci a začlenění modulu `v.in.vfk` do již existující instalace GRASSu.

A.2.1 Instalace celého systému

Přeneseme se tedy do adresáře obsahující zdrojový kód GRASSu

```
$ cd /usr/src/grass6
```

a provedeme konfiguraci systému.

```
$ ./configure
```

Dostupný seznam parametrů získáte pomocí parametru `--help`. Skript `configure` v podstatě ověří, zda jsou v operačním systému přítomny všechny komponenty nutné pro kompilaci a běh aplikace. Množinu těchto závislostí (volitelných) lze ovlivnit právě dostupnými parametry. Pokud ve vašem operačním systému chybí program či knihovna nutná pro běh GRASSu (ať už povinná či volitelná), není nic snadnějšího než ji do systému doplnit. To není ve světě svobodného softwaru sebemenší problém. GRASS není a ve své podstatě nikdy (doufejme) nebude závislý na proprietárních komponentách.

[1] E-mail: `landa@gama.fsv.cvut.cz`.

[2] Systém pro správu verzí.

Konfiguraci tak máme za sebou a můžeme přistoupit ke kompilaci zdrojového kódu.

```
$ make
```

Pokud je přítomný i zdrojový kód modulu `v.in.vfk` (adresář `vector/v.in.vfk/`), bude zkompilován taktéž. Po úspěšné kompilaci nic nebrání GRASS nainstalovat

```
$ make install
```

GRASS by tak měl obsahovat i modul `v.in.vfk`.

A.2.2 Začlenění modulu do existující instalace GRASSu

Instalace samotného modulu je primitivní, stačí se přepnout do adresáře obsahující jeho zdrojový kód, např.

```
$ cd /usr/src/grass6/vector/v.in.vfk
```

a provést kompilaci (současně i instalaci; předpokládáme existující konfiguraci – `configure`)

```
$ INST_NOW=y make
```

Po spuštění GRASSu by tak měl být přístupný i příkaz `v.in.vfk`.

A.2.3 Lokalizace modulu do češtiny

Nativním jazykem výše zmiňovaného modulu je angličtina. A to přesto, že je v podstatě určen pouze pro české uživatele. Hlavním důvodem byla možnost pozdějšího případného začlenění modulu do oficiální distribuce GRASSu.

V této souvislosti tedy vznikl soubor (`v.in.vfk-grassmods_cs.po`) obsahující českou lokalizaci modulu. Tento soubor je umístěn přímo v adresáři se zdrojovým kódem modulu.

Přepneme se do adresáře obsahující kód GRASSu

```
$ cd /usr/src/grass6/locale
```

a připojíme obsah souboru s lokalizací modulu `v.in.vfk` na konec souboru s lokalizací GRASSu

```
$ cat ../vector/v.in.vfk/v.in.vfk-grassmods_cs.po >> po/grassmods_cs.po
```

a vytvoříme soubory s lokalizací (pokud pracujeme již s existující instalací GRASSu, tak přidáme `INST_NOW=y`).

```
$ make mo
```

nakonec zbývá před startem GRASSu nastavit proměnné prostředí.

```
$ export LANG=cs_CZ
```

```
$ export LANGUAGE=cs_CZ
```

Krátká ukázka popisu modulu v češtině:

```
Popis:
 Import dat ve výměnném formátu ISKN (VFK) do GRASSu.

Použití:
 v.in.vfk [-ernduf] [input=name] output=name [type=name[,name,...]]

Přepínače:
  -e   Rozšíření location na základě vstupních dat
  -r   Odstranit všechny nadbytečné tabulky
  -n   Nulová kategorie (-1)
  -d   Datový typ timestamp namísto varchar
  -u   Vytvořit unikátní klíče
  -f   Vytvořit primární a cizí klíče

Parametry:
   input   Soubor VFK určený k importu, pokud není zadán, čte se
           standardní vstup
  output   Název výstupní vektorové mapy
    type   Typ
           volby: point,line,boundary,centroid,area
           výchozí: point,line,boundary,centroid,area
```

A.3 Manuálová stránka modulu `v.in.vfk`

Modul `v.in.vfk` má přirozeně i svoji manuálovou stránku (viz `g.manual v.in.vfk`). Pro zachování kontinuity je v angličtině, lokalizace manuálových stránek jednotlivých modulů totiž nebyla vůbec započata. Nicméně se připravuje český manuál k tomuto modulu, který bude vystaven na informačních stránkách [14].

V případě doplnění modulu `v.in.vfk` do již existující instalace je potřeba tuto manuálovou stránku vytvořit manuálně.

```
$ cd /usr/src/grass6/tools

$ ./build_html_index.sh

$ cp ../dist.i686-pc-linux-gnu/docs/html/v.in.vfk.html $GISBASE/docs/html/
```

A.4 Modifikace vybraných modulů GRASSu v souvislosti s tuto diplomovou prací

Při tvorbě modulu `v.in.vfk` a podpůrných skriptů určených pro snazší práci s importovanou digitální katastrální mapou byly provedeny jisté modifikace zdrojového kódu GRASSu.

Aktuální stav dané věci je uveden na informační stránce k modulu `v.in.vfk` [14]. Nutno podotknout, že zcela zásadní je úprava modulu `v.extract`, ostatní jsou spíše „kosmetické“.

A.4.1 Modul `d.vect`

Během vizualizace mapy v GRASS monitoru se objevil následující problém: při zobrazení hodnot kategorie či atributu prvků (parametr `display=cat|attr`) nebyla brána v potaz uvedená podmínka „where“ (parametr `where`) nebo zadaná hodnota kategorie (`cats`). Podobně se choval i přepínač `-a`, který má za následek vyplnění ploch zadanou barvou (sloupec v atributové tabulce `grassrgb`).

Příčinou tohoto chování modulu je fakt, že je v rámci jedné vrstvy (tj. atributové tabulky) přiřazeno danému vektorovému prvku více kategorií. Tak například centroidům ploch (parcel) je v rámci tabulky obrazy_parcel přiřazen záznam odpovídající definičnímu bodu pozemkové/stavební parcely a označení druhu pozemku. Např. centroid může mít v jedné vrstvě (např. 49) současně kategorii „70“ a „608“. Výtah z příslušné databázové tabulky:

```
  cat | typppd_kod | text
 -----+------------+-------
   70 |        304 |
  608 |         18 | 218/1
```

Mapa vizualizovaná příkazem

```
GRASS:~ > d.vect kml_hp layer=49 where=typppd_kod=18 llayer=49 display=cat \
type=centroid
```

by měla logicky obsahovat pouze kategorie pozemkových parcel (`typppd_kod` = 18). Namísto toho je však zobrazena i kategorie druhu pozemku.

Z výše uvedeného tedy vyplývá, že modul `d.vect` respektuje podmínku „where“ pouze v souvislosti se zobrazením geometrie (`shape`), a nikoliv v souvislosti s hodnotami kategorií (`cat`), atributů (`attr`) a barevnou výplní ploch (přepínač `-a`).

Modifikované soubory:

- `display/d.vect/area.c`
- `display/d.vect/attr.c`
- `display/d.vect/label.c`

A.4.2 Modul `d.what.vect`

Tento modul, jak již jeho název napovídá, umožňuje interaktivní dotazování na vektorovou mapu. V této souvislosti byl přidán nový parametr `layer`, který omezuje výsledek dotazu pouze na specifikovanou vrstvu či vrstvy. Je tak možné zobrazit pouze požadovanou informaci.

Modifikované soubory:

- `d.what.vect/main.c`
- `d.what.vect/what.c`
- `d.what.vect/what.h`

A.4.3 Modul `v.category`

Modul `v.category` je podpůrný program, který umožňuje zpracování kategorií. Lze tak jednotlivé kategorie přidávat, odstraňovat či měnit číslo vrstvy. Kromě modifikace disponuje taktéž volbou pro výpis jednotlivých kategorií a sumarizačních informací do okna příkazového interpretu. Bližší informace získáte na manuálové stránce modulu, kterou jednoduše zobrazíte příkazem `g.manual -m v.category`.

