

# Využitie vybraných technológií pre distribúciu výsledkov zo sčítania obyvateľov, domov a bytov na príklade Bratislavských obcí.

Vladimír Bačík

Úloha internetu, ako prostredia na zdieľanie a šírenie dát rozličného charakteru, je v súčasnej dobe nepopierateľná. Celosvetová počítačová sieť umožňuje prepojenie špecializovaných pracovísk po celom svete, výsledkom čoho je možnosť okamžitej konfrontácie odborníkov z príslušnej oblasti, ako aj distribúcia výsledkov výskumu týchto pracovísk či už odbornej, ale rovnako aj laickej verejnosti. Významnú úlohu v tomto procese zohrávajú samotné technologické prostriedky, na základe ktorých sa uskutočňuje spomenutá výmena informácií. Cieľom príspevku je podať základný prehľad o vybraných technológiách, ktoré sú vo svete internetu veľmi známe, v demografickej praxi sa im však nevenuje toľko pozornosti ako v iných, predovšetkým technických vedných disciplínach. Medzi najzaujímavejšie z nášho pohľadu možno zaradiť vektorový formát SVG (scaleable vector graphics), ktorý sa v poslednej dobe stáva veľmi dôležitým a efektívnym nástrojom internetového mapovania, ako aj databázové systémy, ktoré predstavujú veľmi efektívny nástroj na správu a zdieľanie demografických dát. Na význam oboch technologických nástrojov poukážeme na príklade distribúcie výsledkov zo sčítania obyvateľov, domov a bytov z roku 2001 v prostredí internetu, na modelovom území Bratislavských okresov.

## Databázové systémy

Efektívna organizácia existujúcich dát, je nevyhnutnou súčasťou fungovania všetkých vyspelých organizácií, firiem, pracovísk, rôznych inštitúcií, atď.. Aj v demografickej praxi sa stretávame s nutnosťou organizovania heterogénnych dát, za účelom ich efektívneho využitia pri analýzach rôznorodého charakteru. Okamžitá dostupnosť dát predstavuje dôležitý moment pri rozhodovacích procesoch v modernej spoločnosti.

Databázové systémy sú komplexné štruktúry, pozostávajúce z viacerých elementov. Základným elementom databázových systémov je samotná databáza. Definovanie pojmov z oblasti informatiky je pomerne zložitá, nakoľko tieto podliehajú neustálemu vývoju aplikačného prostredia. V užšom zmysle slova môžeme databázu chápať ako jednoduchú tabuľku, v ktorej sú jednotlivé záznamy zoradené podľa rôznych kritérií. Okrem samotných záznamov je to však aj celý rad operácií, zostáv, nastavení a výberov (SKALKA, J., DRLÍK, M. (2004). Podľa normy ISO je databáza definovaná ako dátová štruktúra pre príjem a uschovanie dát, ktoré sa na požiadanie poskytujú viacerým nezávislým používateľom. Komplexnejšia definícia hodnotí databázu ako súbor informácií, ako sú znaky, čísla, diagramy, ktorého systematická štruktúra umožňuje, aby sa tieto údaje dali vyhľadávať pomocou počítača. Databáza obsahuje štyri základné komponenty (SKALKA, J., DRLÍK, M. 2004) :

- Dátové prvky, ktoré zachytávajú elementárne hodnoty, t.j. vlastnosti v databáze popisovaného objektu (názov obce, kód obce)
- Vzťahy medzi dátovými prvkami, ktoré sú definované zložitejšími dátovými štruktúrami
- Integritné obmedzenia, ktoré si môžeme predstaviť ako explicitne zadané podmienky, ktoré musia údaje uložené v databáze spĺňať
- Schému databázy, čiže popis dát na úrovni, ktorej rozumie používateľ databázy

Databázu teda možno chápať ako štruktúrované úložisko údajov s tým, že údaje a vzťahy medzi nimi sú uložené a definované v rôznych štruktúrach a objektoch. Údaje sú spravidla uložené vo viacerých tabuľkách, medzi ktorými sú relačné väzby (HAVLÍČEK, 1992).

