

Slovenská technická univerzita v Bratislave

Fakulta informatiky a informačných technológií

Študijný program: Počítačové systémy a siete

Rastislav Szabó

Vizualizácia lokalizácie IP adries

Bakalársky projekt

Vedúci bakalárskeho projektu: Ing. Dušan Bernát

máj 2009

Zadanie

Analyzujte možnosti získania informácie o geografickej polohe (napríklad krajina, mesto, ...) zariadenia s danou IP adresou. Navrhните a implementujte nástroj pre vyhľadávanie a prehľadné grafické zobrazenie polohy prislúchajúcej danej IP adrese, ako i zobrazenie ciest (postupností adries) a ďalších agregovaných údajov pre zadané súbory IP adries. Riešenie overte.

Podpísaný Rastislav Szabó čestne vyhlasujem, že bakalársku prácu Vizualizácia lokalizácie IP adries som vypracoval samostatne, na základe poznatkov získaných počas štúdia a informácií z dostupnej literatúry uvedenej v práci.

Uvedenú prácu som vypracoval pod vedením Ing. Dušana Bernáta.

V Bratislave, 15.5.2009

Rastislav Szabó

Podakovanie

Na tomto mieste by som sa rád poďakoval vedúcemu môjho bakalárskeho projektu Ing. Dušanovi Bernátovi, ktorý ma pri vypracovávaní projektu svedomito viedol a zabezpečil výborné technické zázemie pre implementáciu a publikovanie vypracovanej aplikácie.

Vďaka tiež patrí všetkým sto sedemdesiatim priateľom, ktorí sa zapojili do používateľského testovania vypracovanej aplikácie a mnohokrát prispeli cennou myšlienkou, či postrehom.

Anotácia

Slovenská technická univerzita v Bratislave
FAKULTA INFORMATIKY A INFORMAČNÝCH TECHNOLOGÍÍ
Študijný program: Počítačové systémy a siete

Autor: Rastislav Szabó
Bakalársky projekt: VIZUALIZÁCIA LOKALIZÁCIE IP ADRIES
Vedenie bakalárskeho projektu: Ing. Dušan Bernát
máj 2009

Predkladaný bakalársky projekt analyzuje možnosti získania informácií o geografickej polohe zariadenia s danou IP adresou, ako i prípady použitia lokalizácie IP adres v praxi. Podrobne rozoberá 4 metódy lokalizácie IP adres – lokalizáciu na základe *LOC DNS* záznamu, analýzy údajov z *Whois* serverov, analýzy doménového mena a *TXT DNS* záznamu. Popisuje nástroj *traceroute* ako doplnkovú možnosť odhadu lokalizácie IP adresy.

Popísané metódy lokalizácie IP adres využíva konzolová aplikácia pre lokalizáciu IP adres a webová aplikácia pre vizualizáciu lokalizácie IP adres, ktoré sú výsledkom riešenia tohto projektu. Podrobný popis ich návrhu a implementácie sa nachádza v príslušných častiach tejto práce. Výsledky implementovaného algoritmu pre lokalizáciu IP adres boli overené porovnaním s externou databázou, ako i používateľským testovaním formou elektronického dotazníka. Implementované riešenie je dostupné vo forme konzolovej aplikácie pre lokalizáciu IP adres, ako i vo forme webovej aplikácie, ktorá je verejne dostupná na Internete.

Annotation

Slovak University of Technology Bratislava
FACULTY OF INFORMATICS AND INFORMATION TECHNOLOGIES
Degree Course: Computer Systems and Networks

Author: Rastislav Szabó
Bachelor Theses: Visualization of IP addresses localization
Supervisor: Ing. Dušan Bernát
2009, May

This bachelor project analyzes possibilities of obtaining information about the geographical location of a device with the given IP address, as well as the use of the localization of IP addresses in the practice. Project includes detailed analysis of 4 localization methods – location based on LOC DNS entry analysis, the analysis of data from Whois servers, hostname and TXT DNS analysis. It describes the traceroute tool, as additional possibility of estimating the IP addresses localization.

