Univerzita Palackého v Olomouci Přírodovědecká fakulta Katedra geoinformatiky

Bc. Jan SMÉKAL

VYUŽITÍ OLAP NADSTAVEB V GIS

magisterská práce

Vedoucí práce: Ing. Zdena DOBEŠOVÁ, Ph.D.

Olomouc 2009

Prohlašuji tímto, že jsem zadanou magisterskou práci řešil sám a že jsem uvedl veškerou použitou literaturu. Všechna poskytnutá vstupní i výsledná digitální data nebudu bez souhlasu školy poskytovat.

V Olomouci 4. května 2009

podpis

Oboustranná kopie zadání magisterské práce

OBSAH

1. Úvod	
2. Cíl práce	
3. Metody a postupy zpracování	7
4. Technologie online analytical processing	
5. Tvorba zkušebních databází	14
5.1 Databáze teploty a vlhkosti půdy na experimentálním sesuvu	
v Halenkovicích	14
5.2 Databáze znečištění ovzduší v Olomouckém a Moravskoslezském	
kraji	18
5.3 Import prostorových dat do databázových serverů	20
6. Instalace, zprovoznění a možnosti OLAP nadstaveb	22
6.1 OLAP for ArcGIS	22
6.1.1 Instalace a zprovoznění	22
6.1.2 Příprava datové kostky	24
6.1.3 Možnosti nadstavby	27
6.2 PostGeoOLAP	32
6.2.1 Instalace a zprovoznění	32
6.2.2 Příprava datové kostky	33
6.2.3 Možnosti nadstavby	36
6.2.4 Integrace výsledků analýzy do GIS	38
7. Využití technologie OLAP v GIS	40
7.1 Vizualizace	40
7.2 Analýza	41
8. Návrh dalších možností	43
8.1 Využití prostorového dotazu	43
8.1.1 Prostorový dotaz v rámci OLAP analýzy	43
8.1.2 Prostorový dotaz před OLAP analýzou	44
8.1.3 Prostorový dotaz po OLAP analýze	49
8.2 Prostorový OLAP (Spatial OLAP)	50
9. Diskuze	52
10. Závěr	55
11. Použité zdroje	56
Summary	58

1. ÚVOD

Žádný GIS (geografický informační systém) se neobejde bez dat. Běžně jsou v něm využívána geografická data v nejrůznějších formátech, databázové tabulky nebo databáze. Ty mohou obsahovat data (i geografická) ve formě samostatných tabulek nebo mohou být tabulky spojeny relačními vztahy. Na pomyslném vrcholu pak stojí velké geografické databáze, které mohou zahrnovat všechny výše zmíněné formy uložení dat a které současné moderní GIS často využívají. Vedle toho ale existují i jiné formy uložení dat, které jsou používány v jiných odvětvích lidské činnosti. Jednou z nich jsou OLAP (online analytical processing) databáze, které jsou určeny k analýze multidimenzionálních dat a nacházejí uplatnění např. v řízení a marketingu velkých obchodních společností.

V poslední době se objevilo několik řešení, která umožňují data uložená právě v tomto typu databází zpracovat pomocí OLAP analýz a výsledky použít v GIS systémech. Tímto se otevírá prostor pro zkoumání přínosu, jenž tato technologie přináší do prostředí GIS. Tato práce se bude zabývat jednou z částí této problematiky.

2. CÍL PRÁCE

Cílem magisterské práce bylo seznámit se s problematikou OLAP technologie, na základě těchto poznatků zjistit možnosti specializovaných OLAP nadstaveb v aplikacích ArcGIS Desktop (extenze OLAP for ArcGIS) a PostGIS (PostGeoOLAP 0.8.3) a zdokumentovat proces provádění OLAP analýz v jejich prostředí. Pro splnění tohoto cíle musely být vytvořeny ukázkové databáze (datové krychle) s geografickými daty. Vybrané řezy datovými krychlemi byly umístěny v digitální formě do přílohy.

Na základě zjištěných výsledků byly nastíněny možnosti využití výsledků OLAP analýz v GIS a navrženy jejich další alternativy. O magisterské práci byla vytvořena statická internetová stránka, která byla umístěna na internetových stránkách katedry.

3. METODY A POSTUP ZPRACOVÁNÍ

Zpracování této diplomové práce začalo studiem literárních a internetových zdrojů, jehož cílem bylo co nejlepší osvojení si problematiky OLAP databází a analýz. Dále bylo postupně nainstalováno potřebné programové vybavení (ArcGIS Desktop 9.0 společně s databázovým serverem Microsoft SQL Server 2005 ve verzi Enterprise Edition a databázový server PostgreSQL 8.3.0 s nadstavbou pro uložení prostorových dat PostGIS 1.3.2) a OLAP extenze k oběma aplikacím (OLAP for ArcGIS pro ArcGIS Desktop a PostGeoOLAP pro PostGIS).

Microsoft SQL Server 2005 je komerční produkt, tudíž byla v práci použita 180 denní zkušební verze, kterou lze stáhnout z adresy:

http://www.microsoft.com/sqlserver/2005/en/us/trial-software.aspx. Zatímco server PostgreSQL je aplikace typu "open source" a je díky BSD licenci v plné verzi zdarma ke stažení z adresy: http://www.postgresql.org, stejně jako jeho nadstavba PostGIS, která využívá GNU GPL licenci a je volně ke stažení z adresy: http://postgis.refractions.net.

Práce pokračovala seznámením se s prostředím a prací ve výše zmíněných databázových serverech, k čemuž byly využity aplikace pro jejich správu. V případě Micsrosoft SQL Server 2005 se jednalo o aplikaci SQL Server Management Studio a v případě PostgreSQL o program pgAdmin III ve verzi 1.8.2. Oba tyto nástroje jsou přímou součástí instalace příslušného databázového řešení.

Následovala fáze přípravy a zpracování zkušebních OLAP databází tzv. datových krychlí v prostředí obou databázových serverů, pomocí nichž měly být zjišťovány možnosti zkoumaných nadstaveb. V této fázi bylo cílem najít a implementovat takové mechanismy, které by zajistily převod dat, ve formě v jaké byla k dispozici, do struktury specifické pro potřeby OLAP databází. K tomuto účelu byl využit skriptovací jazyk Avenue v prostředí aplikace ArcView 3.1, dvojice aplikací Microsoft Excel a Word ve verzi 2003 a program JanDat verze 2.3.7, který je součástí systému Janitor J/2, který je volně stažitelný z adresy: http://janitor.cenia.cz (celý postup zpracování je uveden v samostatné kapitole 5.1). Poněvadž je tématem této práce využití OLAP technologie v prostředí GIS, bylo nutné najít řešení, která by umožnila uložení prostorových dat v prostředí obou databázových serverů.

Stěžejní část práce se zabývá podrobným popisem instalace, zprovoznění a možností zkoumaných OLAP nadstaveb. Rozebírán je zde také celý postup

zpracování OLAP analýzy v obou řešeních a způsob, jakým lze její výsledky integrovat a použít v GIS aplikacích. Výstupy analýz jsou uvedeny v digitální příloze ve formě řezů datovými krychlemi. Následně pak bylo v krátké kapitole nastíněno, jakým způsobem lze tyto výsledky využít pro vizualizaci a provádění prostorových či neprostorových analýz.

V závěru pak byly představeny návrhy dalších možností spolupráce OLAP technologie s GIS. Jedním z nich je implementace prostorového dotazu do OLAP analýzy, pro kterou byla vytvořena dvojice skriptů v jazyce Python 2.5.1, určených pro použití v prostředí aplikace ArcGIS Desktop 9.3. Zde je nutné zmínit, že ačkoliv byla nadstavba OLAP for ArcGIS provozována v prostředí aplikace ArcGIS Desktop 9.0 (z důvodů uvedených v kapitole 6.1.1), tak implementace prostorového dotazu byla tvořena na jiném počítači pro verzi ArcGIS Desktop 9.3, a to z toho důvodu, že pro zpracování skriptů bylo nutné využít technologii ArcSDE, která nebyla pro verzi 9.0 k dispozici.

Jak je zmíněno výše, v rámci práce byly používány dva počítače s rozdílnou konfigurací, a to Intel Core 2 Duo 2,13 GHz, 3 GB RAM, OS Windows XP SP2 pro zkoumání nadstavby PostGeoOLAP a implementaci prostorového dotazu v prostředí ArcGIS Desktop 9.3 a Intel Celeron 2 GHz, 1,87 GB RAM, OS Windows XP SP2 pro zjišťování možností nadstavby OLAP for ArcGIS.

4. TECHNOLOGIE ONLINE ANALYTICAL PROCESSING

Online Analytical Processing (OLAP) je jednou z technologií Business Intelligence. Pojem Business Intelligence (BI) představuje proces transformace údajů na informace a převod těchto informací na poznatky [7]. Jinými slovy lze říci, že účelem BI je převést velké objemy nepřehledných údajů na jednoznačné poznatky, které jsou potřebné pro koncového uživatele a na jejichž základě lze potom např. učinit konkrétní opatření či rozhodnutí. Technologie BI, mezi které kromě OLAP patří i budování datových skladů (data warehouse) a dolování dat (data mining), se začali prosazovat do praxe okolo roku 1990 a v současné době jsou nedílnou součástí moderních databázových serverů. Samotný termín OLAP zavedl Dr. E. F. Codd a existuje pro něj více definic. Jednou z nich je: "OLAP je volně definovaný řád principů, které poskytují dimenzionální rámec pro podporu rozhodování".

Technologie OLAP je využívána pro analýzu velkého množství multidimenzionálních údajů, které obvykle obsahují časovou složku. Výsledkem analýzy jsou poté přehledné souhrny (sumy, průměry atd.), které mohou být přehledně prezentovány v podobě tabulek, grafů, nebo v případě geograficky vztažených údajů, map a mohou sloužit např. jako podpůrná data v procesu rozhodování či pro hledání jinak obtížně odhalitelných vztahů a závislostí.

OLAP analýza vyžaduje provedení velkého množství výpočtů a agregací, a to může být při velkém množství dat, která jsou tímto způsobem obvykle zpracovávána, časově velmi náročné. V tomto případě se z tohoto důvodu nepoužívají data uložená v běžné relační databázi, jejíž tabulky vyhovují podmínkám druhé a třetí normální formy a jsou tedy atomické a relačně svázané. Tyto databáze jsou pro tento typ analýz z hlediska efektivity nevhodné [7].

Namísto relační se používá multidimenzionální typ databáze, která překonává výše zmíněná omezení uložením dat do multidimenzionálních struktur. V této databázi jsou data uložena převážně v nenormalizovaných tabulkách, které lze rozdělit na tabulky faktů a dimenzí. Tyto tabulky jsou nejčastěji dvourozměrné a jsou navzájem spojeny pomocí relací. Tabulka faktů obsahuje numerické hodnoty (např. naměřené hodnoty či ceny výrobků) a v rámci databáze má obvykle největší objem dat. Její obsah se s průběhem času rozšiřuje o nově získaná data. V dimenzích jsou uvedeny údaje, které popisují, co představují numerické hodnoty v tabulce faktů. Budou-li v tabulce faktů uloženy např. výsledky nějakého měření, pak dimenze mohou být např. tři, a to kdy, kde a co bylo měřeno. Tabulky dimenzí jsou menší,

data jsou v nich relativně stabilní a jejich atributy jsou obvykle uspořádány ve stromové hierarchické struktuře (viz níže uvedený příklad). Tento mechanismus se v praxi používá k tzv. "drilování", což znamená, že je možné určit, pro jakou úroveň dimenze budou počítány souhrnné údaje (globální roční výsledky či detailní denní).

Stromová hierarchická struktura dimenze Čas:

- Rok
 - Měsíc
 - Týden
 - Den atd.

Tabulka faktů je s dimenzemi svázána pomocí cizích klíčů, které každou numerickou hodnotu pojí pomocí relací s primárními klíči v tabulkách dimenzí (viz Obr. 1).



Obr. 1 Obecná struktura OLAP databáze

Analýzou a agregací údajů uvedených v tabulkách faktů a dimenzí, vznikne multidimenzionální datová struktura tzv. datová krychle či kostka. Tu si lze v multidimenzionálním datovém modelu představit jako ekvivalent tabulky v relační databázi. Jednotlivé hrany krychle představují dimenze. Hodnoty (fakta) jsou umístěny na průnicích dimenzí, přičemž jejich počet není teoreticky omezen a jednotlivé dimenze obvykle neobsahují stejné množství údajů.

Jestliže datová krychle obsahuje např. údaje o množství importovaného zboží a dimenze času, původu a dopravy, tak je díky tomuto způsobu uložení dat možné pomocí agregačních funkcí získat souhrnné údaje za určité časové období, vztahující se k určitému původu nebo konkrétnímu způsobu dopravy. Data v této formě jsou pro vedení firmy provádějící import zboží často důležitější a přehlednější než jednotlivé dílčí údaje.



Obr. 2 Ukázka datové krychle s třemi dimenzemi (upraveno podle [1])

Multidimenzionální model, na jehož základě se datová krychle vytváří, může mít různé topologické uspořádání tzv. schéma. Nejběžněji je používáno hvězdicové schéma (star schema) a schéma "sněhové vločky" (snowflake schema) [7]. První zmíněné neobsahuje relační propojení mezi dimenzemi, které jsou v nenormalizované podobě. Díky tomu tento model dosahuje vysokého dotazovacího výkonu, ale jeho vytvoření je v důsledku nenormalizovaných dimenzí časově náročné. Dimenzionální model založený na schématu "sněhové vločky" obsahuje dimenze, z nichž některé jsou tvořené více relačně spojenými tabulkami. Ten má přesně opačné vlastnosti než předchozí. Je snazší na vytvoření, ale dosahuje nižšího dotazovacího výkonu. V obou těchto schématech se nachází pouze jedna tabulka faktů, ale existují i taková, kde jich může být více.