Len vo veľmi zriedkavých prípadoch sa databáza skladá z jednej tabuľky. Spravidla sa jedná o celý súbor tabuliek, pričom každá z nich má vlastnú štruktúru a tieto sú prepojené pomocou samotných relácií, pomocou primárnych a cudzích kľúčov. Medzi tabuľkami možno vytvoriť niekoľko základných druhov relácií (LACKO, L., 2001):

- Relácia 1 : 1 (one to one) - Každý riadok primárnej tabuľky môže byť prepojený práve s jedným riadkom sekundárnej tabuľky. Pre zaistenie takejto relácie je nutné vytvoriť v oboch tabuľkách unikátne kľúče.
- Relácia N : 1 (many to one) – Každý riadok primárnej tabuľky môže byť prepojený s jedným, alebo viacerými riadkami sekundárnej tabuľky (okres - obec)
- Relácia N : M (many to many) – Viac riadkov primárnej tabuľky je spojených s viacerými riadkami sekundárnej tabuľky. Tento vzťah je pomerne komplikovaný a v praxi býva zvyčajne nahradený viacerými reláciami typu N:1

Relačné tabuľky musia spĺňať niektoré elementárne požiadavky, ktoré možno vyjadriť niekoľkými základnými definíciami (HAVLÍČEK, Z., 1992):

- všetky hodnoty v tabuľke musia byť elementárne, tzn. ďalej nedeliteľné na ďalšie údaje (1. normálová forma)
- v tabuľke je 1 až n stĺpcov, pričom ich pozícia je nevýznamná, to znamená že stĺpce môžu byť v ľubovoľnom poradí
- v tabuľke je 0 až n riadkov, pričom ich pozícia je nevýznamná, to znamená že riadky môžu byť v ľubovoľnom poradí
- stĺpec musí byť homogénny, to znamená že obor hodnôt určitého stĺpca musí byť rovnaký (dáta rovnakého druhu)
- každému stĺpcu musí byť pridelené jednoznačné meno (atribút)
- každý riadok je jednoznačne rozlíšiteľný (primárnym kľúčom), v relačnej tabuľke nemôžu existovať dva úplne rovnaké riadky

Pomocou relácií môžeme odstrániť duplicitné záznamy v jednotlivých tabuľkách, čím splníme jednu so základných podmienok databázy o redundancii (nadbytočnosti), t.j. optimálnej štruktúre databázovej tabuľky bez viacnásobného opakovania jednotlivých dát. Optimálnu štruktúru databázových tabuliek nám umožnia predovšetkým procesy normalizácie, ktoré možno chápať ako reverzibilný (vratný) proces postupného nahrádzania danej množiny súborom relácií, ktoré majú jednoduchšiu a regulárnejšiu štruktúru.

Okrem samotných databázových tabuliek, je neoddeliteľnou súčasťou databázových systémov aj programové vybavenie, ktoré riadi všetky prístupy k údajom v databázových tabuľkách. Jedná sa o systém riadenia bázy dát (SRBD). Komunikácia medzi klientom (užívateľom) a SRBD prebieha pomocou jazyka SQL. Medzi najrozšírenejšie SRBD patria viaceré produkty svetových firiem, najznámejšími sú MySQL, ORACLE, SQL Server, IBM DB2 Universal Database, Microsoft Access, Postgre SQL, Sybase Adaptive Server, a mnohé iné. Použitie týchto programov možno realizovať viaceré základné funkcie potrebné pre efektívne využitie dát v databáze (vytváranie bázy dát, aktualizácia bázy dát, cielený výber z bázy dát). Z uvedeného teda možno považovať za dva základné komponenty databázových systémov databázy (jednotlivé tabuľky, v ktorých sú uložené dáta) a SRBD (programové vybavenie). Mnohí autori považujú termíny „databázový systém“ a „databáza“ za synonymum, čo je však veľmi nepresné, nakoľko databázové systémy sú komplexnejšie štruktúry.

Samotné databázové systémy možno rozdeliť do viacerých základných skupín. HALÁSZ a KRÁLIKOVÁ (1999) rozlišujú dve základné skupiny databázových systémov (DBS):

1. podľa použitého logického modelu vyjadrenia väzieb medzi dátami:

- systémy pre správu súborov
  - hierarchické DBS – modely sú uložené na stromových štruktúrach, v ktorých sa môžu vyskytovať iba vzťahy typu 1:1 alebo 1:N
  - sieťové DBS – realizujú sa hlavne vzťahy 1:1 a 1:N, ale aj N:M, čiže každý prvok dát môže byť zviazaný s ľubovoľným iným prvkom
  - relačné DBS – najčastejšie používané
2. podľa architektúry, resp. usporiadania jednotlivých spracovateľských komponentov systému:
- host-terminal – spracovanie dát vykonáva skoro výlučne hositeľský počítač (host). Terminály slúžia len na komunikáciu s užívateľmi, prípadne majú len obmedzené možnosti spracovania dát
  - file-server – charakteristickým rysom je oddelené umiestnenie dát a spracovávajúcich programov. Dáta sú uložené na súborovom serveri, ale spracovanie vykonávajú pripojené užívateľské stanice
  - klient-server – architektúra je založená na myšlienke rozdelenia kompetencií medzi programovými procesmi spracovania dát. Proces server (back-end) obstaráva všetky databázové operácie súvisiace s uchovaním, správou a zabezpečením uložených dát. Proces klient (front-end) komunikuje s užívateľom, prijíma jeho požiadavky, odovzdáva ich vo vhodnej forme databázovému serveru a prezentuje výsledky v žiadanej podobe.