Described methods of the IP addresses localization are used in the console application and also in the web application allowing visualization of IP addresses localization. A detailed description of their design and implementation can be found in relevant sections of this project. The results of the implemented algorithm for the IP addresses localization has been verified by comparison with external database, and by user testing using an electronic questionnaire. The implemented solution is available in the form of the console application, as well as in the form of a web application allowing visualization of the IP addresses localization, which is publicly available on the Internet.

Obsah

Úvod	1
1 Analýza problematiky	2
1.1 Prípady použitia.....	2
1.1.1 Online marketing.....	2
1.1.2 Bezpečnosť.....	2
1.1.3 Právne aspekty.....	2
1.1.4 VoIP a tiesňové hovory.....	3
1.2 Prehľad podobných riešení a produktov.....	3
1.2.1 Webové aplikácie.....	3
1.2.2 Databázy IP adries.....	3
1.2.3 Softvérové produkty a knižnice.....	4
1.3 Možnosti získania informácií o geografickej polohe IP adresy.....	4
1.3.1 Whois databázy.....	4
1.3.2 Doménové meno.....	10
1.3.3 DNS záznamy.....	10
1.3.4 Použitie nástroja traceroute.....	11
1.3.5 Ďalšie možnosti.....	12
1.4 Voľba implementačného prostredia.....	12
1.4.1 Implementačné prostredie lokalizácie.....	12
1.4.2 Implementačné prostredie vizualizácie.....	13
1.4.3 Databázové prostredie.....	15
2 Návrh riešenia	17
2.1 Špecifikácia požiadaviek.....	17
2.2 Návrh lokalizácie IP adries.....	17
2.2.1 Lokalizácia na základe LOC DNS záznamu.....	18
2.2.2 Lokalizácia na základe údajov z Whois databáz.....	19
2.2.3 Lokalizácia na základe analýzy doménového mena.....	21
2.2.4 Lokalizácia na základe TXT DNS záznamu.....	24
2.2.5 Lokalizácia s použitím nástroja traceroute.....	24
2.3 Výpočet vzdialenosti medzi IP adresami.....	24
2.4 Návrh konzolovej aplikácie pre lokalizáciu IP adries.....	25
2.5 Návrh webovej aplikácie pre vizualizáciu lokalizácie IP adries.....	25
3 Implementácia	27
3.1 Použité externé moduly.....	27
3.1.1 Net::DNS.....	27
3.1.2 Net::Whois:Proxy.....	27
3.1.3 Sys::HostIP.....	28
3.2 Viacvláknosť.....	28
3.3 Traceroute.....	28
3.4 Databázové prostredie.....	29
3.5 Vymazávanie dočasných údajov.....	30

3.6 CGI a šablónovací systém.....	30
3.7 Google Maps API.....	31
3.8 Jazykové mutácie webovej aplikácie.....	33
3.9 Bezpečnosť webovej aplikácie.....	33
3.9.1 Overovanie platnosti vstupných údajov.....	33
3.9.2 Ochrana pred zahltením.....	33
3.9.3 Zakrytie citlivých údajov.....	34
4 Overenie riešenia.....	35
4.1 Porovnanie výsledkov lokalizácie s externou databázou.....	35
4.1.1 Vstupné dáta.....	35
4.1.2 Výsledky testovania.....	35
4.2 Používateľské testovanie lokalizácie.....	37
4.2.1 Vstupné dáta.....	37
4.2.2 Výsledky testovania.....	37
4.3 Beta testovanie.....	39
5 Zhodnotenie.....	40
6 Technická dokumentácia.....	41
6.1 Prehľad zdrojových kódov riešenia.....	41
6.1.1 Zdrojové kódy modulov lokalizácie IP adres.....	41
6.1.2 Zdrojové kódy konzolovej aplikácie.....	43
6.1.3 Zdrojové kódy webovej aplikácie.....	43
6.2 Fyzický dátový model.....	44
6.3 Inštalačná príručka.....	47
6.3.1 Inštalačná príručka konzolovej aplikácie.....	47
6.3.2 Inštalačná príručka webovej aplikácie.....	47
6.4 Používateľská príručka.....	48
6.4.1 Používateľská príručka konzolovej aplikácie.....	48
6.4.2 Používateľská príručka webovej aplikácie.....	49
7 Zoznam použitej literatúry.....	55
Zoznam skratiek.....	57
Obsah elektronického média.....	58