Obr. 3 Hvězdicové schéma (nahoře) a schéma "sněhové vločky" (dole)

S rostoucím počtem dimenzí v databázi se zvětšuje také její náročnost na úložnou kapacitu. Dalším faktem, který tomuto jevu může napomáhat je, že se ne vždy na všech průsečících dimenzí nacházejí údaje. Takovéto krychle se označují jako řídké [7]. Z těchto důvodů se v praxi vyvinuly různé technologie sloužící ke kompresi objemu multidimenzionálních databází při OLAP analýze.

Multidimenzionální OLAP (MOLAP)

Pomocí tohoto řešení se vypočítá co nejvíce předběžných výsledků, které jsou spolu s analytickými daty a indexy uloženy do jednotlivých polí multidimenzionální databáze. Ta je organizována tak, aby bylo možné rychle získat údaje z více dimenzí najednou. Výhodou je pak vysoký výkon tohoto řešení a nevýhodou redundance údajů, neboť jsou data uložena také v relační databázi.

Relační databázový OLAP (ROLAP)

Tato technologie ukládá data a metadata do relační databáze. Pomocí metadat se při OLAP analýze generují SQL dotazy, které se používají na získávání dat požadovaných uživatelem. Díky relačnímu způsobu uložení nevzniká problém s redundancí dat.

Hybridní OLAP (HOLAP)

Hybridní OLAP v sobě kombinuje technologie MOLAP a ROLAP, čímž využívá jejich výhody a značně eliminuje jejich nevýhody. V tomto řešení jsou detailní data uložena v relačním a vytvořené agregace v multidimenzionálním typu databáze.

Obecně je nutné u multidimenzionálního typu databáze zmínit tyto výhody:

- rychlý a komplexní přístup k velkému objemu údajů
- přístup k multidimenzionálním a relačním datovým strukturám
- možnost provádění komplexních analýz
- silné schopnosti pro modelování a vytváření prognóz

Nevýhodami jsou pak velká náročnost na diskovou kapacitu při velkém množství vstupních dat a problémy, které nastanou v případě, kdy je nutné strukturu dimenzí změnit tak razantním způsobem, že je třeba změnit časový interval údajů v časové dimenzi (je-li taková). Např. je-li nezbytné přejít od týdenního k dennímu sledování hodnot [7].

5. TVORBA ZKUŠEBNÍCH DATABÁZÍ

5.1 Databáze teploty a vlhkosti půdy na experimentálním sesuvu v Halenkovicích

Vstupní data pro tuto databázi byla získána z bakalářské práce Bc. Jana Geletiče [4]. Data byla uložena ve formátu programu Microsoft Excel (XLS) a obsahovala údaje o osmi sledovaných charakteristikách ze dvou měřících stanic. Jednalo se o teplotu půdy v 5, 10, 20, 35 a 50 cm, vlhkost půdy ve 20 a 40 cm a průtok z horizontálního vrtu (údaje o průtoku nebyly dále zpracovávány, protože netvořily ucelenou časovou řadu). Dále pak byly v datech obsaženy záznamy o čase, ve kterém byla každá hodnota naměřena (rok, měsíc, den, hodina a minuta), přičemž interval mezi jednotlivými měřeními byl 15 minut. Data byla sbírána v období od 23. listopadu 2006 do 2. listopadu 2007.

Aby mohla být vytvořena databáze v prostředí databázového serveru, musela být data přetransformována do podoby dotazu v jazyce SQL. Nejprve byl soubor ve formátu XLS přímo v prostředí programu Microsoft Excel rozčleněn na tři části (naměřené hodnoty - fakta a dimenze času a měřených parametrů), které byly samostatně uloženy a z nichž poté měly vzniknout jednotlivé databázové tabulky. Byly také umazány nepotřebné sloupce s údaji, které nebyly pro výslednou databázi podstatné. Poté byly jednotlivé XLS soubory uloženy do formátu dBase IV (DBF). DBF soubor obsahující naměřené hodnoty musel být upraven pomocí skriptovacího jazyka Avenue v prostředí ArcView 3.1 (viz Obr. 4) do podoby tabulky faktů potřebné pro provádění OLAP analýz (tj., aby každá naměřená hodnota byla uvedena v samostatném záznamu a byla spojena se záznamy z jednotlivých dimenzí pomocí identifikátorů). Výsledná databáze obsahovala tabulku faktů s 372 848 unikátními záznamy a tři tabulky dimenzí (čas, měřené parametry a měřcí stanice).

	A	В	C	D	E		F	G	Н	I.	J	K	Ĺ	M
1	ID	rok	den	hodina	rok-den-ho	dina	t5	t10	t20	t35	t50	Prutok	V1_20cm	V2_40cm
2	1	2006	1123	1300	20061123	3130000	8,90	8,80	8,60	9,00	9,20	29,30	27,90	28,30
3	2	2006	1123	1315	20061123	3131500	9,10	8,80	8,70	9,00	9,20	29,70	27,10	27,90
4	3	2006	1123	1330	20061123	3133000	9,30	8,90	8,70	9,10	9,30	29,20	27,20	28,00
5	4	2006	1123	1345	20061123	3134500	9,40	9,00	8,70	9,10	9,30	29,10	27,40	27,90
6	5	2006	1123	1400	20061123	3140000	9,40	9,00	8,80	9,10	9,30	29,20	27,40	27,90
7	6	2006	1123	1415	20061123	8141500	9,50	9,10	8,80	9,10	9,30	29,10	27,00	28,00
8	7	2006	1123	1430	20061123	3143000	9,40	9,10	8,80	9,10	9,30	29,20	27,10	28,10
9	8	2006	1123	1445	20061123	3144500	9,40	9,10	8,80	9,10	9,20	29,10	27,30	27,90
10	9	2006	1123	1500	20061123	3150000	9,30	9,10	8,80	9,10	9,30	29,00	27,30	27,90
11	10	2006	1123	1515	20061123	3151500	9,30	9,10	8,90	9,10	9,30	29,00	27,20	27,90
12	11	2006	1123	1530	20061123	3153000	9,20	9,20	8,90	9,10	9,30	29,00	27,20	28,10
13	12	2006	1123	1545	20061123	3154500	9,10	9,20	8,90	9,10	9,30	29,10	27,40	27,80
14	13	2006	1123	1600	20061123	3160000	9,10	9,20	8,90	9,10	9,30	29,00	27,20	27,80
				[1,1]	HODNOTA	ID_CA	S ID	PARAN	1 ID_N		D_FAK	TUM		
				1	8.90		1		1	1		1		
				2	8.80		1		2	1		2		
			-	3	8.60		1		3	1		3 🚽		
				4	9.00		1		4	1		4		
				.5	9.20		1		5	1		5		
				6	27,90		1	1	6	1		6		
			-	7	28,30		1	ģ	7	1		7		
				8	9.10		2	į.	1	1		8		
			-	9	8.80		2	ŝ	2	1		9		

Obr. 4 Změna struktury dat ze vstupního souboru ve formátu XLS do podoby tabulky faktů

DBF soubory byly poté načteny do aplikace JanDat ze systému Janitor J/2. Tato aplikace umožňuje přímé uložení souboru ve formátu DBF do formátu databázového SQL dotazu (SQL). Výsledný SQL soubor z programu JanDat je uzpůsoben pro použití v databázovém serveru MySQL ve verzi 4.1 a není jej možné bez úprav použít pro databázové servery využité v této práci. Textový soubor SQL tudíž bylo třeba ještě před jeho aplikací upravit v programu Microsoft Word a to tak, že byla odstraněna hlavička souboru, definice znakové sady, příkaz pro vytvoření tabulky a všechny znaky "`" (viz Obr 5).

<pre># Janitor SQL import&export # version 2.1 # http://www.janitor.cz # # Generated: 15.05.2008 22:20:48 # version MvSOL: 4.1</pre>					
#					
# Header					
CREATE TABLE IF NOT EXISTS 'Merene_pai `ID``DECIMAL(9,0) NOT NULL DEFAULT (`PARAMETR` VARCHAR(12))	rametry'(),	+			
CHARSET = Cp1250]			
COLLATE = cp1250_czech_cs;		82 1	12432		
# Data	INSERT IN	ITO Merene_pa	rametry	(ID,	PARAMETR)
INSERT INTO Merene_parametry (ID ,	VALUES ('	1', 't_5cm	');		
INCEPT INTO MORODO DOROMOTORY' ('ID'	INSERT IN	ITO Merene_pa	rametry	(ID,	PARAMETR)
VALUES ('2' '+ 10cm ')	VALUES ('	2', 't_10cm	');	200326	1778 - 1978 - 1978 - 19 <u>7</u> 4
INSERT INTO `Merene parametry` (`ID`	INSERT IN	ITO Merene_pa	rametry	(ID,	PARAMETR)
VALUES ('3', 't 20cm ');	VALUES ('	3', 't_20cm	·);	21	123
INSERT INTO `Merene parametry` (`ID`.	INSERT IN	ITO Merene_pa	rametry	(ID,	PARAMETR)
VALUES ('4', 't_35cm' ');	VALUES (4', 't_35cm			1997 1 9 -1997 - 1997
INSERT INTO 'Merene_parametry' ('ID',	INSERT IN	ITO Merene_pa	rametry	(ID,	PARAMETR)
VALUES ('5', 't_50cm ');	VALUES (5, t_50cm);	(
INSERT INTO 'Merene_parametry' ('ID',	INSERT IN	lio Merene_pa	rametry	(10,	PARAMEIR)
VALUES ('6', 'V_20cm ');	VALUES (o, v_zucm	J;	(TD	DADAMETRY
INSERTING Merene_parametry (ID ,	VALUES (7' 'v 40cm	ametry.	(10,	PARAMETRI
	THEORD (, , <u>, , , , , , , , , , , , , , , , , </u>	12		

Obr. 5 Příklad výsledku transformace tabulky ve formátu DBF do databázového SQL dotazu v prostředí aplikace JanDat

V obou databázových serverech byla připravena prázdná databáze s nadefinovanými tabulkami a atributy (struktura viz Obr. 6), do které byla pomocí výsledného SQL dotazu naimportována všechna data (u databázového serveru PostgreSQL je nutné, aby všechny názvy tabulek a atributů byly malými písmeny, jinak import neproběhne). Nakonec byly nastaveny primární a cizí klíče tak, aby relační databáze byla ve formátu potřebném pro provádění OLAP analýz.



Obr. 6 Struktura databáze teploty a vlhkosti půdy na experimentálním sesuvu v Halenkovicích

```
hodnoty = av.GetProject.FindDoc("hodnoty.dbf")
cas = av.GetProject.FindDoc("cas.dbf")
hodnotyVTab = hodnoty.GetVTab
casVTab = cas.GetVTab
hodnoty_kod = hodnotyVTab.FindField("Kod")
t5 = hodnotyVTab.FindField("t5")
t10 = hodnotyVTab.FindField("t10")
t20 = hodnotyVTab.FindField("t20")
t35 = hodnotyVTab.FindField("t35")
t50 = hodnotyVTab.FindField("t50")
V1_20cm = hodnotyVTab.FindField("V1_20cm")
V2 40cm = hodnotyVTab.FindField("V2 40cm"
meridlo = hodnotyVTab.FindField("Meridlo")
id = casVTab.FindField("Id")
cas_kod = casVTab.FindField("Kod")
theVTab=VTab.MakeNew("Tabulka_faktu".AsFileName,dBase)
id faktum = Field.Make("Id faktum", #FIELD DOUBLE, 10, 0)
hodnota = Field.Make("Hodnota", #FIELD DECIMAL, 6, 2)
id_cas = Field.Make("Id_cas",#FIELD_DOUBLE,10,0)
id_meridlo = Field.Make("Id_misto", #FIELD_SHORT, 2, 0)
id_charakteristika = Field.Make("Id_param", #FIELD_SHORT, 2, 0)
theVtab.AddFields({id_faktum,hodnota,id_cas,id_meridlo,id_charakteristika})
pocet_casu = casVTab.GetNumRecords
pocet_hodnot = hodnotyVTab.GetNumRecords
posun = 0
for each i in 0 .. (pocet hodnot-1)
  meridlo id = hodnotyVTab.ReturnValue(meridlo,i)
  if (meridlo_id.AsString = "ALA") then
    meridlo_promenna = 1
  else
    meridlo_promenna = 2
  end
  hodnoty_id = hodnotyVTab.ReturnValue(hodnoty_kod,i)
  for each e in 0 .. (pocet_casu - 1)
    cas id = casVTab.ReturnValue(cas kod,e)
    if (hodnoty_id = cas_id) then
      cas_promenna = casVTab.ReturnValue(id,e)
    end
  end
  theVTab.AddRecord
  t5_promenna = hodnotyVTab.ReturnValue(t5,i)
  theVTab.SetValue (id faktum,posun,posun+1)
  theVTab.SetValue (hodnota,posun,t5_promenna)
  theVTab.SetValue (id_cas,posun,cas_promenna)
  theVTab.SetValue (id_meridlo,posun,meridlo_promenna)
  theVTab.SetValue (id_charakteristika,posun,1)
  posun = posun + 1
  Opakuje se ještě šestkrát pro t10, t20, t35, t50, V1 20cm a V2 40cm
```

end

theVTab.SetEditable(false)

Obr. 7 Avenue skript pro úpravu tabulky ve formátu DBF obsahující naměřené hodnoty

5.2 Databáze znečištění ovzduší v Olomouckém a Moravskoslezském kraji

Data pro tuto databázi byla získána z internetových stránek Českého hydrometeorologického ústavu (dostupné z:

http://www.chmi.cz/uoco/isko/tab_roc/tab_roc.html).

Ze zde uvedených měřících stanic bylo vybráno dvanáct (v Olomouckém kraji pět a sedm v Moravskoslezském), které byli v provozu po celé období let 1997 – 2007, z něhož byla data na internetu dostupná a měla po vizuálním prozkoumání nejúplnější časové řady měření hodnot znečištění v denním intervalu (každá stanice měla alespoň jeden výpadek v měření). Z imisí měřených stanicemi byla vybrána trojice SO₂, NO₂ a NO_x.

Surová data byla z internetu vykopírována do programu Microsoft Excel. Dále pak již byl postup zpracování a úpravy dat do potřebného formátu pro vytvoření OLAP databáze obdobný jako u předchozí databáze pouze s tím rozdílem, že kvůli jiné struktuře vstupních dat byl použit odlišný skript v jazyce Avenue. Výsledkem byla tabulka faktů se 162 724 záznamy a tři tabulky dimenzí (čas, imise a měřící stanice).