Vzhľadom na priestorový rozmer jednotlivých záznamov by bolo možné doplniť aj tretiu skupinu databázových systémov. Drvivá väčšina dát má geografický rozmer – súvisia s vybranými geografickými prvkami priestoru (okresy, obce, štáty, rieky,...), vyjadrujú teda určité vlastnosti týchto elementov. Pokiaľ sú však tieto uložené len v jednotlivých tabuľkách bez priestorového prepojenia (texty, čísla,...), možno tieto považovať za nepriestorové databázové systémy. Ak však dôjde k priamemu prepojeniu na geografické elementy (t.z. že v databáze sú uložené informácie o priestorových objektoch, ako je tomu v prostredí GIS) možno hovoriť o priestorových databázových systémoch. Dôležitý rozdiel predstavuje aj samotný SRBD. V priestorových systémoch sa nazýva SRBDPO (systém riadenia bázy dát priestorových objektov) a okrem štandardných databázových funkcií musí umožniť (HALÁSZ, J., KRÁLIKOVÁ, R., 1999):

- definovanie priestorových dátových typov
- formuláciu priestorových dotazov (nájdienie prieniku objektov, vyhľadávanie objektov v určitej vzdialenosti od daného objektu a pod.)
- indexáciu priestorových objektov
- vykonávanie priestorových operácií (napr. priestorové spojenie množín objektov)

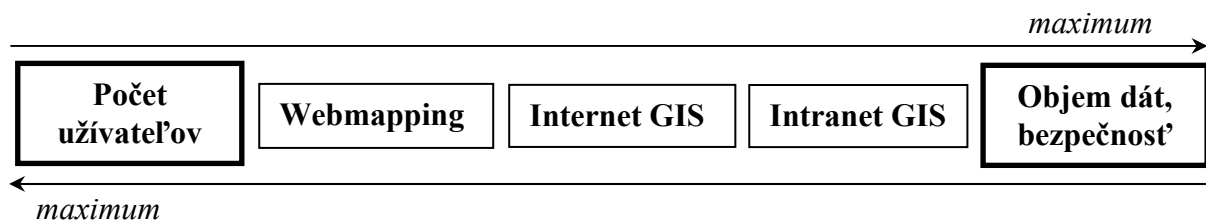
Cieľom tejto časti bolo podať základný prehľad o databázových systémoch, využívaných v prostredí internetu za účelom distribúcie dát koncovým užívateľom. Pochopenie niektorých teoretických momentov je dôležité pri analýze modelovej aplikácie vytvorenej pre Bratislavské okresy. V tejto bol využitý „open source“ SRBD MySQL, ktorý predstavuje jeden z najčastejšie využívaných databázových systémov v prostredí Internetu. Jeho kompatibilita s programovacím jazykom PHP dáva tvorcom aplikácií podobného charakteru k dispozícii veľké množstvo funkcií zameraných na podrobný výpis a organizáciu požadovaných dát.

### **Vektorový formát SVG (scaleable vector graphics) a internetové mapovanie**

Druhou významnou technologickou súčasťou hodnotenej aplikácie je formát SVG. Tento umožňuje prezentovanie mapových výstupov (ako aj iných objektov) vo vektorovom

formáte, čím dochádza k zachovaniu rovnakej kvality aj pri rôzne veľkých priblíženiach konkrétnych objektov. Formát SVG výrazným spôsobom ovplyvnil oblasť internetového mapovania, ktorej sa venuje predovšetkým v západných krajinách výrazná pozornosť práve v súvislosti s distribúciou kartografických výstupov na internete. V geografickej praxi dochádza pomerne často k stotožňovaniu termínov webmapping (internetové mapovanie) a webGIS, resp. internetGIS. Na zásadný rozdiel oboch termínov, resp. technológií poukazujú viacerí autori. HELD, G., SCHAEFFER, O., VOGEL, A. (2003) vo svojej práci zdôrazňujú základný rozdiel v stupni interakcie na strane užívateľa, bezpečnosťou distribuovaných dát ako aj celkovým počtom užívateľov (Obr. 1).