Modifikace tohoto modulu se dotýká pouze volby pro sumarizační výpis (volba `report`). Původní verze zobrazovala pouze číslo vrstvy, např.:

```
LAYER 1:
type       count         min         max
point       6114           1        6114
line           0           0           0
boundary       0           0           0
centroid       0           0           0
area           0           0           0
all         6114           1        6114
```

V případě modifikované verze je výpis doplněn o název připojené databázové tabulky (pokud taková existuje). Následující příklad demonstruje obě dvě varianty: K vrstvě číslo 1 je připojena tabulka kml_souradnice_obrazu, naopak k vrstvě číslo 2 není připojena tabulka žádná.

```
LAYER/TABLE 1/kml_souradnice_obrazu:
type       count         min         max
point       6114           1        6114
line           0           0           0
boundary       0           0           0
centroid       0           0           0
area           0           0           0
all         6114           1        6114
LAYER 2:
type       count         min         max
point       6114           2           8
line           0           0           0
boundary       0           0           0
centroid       0           0           0
area           0           0           0
all         6114           2           8
```

Modifikovaný soubor:

`vector/v.category/main.c`

Status:

začleněno do vývojové větve 6.1

A.4.4 Modul `v.extract`

Nejproblémovější zásah do GRASSu a to hlavně v „koncepční“ rovině. Při rozkladu importované katastrální mapy do jednotlivých tematických map (viz kapitola 6.3) je potřeba vybrat z mapy množinu vrstev (může jich být několik desítek) a vytvořit tak novou vektorovou mapu. Silový přístup, extrahovat jednu vrstvu po druhé, tyto polotovary spojit do jedné mapy a odstranit velké množství duplicitních prvků je jemně řečeno neefektivní a velmi pomalé. Modifikovaná verze `v.extract` tak na rozdíl od originálu podporuje výběr více vrstev. Není však z filozofického hlediska vyřešen vztah k parametrům `where`, `list`, či `type`.

Nicméně platí, že skript `v.vfk.extract` pracuje korektně pouze ve spolupráci s modifikovaným modulem `v.extract`.

Modifikované soubory:

`vector/v.extract/main.c`

`vector/v.extract/extract.c`

A.4.5 Vektorová knihovna – funkce `Vect_set_varray_from_db ()`

Během tvorby mapového výstupu s využitím modulu `ps.map` se vyskytl problém, který byl řešen modifikací knihovní funkce `Vect_set_varray_from_db ()`. Tato funkce nastavuje hodnotu pole „varray“, pokud kategorie objektu daného prvku je kategorií vybranou na základě databázového dotazu.

V nemodifikované verzi funkce je vybrána první kategorie objektu v rámci vrstvy bez kontroly, zda daná kategorie odpovídá výsledku dotazu. Pokud jsou například přiřazeny v rámci jedné vrstvy (tj. atributové tabulky) danému vektorovému elementu dvě kategorie, bude vždy (bez zohlednění výběru) vrácena první hodnota kategorie. Tento problém řeší právě daný patch.

Modifikovaný soubor:

`lib/vector/Vlib/array.c`

Status:

začleněno do hlavní (pro připravovanou verzi 6.0.1) a vývojové větve (6.1)

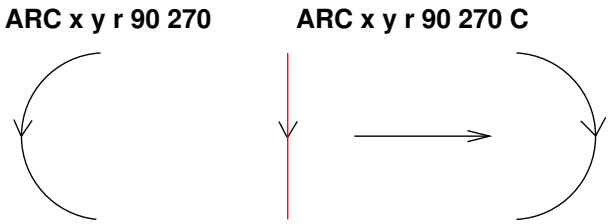
A.4.6 Symboly – klíčové slovo `ARC`

V souvislosti s tvorbou mapových značek pro katastrální mapu byla modifikována knihovní funkce `stroke_chain ()`. Část kruhu (`ARC x y r a1 a2 [C]`[3]) bylo možno vykreslit pouze v jednom směru. Celou situaci postihuje obrázek A.1.

Modifikovaný soubor:

`lib/symbol/stroke.c`

[3]Střed kruhu x, y, poloměr r, počáteční a koncový úhel výseče $a1, a2$, ve směru hodinových ručiček (C).



Obrázek A.1: Grafický symbol – oprava chyby u klíčového slova `ARC`

A.5 Typ prvku prostorových dat

Poznámka: Přiřazení typu vektorového elementu k danému typu prvku prostorových dat není v žádném případě úplné. Odpovídá možnostem testovaných dat během vývoje modulu. Proto je velmi pravděpodobné, že budou časem provedeny v tomto přiřazení jisté změny.

kód	*význam*	*vektorový element*
21900	Hranice parcely standardní	hraniční linie
21920	Hranice parcely pohyblivá, nestálá	hraniční linie
21910	Hranice parcely shora neviditelná	hraniční linie
21950	Hranice parcely podzemní	hraniční linie
21930	Hranice parcely převzatá	hraniční linie
21940	Hranice parcely sporná	hraniční linie
18	Číslo (def.bod) pozemkové parcely	centroid
28	Číslo (def.bod) stavební parcely	centroid
1018	Popisné parcelní číslo	
1032	Čára pro umístění šipky	
1033	Šipka k parcelnímu číslu	
301	Orná půda	centroid
302	Chmelnice	centroid
303	Vinice	centroid
304	Zahrada	centroid
305	Ovocný sad	centroid
306	Louka	centroid
307	Pastvina	centroid
308	Lesní půda bez rozlišení porostu	centroid
314	Park, okrasná zahrada	centroid
315	Hřbitov	centroid
316	Neplodná půda	centroid
701	Povrchová těžba nerostů a surovin	centroid
703	Ložisko slatin a rašelin	centroid
802	Vodní tok širší než 2m	centroid
803	Vodní nádrž, rybník	centroid
804	Močál, bažina	centroid

318	Nemovitá kulturní památka	centroid
21800	Hranice – vnitřní kresba	hraniční linie
21820	Hranice – vnitřní kresba pohyblivá, nestálá	hraniční linie
21810	Hranice – vnitřní kresba shora neviditelná	hraniční linie
21850	Hranice – vnitřní kresba podzemní	hraniční linie
21830	Hranice – vnitřní kresba převzatá	hraniční linie
21840	Hranice – vnitřní kresba sporná	hraniční linie
50100	Osa železniční koleje norm.rozchodu	linie
52100	Visutá lanová dráha	linie
52200	Pozemní lanová dráha	linie
60500	Osa nadzemního vedení	linie
60502	Venkovní silové vedení bez rozlišení	linie
411	Předmět malého rozsahu určený středem	centroid
601	Kovový, betonový stožár	bod
602	Příhradový stožár	bod
604	Stožár vysílací, retranslační stanice	bod
811	Veřejná studna	bod
420	Most, propustek	
402	Budova zděná, betonová, kovová	centroid
403	Budova dřevěná	centroid
409	Kostel, kaple nebo modlitebna	centroid
410	Synagoga	centroid
412	Předmět malého rozsahu bez rozlišení	centroid
404	Budova zděná, betonová, kovová evidovaná v SPI	centroid
405	Budova dřevěná evidovaná v SPI	centroid
21700	Obvod budovy evidované v SPI	hraniční linie
101	Bod polohového bodového pole	bod
102	Bod PBP – pouze podzemní značka	bod
103	Bod jednotné nivelační sítě bod	bod
104	Stabilizovaný bod technické nivelace	bod
105	Hraniční znak	bod
1016	Číslo bodu bodového pole	
1027	Číslo hraničního znaku na státní hranici	
1001	Název města	
1002	Název městského obvodu nebo části	
1004	Název obce	
1005	Název části obce	
1007	Název náměstí, parku	
1008	Název ulice	
1009	Název pozemkové tratě	
1010	Název podružné pozemkové tratě	
1012	Název sousedního státu	