Obr. 8 Struktura databáze znečištění ovzduší v Olomouckém a Moravskoslezském kraji

if ((crok = xrok) and (cmes cis = xmesic) and (cden = 1)) then Opakuje se ještě třicetkrát pro 2 – 31 (číslo se mění v červeně označených místech). faktaVTab.SetValue(fstanice,pocitadlo,xstaniceid) xstaniceid = theFTab.ReturnValue(idstanice,e) xdruhimise = imiseVTab.ReturnValue(druhimise,f) faktaVTab.SetValue(fid,pocitadlo,pocitadlo+1) ximiseid = imiseVTab.ReturnValue(idimise,f) faktaVTab.SetValue(fimise, pocitadlo, ximiseid) cmes_cis = casVTab.ReturnValue(mes_cis,g) xoznaceni = theFTab.ReturnValue(Oznaceni,e) xstanice = zdrojVTab.ReturnValue(zstanice,i) ccasid = casVTab.ReturnValue(idcas,g) faktaVTab.SetValue(fcas, pocitadlo, ccasid) faktaVTab.SetValue(fhodnota,pocitadlo,x xmesic = zdrojVTab.ReturnValue(zmesic,i) ximise = zdrojVTab.ReturnValue(zimise,i) crok = casVTab.ReturnValue(rok,g) cden = casVTab.ReturnValue(den,g) xrok = zdrojVTab.ReturnValue(zrok,i) X31 = zdrojVTab.ReturnValue(z31,i) for each e in 0.. (radky stanice-1) if (xoznaceni = xstanice) then For each g in 0.. (radky cas-1) x1 = zdrojVTab.ReturnValue(z1,i) for each f in 0.. (radky imise-1) if (xdruhimise = ximise) then for each i in 0.. (radky_zdroj-1) pocitadlo = pocitadlo+1 faktaVTab.AddRecord if (x <>0) then end end end end end end end end zdroj = av.GetProject.FindDoc("surova_tabulka.dbf") fstanice = faktaVTab.FindField("Id_stanice") imise = av.GetProject.FindDoc("imise.dbf") fakta = av.GetProject.FindDoc("fakta.dbf") theTheme = theView.GetActiveThemes.Get(0) fhodnota = faktaVTab.FindField("Hodnota") zstanice = zdrojVTab.FindField("Stanice") theShape = theFTab.FindField("Oznaceni") druhimise = imiseVTab.FindField("Imise") oznaceni = theFTab.FindField("Oznaceni") fimise = faktaVTab.FindField("Id_imise") cas = av.GetProject.FindDoc("cas.dbf") mes_cis = casVTab.FindField("Mes cis") radky_stanice = theFTab.GetNumRecords radky_imise = imiseVTab.GetNumRecords radky_zdroj = zdrojVTab.GetNumRecords zimise = zdrojVTab.FindField("Imise") zmesic = zdrojVTab.FindField("Mesic") fcas = faktaVTab.FindField("Id cas") idstanice = theFTab.FindField("Id") idimise = imiseVTab.FindField("Id") radky cas = casVTab.GetNumRecords zrok = zdrojVTab.FindField("Rok") Z31 = zdrojVTab.FindField("31.") fid = faktaVTab.FindField("Id") idcas = casVTab.FindField("Id") z1 = zdrojVTab.FindField("1.") rok = casVTab.FindField("Rok") den = casVTab.FindField("Den") theID=theFTab.FindField("Id" faktaVTab.SetEditable(true) theFTab = theTheme.GetFTab theView = av.GetActiveDoc faktaVTab = fakta.GetVTab imiseVTab = imise.GetVTab zdrojVTab = zdroj.GetVTab casVTab = cas.GetVTab

Obr. 9 Avenue skript pro úpravu tabulky ve formátu DBF obsahující naměřené hodnoty imisí

pocitadlo = 0

faktaVTab.SetEditable(false)

5.3 Import prostorových dat do databázových serverů

Pro databázi vlhkostí a teplot půdy byla prostorová data již k dispozici v souřadném systému S-JTSK, zatímco pro databázi znečištění ovzduší bylo prostorová data nejprve nutné vytvořit ze souřadnic v systému WGS 84 a poté je do systému S-JSTK transformovat.

Databázový server Microsoft SQL Server 2005 uložení prostorových dat neumožňuje. Některé aplikace a řešení však umožňují toto omezení překonat. V této práci bylo pro uložení prostorových dat do tohoto databázového serveru využito jeho spojení s aplikací ArcGIS Desktop 9.3 pomocí technologie ArcSDE. Toto řešení umožňuje vytvářet pomocí ArcCatalogu geodatabáze s prostorovými i neprostorovými daty přímo v prostředí databázového serveru, k čemuž se využívá možnost Database Servers. Při přidání nového připojení stačí pouze vyplnit název databázové části serveru.

Při využití databázového serveru PostgreSQL s nadstavbou pro uložení prostorových dat PostGIS je nutné před importem prostorových dat v souřadném systému S-JTSK vložit korektní definici tohoto systému do databáze pomocí SQL skriptu. V řešení PostgreSQL/PostGIS je uvedena definice EPSG 2065. Pro data z našeho území je ale korektní definice ESRI 102067 [20]. Pro import prostorových dat do PostgreSQL/PostGIS byla nejprve vyzkoušena varianta s použitím programu JanMap ze systému Janitor J/2. Ačkoliv by tato aplikace měla import umožňovat, tak v tomto případě požadovaná operace skončila neúspěchem. Aplikace při importu zobrazila hlášku "Oggr dataset is null" a hlásila, že se nelze připojit k databázi, přičemž test spojení s databází hlásil, že je vše v pořádku.

Namísto toho tedy musela být použita varianta s využitím volně šiřitelného programu Quantum GIS ve verzi 0.10, který je ke stažení na adrese: http://download.ggis.org. V něm bylo nutné v okně *Správce zásuvných modulů* ze záložky *Zásuvné moduly* aktivovat modul SPIT (Shapefile to PostgreSQL/PostGIS Import Tool), pomocí nějž byl import prostorových dat do databáze v pořádku proveden. V prostředí modulu po nastavení připojení ke konkrétní databázi, do níž se budou prostorová data ukládat, je možné nastavit jméno atributového pole, které bude obsahovat geografickou informaci.

Dále zde lze nastavit identifikátor souřadného systému (SRID), název nového atributového pole, které bude obsahovat primární klíč a schéma databázového serveru. Poté se již jen přidá požadovaná vrstva či jejich skupina a tlačítkem *OK* se provede samotný import. Zajímavostí je, že poté se již bylo možné z aplikace JanMap do databáze k importovaným datům bez problémů připojit a v prostředí aplikace je zobrazit.

```
INSERT INTO "spatial_ref_sys"
("srid","auth_name","auth_srid","srtext","proj4text")
VALUES(102067, 'ESRI', 102067,
'PROJCS["S-JTSK_Krovak_East_North",GEOGCS["GCS_S_JTSK",DATUM["D_S_JTSK",
SPHEROID["Bessel_1841",6377397.155,299.1528128]],PRIMEM["Greenwich",0],
UNIT["Degree",0.017453292519943295]],PROJECTION["Krovak"],
PARAMETER["False_Easting",0],
PARAMETER["False_Northing",0], PARAMETER["Pseudo_Standard_Parallel_1",78.5],
PARAMETER["Scale_Factor",0.9999],
PARAMETER["Azimuth", 30.28813975277778],
PARAMETER["Longitude_Of_Center",24.83333333333333],
PARAMETER["Latitude_Of_Center",49.5],
PARAMETER["X_Scale",-1],
PARAMETER["Y_Scale",1],"
PARAMETER["XY_Plane_Rotation",90],
UNIT["Meter",1]]',
'+proj=krovak +lat_0=49.5 +lon_0=24.83333333333333333 +alpha=30.28813975277778
+k=0.9999 +x_0=0 +y_0=0 +ellps=bessel +units=m');
```

Correction of the second state of the second	Obr.	10 SQL	příkaz pi	ro vložení s	ouřadného	systému	S-JTSK	do Postgr	eSQL/Post	GIS [20]
--	------	--------	-----------	--------------	-----------	---------	--------	-----------	-----------	----------

Znecisteni						•
				Nové	Editovat	Odebrat
mport options and shapefile list						
🗙 Use Default Geometry Column Name	or specify here				the_ge	om
Use Default SRID or specify here				102067	A. 0.144]\$
Primary Key Column Name				gid		
Slobální schéma			put	lic		
Jméno souboru	Třída objektu	Objekty	DB Relation Name	Schéma		
E:\Honza\Znecisteni\stanice_final.shp	POINT	14	stanice_final	public		
		1	h.	<u>()</u>		
				DØdst]	Odebrat	Odebrat uže

Obr. 11 Prostředí modulu SPIT v aplikaci QGIS 0.10

6. INSTALACE, ZPROVOZNĚNÍ A MOŽNOSTI OLAP NADSTAVEB

6.1 OLAP for ArcGIS

6.1.1 Instalace a zprovoznění

Tato extenze je volně stažitelná ze stránek společnosti ESRI, která ji také vytvořila a je dostupná z adresy:

http://www.esri.com/software/arcgis/extensions/olap/index.html. Extenze je k dispozici ve dvou provedeních pro platformu Windows (stav k dubnu 2009). První OLAP for ArcGIS 9.1 je určeno pro ArcGIS Desktop ve verzích 9.0 a 9.1 a druhé OLAP for ArcGIS 9.2 pro ArcGIS Desktop 9.2. Podstatné je, že samotná extenze pro využití OLAP technologie nestačí, ale uživatel musí mít navíc nainstalován databázový server, přičemž jsou podporovány tyto tři:

- Microsoft SQL Server s komponentou Analysis Services (použitý v této práci)
- SAS OLAP Server
- SAP Business Information Warehouse (SAP BW)

OLAP for ArcGIS 9.1 vyžaduje pro svoji funkčnost buď ArcGIS Desktop 9.0 Service Pack 3 nebo ArcGIS Desktop 9.1, přičemž obě tato řešení musí mít doinstalovánu technologii .*NET Support*. Nutností je také instalace rozhraní *Microsoft* .*NET Framework 1.1* (ke stažení z adresy:

http://www.microsoft.com/downloads/details.aspx?FamilyID=262d25e3-f589-4842-8157-034d1e7cf3a3&displaylang=en). OLAP for ArcGIS 9.2 vyžaduje ArcGIS Desktop 9.2 Service Pack 2 opět s technologií .NET Support a Microsoft .NET Framework 2.0 (ke stažení z adresy:

http://www.microsoft.com/downloads/details.aspx?FamilyID=0856eacb-4362-4b0d-8edd-aab15c5e04f5&DisplayLang=en).

Obě tyto extenze také potřebují komponentu *Microsoft Data Access Components (MDAC) 2.8* (ke stažení z adresy:

http://www.microsoft.com/downloads/details.aspx?FamilyID=6c050fe3-c795-4b7db037-185d0506396c&displaylang=en) a rozhraní *OLAP for OLE DB Provider* (v případě použití databázového serveru Microsoft SQL Server je toto rozhraní obsaženo již v instalaci *MDAC 2.8* a není ji tedy třeba provádět zvlášť) [11].

🖶 ArcGIS Setup	
Select Features Please select which features you would like to	o install.
Applications ArcCatalog ArcMap ArcReader Styles Templates Tablet PC Support NET Support Extensions	Feature Description: Select this feature to install the ArcGIS .NET primary interop assemblies into the global assembly cache. These are required to develop and use ArcGIS .NET customizations. This feature will remain on your local hard drive. This feature requires 0KB on your hard drive.
C:\ArcGIS90\ArcGIS\	
Help Disk Cost Reset	< <u>B</u> ack <u>N</u> ext > Cancel

Obr. 12 Instalace .NET Support do aplikace ArcGIS Desktop

Extenze se od sebe neliší funkcionalitou. V této práci bude dále popisována nadstavba OLAP for ArcGIS verzi 9.1 v prostředí aplikace ArcGIS Desktop 9.0. Původně měla být využita druhá verze, ale tu se nepovedlo nainstalovat ani na jednom ze tří zkušebních počítačů (buď se nepodařilo nainstalovat Service Pack 2 k ArcGIS Desktop 9.2 nebo se instalace extenze ukončila chybovou hláškou). S extenzí se pracuje v prostředí ArcCatalogu, kde musí být po instalaci zpřístupněna v záložce *Extensions*. Poté v ArcCatalogu přibude možnost připojení k OLAP databázi, která musí být uložena na jednom ze tří podporovaných databázových serverů. Dále je postup popisován pro databázový server Microsoft SQL Server 2005, který byl použit pro potřeby této práce.

Po kliknutí na možnost Add OLAP Connection se zobrazí okno, kde se nastavuje připojení k databázovému serveru. Z první nabídky se vybere možnost Microsoft OLE DB Provider for Analysis Services 9.0 a po kliknutí na tlačítko Next je nutné nastavit zdroj dat (název serveru na kterém je spuštěn Microsoft SQL Server 2005 s komponentou Analysis Services). Zajímavostí je, že manuál k této extenzi v tomto kroku říká, že má být použita možnost Microsoft OLE DB Provider for OLAP Services 8.0 namísto výše zmíněného Analysis Services 9.0. S touto volbou ale extenze nefungovala. Při testu připojení ke zdroji dat se objevila pouze hláška "The server is either not started or too busy" a po stisknutí tlačítka OK došlo k samovolnému přepnutí této volby na Analysis Services 9.0, se kterou nadstavba v pořádku fungovala.



Obr. 13 Zpřístupnění extenze OLAP for ArcGIS v ArcCatalogu a nastavení připojení k databázovému serveru

Po potvrzení nastavení připojení tlačítkem *OK* je nutné již jen vyplnit název tohoto připojení, které se uloží a zobrazí v prostředí ArcCatalogu. Zde se také nově zpřístupní funkce *Add Table*. Po jejím spuštění se zobrazí nabídka, ze které se vybere datová kostka, která bude následně analyzována. Po vybrání jedné z nich se spustí aplikace ODBO Table Wizard, která pracuje na základě komponenty Microsoft Office PivotTable, která je součástí instalace aplikace Microsoft Office Excel.