Obr. 1: Vzťah aplikácii webmappingu, internetGISov a intranetGISov (upravené podľa HELD, G., SCHAEFFER, O., VOGEL, A. (2003))

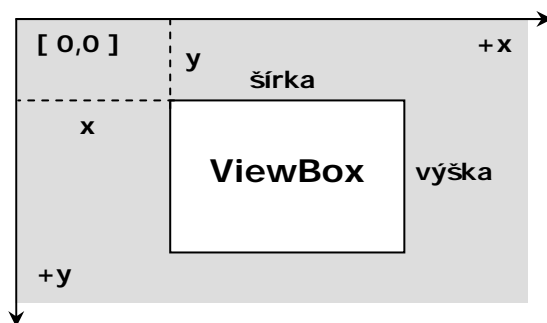


Dôležitý rozdiel predstavuje aj technologická výbava internetového servera, na ktorom je pri aplikáciách typu InternetGIS nainštalovaný špeciálny mapový server (ArcIMS, MapExtreme, Autodesk Mapguide, atď.) Aplikácie internetového mapovania majú jednoduché intuitívne ovládanie, zamerané zväčša na základné funkcie, ktorú sú však postačujúce zvolenému účelu ako aj cieľovej skupine užívateľov. Neustálym vývojom štandardov a podporou databázových systémov ako aj rozličných skriptovacích jazykov sa však čoraz častejšie stretávame s aplikáciami, ktoré sa svojou funkcionalitou výrazne približujú samotným aplikáciám typu InternetGIS. Rastúci význam SVG je zrejmy aj z podpory exportu dát do tohto formátu známymi tvorcami rozličných GIS programov. Jedná sa či už o priamu možnosť konverzie dát do SVG (ArcGIS 9.0), alebo o využitie špeciálnych nastavení umožňujúcich export dát do formátu SVG (MapToSVG, MapViewSVG, atď.)

Samotný formát SVG je podľa konzorcia W3C definovaný ako jazyk na popísanie dvojdimenzionálnej grafiky a grafických aplikácií v XML (<http://www.w3.org/Graphics/SVG/Overview.html>). SVG definuje tri základné typy grafických objektov: vektorové tvary (vector graphics shapes), rastrové obrazy (raster images) a textové objekty. Pri vytváraní objektov v uvedenom formáte treba dodržiavať isté syntaktické pravidlá, podobne ako pri iných internetových štandardoch. Na tomto mieste nebudeme podrobne vysvetľovať princíp vykreslenia jednotlivých geometrických primitív, ukážeme si len základné syntaktické postupy používané v objektoch typu „path“ (cesty, plochy), pomocou ktorých štandardne definujeme plošné (administratívne) jednotky ako obce, okresy, kraje, a načrtne problematiku používaného súradnicového systému ako aj vkladania SVG objektov do existujúcej internetovej stránky.

Súradnicový systém používaný vo formáte SVG má štandardne počiatok v ľavom hornom rohu, pričom kladné hodnoty osi X sú v smere zľava doprava a kladné hodnoty osi Y sú v smere zhora nadol (obr. 2). Využitím atribútu „transform“ možno súradnicový systém otočiť podľa potreby.

Obr. 2: Súradnicový systém používaný v SVG (podľa HELD, G., 2003)



Rozsah zobrazovanej oblasti je možné upraviť definovaním tzv. „viewboxu“, ktorého zápis môže mať nasledujúcu podobu:

```
<svg viewBox="70 50 200 100">
```

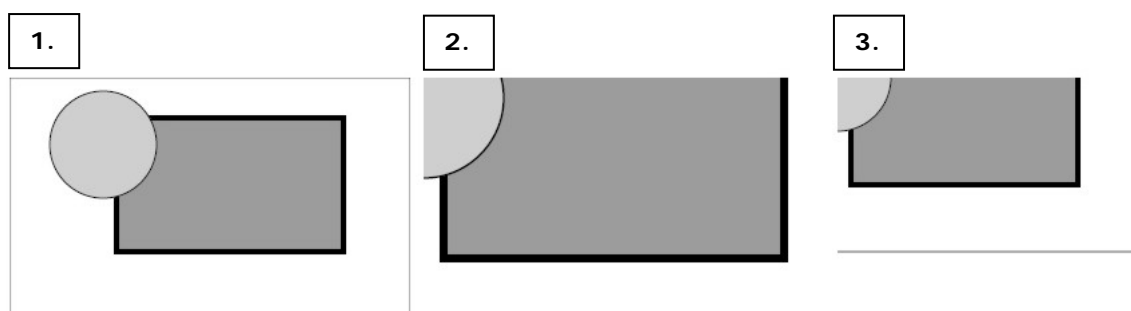
Prvá hodnota predstavuje X súradnicu začiatku, druhá Y súradnicu začiatku, a ďalšie dve hodnoty popisujú šírku a výšku zobrazovanej oblasti. Praktické znázornenie rozdielov v zobrazení elementov pri definovaní viewBoxu je na obr. 3. V uvedenom prípade boli vykreslené tri objekty, ktorých zápis je nasledujúci:

```
<rect fill="#9c9c9c" stroke="black" stroke-width="4" x="80" y="30" width="170" height="100" />
<circle fill="#cecfce" stroke="black" cx="70" cy="50" r="40"/>
<rect fill="none" stroke="black" stroke-width="1" x="0" y="0" width="300" height="180" />
```