1013	Název řeky sloužící k vodní dopravě	
1014	Název řeky, jezera, vel.rybníku, přehrady	
1015	Název potoka, rybníku	
408	Čára jako výplň schodiště	
1060	Symbol vodního toku užšího než 2m	
20600	Hranice kat.území (bez značky)	hraniční linie
20602	Hranice kat.území (značka uprostřed)	hraniční linie
20603	Hranice kat.území (značka na začátku)	hraniční linie
20604	Hranice kat.území (značka na konci)	hraniční linie
20605	Hranice kat.území (na obou koncích)	hraniční linie
20610	Hranice kat.území shora neviditelná (bez značky)	hraniční linie
20612	Hranice kat.území shora neviditelná (značka uprostřed)	hraniční linie
20613	Hranice kat.území shora neviditelná (značka na začátku)	hraniční linie
20614	Hranice kat.území shora neviditelná (značka na konci)	hraniční linie
20615	Hranice kat.území shora neviditelná (na obou koncích)	hraniční linie
20620	Hranice kat.území pohyblivá, nestálá (bez značky)	hraniční linie
20622	Hranice kat.území pohyblivá, nestálá (značka uprostřed)	hraniční linie
20623	Hranice kat.území pohyblivá, nestálá (značka na začátku)	hraniční linie
20624	Hranice kat.území pohyblivá, nestálá (značka na konci)	hraniční linie
20625	Hranice kat.území pohyblivá, nestálá (na obou koncích)	hraniční linie
20630	Hranice kat.území převzatá (bez značky)	hraniční linie
20632	Hranice kat.území převzatá (značka uprostřed)	hraniční linie
20633	Hranice kat.území převzatá (značka na začátku)	hraniční linie
20634	Hranice kat.území převzatá (značka na konci)	hraniční linie
20635	Hranice kat.území převzatá (na obou koncích)	hraniční linie
20640	Hranice kat.území sporná (bez značky)	hraniční linie
20642	Hranice kat.území sporná (značka uprostřed)	hraniční linie
20643	Hranice kat.území sporná (značka na začátku)	hraniční linie
20644	Hranice kat.území sporná (značka na konci)	hraniční linie
20645	Hranice kat.území sporná (na obou koncích)	hraniční linie
20650	Hranice kat.území podzemní (bez značky)	hraniční linie
20652	Hranice kat.území podzemní (značka uprostřed)	hraniční linie
20653	Hranice kat.území podzemní (značka na začátku)	hraniční linie
20654	Hranice kat.území podzemní (značka na konci)	hraniční linie
20655	Hranice kat.území podzemní (na obou koncích)	hraniční linie
20500	Hranice obce (bez značky)	hraniční linie
20502	Hranice obce (značka uprostřed)	hraniční linie
20503	Hranice obce (značka na začátku)	hraniční linie
20504	Hranice obce (značka na konci)	hraniční linie
20505	Hranice obce (na obou koncích)	hraniční linie
20510	Hranice obce shora neviditelná (bez značky)	hraniční linie
20512	Hranice obce shora neviditelná (značka uprostřed)	hraniční linie

20513	Hranice obce shora neviditelná (značka na začátku)	hraniční linie
20514	Hranice obce shora neviditelná (značka na konci)	hraniční linie
20515	Hranice obce shora neviditelná (na obou koncích)	hraniční linie
20520	Hranice obce pohyblivá, nestálá (bez značky)	hraniční linie
20522	Hranice obce pohyblivá, nestálá (značka uprostřed)	hraniční linie
20523	Hranice obce pohyblivá, nestálá (značka na začátku)	hraniční linie
20524	Hranice obce pohyblivá, nestálá (značka na konci)	hraniční linie
20525	Hranice obce pohyblivá, nestálá (na obou koncích)	hraniční linie
20530	Hranice obce převzatá (bez značky)	hraniční linie
20532	Hranice obce převzatá (značka uprostřed)	hraniční linie
20533	Hranice obce převzatá (značka na začátku)	hraniční linie
20534	Hranice obce převzatá (značka na konci)	hraniční linie
20535	Hranice obce převzatá (na obou koncích)	hraniční linie
20540	Hranice obce sporná (bez značky)	hraniční linie
20542	Hranice obce sporná (značka uprostřed)	hraniční linie
20543	Hranice obce sporná (značka na začátku)	hraniční linie
20544	Hranice obce sporná (značka na konci)	hraniční linie
20545	Hranice obce sporná (na obou koncích)	hraniční linie
20550	Hranice obce podzemní (bez značky)	hraniční linie
20552	Hranice obce podzemní (značka uprostřed)	hraniční linie
20553	Hranice obce podzemní (značka na začátku)	hraniční linie
20554	Hranice obce podzemní (značka na konci)	hraniční linie
20555	Hranice obce podzemní (na obou koncích)	hraniční linie
20400	Hranice okresu (bez značky)	hraniční linie
20402	Hranice okresu (značka uprostřed)	hraniční linie
20403	Hranice okresu (značka na začátku)	hraniční linie
20404	Hranice okresu (značka na konci)	hraniční linie
20405	Hranice okresu (na obou koncích)	hraniční linie
20410	Hranice okresu shora neviditelná (bez značky)	hraniční linie
20412	Hranice okresu shora neviditelná (značka uprostřed)	hraniční linie
20413	Hranice okresu shora neviditelná (značka na začátku)	hraniční linie
20414	Hranice okresu shora neviditelná (značka na konci)	hraniční linie
20415	Hranice okresu shora neviditelná (na obou koncích)	hraniční linie
20420	Hranice okresu pohyblivá, nestálá (bez značky)	hraniční linie
20422	Hranice okresu pohyblivá, nestálá (značka uprostřed)	hraniční linie
20423	Hranice okresu pohyblivá, nestálá (značka na začátku)	hraniční linie
20424	Hranice okresu pohyblivá, nestálá (značka na konci)	hraniční linie
20425	Hranice okresu pohyblivá, nestálá (na obou koncích)	hraniční linie
20430	Hranice okresu převzatá (bez značky)	hraniční linie
20432	Hranice okresu převzatá (značka uprostřed)	hraniční linie
20433	Hranice okresu převzatá (značka na začátku)	hraniční linie
20434	Hranice okresu převzatá (značka na konci)	hraniční linie