6.1.2 Příprava datové kostky

Extenze pracuje s daty, která jsou uložena v prostředí databázovém serveru. Proto je tedy třeba nejprve v databázové komponentě Microsoft SQL Serveru 2005 vytvořit relační databázi ve formátu, který je vyžadován pro zpracování OLAP analýz (viz kapitola 4), a naplnit ji daty. Tím úloha této komponenty končí a je možné spustit aplikaci SQL Server Business Intelligence Development Studio, která je součástí instalace komponenty MS SQL Server Analysis Services a která slouží pro tvorbu a správu datových kostek (v jejím prostředí lze provádět i OLAP analýzy). Zde se po stisknutí tlačítka *New* vybere možnost *Analysis Services Project*.

Následně se v pravé části obrazovky objeví okno s názvem *Solution Explorer*. V něm se nejprve pravým tlačítkem myši stiskne možnost *Data Sources* a vybere se jediná dostupná nabídka. Objeví se nové okno, ve kterém se po stisknutí tlačítka *New* vyplní název Microsoft SQL Serveru 2005, v němž je uložena zdrojová relační databáze (lze se takto připojit i k externím datům např. v databázi Oracle), jejíž název se následně vybere z rozbalovací nabídky. Rozbalovací nabídka *Provider* zůstane nastavena na výchozí hodnotě a je možné stisknout *OK*. Zvolená databáze se objeví v okně *Data connections*, tudíž se vybere a stiskne tlačítko *Next*. Zbývá už jen v dalším okně zvolit možnost *Use the service account*, která značí, že pro přístup k datům bude použit účet použitý v databázové části aplikace, a napsat název vytvořeného datového zdroje.

Další volbou v okně *Solution Explorer* je *Data Source Views*. Po stisknutí pravého tlačítka myši se vybere jediná dostupná nabídka. V okně *Relational data sources* se objeví název předtím vytvořeného datového zdroje (*Data Source*). Po jeho vybrání a stisku tlačítka *Next* se otevře okno se všemi dostupnými tabulkami ve zdrojové databázi. Ty, které budou použity pro vytvoření datové kostky a následnou analýzu, se přesunou z levé části do pravé a vyplní se název.

Stejný postup jako v předchozích dvou případech se použije na volbu *Cubes* z okna *Solution Explorer*. Objeví se okno, ve kterém je možné vybrat, zda se datová kostka vytvoří na základě nějakého datového zdroje nebo bez něj. Poněvadž v tomto případě je datový zdroj k dispozici, zvolí se první možnost. Po vypnutí volby *Auto build* (tuto možnost lze nechat zatrhnutou a datovou kostku, kterou vytvoří počítač pak pouze upravit) se přejde k dalšímu oknu. Zde se zvolí dostupný zdroj dat a v dalším okně se rozdělí tabulky na dimenze a na tabulku(y) faktů. V horní části se ještě z rozbalovací nabídky vybere tabulka, která bude představovat časovou dimenzi (je-li taková). Následně se přiřadí atributy z tabulky představující časovou dimenzi k odpovídajícím hodnotám v levé části okna. V dalších dvou oknech se určí atribut(y) z tabulky faktů a dimenzí, které budou použity pro tvorbu datové kostky a zvolí se její název. Pak už jen stačí v okně *Solution Explorer* na nově vytvořené datové kostce stisknout pravé tlačítko myši a zvolit možnost *Process* a kostka je připravena k OLAP analýze v prostředí nadstavby.



Obr. 14 Aplikace SQL Server Business Intelligence Development Studio s oknem Solution Explorer

Před stisknutím tlačítka je ještě dobré pomocí utility SQL Server Configuration Manager, která je dostupná přes Start/Programy/Microsoft SQL Server/Configuration Tools, zkontrolovat, zda je spuštěná komponenta MS SQL Server Analysis Services a případně ji spustit. Pokud totiž komponenta není spuštěna, požadovaná operace *Process* se neprovede a skončí chybovou hláškou.

Důležité je ještě zmínit, že výše uvedeným úkonem dojde k uložení informací o datové kostce a zdrojích dat do zvláštní databáze uvnitř komponenty Analysis Services, která svým názvem odpovídá názvu projektu, který se zadal hned po spuštění aplikace SQL Server Business Intelligence Development Studio. Bude-li třeba vytvořit novou datovou kostku na základě jiné relační databáze, je nutné, aby byla umístěna ve stejném projektu a tudíž ve stejné databázi v komponentě Analysis Services. Extenze OLAP for ArcGIS umí totiž pro OLAP analýzu nabídnout datové kostky pouze z jedné databáze (projektu), přičemž si konkrétní databázi nelze žádným způsobem vybrat. Existuje pouze jedna varianta, která přichází v úvahu, pro práci s více databázemi. Ty je možné mít všechny pomocí funkcí *Back up* a *Restore* zálohované a obnovenou mít vždy jen jednu pracovní, zatímco ostatní jsou odstraněné.



Obr. 15 Databáze Znečištění v komponentě Analysis Services se dvěma datovými kostkami a funkcemi Back Up a Restore dostupnými přes pravé tlačítko myši

6.1.3 Možnosti nadstavby

Téměř všechna nastavení, kterými lze stávající datovou kostku či OLAP analýzu v rámci této extenze upravit, se provádí v prostředí aplikace SQL Server Business Intelligence Development Studio, ve které se prováděla také tvorba datové kostky. Proto budou na následujících řádcích zmíněny některé možnosti tohoto programu. Po zpracování kostky pomocí funkce *Process* a dvojím poklepáním levého tlačítka myši na její název v okně *Solution Explorer*, se objeví nové okno s několika záložkami v horní části.



Obr. 16 Prostředí aplikace SQL Server Business Intelligence Development Studio

První z nich se nazývá *Cube Structure* a je zároveň nejdůležitější. Pomocí tlačítka *New Measure* lze zvolit, které agregační funkce mohou být použity při výpočtu souhrnných údajů v rámci OLAP analýzy. K dispozici je jich dvanáct:

- Suma (Sum)
- Počet řádků (Count of Rows)
- Počet neprázdných hodnot (Count of non-empty values)
- Minimum (Minimum)
- Maximum (Maximum)
- Počet jedinečných hodnot (Distinct count)
- Finanční bilance (Account) vyžaduje speciální "account" dimenzi
- Průměr za časové období (Average over time) vyžaduje časovou dimenzi
- První hodnota (First value)
- Poslední hodnota (Last value)
- První neprázdná hodnota (First non-empty value)
- Poslední neprázdná hodnota (Last non-empty value)

V levé spodní části obrazovky je umístěno okno s názvem *Dimensions*, které umožňuje pomocí dvou záložek nastavovat a upravovat hierarchické struktury a použité atributy v jednotlivých dimenzích. Stačí si vybrat dimenzi, která bude upravována a stisknout tlačítko *Edit*. Objeví se nové okno rozdělené do tří částí. V levé části je možné odebírat a přidávat atributy, které budou dostupné pro OLAP analýzu. Odebírá se stisknutím pravého tlačítka na odstraňovaném atributu a použití možnosti *Delete*. Naopak přidává se prostým přetažením zvoleného atributu z pravé části, ve které jsou zobrazeny všechny dostupné atributy zvolené dimenze, do levé.

V prostřední části lze vytvořit či upravit hierarchickou strukturu a to opět pomocí přetažení atributů z pravé části do zvolené struktury (pokud bude přetažen mimo ni, pak se vytvoří nová struktura). Jako první se vždy nastavuje nejobecnější hierarchická úroveň (např. rok) a pokračuje se přes více detailní k nejdetailnější úrovni (např. den). Počet úrovní není aplikací omezen.

V druhé záložce s názvem *Dimensions* je možné upravovat z jakých tabulek faktů a dimenzí bude stávající datová kostka složena. Lze zde jakoukoliv dostupnou tabulku vypnout či zapnout nebo novou z dostupných datových zdrojů připojit pomocí tlačítka *Add Cube Dimension*.

Způsob, jakým bude uložena multidimenzionální databáze používaná pro OLAP analýzu, lze nastavit či změnit v záložce s názvem *Partitions*. Po stisknutí možnosti *Design Aggregations* se objeví nové dialogové okno. V něm je na výběr z technologií ROLAP, HOLAP a několika druhů MOLAP. Po výběru jedné z nich stačí v dalším okně stisknout pouze tlačítko *Count* a počkat až počítač provede výpočet.

V posledním okně se provádí výpočet počtu agregací, které budou předpřipraveny a uloženy v databázi. Lze nastavit, jakou má mít výsledná multidimenzionální databáze celkovou velikost (v MB), jaký počet předběžných agregací má být vytvořen (v % z počtu všech možných agregací) bez ohledu na požadovanou diskovou kapacitu. Uživatel může popřípadě sám zastavit výpočet pomocí tlačítka *Stop*, jakmile některý ze zmíněných ukazatelů dosáhne požadované hodnoty. Uživatel vidí vývoj ukazatelů prostřednictvím grafu na obrazovce. Výpočet se spustí tlačítkem *Start*, a když je ukončen, celý dialog se uzavře. Např. databáze dat teplot a vlhkostí půdy uložená pomocí technologie MOLAP zabere na disku pouze 1,2 MB při 99% agregací, zatímco databáze znečištění 1,7MB při 91%. Více o jednotlivých technologiích viz kapitola 4.

V poslední záložce s názvem *Browser* již uživatel může provést samotnou OLAP analýzu a nadefinovat její parametry. K tomuto účelu ale slouží také samotná extenze OLAP for ArcGIS, tedy konkrétně její aplikace ODBO Table Wizard, která se spouští z prostředí ArcCatalogu. Proto bude dále popisována tato aplikace. Program SQL Server Business Intelligence Development Studio sice nabízí mnohem více funkcí a možností než bylo v této práci zmíněno. Ty ale přesahují její rámec a zaměření.

Po spuštění aplikace ODBO Table Wizard se zobrazí také okno *PivotTable Field List*, ve kterém jsou zobrazeny všechny agregační funkce a dimenze se svými atributy, které byly nadefinovány pro analyzovanou datovou kostku v prostředí programu SQL Server Business Intelligence Development Studio.

OLAP analýza se provádí jednoduše pomocí přetažení požadovaných atributů do prostoru řádků, sloupců a filtrů kontingenční tabulky, přičemž pokud má být výsledná tabulka napojena v prostředí ArcGIS Desktop na související prostorová data, pak musí být do oblasti řádků umístěn odpovídající atribut z prostorové dimenze, který toto spojení umožní (shodný identifikátor). Stejným způsobem se přetáhne zvolená agregační funkce (může jich být použito i více najednou) do části kontingenční tabulky, ve které se zobrazují samotné hodnoty. V závislosti na vybraných parametrech se okamžitě zobrazí požadovaná výsledná souhrnná data.

Z atributů zvolených jako filtr, lze vybírat konkrétní hodnoty, které slouží pro specifikaci kritérií OLAP analýzy. Získaná souhrnná data lze zobrazit pomocí tlačítka *Show As* také jako procentuální podíl vybraného jevu (např. nejvyšší hodnota z řádku, sloupce či celkově). Dále je možné výsledek analýzy exportovat do aplikace Microsoft Excel. Pomocí funkce *MDX* ze záložky *View* se zobrazí MDX dotaz, na jehož základě byl získán výsledek OLAP analýzy. Tento typ dotazů používá komponenta Analysis Services Microsoft SQL Serveru 2005. Celý proces funguje tak, že na základě vybraných atributových polí umístěných do kontingenční tabulky, se vygeneruje dotaz, který se provede v prostředí komponenty Analysis Services a získaný výsledek odešle a zobrazí v prostředí nadstavby.

Nejdůležitější je ale možnost pomocí volby *Save as* ze záložky *File* výsledek analýzy uložit. Ten se uloží ve formě tabulky, která je dostupná z prostředí ArcCatalogu (nejde o fyzický soubor) pod odpovídajícím připojením k OLAP databázi. Tabulku je možné obvyklým způsobem zobrazit v aplikaci ArcGIS Desktop a připojit ji k odpovídajícím prostorovým datům. Z toho vyplývá, že tato extenze vůbec nevyžaduje uložení prostorových dat v OLAP databázi.

Velkou výhodou je fakt, že pokud je v oblasti sloupců umístěno více atributových polí (jednotlivá pole je třeba zviditelnit pomocí tlačítka +) či je použito více agregačních funkcí, zobrazí se všechny i v uložené tabulce. Výsledné pole má pak název ve tvaru "pole1.pole2.polex.agregační_funkce". Pole, které je umístěné, ale není zviditelněné, není bráno v potaz. Díky této funkci není nutné pro každou sledovanou kombinaci atributů tvořit zvláštní tabulku. Funkce má ale i svá omezení. Aby šlo výsledek OLAP analýzy uložit může být použit pouze jeden filtr a v oblasti sloupců nesmí být více atributových polí ze stejné dimenze. V opačném případě se výsledky sice zobrazí, ale tabulka se buď vůbec neuloží nebo se uloží ale kvůli chybě nelze zobrazit její obsah. Navíc je užitečné obsah uložené tabulky zkontrolovat, protože v některých případech došlo k nevysvětlitelné ztrátě dat (neuložila se některá pole).