V konkrétnych prípadoch je viewBox (zobrazovaná oblasť) definovaný nasledovne:

1. `<svg width="300" height="180"> // viewBox nie je definovaný`
2. `<svg width="300" height="180" viewBox="70 50 200 100">`
3. `<svg width="300" height="180" viewBox="70 50 0 0">`

Obr. 3: Zobrazenie elementov definovaním „viewBoxu“



Ďalším dôležitým krokom pri aplikáciách využívajúcich zápis objektov vo formáte SVG je vloženie takto definovaných elementov do existujúcej internetovej stránky. Existujú tri základné možnosti (tagy) používané v jazyku HTML (Hypertext Markup Language):

- **EMBED** `<embed src="subor.svg" width="350" height="176" type="image/svg+xml" name="map">`
- **OBJECT** `<object type="image/svg+xml" name="map" data="subor.svg" width="350" height="176"></object>`

- `IFRAME <iframe src="subor.svg" width="350" height="176" name="map"></iframe>`

Ako bolo naznačené v úvode tejto časti, pomocou syntaktických pravidiel formátu SVG možno definovať všetky základné druhy objektov (kruhy, body, štvorce, elipsy, krivky, lomené čiary, plochy). Pre korektné zobrazenie týchto objektov sú dôležité súradnice X,Y vo vzťahu k zobrazovanej ploche definovanej „viewBoxom“, ďalej atribúty výšky a šírky, ako aj príslušné diferenciacné atribúty (skosenie hrán v smere osi X, veľká poloos elipsy, polomer kruhu, atď.) V geografickej (demografickej) praxi pracujeme zvyčajne s plošnými jednotkami (obce, okresy), ku ktorým sa viažu originálne dáta, pričom tieto objekty sa v o formáte SVG označujú ako objekty typu „path“ (cesty, plochy). Základnú syntax pre zobrazenie elementu typu <path> možno zapísať nasledovne:

```
<path d="[zoznam bodov a príkazov pre znázornenie konkrétnych uzlov]"/>
```

Pri konverzii objektov z rozličných špecializovaných programov (Corel, ArcView9.x) za účelom exportu do formátu SVG sa zvyčajne využívajú tri základné druhy príkazov, podľa toho na aké miesto v objekte sa vzťahujú (BARTH, Y, 2004, FIBINGER, I., 2001):

- príkazy pre začiatkové a koncové uzly

- M [X] [Y]

*Moveto – príkaz nastaví polohu virtuálneho pera na súradnice X,Y bez kreslenia čiary*

- Z

*Closepath – príkaz na uzatvorenie vektorovej plochy (spojí koncový a začiatkový bod, bez nutnosti opätovného uvedenia súradníc prvého bodu)*

- príkazy pre línie s uhlovými bodmi

- L [X] [Y] / l [?x] [?y]

*Lineto – kreslí úsečku z počiatočného bodu na nové súradnice Z,Y*

Okrem samotného vykreslenia elementov je možné tieto ľubovoľným spôsobom transformovať. Konkrétne transformácie sa zaznamenávajú do atribútu „transform“, pričom hodnotou atribútu je zoznam transformácií, medzi ktoré možno zaradiť (FIBINGER, I., 2001):

- Preloženie (Translate)

Slúži na presun grafických elementov v smere XY

```
translate(tx,ty) -> translate(30,70)
```

- Zmena mierky (Scale)

Grafické elementy môžu byť zmenšené alebo zväčšené v smere X,Y

```
scale(sx,sy) -> scale(3,1)
```

- Rotácia (Rotate)

Grafické elementy môžu byť otočené o zadaný uhol, okolo ľubovoľného streda otáčania (ak nie je stred otočenia zadaný, jeho východisková hodnota je 0,0)

```
rotate(angle [centerX centerY]) -> rotate(10,30,70)
```

- Naklonenie (Skew)