Úvod

Celosvetová sieť Internet sa v súčasnosti stáva neoddeliteľnou časťou nášho každodenného života. Z pôvodného prostriedku vedy sa dnes stáva centrum informácií, obchodu i zábavy.

Komunikácia medzi jednotlivými zariadeniami pripojenými v Internete prebieha v súčasnosti predovšetkým prostredníctvom IP protokolu (protokol 3. vrstvy referenčného modelu OSI). Pre umožnenie komunikácie zariadení v Internete je nutné každé zariadenie jednoznačne identifikovať. Na tento účel sa používa IP adresa – 32 bitová jedinečná logická adresa sieťového rozhrania (IP protokol verzia 4).

S rastúcou popularitou a zvyšovaním počtu používateľov i služieb Internetu narastá význam identifikácie jednotlivých používateľov nielen na základe IP adresy, ale aj na základe príslušnosti ku geografickej lokácii používateľovho zariadenia. Ak poznáme IP adresu konkrétneho zariadenia, je možné na základe istých postupov odhadnúť i jeho geografickú polohu.

Predkladaná práca obsahuje analýzu prípadov použitia lokalizácie IP adries v praxi, analýzu a rozbor jednotlivých postupov a metód, ktorými je možné odhadnúť lokalizáciu zariadenia s danou IP adresou.

V práci sa ďalej nachádza podrobný návrh procesu lokalizácie IP adries použitím metód rozobraných v analýze problematiky, ako i návrh konzolovej aplikácie pre lokalizáciu IP adries a webovej aplikácie pre vizualizáciu lokalizácie IP adries.

Ďalšia časť dokumentu popisuje samotnú implementáciu navrhnutého riešenia, overenie výsledkov lokalizácie IP adries, ako i funkcionality samotnej aplikácie.

Záverečnú časť dokumentu tvorí technická dokumentácia obsahujúca prehľad zdrojových kódov riešenia, inštalačnú a používateľskú príručku.

1 Analýza problematiky

Táto časť dokumentu je zameraná na analýzu možností získania informácie o geografickej polohe zariadenia s danou IP adresou. Predstavuje niekoľko prípadov použitia lokalizácie IP adres, prehľad existujúcich riešení ponúkajúcich túto funkcionality. Záverečná časť kapitoly je zameraná na analýzu implementačných prostredí, ktoré by mohli byť použité pri návrhu aplikácie pre vizualizáciu lokalizácie IP adres.

1.1 Prípady použitia

V praxi existuje mnoho situácií, v ktorých môže byť užitočné lokalizovať neznámu IP adresu. Jednotlivé situácie kladú rozličný dôraz na presnosť geografickej lokalizácie. V niektorých prípadoch postačuje určiť krajinu, v ktorej sa používateľ nachádza, v iných sa kladú požiadavky na veľmi presnú lokalizáciu [1].

1.1.1 Online marketing

Veľmi užitočným nástrojom v online marketingu je cielenie reklamných kampaní na základe geografickej polohy návštevníka stránky. Poskytovatelia online marketingových služieb, akým je napríklad *Google AdWords* [2], používajú zacielenie na regionálne a miestne oblasti, pomocou ktorého je možné osloviť iba tých zákazníkov, ktorí sú najvhodnejší z hľadiska ich lokalizácie, prípadne propagovať isté produkty na základe geografickej polohy potenciálneho zákazníka.