Obr. 17 MDX dotaz pro výběr největší hodnoty znečištění za jednotlivé měsíce ve sledovaném období

😟 ODBO Table \						PivotTable Field List 🛛 🛛 🕅
File View						Drag items to the PivotTable list
98 Ba 41 Z.		街 📾 % •	-1 +1 + 1 + 1 •	6 B B 2		Vinecisten
			(m) • [/.			- 10 HODNOTA - 10 Maximum HODNOT#
	MES CIS -	100	100	122	1.42	
	1	2	3	4	5	Pocet
Imise - IMISE *	Maximum HOI	DNOTA Maxim	um HODNOTA Maximu	um HODNOTA Maximun	HODNOTA M	01 10 Prumer
N02		147	114,6	84,2	54	🕀 🗐 Cas
NOx		440	240,7	158,6	89,4	DEN E
SO2		360	144	91	58	🗄 🗐 Hierarchy
Grand Total		440	240,7	158,6	89,4	🗉 🗐 MES CIS
					10	😠 🗐 MESIC
						II I ROK
						🗉 🗐 Hierarchy
						🕣 🗐 Imise
						🗉 🗐 Imise - IMISE 🛛 🚟
						Hierarchy
						🕀 🗐 Merici Stanice 🛛 😽
	<				>	Add to Row Area 💌

Obr. 18 Prostředí aplikace ODBO Table Wizard s výsledkem OLAP analýzy definované na Obr. 17

6.2 PostGeoOLAP

6.2.1 Instalace a zprovoznění

Tato OLAP extenze je určena pro databázový server PostgreSQL s nadstavbou umožňující uložení prostorových dat PostGIS a je spustitelná pod operačními systémy Windows a Linux. Domovská stránka projektu je na adrese: http://postgeoolap.sourceforge.net/ingles/index.html. Pod tvorbou této aplikace jsou podepsáni tři vědečtí pracovníci z různých akademických pracovišť v Brazílii (Giovanni Colonese, Rodrigo Soares Manhães a Alan Carvalho Galante) a její poslední verze pochází z roku 2005 (PostGeoOLAP 0.8.3). Ta byla vytvořena v programovacím jazyce Java.

Starší verze byly programovány pomocí jazyka VB.Net. Podle vyjádření autorů je projekt v důsledku nedostatku času v této chvíli ve stadiu spánku, ale jeho pokračování není vyloučeno. Díky licenci GNU GPL, jíž je aplikace opatřena, jsou ke starší i aktuální verzi dostupné na internetu kompletní zdrojové kódy, které lze libovolně upravovat. Extenze je stažitelná z adresy:

http://sourceforge.net/project/showfiles.php?group_id=107278&package_id=1158 49&release_id=236893. Instalace technologie Java je také jedinou podmínkou pro fungování této extenze. Tu není třeba ani instalovat, stačí pouze rozbalit stažený ZIP archiv a spustit soubor "startup.bat".

Před prvním spuštěním programu je ale ještě nutné provést několik úkonů. Prvním z nich je vytvoření nové prázdné databáze s názvem "pgometadata" v prostředí databázového serveru PostgreSQL, do které aplikace ukládá údaje o uživatelských nastaveních a volbách např. datové zdroje, agregace, definice datových kostek atd. Dalším krokem je editace souboru "metadata.xml", který je umístěn v kořenovém adresáři programu. V něm je nutné vyplnit IP adresu databázového serveru, název databáze s metadaty ("pgometadata" viz výše), uživatelské jméno a heslo pro přístup k databázovému serveru. Nyní je již možné aplikaci spustit a v horní liště zvolit možnost *Metadata* a poté *Generate Metadata*. Pomocí této funkce se v prázdné databázi "pgometadata" vytvoří potřebná datová struktura, program se sám ukončí a je připraven k použití [12].

```
<?xml version="1.0" encoding="utf-8" ?>
<metadata>
  <server>
        <server-location>localhost</server-location>
        <database-name>pgometadata</database-name>
        <user>postgres</user>
        <password>heslo</password>
        </server>
    </metadata>
</metadata>
```

Obr. 19 Struktura souboru metadata.xml

6.2.2 Příprava datové kostky

Prvním krokem při práci s aplikací je zvolení záložky *Schema* z horní lišty a z ní následné vybrání možnosti *Select Schema*. V tomto okně je možné nastavit nebo upravit zdroje dat, které budou použity pro tvorbu datové kostky. Při nastavení nového zdroje dat je třeba v okně vyplnit IP adresu databázového serveru, v němž jsou data uložena, uživatelské jméno a heslo k databázovému serveru, název databáze s daty (je nutné dodržet malá a velká písmena), název tabulky, která obsahuje prostorová data a identifikátor jejich souřadnicového systému (SRID).

Server	127.0.0.1
User	postgres
Password	•••••
Database	Halenkovice
Мар	/home/pokus
Map SRID	4326

Obr. 20 Nastavení zdroje dat pro OLAP analýzu v aplikaci PostGeoOLAP

Po připojení se ke zvolenému zdroji dat následuje samotné vytvoření nové datové kostky, které je dostupné přes záložku *Schema* a možnost *Create cube*. V prvním okně se zvolí název nové kostky a minimální počet řádků, pro které bude prováděna agregace. Je-li zvoleno číslo 1, budou při zpracování datové kostky vypočítány všechny možné agregace. Jsou li k dispozici např. denní údaje a je

zvoleno číslo větší než 31, tak nebudou zpracovány měsíční agregace těchto údajů, ale pouze roční, přičemž měsíční budou vypočítávány až při OLAP analýze datové kostky. Tento mechanismus slouží ke zkrácení času nutného pro zpracování datové kostky.

V dalším okně se vybere tabulka faktů z dostupných tabulek v databázi a z ní atribut, který obsahuje hodnoty (fakta). Následuje výběr tabulek, které budou představovat v datové kostce dimenze. U každé z nich je možné pro jejich jednotlivé atributy nastavit úroveň hierarchie v rozmezí čísel 1-9. Přičemž číslice 9 představuje nejdetailnější úroveň a číslice 1 naopak tu nejobecnější. Tento proces se opakuje do té doby, než jsou nastaveny všechny potřebné dimenze a ukončí se stisknutím tlačítka *Cancel* v okně výběru tabulek dimenzí. Již jednou zvolené dimenze, tabulku faktů, stejně jako celou vytvořenou datovou kostku nelze v prostředí aplikace odstranit. Lze toho docílit pouze pomocí vymazání příslušných údajů z tabulky "pgometada" v prostředí databázového serveru.

V hlavním okně aplikace se po dvojím kliknutí na datový zdroj objeví dostupné datové kostky. Ve stromové struktuře jsou u každé uvedeny její tabulky se svými atributy. Přes pravé tlačítko myši je dostupných několik voleb. Pomocí volby *Add dimesion* lze do stávající kostky přidat další dimenzi, funkce *Add fact* pak umožňuje přidat tabulku faktů, ale pouze v případě, že ji kostka ještě neobsahuje (není povoleno, aby datová kostka obsahoval více než jednu). K úpravě stávající hierarchické struktury atributů dimenzí je určena možnost *Edit hierarchy*. Funkce *Add non-agregable dimension* slouží pro přidání dimenze, pro kterou nebudou vytvořeny agregace, do datové kostky. Tato funkce ale z neznámého důvodu nefunguje. Je-li taková dimenze zařazena, pak nelze datovou kostku analyzovat.

Poslední dvě volby *Process Cube* a *Analyze data* jsou nejdůležitější. První z nich v nadefinované kostce vytvoří agregace a připraví ji tak pro samotnou OLAP analýzu. Druhá zase spustí speciální okno, v němž uživatel nastavuje parametry analýzy a získává potřebné souhrnné údaje. Aplikace vyžaduje, aby byly všechny názvy atributů jednotlivých tabulek odlišné, jinak neumožní data analyzovat.

	📓 Create New Cube		
	Schema	Znecisteni	
	Cube name	Znecisteni	
	Minimum rows for aggregation		
		Cancel	
			-
🅌 Select Fact Table			X
Tables		Attributes	
Merici_stanice		kan hodnota	
aeometry columns		id id	
imis	C	A id inise	
spatial_ref_sys		id_stanice	
4			4
	Ok	Cancel	
Soloct Itom from F	ect Table		
East table attributes	let rable	Ni merical Trans	
id		> bodnota	1
id_cas			
id_imise id_stanice			
		<<<	
	Ok	Cancel	
Select Dimension			×
Tables	i	Attributes	
Merici_stanice		iden id	
geometry_columns		mes_cis	
imis spatial ref sys		>>> mesic	
tab_faktu			
		Cance	
😹 Select Hi	erarchy - Dimension: ca	s 🔀	
In the table a	bove, enter the hierarchy level	of each attribute in this dimension. Higher level	
attributes mu values, up to	st get lower values and attribut maximum of [9]. Several attribut	tes with smaller granularities must get bigger utes can share the same level in a dimension. Ex:	
StoreName [9), StoreID [9], StoreDistrict [8]], StoreCity [7].	
Name		Hierarchic Level	
den id		8	
mes_cis		7	
mesic		7	
rok		P	

Obr. 21 Postup vytvoření nové datové kostky v aplikaci PostGeoOLAP



Obr. 22 Hlavní okno aplikace PostGeoOLAP s funkcemi dostupnými pře pravé tlačítko myši

6.2.3 Možnosti nadstavby

Extenze pracuje pouze s relačními databázemi v hvězdicovém schématu, přičemž využívá technologii ROLAP. Počet hierarchických úrovní pro atributy dimenzí je omezen na 9, jak již bylo uvedeno výše. Nyní již k samotné OLAP analýze, která se odehrává v samostatném okně s názvem *Data Analysis*. Okno se skládá ze čtyř částí. V pravé horní jsou ve stromové struktuře zobrazeny všechny tabulky se svými atributy. Zde jsou také dostupné dvě funkce, a to po stisknutí pravého tlačítka myši na vybraném atributu.

První z nich *Criteria* umožňuje pomocí dialogového okna, specifikovat kritéria OLAP analýzy. Např. lze získat souhrnné hodnoty pouze pro zájmové období, oblast či sledovaný jev. Tato kritéria je možné použít v neomezeném množství, přičemž jsou mezi sebou navzájem spojena některým z dostupných logických operátorů (AND, OR, AND NOT, OR NOT). Nadefinování kritéria se provede výběrem logického operátoru, jednoho z porovnávacích operátorů (=, >, <, >=, <=, LIKE) a dvojím stisknutím levého tlačítka myši na zvolené hodnotě atributu. Po potvrzení tlačítkem OK se zvolené kritérium i všechna další objeví v levé dolní části okna, kde je lze případně upravit. Druhá funkce se jmenuje *Clausula WHERE* a má stejný účel jako předcházející s tím rozdílem, že kritérium není stanoveno pomocí dialogového okna, ale je vypsáno ručně (funkce tedy není omezena na existující hodnoty stávajících atributů jako předchozí).

() AND	O OR	O AND NOT	O OR NOT
'In order to a values at list a criterion, exec	Comparating dd a criterium to bove. If you do ute the steps o	Operator = o query, do > esire more t < of this frame >=	one of desired /alue or e time,
1997 1998 1999	iyan mun Anar	<= LIKE	
2000 2001 2002			
2003 2004			
		1 Innel	

Obr. 23 Uživatelské prostředí funkce Criteria

Atributy, které chce uživatel zahrnout do analýzy, se jednoduše přesunou pomocí dvojího stisknutí levého tlačítka myši do pravé spodní části. Jakmile je OLAP analýza pomocí atributů nadefinována, objeví se zde po stisknutí tlačítka *Execute* výsledky. Aplikace v tento okamžik předpokládá, že jsou všechny názvy tabulek i jejich atributů v databázi uvedeny malými písmeny, jinak dojde k chybě a výsledky se nezobrazí. Jedinou agregační funkcí, kterou tato aplikace disponuje, je suma hodnot (Sum), což její praktické využití velmi výrazně omezuje. Lze ji změnit pouze zásahem do zdrojového kódu, ale aplikace neobsahuje žádné uživatelské rozhraní, které by umožňovalo výběr funkce, tudíž by muselo být navíc nově vytvořeno.

Tlačítko *New Query* vymaže výsledky analýzy a umožní její nové zadání. Tlačítko *Show Displays* slouží dle autorů k zobrazení podkladových prostorových dat. V případě zatrhnutí políčka *Show Results on Map* by se měly výsledky OLAP analýzy zobrazit také v grafické podobě v mapovém poli v pravé horní části okna. Tuto funkci stejně jako *Show Displays* se ani přes konzultaci s autory však nepodařilo zprovoznit. Poněvadž ale výsledky analýzy, které jsou zobrazeny velmi nepřehledně, nelze v prostředí aplikace bohužel ani uložit, není tento nedostatek klíčový. A to z toho důvodu, že je nutné souhrnná data dostat do podoby, kdy by byla využitelná pro další zpracování v GIS systémech a kde mohou být také vizualizována. V praxi tak pro provedení OLAP analýzy v prostředí této nadstavby nejsou třeba prostorová data stejně jako v případě extenze OLAP for ArcGIS.

6.2.4 Integrace výsledků analýzy do GIS

Integrace výsledků je možná díky tomu, že je lze vybrat a zkopírovat do prostředí nějakého tabulkového procesoru. Jedinou podmínkou pro jeho výběr je, že musí být možné v jeho prostředí vytvořit kontingenční tabulku. V této práci byl pro tento úkon použit program Microsoft Excel 2003 v české lokalizaci. V něm bylo nutné pojmenovat jednotlivé sloupce s daty a na jejich základě vytvořit pomocí funkce *Kontingenční tabulka a graf* ze záložky *Data* výše zmíněnou kontingenční tabulku. Obsahuje-li sloupec souhrnných hodnot z analýzy desetinná čísla, pak je nutné pomocí možnosti *Nahradit* ze záložky *Úpravy*, změnit desetinný oddělovač z tečky na čárku. Názvy datových sloupců se objeví v okně *Seznam polí kontingenční tabulky*. Do řádkového pole je třeba umístit zástupce geografické složky (identifikátor, přes který se budou výsledky připojovat k prostorovým datům), do sloupcového pole se vloží sledovaný jev a do pole hodnot sloupec se souhrnnými údaji.