20435	Hranice okresu převzatá (na obou koncích)	hraniční linie
20440	Hranice okresu sporná (bez značky)	hraniční linie
20442	Hranice okresu sporná (značka uprostřed)	hraniční linie
20443	Hranice okresu sporná (značka na začátku)	hraniční linie
20444	Hranice okresu sporná (značka na konci)	hraniční linie
20445	Hranice okresu sporná (na obou koncích)	hraniční linie
20450	Hranice okresu podzemní (bez značky)	hraniční linie
20452	Hranice okresu podzemní (značka uprostřed)	hraniční linie
20453	Hranice okresu podzemní (značka na začátku)	hraniční linie
20454	Hranice okresu podzemní (značka na konci)	hraniční linie
20455	Hranice okresu podzemní (na obou koncích)	hraniční linie
20300	Hranice kraje (bez značky)	hraniční linie
20302	Hranice kraje (značka uprostřed)	hraniční linie
20303	Hranice kraje (značka na začátku)	hraniční linie
20304	Hranice kraje (značka na konci)	hraniční linie
20305	Hranice kraje (na obou koncích)	hraniční linie
20310	Hranice kraje shora neviditelná (bez značky)	hraniční linie
20312	Hranice kraje shora neviditelná (značka uprostřed)	hraniční linie
20313	Hranice kraje shora neviditelná (značka na začátku)	hraniční linie
20314	Hranice kraje shora neviditelná (značka na konci)	hraniční linie
20315	Hranice kraje shora neviditelná (na obou koncích)	hraniční linie
20320	Hranice kraje pohyblivá, nestálá (bez značky)	hraniční linie
20322	Hranice kraje pohyblivá, nestálá (značka uprostřed)	hraniční linie
20323	Hranice kraje pohyblivá, nestálá (značka na začátku)	hraniční linie
20324	Hranice kraje pohyblivá, nestálá (značka na konci)	hraniční linie
20325	Hranice kraje pohyblivá, nestálá (na obou koncích)	hraniční linie
20330	Hranice kraje převzatá (bez značky)	hraniční linie
20332	Hranice kraje převzatá (značka uprostřed)	hraniční linie
20333	Hranice kraje převzatá (značka na začátku)	hraniční linie
20334	Hranice kraje převzatá (značka na konci)	hraniční linie
20335	Hranice kraje převzatá (na obou koncích)	hraniční linie
20340	Hranice kraje sporná (bez značky)	hraniční linie
20342	Hranice kraje sporná (značka uprostřed)	hraniční linie
20343	Hranice kraje sporná (značka na začátku)	hraniční linie
20344	Hranice kraje sporná (značka na konci)	hraniční linie
20345	Hranice kraje sporná (na obou koncích)	hraniční linie
20350	Hranice kraje podzemní (bez značky)	hraniční linie
20352	Hranice kraje podzemní (značka uprostřed)	hraniční linie
20353	Hranice kraje podzemní (značka na začátku)	hraniční linie
20354	Hranice kraje podzemní (značka na konci)	hraniční linie
20355	Hranice kraje podzemní (na obou koncích)	hraniční linie
20100	Hranice státní (bez značky)	hraniční linie

20102	Hranice státní (značka uprostřed)	hraniční linie
20103	Hranice státní (značka na začátku)	hraniční linie
20104	Hranice státní (značka na konci)	hraniční linie
20105	Hranice státní (na obou koncích)	hraniční linie
20110	Hranice státní shora neviditelná (bez značky)	hraniční linie
20112	Hranice státní shora neviditelná (značka uprostřed)	hraniční linie
20113	Hranice státní shora neviditelná (značka na začátku)	hraniční linie
20114	Hranice státní shora neviditelná (značka na konci)	hraniční linie
20115	Hranice státní shora neviditelná (na obou koncích)	hraniční linie
20120	Hranice státní pohyblivá, nestálá (bez značky)	hraniční linie
20122	Hranice státní pohyblivá, nestálá (značka uprostřed)	hraniční linie
20123	Hranice státní pohyblivá, nestálá (značka na začátku)	hraniční linie
20124	Hranice státní pohyblivá, nestálá (značka na konci)	hraniční linie
20125	Hranice státní pohyblivá, nestálá (na obou koncích)	hraniční linie
20130	Hranice státní převzatá (bez značky)	hraniční linie
20132	Hranice státní převzatá (značka uprostřed)	hraniční linie
20133	Hranice státní převzatá (značka na začátku)	hraniční linie
20134	Hranice státní převzatá (značka na konci)	hraniční linie
20135	Hranice státní převzatá (na obou koncích)	hraniční linie
20140	Hranice státní sporná (bez značky)	hraniční linie
20142	Hranice státní sporná (značka uprostřed)	hraniční linie
20143	Hranice státní sporná (značka na začátku)	hraniční linie
20144	Hranice státní sporná (značka na konci)	hraniční linie
20145	Hranice státní sporná (na obou koncích)	hraniční linie
20150	Hranice státní podzemní (bez značky)	hraniční linie
20152	Hranice státní podzemní (značka uprostřed)	hraniční linie
20153	Hranice státní podzemní (značka na začátku)	hraniční linie
20154	Hranice státní podzemní (značka na konci)	hraniční linie
20155	Hranice státní podzemní (na obou koncích)	hraniční linie
22300	Hranice chráněného území	hraniční linie
22400	Hranice ochranného pásma	hraniční linie
1040	Linie obvodů bonitovaných a nebonitovaných ploch	hraniční linie
1042	Kód BPEJ	centroid
1043	Kód nebonitované plochy	centroid
1050	Nezařazená linie po migraci	linie
1051	Nezařazený text po migraci	
1052	Nezařazený symbol po migraci	
1019	Popisné parcelní číslo (volné)	
1028	Čára pro umístění šipky (volná)	
1029	Šipka k parcelnímu číslu (volná)	
22000	Hranice - vnitřní kresba bez topologie	hraniční linie

A.6 Seznam databázových tabulek

Tabulka A.2: Seznam podkladových databázových tabulek a související číslo vrstvy

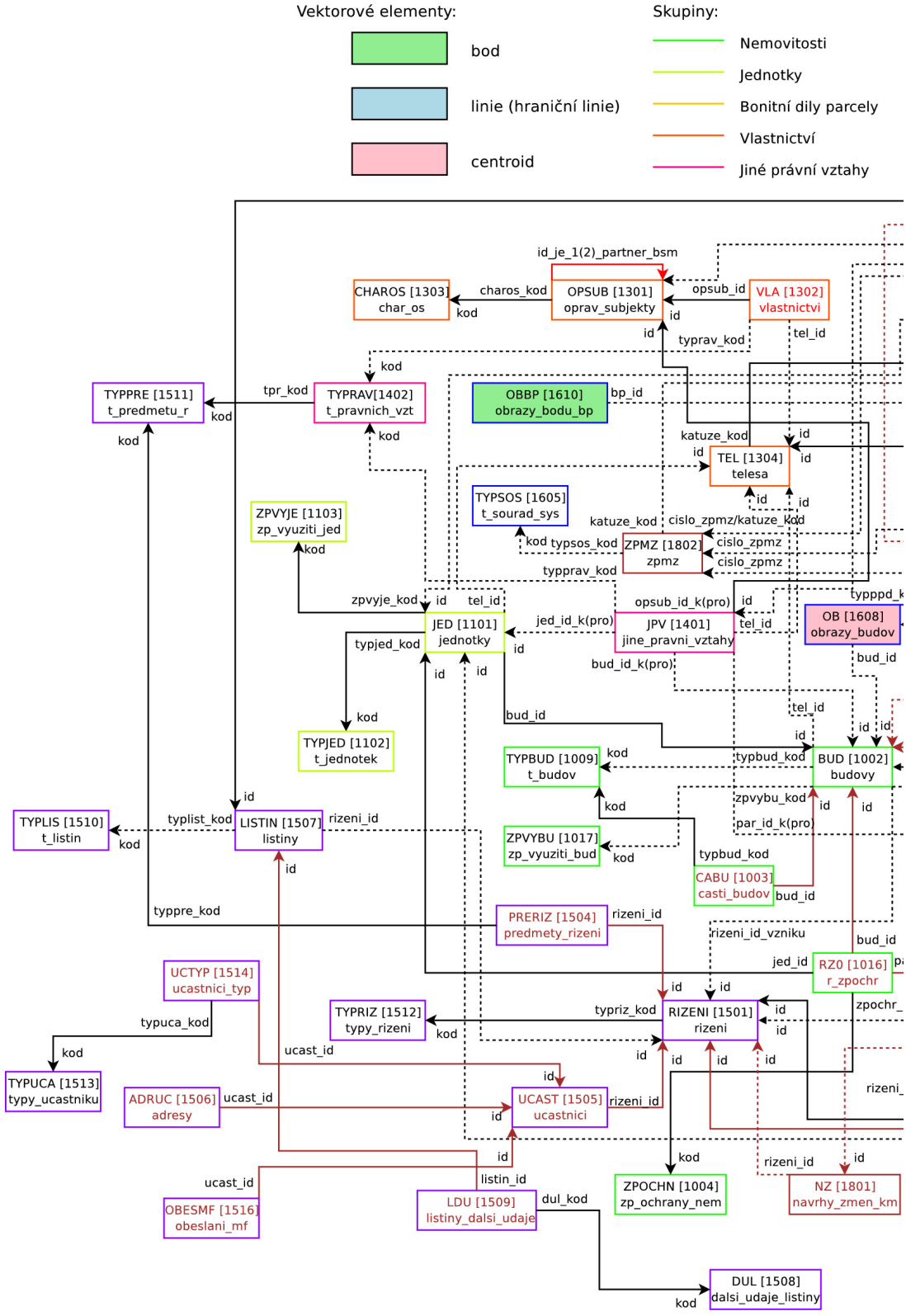
číslo vrstvy	položka	tabulka	popis
Nemovitosti NEMO			
1001	PAR	parcely	parcely
1002	BUD	budovy	budovy
1003	CABU	casti budov	části budov
1004	ZPOCHN*	zp_ochrany_nem	číselník způsobů ochrany nemovitosti
1005	DRUPOZ*	d_pozemku	číselník druhů pozemku
1006	ZPVYPO*	zp_vyuziti_poz	číselník způsobů využití pozemku
1007	ZDPAZE*	zdroje_parcel_ze	číselník zdrojů parcel ZE
1008	ZPURVY*	zp_urceni_vymery	číselník způsobů určení výměry
1009	TYPBUD*	t_budov	číselník typů budov
1010	MAPLIS*	mapove_listy	číselník mapových listů
1011	KATUZE*	katastr_uzemi	číselník katastrálních území
1012	OBCE*	obce	číselník obcí
1013	CASOBC*	casti_obci	číselník částí obce
1014	OKRESY*	okresy	číselník okresů
1015	KRAJE*	kraje	číselník krajů
1016	RZO	r_zpochr	přiřazení způsobu ochrany k nemovitostem
1017	ZPVYBU*	zp_vyuziti_bud	způsob využití budov
Jednotky JEDN			
1101	JED	jednotky	jednotky
1102	TYPJED*	t_jednotek	číselník typů jednotek
1103	ZPVYJE*	zp_vyuziti_jed	způsob využití jednotek
Bonitní díly parcely BDPA			
1201	BDP	bonit_dily_parc	bonitní díly parcel
Vlastnictví VLST			
1301	OPSUB	oprav_subjekty	oprávněné subjekty
1302	VLA	vlastnictvi	vlastnictví
1303	CHAROS*	char_os	číselník charakteristik oprávněných subjektů
1304	TEL	telesa	katastrální tělesa
Jiné právní vztahy JPVZ			
1401	JPV	jine_prav_vztahy	jiné právní vztahy
1402	TYPRAV*	t_pravnich_vzt	číselník typů právních vztahů
Řízení RIZE			
1501	RIZENI*	rizeni	řízení (vklad, záznam)
1502	RIZKU*	rizeni_ku	vazba řízení – katastrální území
1503	OBJRIZ*	objekty_rizeni	objekty řízení (parcely, budovy)
1504	PRERIZ*	predmety_rizeni	předměty řízení