Pre autorov a správcov internetových stránok môže byť tiež veľmi užitočné poznať geografické rozloženie návštevníkov ich stránok. Tieto informácie poskytujú i mnohé nástroje určené pre monitorovanie návštevnosti internetových stránok.

1.1.2 Bezpečnosť

Z hľadiska bezpečnosti môže byť užitočné ukladať IP adresu a následne ju lokalizovať napríklad v prípade vykonania nedovolených transakcií v sieťovej aplikácii, prípadne pri spätnej identifikácii prípadného útočníka systému (i keď možno predpokladať, že zachytená IP adresa útočníka s veľmi veľkou pravdepodobnosťou nebude zodpovedať skutočnej IP adrese jeho zariadenia).

Ďalší význam nadobúda lokalizácia IP adres v oblasti bankového sektoru, napríklad pri platbe kreditnou kartou na Internete. Na základe IP adresy kupujúceho v internetovom obchode je možné zistiť, kde sa zákazník takéhoto obchodu nachádza. Ak je tou istou kreditnou kartou v krátkych časových intervaloch platené na rôznych miestach sveta, je to indikátorom toho, že túto kartu používa viac ako jeden používateľ, čo nie je v prípade súkromných kreditných kariet obvyklým javom a môže to indikovať jej zneužitie [1].

1.1.3 Právne aspekty

Niektoré právne predpisy, zmluvy a normy prikazujú napríklad obmedziť poskytovanie internetových služieb len na určitú oblasť. Charakteristickým príkladom je prenos televízneho vysielania prostredníctvom Internetu [1]. Ak má televízia zakúpené vysielacie práva s platnosťou len na území príslušného štátu a chce vysielanie sprístupniť aj na Internete, musí zabezpečiť, aby sa k vysielaniu nedostali návštevníci zo zahraničia.

Ďalším príkladom takýchto internetových služieb by mohli byť napríklad lekárne, ktoré nemôžu distribuovať lieky mimo hraníc príslušného štátu, prípadne stávkové kancelárie, ktoré môžu

prijímať stávky len v rámci povolenia v určitom štáte.

1.1.4 VoIP a tiesňové hovory

V prípade klasických telefónnych liniek je každému používateľovi pridelené pevné telefónne číslo, na základe ktorého vie operátor presne určiť lokalizáciu volajúceho. Pri použití služby **VoIP** (*Voice over IP*) však môže používateľ toho istého čísla telefonovať z ktoréhokoľvek miesta na svete. Pre určenie jeho polohy môže byť použitá lokalizácia na základe jeho IP adresy.

Lokalizácia volajúceho je dôležitá hlavne v prípade tiesňových volaní. Hovor môže uskutočňovať vystrašená osoba v panike, ktorá zabudne uviesť presnú pozíciu, prípadne môže byť hovor prerušený. Hlavne v tomto prípade je žiaduca veľmi vysoká presnosť lokalizácie.

1.2 Prehľad podobných riešení a produktov

Význam tejto problematiky potvrdzuje i to, že v súčasnosti už existuje množstvo nástrojov určených k lokalizácii IP adres. Je však veľmi ťažké odhadnúť správnosť ich funkcionality, čo potvrdzuje i fakt, že mnohé z nich vracajú pri rovnakých vstupoch rôzne výsledky.

1.2.1 Webové aplikácie

Veľkú časť existujúcich riešení tvoria webové aplikácie. Mnohé z nich ponúkajú vizualizáciu lokalizácie IP adresy na mape, väčšina z nich na tento účel využíva voľne prístupné *Google Maps API*. Len malý počet nájdených riešení však poskytuje i zobrazovanie ciest (postupností) IP adres.