Grafické elementy sú naklonené o zadaný uhol pozdĺž osi X, resp. Y.

```
skewX(angle) -> skewX(30)
```

```
skewY(anlgle) -> skewY(30)
```

Možnosti využitia vektorového formátu sú samozrejme oveľa rozsiahlejšie. Vďaka kompatibiliti s programovacím jazykom javascript (ecmascript) je možné využívať aj funkcie určené na animáciu vybraných objektov, výsledkom čoho sú interaktívne a užívateľsky veľmi

atraktívne aplikácie. Pri týchto však treba disponovať rozsiahlymi znalosťami z oblasti programovania a z hľadiska uvedeného príspevku nie je nutné sa tejto oblasti venovať. Hlavným cieľom bolo poukázať na základné charakteristiky formátu SVG, ktorý bol v hodnotenej aplikácii využitý na kartografickú prezentáciu Bratislavských obcí ako aj vybraných údajov zo sčítania v podobe jednoduchých tematických máp, v ktorých bola využitá metóda kartogramu.

### **Modelová aplikácia vytvorená pre obce Bratislavských okresov<sup>1</sup>**

Uvedené technológie, teda databázové systémy a formát SVG, boli využité s cieľom distribúcie vybraných ukazovateľov zo sčítania obyvateľov, domov a bytov z roku 2001 na modelovom území. Jedná sa v podstate o dve nezávislé platformy, prepojené medzi sebou pomocou štandardného identifikátora používaného v relačných databázach na úrovni obcí, ktorý predstavoval samotný kód obce. V aplikácii boli využité skripty napísané Andreasom Neumannom, ktorý ich použil vo svojom projekte „Vienna – Social patterns and structures“.

Jednou z dvoch hlavných častí aplikácie je kartografické znázornenie vybraných výsledkov zo sčítania na mape, v ktorej samotné objekty sú definované vo formáte SVG. Dôležitú súčasť však tvoria skripty (pozri hore), pomocou ktorých sú realizované viaceré dôležité funkcie v mape. Tieto sú zaznamenané v samotnom súbore (\*.js), pričom medzi základné možno zaradiť:

- `colorit` (vyfarbenie obcí podľa príslušnej hodnoty ukazovateľa)
- `toggleSVGElements` (zapnutie v vypnutie viditeľnosti vrstiev – obce, okresy, názvy obcí)
- `showDistrict` / `emptyDistrict` (zvýraznenie vybraného objektu pri prechode myšou, zobrazenie príslušnej hodnoty v podobe textového výpisu)
- `zoomIt` (priblíženie mapového výrezu, toto je realizované pomocou rozbalovacieho menu, ktoré je prepojené s prehľadovou mapou)
- `beginPan(evt)` / `doPan` / `endPan` (presun konkrétneho výrezu pri priblížení v oblasti prehľadovej mapy)

Zobrazované dáta sú uložené podobne ako skripty v samostatnom súbore (\*.js) ako premenné. Stručný zápis dát vyzerá nasledovne:

```
var poly529427 = ['Záhorská Bystrica', 'Bratislava4', 104, 0, 48.08, ..ostatné dáta ..];
var poly529371 = ['Devínska Nová Ves', 'Bratislava4', 104, 0, 48.43, ..ostatné dáta..];
.... ostatné obce
var poly529494 = ['Rusovce', 'Bratislava5', 105, 0, 49.84, ..ostatné dáta..];
var poly529435 = ['Čunovo', 'Bratislava5', 105, 0, 50.71, ..ostatné dáta..];

var ids=[529427, 529371, ..., 529494, 529435];
```

Konkrétne hodnoty vystupujú ako premenné v skriptovom súbore, pričom reálne hodnoty zo súboru sa v mape zobrazujú na základe priameho prepojenia s mapovým objektom (obcou) pomocou zápisu v tvare:

```
<path id="bebaut529427" onmouseover="showDistrict(529427,1)"
onmouseout="emptyDistrict(1)" d="M378.90225,75.36636L376.98254,....
súradnice..z"></path>
```

---

<sup>1</sup> Online podoba výslednej aplikácie je na stránke <http://www.sodbtn.sk/svg/bratislava>