Pokračování na další stránce

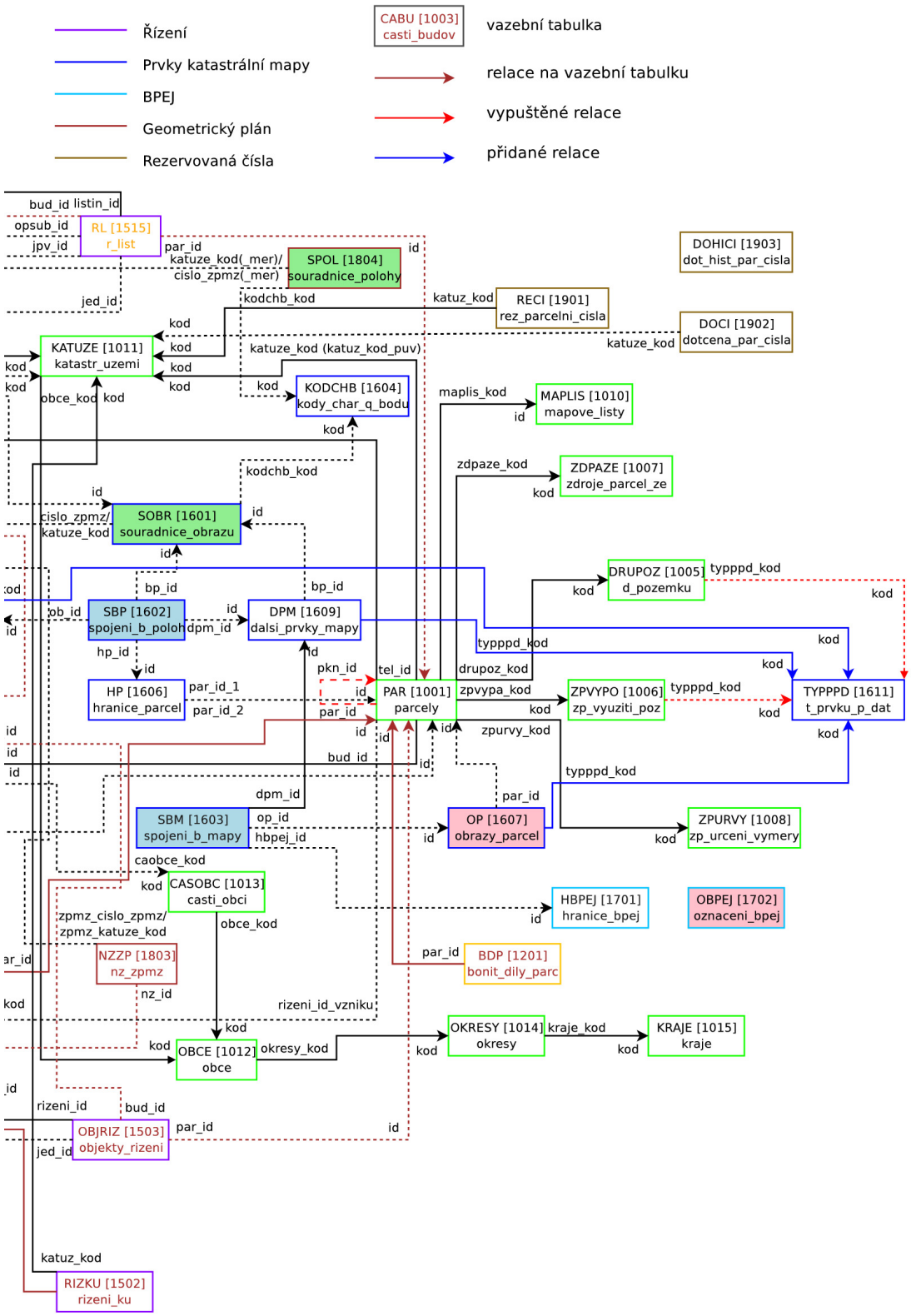
Pokračování

číslo vrstvy	položka	tabulka	popis
1504	UCAST*	ucastnici	účastníci řízení
1506	ADRUC*	adresy	adresy účastníků řízení
1507	LISTIN*	listiny	listiny
1508	DUL*	dalsi_udaje_listiny	další údaje listin
1509	LDU*	listiny_dalsi_udaje	vazba listiny – další údaje listin
1510	TYPLIS*	t_listin	číselník typů listin
1511	TYPPRE*	t_predmetu_r	číselník typů předmětu řízení
1512	TYPRIZ*	typy_rizeni	typy řízení
1513	TYPUCA*	typy_ucastniku	typy účastníků řízení
1514	UCTYP*	ucastnici_typ	vazba mezi účastníky a typy účastníků řízení
1515	RL	r_list	přiřazení listin k nemovitostem, vlastnictví a jiným právním vztahům
1516	OBESMF*	obeslani_mf	obeslání účastníků řízení
Prvky katastrální mapy PKMP			
1601	SOBR*	souradnice_obrazu	souřadnice obrazu bodů polohopisu v mapě
1602	SBP	spojeni_b_poloh	spojení bodů polohopisu – definuje polohopisné liniové prvky
1603	SBM	spojeni_b_mapy	spojení bodů mapy – definuje nepolohopisné liniové prvky
1604	KODCHB*	kody_char_q_bodu	číselník kódů charakteristiky kvality bodu
1605	TYPSOS*	t_sourad_sys	číselník typů souřadnicových systémů
1606	HP	hranice_parcel	hranice parcel
1607	OP	obrazy_parcel	obrazy parcel (parcelní číslo, značka druhu pozemku)
1608	OB	obrazy_budov	obrazy budov (obvod budovy, značka druhu budovy)
1609	DPM	dalsi_prvky_mapy	další prvky mapy
1610	OBBP	obrazy_bodu_bp	obrazy bodů BP
1611	TYPPPD*	t_prvku_p_dat	číselník typů prvku prostorových dat
BPEJ BPEJ			
1701	HBPEJ	hranice_bpej	hranice BPEJ
1702	OBPEJ	oznaceni_bpej	označení BPEJ
Geometrický plán GMPL			
1801	NZ	navrhy_zmen_km	hlavičky geometrických plánů a ostatních změn KM
1802	ZPMZ	zpmz	hlavičky ZPMZ
1803	NZZP	nz_zpmz	vazební tabulka návrhy změn KM – ZPMZ
1804	SPOL	souradnice_polohy	souřadnice polohy bodů polohopisu (měřené)
Rezervovaná čísla REZE			
1901	RECI	rez_parcelni_cisla	rezervovaná parcelní čísla
1902	DOCI	dotcena_par_cisla	dotčená parcelní čísla
1903	DOHICI	dot_hist_par_cisla	dotčená historická parcelní čísla

A.7 Grafické znázornění vazeb mezi tabulkami



Obrázek A.2: Grafické znázornění vazeb mezi tabulkami


Řízení
Prvky katastrální mapy
BPEJ
Geometrický plán
Rezervovaná čísla
CABU [1003] casti_budov
vazební tabulka
relace na vazební tabulku
vypuštěné relace
přidané relace
RL [1515] r_list
SPOL [1804] souradnice_polohy
DOHICI [1903] dot_hist_par_cisla
RECI [1901] rez_parcelni_cisla
DOCI [1902] dotcena_par_cisla
KATUZE [1011] katastr_uzemi
KODCHB [1604] kody_char_q_bodu
MAPLIS [1010] mapove_listy
ZDPAZE [1007] zdroje_parcel_ze
SOBR [1601] souradnice_obrazu
DRUPOZ [1005] d_pozemku
SBP [1602] spojeni_b_poloh
DPM [1609] dalsi_prvky_mapy
HP [1606] hranice_parcel
PAR [1001] parcely
ZPVYPO [1006] zp_vyuziti_poz
TYPPPD [1611] t_prvku_p_dat
SBM [1603] spojeni_b_mapy
OP [1607] obrazy_parcel