Portál **WhatsMyIPAddress** [3] používateľovi hneď po vstupe na stránku zobrazí jeho IP adresu a jej polohu na mape *Google Maps*. Autori na stránke uvádzajú, že na lokalizáciu IP adresy používajú *Whois* databázy a spätný DNS preklad IP adresy na doménové meno (kapitola 1.4.). Stránka ponúka aj vizuálne trasovanie IP adresy (*traceroute*), teda zobrazenie cesty medzi serverom tohto webu a zariadením s danou IP adresou.

Find Ip Address [4] má základnú funkcionality veľmi podobnú ako predošlá stránka, okrem lokalizácie IP adresy a zobrazenia lokalizácie na mape *Google Maps* ponúka aj niektoré doplnkové služby, napríklad výpis kompletného záznamu z *Whois* serveru, výpis DNS záznamov, spätný DNS preklad, alebo meranie odozvy nástroja *ping* (*Packet InterNet Groper*). Autori však neuvádzajú, akým spôsobom dochádza k lokalizácii IP adres.

IP Address Lookup [5] je príkladom nástroja, ktorý nepoužíva vlastné lokalizačné techniky. Autori tohto webu uvádzajú, že na lokalizáciu IP adres je použitá externá databáza IP adres patriaca inej spoločnosti.

Na Internete existuje obrovské množstvo podobných aplikácií. Jednotlivé nástroje sa veľmi líšia z hľadiska presnosti lokalizácie. Mnohé z nich sú schopné určiť iba krajinu lokalizácie, prípadne za mesto lokalizácie nesprávne označujú iné mesto tej istej krajiny, niektoré však dokážu IP adresy lokalizovať pomerne presne. Na základe výsledkov je však ťažké určiť, aká technológia sa pri určovaní geografickej polohy využíva, a preto je ťažké odhadovať dôveryhodnosť ich výsledkov.

1.2.2 Databázy IP adres

Na Internete sú dostupné rôzne databázy, ktoré umožňujú IP adrese priradiť miesto lokalizácie, prípadne i geografické súradnice.

Veľmi známe je komerčné riešenie spoločnosti **MaxMind, Inc.** – databáza **GeoIP** [6]. Časť databázy, ktorá je však menej často aktualizovaná je dostupná i bezplatne, vo forme databázových

súborov, ako i vo forme API funkcií. Autori však na stránke neuvádzajú, akým spôsobom sa IP adresy lokalizujú.

Ďalším známym internetovým projektom je **hostip.info** [7], ktorý je založený na komunitnom systéme. Každý návštevník tejto stránky môže uložiť svoju lokalizáciu, čím sa do databázy pridá jeho IP adresa spolu so zadanou lokalizáciou. K dispozícii je kompletná databáza, lokalizácia IP adresy je možná i prostredníctvom *GET* požiadaviek *HTTP* protokolu.

Problémom takýchto databáz je fakt, že IP adresy vo väčšine prípadov nie sú statické. Rovnaká IP adresa môže byť napríklad dynamicky pridelovaná rôznym používateľom v rámci jedného poskytovateľa internetového pripojenia. Jednotlivým subjektom takisto môže byť IP adresa odobraná a následne pridelená inému subjektu. Databáza obsahujúca lokalizácie takýchto IP adres tak bude takmer určite zastaraná, pretože manuálne udržiavanie tak veľkej databázy je pri dnešnom trende neustáleho nárastu počtu uzlov v Internete prakticky nemožné.

1.2.3 Softvérové produkty a knižnice

Podstatnú časť existujúcich riešení tvoria hotové softvérové produkty, často vo forme komerčných aplikácií. Existujú i voľne šíriteľné knižnice napísané v rôznych programovacích či skriptovacích jazykoch, ktoré je možné použiť pri vývoji vlastnej aplikácie. Mnohé z nich sú však často iba nástrojmi na čítanie zo statických databáz popísaných v predchádzajúcej časti. Na stránke *MaxMind, Inc.* [6] sú dostupné riešenia používajúce *GeoIP* databázu integrované do mnohých jazykov, ako napríklad *Perl*, *Python*, *Ruby*, *PHP*, *C*, *C#* či *Java*.