Poslední úpravy se provedou pomocí stisknutí pravého tlačítka myši na poli hodnot a zvolením možnosti *Nastavení pole*. Zde je třeba změnit hodnotu volby *Souhrn* (automaticky je zde nastavena hodnota "Počet") na "Součet". Pro každou kombinaci řádků a sloupců existuje pouze jedna hodnota, tudíž k žádnému reálnému sčítání nedojde a zobrazí se právě ta jediná (v tomto případě by tedy bylo možné použít i hodnoty "Maximum" a "Minimum" a výsledek by se nezměnil). V případě, že jsou výsledkem desetinná čísla, je ještě nutné upravit nastavení v nabídce *Číslo*, kde se vybere hodnota "Číslo" a zvolí odpovídající počet desetinných míst. Nyní již stačí jen stisknout tlačítko *Aktualizovat data* a výsledky již budou v podobě, kdy je lze připojit ke geografickým datům v GIS díky exportu do formátu DBF a jejich spojením přes identifikátor např. pomocí funkce *Join* v ArcGIS Desktop.

Stejně tak tento způsob zpracování předpokládá, že bude OLAP analýza v aplikaci PostGeoOLAP úžeji specifikována, poněvadž sloupce výsledné tabulky mohou sledovat pouze jednu zájmovou oblast, což je rozdíl oproti extenzi OLAP for ArcGIS, která toto umožňuje. Např. je možné zobrazit souhrnné hodnoty pro měsíce v roce, ale rok a co má být sledováno je nutné specifikovat pomocí kritérií v extenzi. Druhou možností je tuto specifikaci provést až v kontingenční tabulce.

sum_hodnota		id_stan			rok		
3987.50		8			2004		
5790.40		13			2007		
345.40		8			2007		
5713.30		13			2004		
965.89		1			2000		
5234.40		13			2006		
3248.77		4			2001		
.0522.50		8			2006		
3537.32		9			2001		
3098.14		5	_		2001		1/0
2598.71		7			2001		~
Součet z Hodnota	Rok 🔽						
ld_stanice	1997	1998	1 <mark>9</mark> 99	2000	2001	2002	2003
	8461,57	10977,24	7566,79	7965,89	7811,81	9160,35	9885,80
1	2 11209,63	9378,87	8307,44	8248,42	9068,27	9205,01	9902,30
	3 4311,19	3605,83	3030,66	2906,99	2852,06	3022,48	3428,90
	10159,19	9087,05	8561,83	8275,40	8248,77	8395,32	8693,00
	5 9995,60	7837,00	7572,99	6967,76	8098,14	7605,82	8491,80
e e	6 7938,87	6675,51	5683,83	5960,34	6996,30	6247,83	7227,00
	7 3346,25	2576,65	2586,27	2300,63	2598,71	2430,56	2950,30
8	3 10638,41	9037,12	9035,46	8143,88	8768,52	8964,64	9718,70
<u>(</u>	10295,47	8653,73	7829,23	7607,87	8537,32	8300,68	8880,50
10	7857,03	6454,04	5482,34	5556,52	6568,34	6581,19	7277,00
1.	11505,00	9566,00	9169,00	8593,00	9742,11	9446,55	9833,50
12	2 10473,00	8673,22	8123,05	8364,67	8555,32	9083,09	9702,00
13	6230,70	5375,64	5241,59	5345,76	5798,71	6092,17	6012,50
1.	10028,28	9011,05	8153,72	8516,53	9224,44	9466,31	10387,30
Celkový součet	122450.19	106908.95	96344,20	94753.66	102868.82	104002.00	112390.60

Obr. 24 Změna formátu výsledků OLAP analýzy v aplikaci PostGeoOLAP do tvaru pro připojení k prostorovým datům v GIS

7. VYUŽITÍ TECHNOLOGIE OLAP V GIS

7.1 Vizualizace

Jak již bylo řečeno v předchozí kapitole, výstupem z OLAP analýzy je tabulka s daty, kterou je potřeba nejprve napojit na prostorová data. Ty je pak již možné prostorově prezentovat formou mapy. V případě využití nadstavby OLAP for ArcGIS lze předpokládat, že data budou vizualizována v prostředí aplikace ArcGIS Desktop. Při použití druhého řešení záleží na výběru uživatele. V každém případě možnosti vizualizace jsou dány a zároveň omezeny pouze vizualizačními a kartografickými možnostmi použité aplikace a typem prostorových dat (bod, linie, polygon). Kromě jednoduchých vizualizačních metod (bod – velikost, barva, linie – tloušťka, barva, polygon - pseudokartogram) nabízí povaha dat, v případě existence časové dimenze, využití metody kartodiagramu, vytvoření série map pro jednotlivé časové úseky či prezentaci výsledků formou animace. Kromě mapové formy je samozřejmě možné data prezentovat neprostorově grafem nebo tabulkou či kombinací všech výše zmíněných metod.



Obr. 25 Průměrné měsíční znečištění ovzduší imisemi v µg SO2 na Ostravsku v letech 1997 - 2007

7.2 Analýza

Největším přínos technologie OLAP pro GIS jako takový je v získávání podkladových dat určených pro další analýzy, přičemž tento potenciál není dosud v praxi příliš využíván. Je snadné si představit automatický systém zaznamenávající hodnoty vybraných měřených charakteristik na experimentálním území (jako je např. měření teplot a vlhkostí půdy v Halenkovicích). Naměřené hodnoty jsou poté převáděny do databáze, kde je na nich provedena OLAP analýza a agregované hodnoty slouží jako vstup do výpočetních modelů. Zároveň jsou jejich prostorové reprezentace využívány k prostorovým analýzám, v jejichž rámci jsou kombinovány s dalšími daty a je zkoumána jejich vzájemná prostorová závislost. Příkladem může být využití agregovaných dat znečištění ovzduší spolu s údaji o směru větru a srážkách při zkoumání vlivu znečištění ovzduší na lesní porosty.

Poněvadž hlavní využití OLAP analýz je v současnosti ve zpracování neprostorových dat a slouží ke sledování ekonomických výsledků velkých společností, je vhodné uvést příklad, jak tuto oblast inovovat použitím této analýzy na prostorových datech. V oblasti geomarketingu je zaveden termín "mozaika", který představuje rozložení jednotlivých oblastí celého prostoru (např. sčítací okrsky) pomocí statistických metod do několika kategorií, podle demografického složení obyvatel [5]. Tímto způsobem je podáván marketingový obraz o této oblasti (např. manažerská sídla, panelákoví dělníci, smíšený venkov, atd.). Existuje-li poté společnost, která provozuje síť obchodů a má evidenci svých zákazníků, je velmi jednoduché podle adresy tyto zákazníky v "mozaice" lokalizovat a pomocí prostorových analýz a souhrnných dat o obchodních výsledcích v jednotlivých pobočkách zjistit, na jakou kategorii obyvatelstva zaměřit nejlépe novou marketingovou kampaň či jakým směrem upravit nabídku sortimentu.

Neprostorově pak lze data z OLAP analýz statisticky využít pro zjišťování nejrůznějších korelací, regresí, trendů, atd.



Obr. 26 Geomarketingová "mozaika" [5]

8. NÁVRH DALŠÍCH MOŽNOSTÍ

8.1 Využití prostorového dotazu

Jak již bylo v této práci zmíněno, ani jedna z OLAP nadstaveb nevyžaduje nutně pro provedení analýzy prostorová data. V datové kostce ale být mohou. Jako jejich velmi užitečné využití se jeví zařazení prostorového dotazu do průběhu analýzy, které může být v některých případech velmi příhodné. Především za předpokladu že jsou data v databázi získána z mnoha míst na relativně velkém území. Jako příklad je možné použít stanice měřící znečištění ovzduší, které jsou rozmístěny po celém území republiky, a jejich počet dosahuje několika desítek. V případě, že by se řešila analýza vlivu znečištění ovzduší na určité místo či oblast, bylo by vhodné nespecifikovat OLAP analýzu pouze vybráním zájmových atributů, ale také prostorovým dotazem, jakým může být "Vyber všechny stanice do 15 km od daného místa" či "Vyber všechny stanice, které jsou v severním směru od daného místa a vzdálené maximálně 30 km (protože je známo, že v místě převládají severní větry)". Těchto příkladů by bylo možné najít mnoho, přičemž způsob využití závisí pouze na množství, kvalitě a typu dat, které jsou k dispozici. V podstatě existují tři možnosti, jak prostorový dotaz do OLAP technologie integrovat.

8.1.1 Prostorový dotaz v rámci OLAP analýzy

První alternativou je doprogramování této možnosti do aplikace, v níž se samotná analýza provádí. Ve výsledku by systém fungoval tak, že zatímco dosud uživatel vybíral pouze jaké atributy, případně pouze jejich určité hodnoty, budou vstupovat do analýzy, přibude možnost definovat vstupní objekty z prostorové dimenze pomocí prostorového dotazu. Ať už interaktivně v grafickém prostředí nebo pomocí prostorových vztahů mezi vrstvami (průnik, obsažení, vzdálenost od objektů atd.). Tato varianta je technicky velmi náročná a předpokládá existenci databázového serveru, který umožňuje nad prostorovými daty tyto dotazy provádět či GIS, který dokáže výsledek dotazu databázovému serveru předat a schopný programátorský tým. Takové servery v současné době již existují. Jedná se např. o MS SQL Server 2008, Oracle Spatial atd. Nabídka jejich prostorových dotazů ale není tak široká jako v profesionální GIS aplikaci.

Tato varianta má ještě jedno úskalí. V případě, že by taková aplikace pracovala na podobném principu jako v této práci představené extenze, bylo by nutné všechna prostorová data potřebná k provedení dotazu v databázi umístit do datové kostky, což je nereálné. Z toho důvodu by bylo třeba naprogramovat úplně nové rozhraní, které by umožnilo dotaz provést s daty z libovolných externích zdrojů (geodatabáze, shapefile atd.), zatímco v datové kostce by byla umístěna pouze prostorová dimenze. Prakticky by tak vznikla zcela jiná aplikace kombinující v sobě některé vlastnosti klasických GIS programů a specializovaných OLAP aplikací, která by ovšem spolupracovala pouze s jediným konkrétním databázovým serverem. Kompatibilita s více z nich je z důvodu jejich velké vzájemné odlišnosti prakticky nemožná. Velkou výhodou tohoto řešení je fakt, že se fyzicky nezasahuje do dat v datové kostce a že by bylo možné výsledky analýzy zobrazit i v jednoduché prohlížečce (jedinou nepostradatelnou funkcí by bylo napojení tabulky ke geometrii), která by mohla být i přímo integrovanou součástí aplikace. Na podobném principu byl na fakultě elektrotechnické ČVUT vytvořen systém GOLAP [6].

8.1.2 Prostorový dotaz před OLAP analýzou

Druhá možnost předpokládá, že prostorový dotaz je zpracován před OLAP analýzou. Ta je pak provedena až na vybraných datech. Výhodou tohoto řešení oproti předchozímu je snadné použití velkého množství externích dat při provádění prostorového dotazu a široké možnosti variant dotazu. Je několik způsobů jak tuto variantu prakticky realizovat. Velmi také záleží na tom, jaký je použit databázový server, aplikace pro OLAP analýzu a GIS systém. Zde budou uvedena praktická řešení pouze pro OLAP nadstavbu aplikace ArcGIS Desktop. Ta jsou teoreticky použitelná i pro druhé řešení, pouze je nutný odlišný způsob implementace s využitím jiného programového vybavení.



Obr. 27 Obecný princip řešení využití prostorového dotazu před OLAP analýzou

• V této práci je použito propojení aplikace ArcGIS Desktop 9.3 s databázovým serverem Microsoft SQL Server 2005 přes rozhraní ArcSDE. Tudíž nejjednodušší varianta představuje zobrazení prostorové dimenze v prostředí ArcGIS Desktop, v němž se jakýmkoliv způsobem provede prostorový dotaz, a jemu nevyhovující prostorové objekty se snadno vymažou. Aby mohl být jakýkoliv objekt v geodatabázi upravován je nezbytné spustit ArcCatalog a v něm na tomto objektu stisknout pravé tlačítko myši a zvolit možnost *Register As Versioned*. Poněvadž se jedná o fyzický zásah do dat v datové kostce, je nezbytné mít tuto dimenzi zálohovanou a po provedení analýzy s upravenou dimenzí ji znovu uvést do původní kompletní podoby.

• Druhá varianta je podobná předcházející. Jen se nezasahuje do prostorové dimenze, ale přímo do tabulky faktů, odkud jsou vymazány záznamy, které nesouvisí s objekty vybranými prostorovým dotazem z této dimenze. Pro toto řešení byl vytvořen skript, který výběr a smazání nepotřebných záznamů automaticky provádí. Skript byl napsán pomocí skriptovacího jazyka Python 2.5.1, který je součástí instalace aplikace ArcGIS Desktop 9.3, z jehož prostředí je také spustitelný. Pro ladění skriptu byla využita aplikace IDLE 1.2.1 (Python's Integrated DeveLopment Environment), která je součástí instalace skriptovacího jazyka.

Poněvadž zdrojový kód obsahuje několik příkazů *GetParameterAsText()*, které umožňují uživatelské nastavení některých vstupních parametrů, je nutné při přidávání skriptu do ToolBoxu v prostředí aplikace ArcMap provést jejich definici. Celkem jsou čtyři, přičemž všechny jsou povinné. Pro vrstvu tvořenou daty z prostorové dimenze je nastaven datový typ "Feature Layer", pro tabulku faktů "Table View" a pro jejich atributová pole "Field". U atributových polí je nastaveno, z jakých dat má být získán jejich seznam.