ZPURVY [1008] zp_urceni_vymery
CASOBC [1013] casti_obci
HBPEJ [1701] hranice_bpej
OBPEJ [1702] oznaceni_bpej
NZZP [1803] nz_zpmz
BDP [1201] bonit_dily_parc
OKRESY [1014] okresy
KRAJE [1015] kraje
OBCE [1012] obce
OBJRIZ [1503] objekty_rizeni
RIZKU [1502] rizeni_ku

A.8 Obsah přiloženého CD

dp/ – text diplomové práce

- zdrojový kód LaTeX, obrázky ve formátu FIG, EPS, PNG, DIA
- ve formátu PDF
- ve formátu PS

mapove_znacky/ – mapové značky katastrální mapy

patches/ – patche popsané v dodatku A.4

skripty/ – sada podpůrných skriptů, jedná se především o `v.vfk.extract`

test_data/ – testovací data

bylany/ – *lokace* obsahující importovanou DKM

shp/ – ukázkové mapy ve formátu ESRI ShapeFile exportované z *lokace* „bylany“

vfk/ – vstupní soubor ve formátu VFK

v.in.vfk/ – zdrojový kód modulu `v.in.vfk` včetně souboru s lokalizací

Příloha B

Problematika mapového výstupu

B.1 Testovací data

Jako primární testovací data byl použit volně dostupný dataset z webových stránek ČÚZK [34]. V sekci „Poskytování údajů“ → „Výstupy dat ISKN ve výměnných formátech“ jsou k nalezení vzorky dat VFK. Jde o soubory `Export1-4.vfk` (první až čtvrtá skupina datových bloků), `Export1-6.vfk` (první až šestá skupina), `Export1-6+mapa.vfk` (první až sedmá skupina), `Exportmapa.vfk` (sedmá skupina) a `Exportvse.vfk` (první až devátá skupina). Jelikož první dva soubory neobsahují žádnou grafickou informaci, budeme je ignorovat. Zvolíme tedy nejúplnější soubor – `Exportvse.vfk`.

Pro účel vývoje a rozsáhlejšího testování modulu `v.in.vfk` byla později poskytnuta ze strany ČÚZK další data. Tento dataset pokrývá přibližně 20 % okresu Litoměřice – tj. 34 723 parcel!

B.2 Příklady mapových výstupů

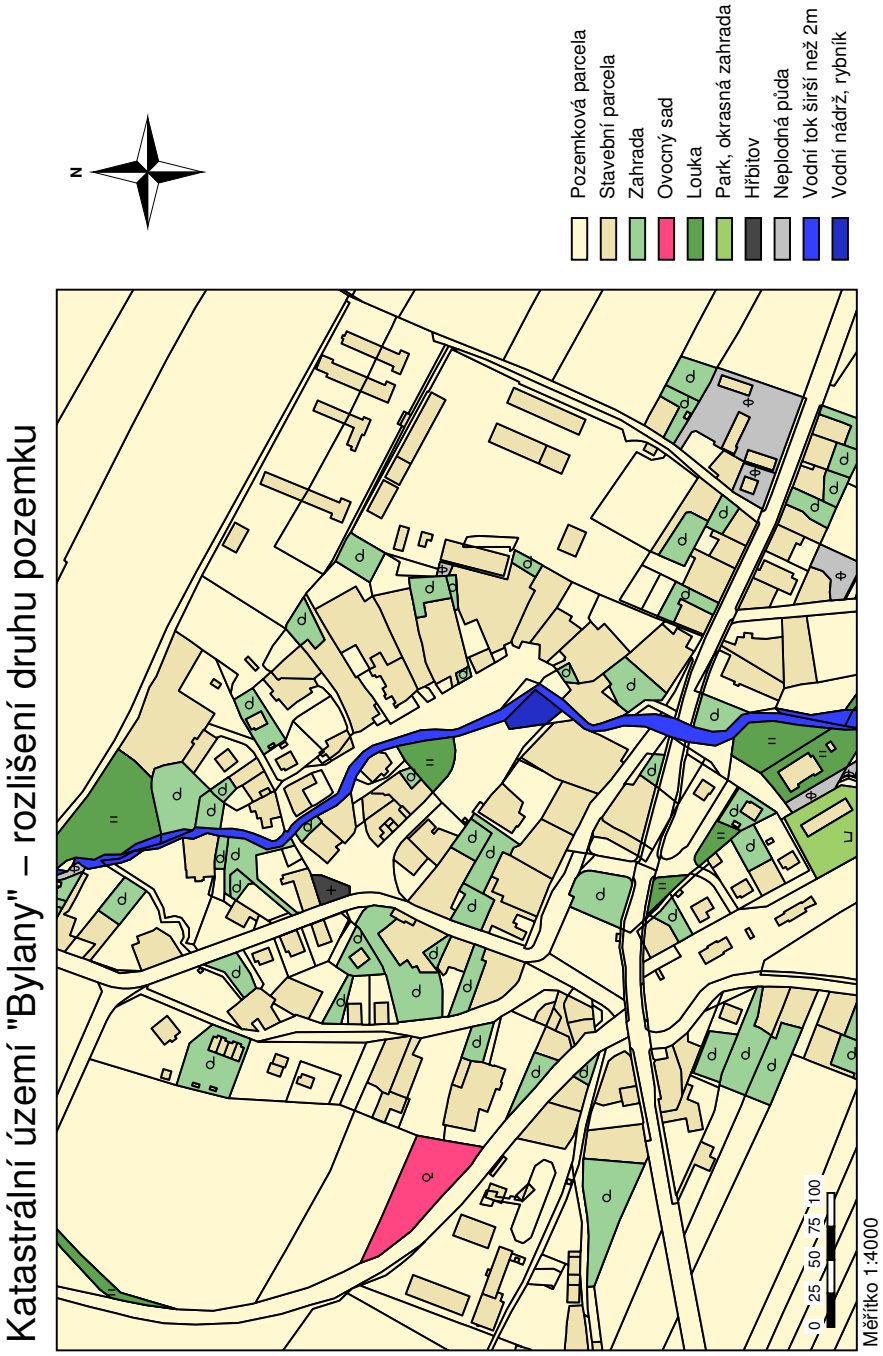
Ukázka řídícího souboru pro modul `ps.map` (výsledek je zobrazen na obrázku B.2):

```
paper a4
end

scale 1:2000

maploc 0.5 1

vlines bylany_dpmt
       color black
       width 0.4
end

vareas bylany_ob
       color black
```

```
       width 0.7
       fcolor none
end

vareas bylany_op
       color black
       width 1
       fcolor none
end

vpoints bylany_op
type centroid
layer 50
where kod=304
symbol dkm/zahrada
color black
size 5
width 1
end

vpoints bylany_op
type centroid
layer 50
where kod=305
symbol dkm/ovocny_sad
color black
size 5
width 1
end

vpoints bylany_op
type centroid
layer 50
where kod=306
symbol dkm/tt_porost
color black
size 5
width 1
end

vpoints bylany_op
type centroid
layer 50
where kod=316
symbol dkm/hrbitov
color black
size 5
width 1
```

```
end

labels op
end

text 0% 101.5% Katastrální mapa "Bylany"
   color black
   width 1
   size 15
   ref lower left
end

text 0% 0% Měřítko 1:2000
   color black
   width 1
   size 10
   ref upper left
end

eps 105% 90%
    epsfile ./kompas.eps
    scale 0.3
end
```