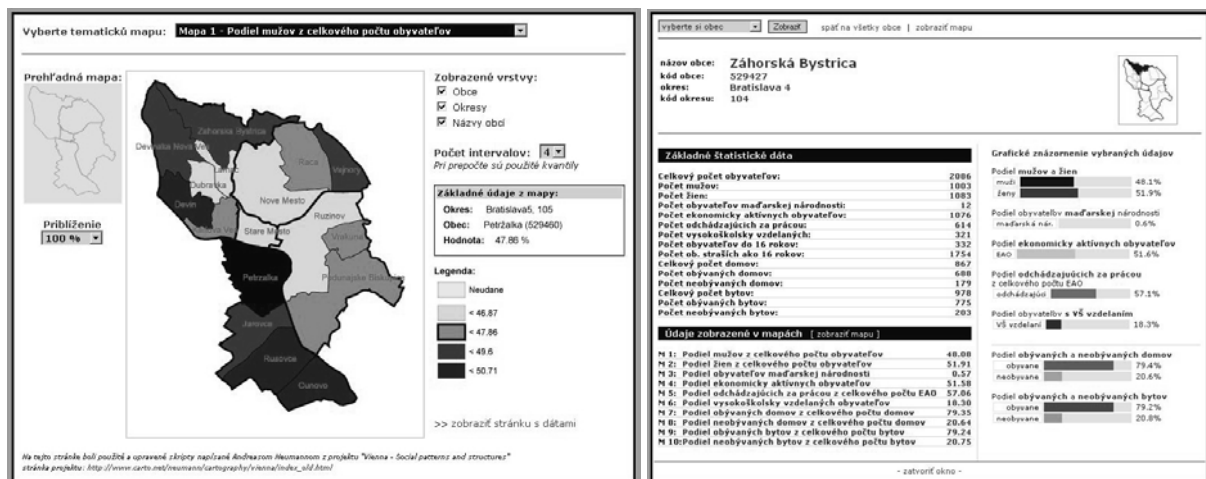
Výsledná aplikácia obsahuje 10 tematických máp vyjadrujúcich rozličné ukazovatele zo sčítania. Pri zobrazovaní dát uložených v samotnom súbore, vzťahujúcich sa ku konkrétnej mape je využitá funkcia `showDistrict`. Prepojenie objektu s dátami uloženými v databáze je možné pomocou štandardného príkazu jazyka XML „`xlink`“, kde hodnota kódu obce vstupujúca ako premenná umožní pomocou jazyka SQL zobrazenie dát vzťahujúcich sa k tejto územnej jednotke. Reálny zápis vyzerať nasledovne:

```
<a xlink:href="http://www.sodbtn.sk/svg/bratislava/obec.php?kod=529427">
  // link v mape, ktorým je prepojený objekt s databázou

select * from bratislava WHERE kod = '$kod'
  // výpis dát na stránke obce, na základe príslušného kódu obce
```

Databázová tabuľka uložená v databáze na serveri, obsahuje základné vybrané dáta zo sčítania. Ich výpis na strane klienta využíva syntaktické pravidlá jazyka SQL, v súvislosti so spomenutou premennou, ktorú predstavuje kód obce. Tieto sú následne zobrazené v podobe tabuľky na stránke konkrétnej obce. Okrem toho bola využitá aj možnosť grafickej vizualizácie vybraných javov, pričom za týmto účelom bol využitý skript „`htmlgraphs`“ od autora Gerda Tentlera. Konkrétna podoba výslednej aplikácie je znázornená na obr. 4.

Obr. 4: Ukážky výslednej aplikácie (kartografická vizualizácia, výpis dát z databázy)