	Display Name		Data Type			
	Vrstva prostor Atributové pol Tabulka faktů	ové d e vrst	Feature Layer Field Table View		≂f	
	Atributové pol	e tab	Field		-	
	J					
	Click any paramete	er above t	to see its properties below.			
	Parameter Prope	rties				
	Property	Value				
	Type	Require	d			
	MultiValue	No				
	Default					
	Filter	None				
	Obtained from	Tabulka	_faktů			
	To add a new para	meter, ty	pe the name into an empty row in t	he		
	To add a new para name column, click then edit the Parar	meter, ty in the Da neter Pro	rpe the name into an empty row in t ta Type column to choose a data ty perties.	he pe,		
	To add a new para name column, click then edit the Parar	meter, ty in the Da neter Pro	rpe the name into an empty row in t ta Type column to choose a data ty perties.	he pe,	žít	
	To add a new para name column, click then edit the Parar	meter, ty in the Da neter Pro	pe the name into an empty row in to the tart type column to choose a data typerties.	he pe,	80	
2	To add a new para name column, click then edit the Parar	meter, ty in the Da neter Pro	rpe the name into an empty row in to ta Type column to choose a data ty perties.	he pe, Pou	žít	
prosto	To add a new para name column, click then edit the Parar	meter, ty in the Da neter Pro	rpe the name into an empty row in t ta Type column to choose a data ty perties. DK Stomo	he pe, Pou	žít Vky	
prostor steni.Di	To add a new para name column, click then edit the Parar	meter, ty in the Da meter Pro	rpe the name into an empty row in t ta Type column to choose a data ty perties. DK Stomo nocí prostov vého dotazu vybrán	he pe, Pou / Zájmové pr		
) prostoi steni.Di	To add a new para name column, click then edit the Parar rové dimenze, z které BO.Merici_stanice le vrstvy prostorové c	meter, ty in the Da meter Pro jsou porr	rpe the name into an empty row in t ta Type column to choose a data ty perties.	he pe, Pou	žít vky	
prosto steni.Dľ tové po	To add a new para name column, click then edit the Parar rové dimenze, z které BO.Merici_stanice le vrstvy prostorové c	meter, ty in the Da neter Pro jsou pon dimenze c	rpe the name into an empty row in t ta Type column to choose a data ty perties. DK Storno nocí prostov vého dotazu vybrán obsahující primární Klíč	he pe, Pou / 2ájmové pr		
prosto steni.Dí tové po a faktů	To add a new para name column, click then edit the Parar rové dimenze, z které BO.Merici_stanice le vrstvy prostorové c	meter, ty in the Da neter Pro jsou pon dimenze c	rpe the name into an empty row in t ta Type column to choose a data ty perties.	he pe, Pou	sn vky	
prosto steni.Dí tové po a faktů steni.Dí	To add a new para name column, click, then edit the Parar rové dimenze, z které BO.Merici_stanice le vrstvy prostorové c	meter, ty in the Da neter Pro jsou por dimenze c	rpe the name into an empty row in t ta Type column to choose a data ty perties. DK Storno nocí prostor vého dotazu vybrány obsahující primární klíč	he pe, Pou	sn vky	

Obr. 28 Nastavení python skriptu v aplikaci ArcMap a jeho dialogové okno pro zadávání parametrů

```
import arcgisscripting, sys, os
gp = arcgisscripting.create(9.3)
gp.OverwriteOutput = "True"
vrstva = gp.GetParameterAsText(0)
pole1 = gp.GetParameterAsText(1)
gp.MakeFeatureLayer(vrstva,"lyr")
rows = gp.SearchCursor("lyr")
row = rows.Next()
seznam = []
while row:
            seznam.append(row.GetValue(pole1))
            row = rows.Next()
tabulka = gp.GetParameterAsText(2)
pole2 = gp.GetParameterAsText(3)
radky = gp.UpdateCursor(tabulka)
radek = radky.Next()
kontrola = 0
while radek:
    jednicka = radek.GetValue(pole2)
    for dvojka in seznam:
        if jednicka == dvojka:
            kontrola = 1
    if kontrola <> 1:
       radky.DeleteRow(radek)
    kontrola = 0
    radek = radky.Next()
del radky
# obdobným způsobem jsou smazány všechny použité proměnné
```

Z multidimenzionální databáze se v aplikaci ArcMap zobrazí vrstva tvořená z dat prostorové dimenze a tabulka faktů. Nad prostorovou dimenzí se provede požadovaný dotaz, ke kterému je možné využít jakoukoliv jinou externí vrstvu či data. Jakmile jsou pomocí dotazu označeny vybrané objekty z prostorové dimenze, spustí se samotný skript. V něm je nutné nastavit vrstvu prostorové dimenze s vybranými objekty a její atributové pole obsahující primární klíč. Dále se zvolí tabulka faktů a z ní atributové pole obsahující cizí klíč vrstvy prostorové dimenze. Na základě tohoto nastavení skript provede požadovanou operaci. Tímto procesem dojde ke značnému zmenšení objemu dat v tabulce faktů a tím ke zrychlení tvorby datové kostky, souvisejících agregací předcházející OLAP analýze a i samotné analýzy. Tato varianta předpokládá zálohování tabulky faktů, protože smazaná data jsou nevratně ztracena.

Obr. 29 Python skript určený k mazání záznamů v tabulce faktů z prostředí ArcGIS Desktop



Obr. 30 Diagram integrace prostorového dotazu do OLAP technologie pomocí python skriptu

Tento způsob řešení je ale relativně časově náročný. Doba, za kterou je skript proveden, závisí na počtu záznamů v tabulce faktů a těch, které mají být smazány, přičemž tato závislost není lineární. Se zvyšujícím se počtem mazaných záznamů se tato doba prudce zvyšuje. Např. je-li mazáno 650 000 záznamů z milionu je doba průběhu skriptu zhruba 10 minut, zatímco při mazání 900 000 záznamů už 40 minut. U tabulky faktů, která obsahovala deset milionů záznamů a mělo v ní být smazáno opět 900 000 záznamů, pak již skript pravděpodobně z hardwarových důvodů neproběhl vůbec.

Výhoda této varianty oproti předchozí je v tom, že omezení tabulky faktů snižuje dobu nutnou ke zpracování datové kostky před OLAP analýzou a velikost místa na disku, kterou kostka vyžaduje. Je tedy možné vytvořit datovou kostku, ve které bude umístěn pouze výřez z tabulky faktů, který se bude měnit v závislosti na požadovaných výsledcích. Poněvadž ale skript nedokáže zpracovat tabulku faktů s neomezeným počtem záznamů jsou možnosti této varianty značně omezené.

• Třetí varianta je založena také na mazání záznamů z tabulky faktů, které je ale tentokrát provedeno v prostředí databázového serveru pomocí příslušného SQL dotazu, jenž je vygenerován skriptem vytvořeným opět v jazyce Python. Nastavení skriptu v prostředí ArcGIS Desktop je totožné jako v předchozím případě pouze s tím rozdílem, že přibývá parametr "výstup" (datový typ je nastaven na "Text File" a proměnná *Direction* na hodnotu "Output"). Použití skriptu je také stejné. Uživatel musí navíc zvolit pouze soubor ve formátu TXT, do kterého se požadovaný dotaz vytvoří. Poté již stačí jen spustit aplikaci pro správu databázové komponenty serveru Microsoft SQL Server 2005, vygenerovaný dotaz zkopírovat a použít jej na tabulku faktů.

Doba nutná pro průběh skriptu je závislá pouze na počtu záznamů v tabulce faktů, zatímco počet mazaných záznamů ovlivňuje dobu, za kterou je zpracován vygenerovaný SQL dotaz v prostředí databázového serveru. Celkově je toto řešení mnohem rychlejší než v předcházejícím případě. Vymazání jakéhokoliv počtu záznamů z tabulky faktů obsahující milion záznamů trvá méně než minutu. U tabulky faktů s deseti miliony záznamů pak tato operace zabere 5 - 7 minut. Výhody tohoto řešení jsou tedy díky stejné logice obdobné jako v předcházejícím případě, přičemž navíc překonává jeho nedostatky. Z tohoto důvodu se jeví jeho využití před OLAP analýzou jako relativně nejvhodnější.

```
import arcgisscripting, sys, os
                                       final = []
gp = arcgisscripting.create(9.3)
                                       kontrola = 0
gp.OverwriteOutput = "True"
                                       for srovnanil in seznamek:
                                           for srovnani2 in seznam:
vstup = gp.GetParameterAsText(0)
                                               if srovnani1 == srovnani2:
vrstva = gp.GetParameterAsText(1)
                                                   kontrola = 1
pole1 = gp.GetParameterAsText(2)
                                           if kontrola <> 1:
gp.MakeFeatureLayer(vrstva,"lyr")
                                               final.append(srovnani1)
rows = gp.SearchCursor("lyr")
                                           kontrola = 0
row = rows.Next()
seznam = []
                                       f=open(vstup, 'w')
                                        zacatek = 'DELETE FROM '+tabulka+' WHERE \n'
while row:
                                        a = str(zacatek)
    seznam.append(row.GetValue(pole1)) f.write(a)
   row = rows.Next()
                                       for x in range(len(final)-1):
tabulka = gp.GetParameterAsText(3)
                                           hodnota = final[x]
pole2 = gp.GetParameterAsText(4)
                                           hodnota str = str(hodnota)
pole2 str = str(pole2)
                                           prostredek = pole2 str+'='+hodnota str+' OR\n'
radky = gp.SearchCursor(tabulka)
                                           b = str(prostredek)
radek = radky.Next()
                                           f.write(b)
seznamek = set()
                                       konec = pole2 str+'='+str(final[len(final)-1])
while radek:
                                        c = str(konec)
    seznamek.add(radek.GetValue(pole2)) f.write(c)
    radek = radky.Next()
                                        f.close()
                                        del radky
                                       # obdobným způsobem jsou smazány všechny použité proměnné
```

Obr. 31 Python skript pro vytvoření SQL dotazu určeného k mazání záznamů z tabulky faktů

8.1.3 Prostorový dotaz po OLAP analýze

Poslední varianta používá prostorový dotaz až po získání výsledků analýzy přímo v prostředí GIS aplikace. Ty se ve formě tabulky připojí ke všem prostorovým objektům. V případě, že je objektů málo, není nijak obtížné několik zájmových z nich vybrat v tabulce či interaktivně graficky a z nich vytvořit novou vrstvu či pouze vymazat nepotřebné. V opačném případě by tato metoda byla velmi neefektivní. Pak je již nutné použít samotný prostorový dotaz z nabídky, kterou konkrétní GIS aplikace umožňuje. Toto řešení je z výše zmíněných nejjednodušší na provedení, což je jeho hlavní výhodou. Jedinou teoretickou nevýhodou je nutnost vlastnit nějakou robustní GIS aplikaci, která umožní širokou škálu prostorových dotazů.

8.2 Prostorový OLAP (Spatial OLAP)

Mrg inz. Jędrzej Gąsiorowski ve své práci představuje myšlenku tzv. prostorové varianty OLAP [3]. Dvě dimenze datové kostky jsou v tomto případě rezervovány pro dvourozměrné prostorové vrstvy jak v rastrové tak vektorové podobě. Ostatní dimenze mohou být představovány např. časovou složkou, úrovní generalizace dat atd. (struktura viz Obr. 32) Základem této myšlenky je existence časové řady prostorových vrstev, přičemž hodnoty v nich obsažené by bylo možné stejně jako v běžném OLAP agregovat a získávat tak přehlednější údaje.

Tabulka faktů by v tomto případě neobsahovala žádné hodnoty, ale pouze identifikátory vrstvy a s ní souvisejících dimenzí, protože všechny hodnoty už jsou obsaženy v prostorových vrstvách. Uživatel by pak měl možnost určit zájmový typ vrstev, v případě jejich vektorové formy i jejich jakýkoliv atribut, časový úsek za jaký chce získat agregované hodnoty, podrobnost dat pomocí úrovně generalizace atd. Tudíž prostorová OLAP analýza s rastrovými daty množství ozonu v atmosféře ve formě družicových snímků s denním intervalem by mohla vypadat např. takto: "Chci znát měsíční průměry hodnot ozonu v rozlišení 1km na pixel".

Aby mohla být tato technologie zrealizována, bylo by třeba vytvořit specializovanou aplikaci, která by v sobě kombinovala možnosti GIS a OLAP. Musela by s prostorovými vrstvami dokázat provádět operace mapové algebry a zároveň výslednou vrstvu také vizualizovat a uložit do standardizovaného formátu. Uživatel by také měl mít možnost si interaktivně zvolit prostorový výřez, z něhož by se výsledné agregované hodnoty počítaly.

Tato technologie má ale i svá omezení. Prostorové vrstvy sledující jeden jev by musely mít jednotný prostorový rozsah a rastrové vrstvy navíc ještě ve stejné úrovni generalizace stejnou velikost pixelu, aby bylo možné korektně provést operace mapové algebry. Dalším problémem by také mohla být velikost místa na disku, kterou by v rámci databáze vyžadovaly prostorové vrstvy. Při jejich velkém množství, které je např. díky pravidelnému družicovému snímkování celého povrchu Země reálné, by mohla dosáhnout obrovských hodnot.



Obr. 32 Datová kostka určená pro SOLAP [3]

9. DISKUZE

Tato práce by ve výsledku měla přinést odpověď na to, zda může technologie OLAP přinést nové možnosti do prostředí GIS a vyplatí-li se ji vůbec používat. Dá se říct, že ano. Její největší přínos se nachází ve zpracování velkého množství nepřehledných údajů do podoby lépe interpretovatelných a zpracovatelných souhrnných hodnot, které pak mohou být v prostředí GIS vizualizovány a dále analyzovány.

Dalším důležitým faktorem této problematiky je určení hranice, kdy se data již vyplatí zpracovávat pomocí OLAP analýz a kdy k tomu stačí použít pouze běžný tabulkový procesor či statistický program. Tuto hranici není možné vymezit ostře. Záleží na množství, povaze dat a hlavně na množství agregovaných údajů, které je z nich potřeba získat. Lze ale uvést několik faktů, které by měl řešitel projektu vzít v úvahu za předpokladu, že chce OLAP technologii úspěšně využít:

• Fáze přípravy dat je časově velmi náročná. Například úprava dat a plnění databáze vlhkostí a teplot půdy trvala skoro 24 hodin čistého času s tím, že výsledná tabulka faktů obsahuje pouze něco přes 350 000 záznamů, přitom je v praxi běžné, že se toto číslo pohybuje v řádech desítek milionů. Jsou-li data zaznamenávána automaticky (např. nějakou měřící stanicí), pak už by se měla průběžně ukládat přímo do databáze ve formě vyžadované pro OLAP analýzu. Tento způsob zpracování dat je z časového hlediska naprosto ideální. Jsou-li naopak data získána už v nějaké podobě z externího zdroje, pak je nutné, aby řešitel vytvořil mechanismus, který automaticky provede převod dat do požadované formy. Je ale třeba předpokládat, že tato operace si při velkém množství dat vyžádá, jak je již zmíněno výše, nemalý časový prostor, který je samozřejmě přímo závislý také na výkonu počítače, jenž tuto operace provádí.