Obrázek B.1: Příklad mapového výstupu vytvořeného pomocí modulu `ps.map`



Obrázek B.2: Příklad mapového výstupu vytvořeného pomocí modulu `ps.map` – rozlišení druhu pozemku

B.3 Vybrané mapové značky katastrální mapy

Soubory s definicí mapových značek mohou být uloženy přímo v daném *mapsetu*

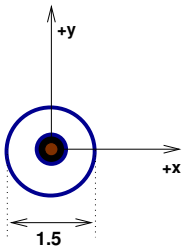
`$GISDBASE/$LOCATION_NAME/$MAPSET/symbol/dkm/`

nebo v základním adresáři instalace GRASSu

`$GISBASE/etc/symbol/dkm/`.

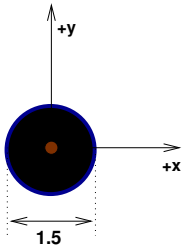
B.3.1 Body bodových polí a hraniční znaky

Bod polohového bodového pole [101] a bod PBP – pouze podzemní značka [102]/ `bod_pbp`



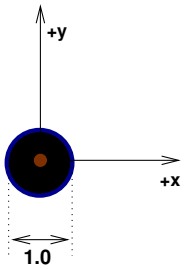

```
VERSION 1.0
BOX -1 -1 1 1
POLYGON
  RING
    FCOLOR NONE
    ARC 0 0 0.75 360 0 C
  END
END
POLYGON
  RING
    FCOLOR 0 0 0
    ARC 0 0 0.25 360 0 C
  END
END
```

Bod jednotné nivelační sítě [103] a stabilizovaný bod technické nivelace [104]/ `bod_jns`




```
VERSION 1.0
BOX -1 -1 1 1
POLYGON
  RING
    FCOLOR 0 0 0
    ARC 0 0 0.75 360 0 C
  END
END
```

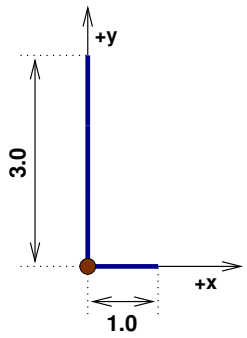
Hraniční znak [105]/`hranicni_znak`




```
VERSION 1.0
BOX -1 -1 1 1
POLYGON
  RING
    FCOLOR 0 0 0
    ARC 0 0 0.5 360 0 C
  END
END
```

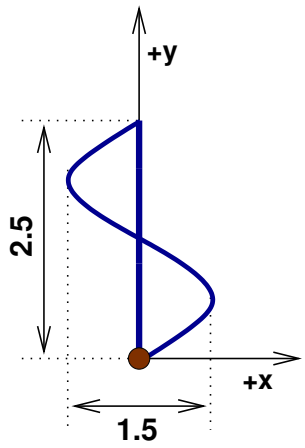
B.3.2 Druh pozemků a způsob jejich využití

Chmelnice [302]/`chmelnice`



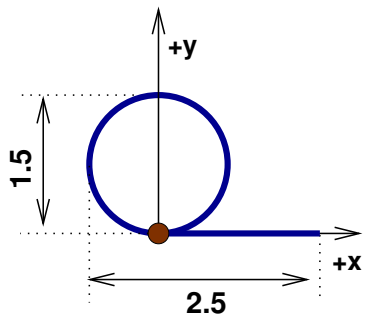

```
VERSION 1.0
BOX -1 -1 1 1
STRING
  LINE
     0 0
     1 0
  END
  LINE
     0 0
     0 3
  END
END
```

Vinice [303]/`vinice`



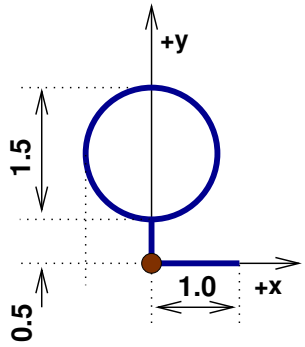

```
VERSION 1.0
BOX -1 -1 1 1
POLYGON
  RING
   FCOLOR NONE
   LINE
     0     0
     0.2   0.1
     0.4   0.3
     0.525 0.4
     0.625 0.625
     0.525 0.825
     0.4   0.925
     0.2   1.15
     0     1.25
     -0.2   1.35
     -0.4   1.55
     -0.525 1.65
     -0.625 1.825
     -0.525 2.025
     -0.4   2.125
     -0.2   2.35
     -0     2.50
   END
  END
END
STRING
  LINE
     0  0
     0  2.5
  END
END
```

Zahrada [304]/`zahrada`



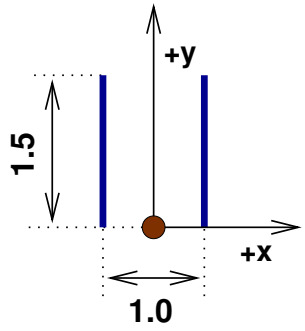

```
VERSION 1.0
BOX -1 -1 1 1
POLYGON
  RING
    FCOLOR NONE
    ARC 0 0.75 0.75 360 0 C
  END
END
STRING
  LINE
     0    0
     1.75 0
  END
END
```

Ovocný sad [305]/ovocny_sad



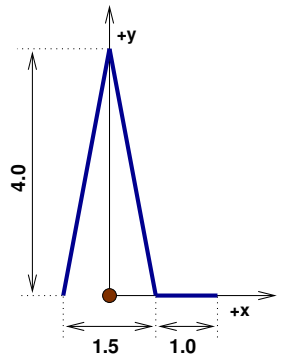

```
VERSION 1.0
BOX -1 -1 1 1
POLYGON
  RING
    FCOLOR NONE
    ARC 0 1.25 0.75 360 0 C
  END
END
STRING
  LINE
     0  0.5
     0  0
     1  0
  END
END
```