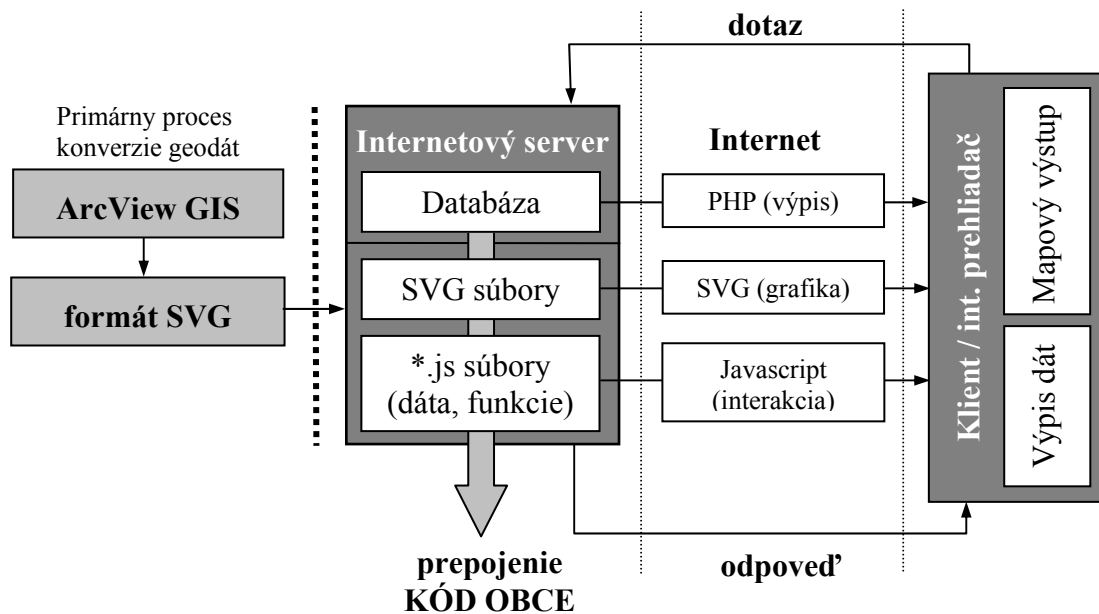


## Záverčné zhodnotenie

Využitie databázového systému v kombinácii s formátom SVG predstavuje zaujímavé možnosti distribúcie výsledkov zo sčítania širokému okruhu záujemcov o tieto výsledky. Schematické znázornenie celej aplikácie je uvedené v závere tejto časti (obr. 5). Z hľadiska dostupnosti jednotlivých technologických prostriedkov možno konštatovať, že uvedené technológie predstavujú v súčasnosti svetový štandard, pričom ich využitím možno dosiahnuť veľmi uspokojivého výsledku pri minimálnej cenovej zaťažnosti. Najnákladnejšou časťou celého systému tak aj v tomto prípade zostávajú vstupné dáta. Na tomto mieste však treba konštatovať, že v Slovenských podmienkach sú aplikácie podobného typu, najmä z pohľadu využitia formátu SVG, skôr ojedinelou záležitosťou ako bežným javom. Preto tu vzniká zaujímavý priestor pre technicky orientovaných demografov, aby participovali na popularizácii tohto, v západných krajinách veľmi rozšíreného nástroja internetového mapovania.



Obr. 5: Schematické znázornenie hodnotenej aplikácie



## Literatúra

- BARTH, Y. (2004):** *Nutzung von Servlets, JavaServer Pages, XML und XSL zur SVG-basierten Visualisierung raumbezogener Daten*, Diplomarbeit, Fachhochschule Stuttgart, 103 str., dostupné na: [http://www.carto.net/papers/yvonne\\_barth/2004\\_yvonne\\_barth\\_dfkviewer.pdf](http://www.carto.net/papers/yvonne_barth/2004_yvonne_barth_dfkviewer.pdf), prístup: 12. 4. 2006
- FIBINGER, I. (2001):** *Scalable Vector Graphics (SVG) Untersuchung des XML-Standards f.r zweidimensionale Vektorgraphiken und Erstellung eines SVG-Tutorials*, Diplomarbeit, Fachhochschule Karlsruhe, 115 str., dostupné na: [http://www.carto.net/papers/iris\\_fibinger/2001\\_03\\_iris\\_fibinger\\_svg\\_diplomarbeit.pdf](http://www.carto.net/papers/iris_fibinger/2001_03_iris_fibinger_svg_diplomarbeit.pdf), prístup: 16. mája 2006
- HAVLÍČEK, Z. (1992):** *Databázové systémy Paradox*, Grada a.s., Praha, ISBN 80-85424-97-5, 136 str.
- HALÁSZ, J., KRÁLIKOVÁ, R. (1999):** *Informačné systémy a databázové technológie*, Technická Univerzita Košice, ISBN 80-7099-407-X, 153 str.
- HELD, G., SCHAEFFER, O., VOGEL, A. (2003):** *Internet GIS mit SVG*, dostupné na: [http://www.geographie.hu-berlin.de/gk/gk/leute/publik/ringvl\\_03/vogel.pdf](http://www.geographie.hu-berlin.de/gk/gk/leute/publik/ringvl_03/vogel.pdf), prístup: 15. januára 2004
- LACKO, L. (2001):** *Web a databáze, programujeme internetové aplikace*, Computer Press, Praha, ISBN 80-7226-555-5, 250 str.
- NEUMANN, A., WINTER, A. (2003):** *Vector-based Web Cartography: Enabler SVG*, dostupné na: [http://www.carto.net/papers/svg/index\\_e.shtml](http://www.carto.net/papers/svg/index_e.shtml), prístup: 12. januára 2004
- SKALKA, J., DRLÍK, M. (2004):** *Databázové systémy*, Nitra, 28 str.

---

Mgr. Vladimír Bačík  
 Katedra humánnej geografie a demogeografie, PrifUK Bratislava  
 Mlynská dolina, 842 15  
 bacik@fns.uniba.sk  
<http://www.fns.uniba.sk/~bacik>