• Druhým úskalím je fáze výpočtu předběžných agregací, při zpracování datové kostky před samotnou OLAP analýzou, která může ve výsledku trvat podle množství dat a agregací i několik hodin. Je-li projekt zpracováván s ohraničeným množstvím dat (projekt má přesně dané období, kdy jsou data získávána a poté jsou již jen analyzována), pak je možné tuto skutečnost zanedbat, protože tato fáze proběhne pouze jednou. Poté je datová kostka pouze analyzována, což už je relativně rychlá operace. Je-li ale třeba analyzovat kontinuálně přibývající data (např. data z automatické stanice), pak je vhodné vytvořit mechanismus, který zajistí, že tato fáze proběhne jednou za daný časový okamžik a výsledky analýzy se automaticky či

manuálně uloží do samostatné databáze určené k jejich následnému porovnávání. Tímto způsobem je možné datovou kostku zpracovat například jednou denně přes noc a ráno mít již aktuální výsledky.

V práci se vyskytl problém se spojením extenze OLAP for ArcGIS a aplikace ArcGIS Desktop propojené s databázovým serverem Microsoft SQL Server 2005 pomocí technologie ArcSDE do jednoho funkčního celku. Tato skutečnost byla způsobena především nutností instalace Service Pack 2 pro ArcGIS Desktop 9.2, kterou se i přes studium problému na odborných fórech nepodařilo realizovat. Problém byl pravděpodobně způsoben instalačním médiem aplikace ArcGIS Desktop 9.2, které bylo k dispozici, protože u aplikace nainstalované z odlišného média proběhla tato operace v pořádku. Přes všechny tyto problémy se jeví použití této varianty jako nejlepší řešení, ale úspěšně je možné využít i nižší verze programu. V případě že by toto řešení mělo být nasazeno v rámci zpracování projektu, je nutné vzít tuto problematiku v úvahu před začátkem prací a najít optimálně fungující alternativu. Nadstavba OLAP for ArcGIS nebyla pro aktuální verzi ArcGIS Desktop 9.3 potvrzena a její uvedení není tedy vůbec jisté. Z tohoto důvodu byly skripty v jazyce Python vytvořené v rámci této práce přepracovány i do ostatních variant aplikace ArcGIS Desktop řady 9 a jsou uvedeny v digitální příloze.

Jak vyplývá z kapitoly o zkoumání možností extenzí, pro zpracování OLAP analýzy nejsou nezbytně třeba žádná prostorová data. Zde se tedy nabízí otázka, čím jsou tyto nadstavby lepší než komponenty zpracovávající tento typ analýz obsažené v některých databázových serverech. Odpověď zní, že prakticky ničím. Nadstavba OLAP for ArcGIS přináší pouze komfortní napojení výsledku analýzy na prostorová data, bez nutnosti jakýchkoliv dalších úprav. Tato výhoda je ale kompenzována vysokými náklady na pořízení aplikace ArcGIS Desktop, kterou extenze pro svoji funkčnost vyžaduje.

Pro řešení PostGeoOLAP hovoří naopak jeho finanční dostupnost, poněvadž jsou všechny jeho komponenty typu "open source" a tudíž zdarma ke stažení. Nevýhodou pak je zdlouhavá úprava výsledků analýzy do podoby použitelné v GIS systémech a jediná agregační funkce Sum, která využitelnost tohoto systému v praxi degraduje na minimum. Pokud by autoři tyto nedostatky odstranili nebo se našel programátor schopný uzpůsobit zdrojový kód nadstavby, pak by toto řešení bylo velmi zajímavou alternativou.

V současnosti se jako nejlepší varianta jeví právě zpracování OLAP analýz v prostředí některého z databázových serverů, jež to umožňují, vykopírování výsledků do prostředí tabulkového procesoru a jejich následné napojení na prostorová data v prostředí GIS. Odpadá nutnost zakoupení drahého profesionálního GIS nástroje, poněvadž je možné napojení dat, prostorové analýzy a kartografické výstupy zpracovat v některém z nekomerčních řešení. V případě Microsoft SQL Serveru 2005 s komponentou Analysis Services a aplikací SQL Server Business Intelligence Development Studio, jež byly použity v této práci, lze říci, že jako celek nabízejí i daleko více možností, co se týče zpracování OLAP analýz, než samotná extenze OLAP for ArcGIS, jež tento databázový server také využívá. Ta totiž nedokáže využít funkcionality programu SQL Server Business Intelligence Development Studio, který například umožňuje navíc definovat vlastní nové agregační funkce pomocí specifického skriptovacího jazyka, z výsledků odvozovat další hodnoty a mnoho dalších operací.

Je nepopiratelným faktem, že existuje prostor pro další vývoj spolupráce OLAP technologie a GIS viz kapitola 8. Velkou perspektivu má vytvoření systému kombinujícího v sobě možnosti GIS a OLAP, který by bylo možné využít pro zrychlení výpočtu náročných prostorových analýz. Tyto systémy jsou ale zatím pouze experimentální a jejich vývoj je velmi náročný. Mnohem jednodušší cestou, jak integrovat prostorový dotaz do OLAP analýzy, je použít jej před jejím zpracováním. K tomuto účelu by bylo třeba upravit stávající extenzi OLAP for ArcGIS nebo vytvořit zcela novou. Tento přístup totiž předpokládá spolupráci GIS aplikace a konkrétního databázového serveru. Nadstavba by zpracovala výsledek prostorového dotazu v GIS prostředí a předala jej databázovému serveru, který by jej použil pro úpravu dotazu zpracovávajícího OLAP analýzu (v případě Microsoft SQL Serveru 2005 by se jednalo o úpravu MDX dotazu). Toto řešení je založeno na podobném principu, jako je představeno v této práci v kapitole 8.1.2. Výhodou oproti němu je, že by nedocházelo k fyzickému zásahu do obsahu datové kostky. Dokud ale nebude takové programové vybavení k dispozici, zůstane uvedený fyzický zásah pro integraci prostorového dotazu nezbytný.

10. ZÁVĚR

V této magisterské práci byl zkoumán přínos OLAP technologie pro úlohy zpracovávané v GIS. Nejprve byly zjišťovány a popisovány možnosti dvou OLAP nadstaveb, a to OLAP for ArcGIS aplikace ArcGIS Desktop a PostGeoOLAP řešení PostgreSQL/PostGIS. Pro tento účel byly vytvořeny dvě zkušební datové kostky, jejichž řezy byly uvedeny v digitální příloze.

Aby mohly být získány výsledky OLAP analýz z těchto extenzí, není třeba, aby analyzovaná datová kostka obsahovala prostorová data, protože na ta jsou výsledky napojovány až v prostředí GIS. Z tohoto důvodu oproti provedení OLAP analýzy v prostředí databázového serveru, který to umožňuje, nepřináší nadstavby prakticky žádnou další funkcionalitu a neexistuje tedy podstatný důvod proč je používat. Navíc databázové servery často poskytují širší možnosti provedení OLAP analýzy (např. Microsoft SQL Server 2005).

OLAP technologie se jeví jako velmi účinný nástroj pro zpracování velkého množství multidimenzionálních dat, se kterými se v GIS pracuje velmi často. Získané agregované hodnoty pak mohou být pomocí GIS vizualizovány nebo dále analyzovány ať už prostorově či neprostorově. V práci byla diskutována problematika způsobu úspěšného nasazení OLAP technologie do průběhu zpracování reálného projektu.

V závěru pak byly prezentovány možnosti dalšího vývoje spolupráce OLAP technologie a GIS, a to zejména různé formy implementace prostorového dotazu do průběhu analýzy, které byly dále rozebrány v diskusní části práce. Byla zde také nastíněna myšlenka využití OLAP analýz pro získání souhrnných hodnot z prostorových vrstev ve vektorové či rastrové podobě namísto dat v tabelární formě.

11. POUŽITÉ ZDROJE

- [1] Accessing OLAP using ASP.NET [online].
 Poslední revize 2004-06-23 [cit. 2009-03-26]. Dostupné z: http://www.aspfree.com/c/a/MS-SQL-Server/Accessing-OLAP-using-ASP-dot-NET/
- [2] Dobešová, Z.: *Databázové systémy v GIS*. Vydavatelství Univerzity Palackého, Olomouc, 2004, 76 s., ISBN 80-244-0891-0.
- [3] Gasiorowski, J.: Analytical Ttools for Business Inteligence in Spatial Databases. Annals of Geomatics 2008, Volume VI., Number 1. Warsaw, Polish association for spatial information, s. 81-90., ISSN 1731-5522.
- [4] Geletič, J.: Charakteristika přírodního prostředí modelové lokality Halenkovice na základě analýzy časových řad. Bakalářská práce, Katedra geoinformatiky PřF UP, 2008.
- [5] Geomarketing: Efektivní segmentace trhu a zákazníků [online].
 Poslední revize 2006-10-11 [cit. 2009-03-28]. Dostupné z:
 http://www.admaz.cz/konference/konf_2006_10_11/download/05-kucera.pdf>
- [6] Kouba, Z.: On-line analýza geografické informace. České vysoké učení technické, Praha, 2003, 26 s., ISBN 80-01-02907-7.
- [7] Lacko, L.: *Datové sklady, analýzy OLAP a dolování dat*. Computer Press, Brno, 2003, 486 s., ISBN 80-7226-969-0.
- [8] Mařík, V., Štěpánková, O., Lažanský, J. a kolektiv: *Umělá inteligence (4)*.
 Academia, Praha, 2002, 475 s., ISBN 80-200-1044-0.
- [9] Microsoft SQL Server 2005 [online].
 Poslední revize 2009-2-15 [cit. 2009-02-28].
 Dostupné z: http://www.microsoft.com/sqlserver/2005/en/us/default.aspx>
- [10] Momjian, B.: PostgreSQL: Praktický průvodce. Computer Press, Brno, 2003, 402 s., ISBN 80-7226-954-2.
- [11] OLAP for ArcGIS [online]. Poslední revize 2007-10-21 [cit. 2009-02-12].
 Dostupné z: http://www.esri.com/software/arcgis/extensions/olap/index.html
- [12] PostGeoOLAP [online]. Poslední revize 2005-8-18 [cit. 2009-02-17].Dostupné z: http://postgeoolap.sourceforge.net/ingles/index.html
- [13] PostGIS [online]. Poslední revize 2008-12-15 [cit. 2009-02-16]. Dostupné z: http://postgis.refractions.net/>
- [14] PostgreSQL [online]. Poslední revize 2009-02-16 [cit. 2009-02-16].
 Dostupné z: http://www.postgresql.org/>

- [15] Quantum GIS Downloads [online].
 Poslední revize 2009-2-18 [cit. 2009-02-25].
 Dostupné z: http://postgeoolap.sourceforge.net/ingles/index.html
- [16] Rigaux, P., Scholl, M., Voisard, A.: Spatial Databases With Application to GIS. Morgan Kaufmann Publishers, San Francisco, 2002, 410 s., ISBN 1-55860-588-6.
- [17] Shekhar, S., Chawla, S.: Spatial Database: A tour. Pearson Education Inc., New Jersey, 2003, 262 s., ISBN 0-13-017480-7.
- [18] SourceForge.net: PostGeoOLAP [online].
 Poslední revize 2005-8-5 [cit. 2009-02-17].
 Dostupné z: http://sourceforge.net/project/showfiles.php?group_id=107278
- [19] Úsek ochrany a čistoty ovzduší ČHMU tabelární ročenky [online].
 Poslední revize 2008-6-22 [cit. 2008-09-28].
 Dostupné z: http://www.chmi.cz/uoco/isko/tab_roc/tab_roc.html
- [20] Vložení korektní projekce S-JTSK do PostGIS [online].
 Poslední revize 2005-8-18 [cit. 2009-02-17]. Dostupné z:
 http://janitor.cenia.cz/www/public/manual/postgresql_install/ar01s04.html
- [21] Voženílek, V.: *Diplomové práce z geoinformatiky*. Vydavatelství Univerzity Palackého, Olomouc, 2002, 31 s., ISBN 80-244-0469-9.

SUMMARY

This thesis presents the results of the final part of the master study program on Geoinformatics at the Faculty of Science, Palacky University in Olomouc.

The contribution of OLAP technology for GIS was examined in this thesis. Possibilities of two OLAP extensions (OLAP for ArcGIS for ArcGIS Desktop 9.x and PostGeoOLAP for PostgreSQL/PostGIS) there were found out and described. Two test data cubes were created for this purpose.

Data cubes don't have to contain spatial data, because results of OLAP analysis are joined with them only in GIS environment. For this reason OLAP extensions don't have no other functionality towards database servers, which can perform OLAP analysis in own environment. Databases servers often have more possibilities in execution of OLAP analysis, so there isn't considerable reason for using these extensions.

OLAP technology is shown as very effective tool for processing of considerable quantity of multidimensional data, which are often used in GIS. Resulting aggregated values can be visualized in GIS environment or analyzed in spatial or non-spatial form. In thesis the problem of successful using of OLAP technology during real project processing was discussed.

In the end there was presented other possibilities of development of OLAP technology and GIS cooperation. It is for example implementation of spatial query into OLAP analysis, which was mentioned in discussion part of thesis too. There was described possibility, how to obtain aggregated values from spatial layers instead of data in table form by OLAP analysis.

PŘÍLOHY

Seznam příloh

1. Přílohové DVD magisterské práce volná (DVD-ROM).

Struktura přílohy DVD-ROM

Adresáře:

Instalační soubory – instalační soubory aplikací a utilit využitých v průběhu práce Řezy – řezy datovými krychlemi napojené na prostorová data ve formátu SHP Skripty a zdrojová data – skripty vytvořené v průběhu práce SQL dotazy – SQL dotazy pro vytvoření samostatných databázových tabulek Zálohy databází – zálohy zkušebních databází

Jednotlivé soubory jsou popsány v metadatovém souboru "Metadata.txt" umístěném v kořenovém adresáři přílohového DVD magisterské práce.