Trvalý travní porost (louka [306]/pastvina [307])/tt_porost



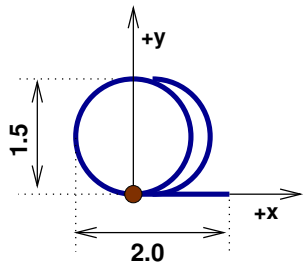

```
VERSION 1.0
BOX -1 -1 1 1
STRING
  LINE
     -0.5 0
     -0.5 1.5
  END
END
STRING
  LINE
     0.5 0
     0.5 1.5
  END
END
```

Lesní půda bez rozlišení druhu porostu [308]/lesni_puda_brdp



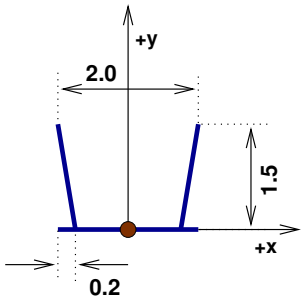

```
VERSION 1.0
BOX -1 -1 1 1
STRING
  LINE
     -0.75 0
      0    4
      0.75 0
      1.75 0
  END
END
```

Lesní půda s křovinatým porostem/lesni_puda_skp



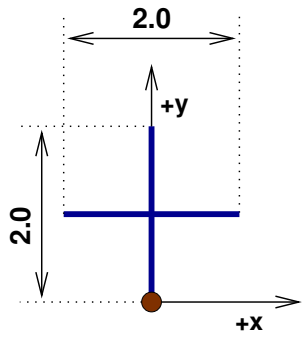

```
VERSION 1.0
BOX -1 -1 1 1
POLYGON
  RING
    FCOLOR NONE
    ARC 0 0.75 0.75 0 360
  END
END
STRING
    ARC 0.25 0.75 0.75 100 270 C
END
STRING
  LINE
     0    0
     1.15 0
  END
END
```

Park, okrasná zahrada [314]/park



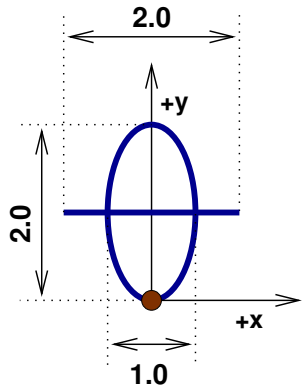

```
VERSION 1.0
BOX -1 -1 1 1
STRING
  LINE
     -1 0
     1 0
  END
END
  LINE
     -0.8 0
     -1   1.5
  END
END
STRING
  LINE
      0.8 0
      1   1.5
  END
END
```

Hřbitov [315]/hrbitov



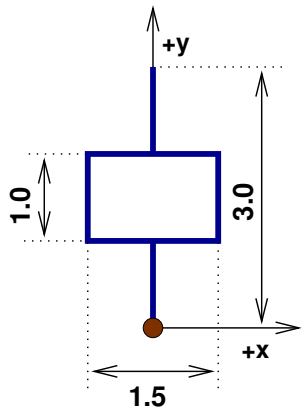

```
VERSION 1.0
BOX -1 -1 1 1
STRING
  LINE
     0 0
     0 2
  END
END
STRING
  LINE
     -1 1
      1  1
  END
END
```

Neplodná půda [316]/neplodna_puda




```
VERSION 1.0
BOX -1 -1 1 1
POLYGON
   RING
      FCOLOR NONE
      LINE
         0.0 2.0
         0.1 1.98
         0.2 1.9
         0.3 1.8
         0.4 1.6
         0.5 1.0
         0.4 0.6
         0.3 0.2
         0.2 0.1
         0.1 0.02
         0.0 0.0
         -0.1 0.02
         -0.2 0.1
         -0.3 0.2
         -0.4 0.6
         -0.5 1.0
         -0.4 1.6
         -0.3 1.8
         -0.2 1.9
         -0.1 1.98
      END
  END
END
STRING
  LINE
     -1 1
      1 1
  END
END
```

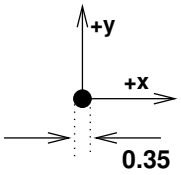
Nemovitá kulturní památka [318]/nem_kul_pam




```
VERSION 1.0
BOX -1 -1 1 1
STRING
  LINE
      0 0
      0 1
  END
END
STRING
  LINE
      1 1
      1 2
     -1 2
     -1 1
      1 1
  END
END
STRING
  LINE
      0 2
      0 3
  END
END
```

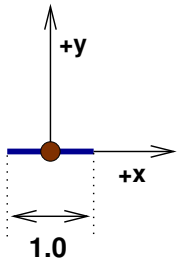
B.3.3 Stavební objekty

Budova zděná, betonová, kovová [402]/bud_zdena



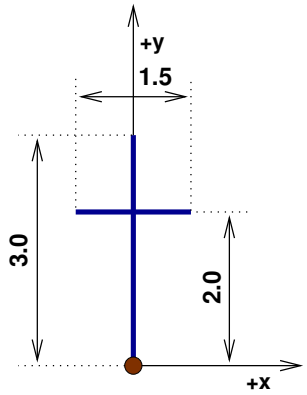

```
VERSION 1.0
BOX -1 -1 1 1
POLYGON
  RING
    FCOLOR 0 0 0
    ARC 0 0.175 0.175 360 0 C
  END
END
```

Budova dřevěná [403]/bud_drevena



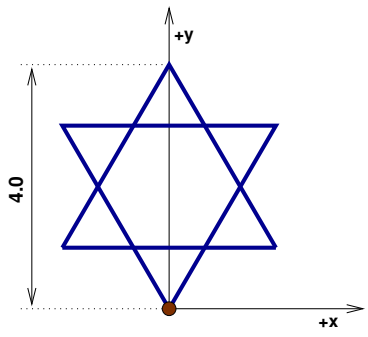

```
VERSION 1.0
BOX -1 -1 1 1
STRING
  LINE
     -0.5 0
      0.5 0
  END
END
```

Kostel, kaple nebo modlitebna [409]/kostel



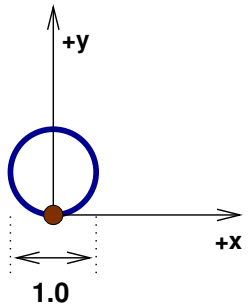

```
VERSION 1.0
BOX -1 -1 1 1
STRING
  LINE
     0 0
     0 3
  END
END
STRING
  LINE
      -0.75 2
       0.75 2
  END
END
```

Synagoga [410]/synagoga



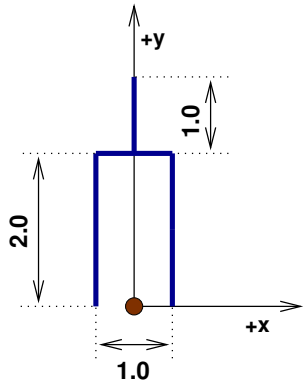

```
VERSION 1.0
BOX -1 -1 1 1
STRING
  LINE
     -1.75 1
      1.75 1
      0    4
     -1.75 1
  END
END
STRING
  LINE
     0    0
     1.75 3
    -1.75 3
     0    0
  END
END
```

Předmět malého rozsahu určený středem [411]/predmet_mrus




```
VERSION 1.0
BOX -1 -1 1 1
POLYGON
  RING
    FCOLOR NONE
    ARC 0 0.5 0.5 360 0 C
  END
END
```

Předmět malého rozsahu bez rozlišení [412]/predmet_mrbr




```
VERSION 1.0
BOX -1 -1 1 1
STRING
  LINE
     -0.5 0
     -0.5 2
      0.5 2
      0.5 0
  END
END
STRING
  LINE
      0 2
      0 3
  END
END
```