1.3 Možnosti získania informácií o geografickej polohe IP adresy

Pre bežného používateľa Internetu v praxi neexistuje žiadny zaručený spôsob priradenia geografickej polohy k zadanej IP adrese [1]. Existuje však niekoľko spôsobov, ktorými je možné jej polohu aspoň približne odhadnúť.

1.3.1 Whois databázy

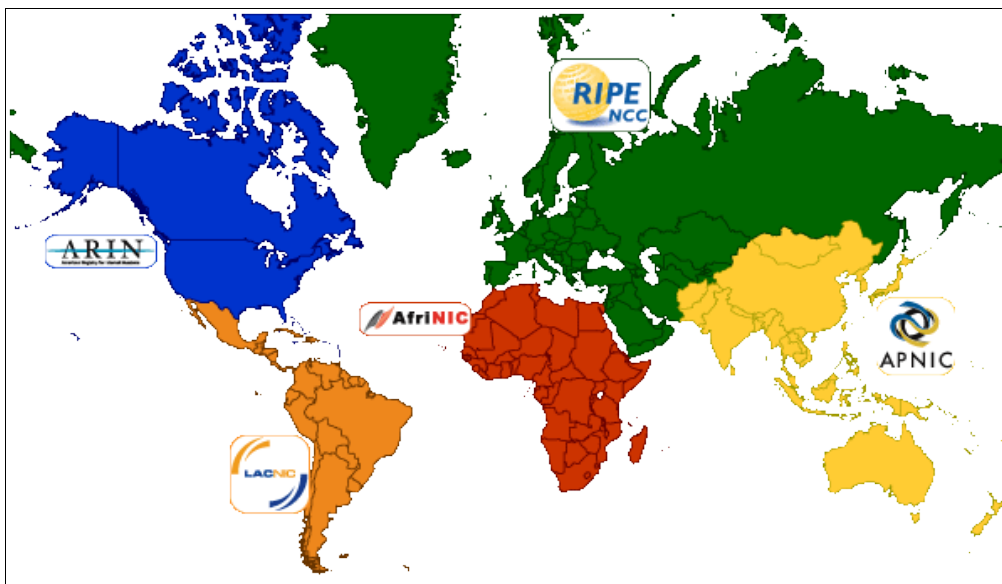
V sieti Internet je potrebné jednoznačne pridelovať IP adresy, čísla autonómnych systémov a doménové mená. Keďže neexistuje žiadna centrálna autorita, ktorá by riadila Internet, bolo založených niekoľko medzinárodných organizácií, ktoré tieto služby zabezpečujú [8].

Najvyššou autoritou v Internete je **The Internet Assigned Numbers Authority (IANA)**. Rozdeľuje rozsah IP adres na veľké intervaly, ktoré prideluje regionálnym internetovým registrátorom (*Regional International Registries – RIR*), ktorí spravujú jednotlivé väčšie oblasti sveta.

Jednotliví regionálni registrátori pridelujú IP adresy a doménové mená lokálnym registrátorom (väčšinou poskytovateľom internetového pripojenia a telekomunikačným spoločnostiam) a vedú záznamy o pridelených adresách a jednotlivých subjektoch vo svojich databázach.

Regionálnymi internetovými registrátormi sú (Obr. 1):

- **ARIN** (*American Registry for Internet Numbers*) – Amerika
- **AfriNIC** (*African Network Information Center*) – Afrika
- **APNIC** (*Asia Pacific Network Information Centre*) – Ázijsko-pacifická oblasť
- **LACNIC** (*Latin American and Caribbean IP Address Regional Registry*) – Latinská Amerika a Karibská oblasť
- **RIPE NCC** (*Reseaux IP Europeens Network Coordination Centre*) – Európa



Obr. 1: Mapa regionálnych internetových registrátorov

Časti databáz regionálnych registrátorov sú verejne prístupné prostredníctvom ich *Whois* serverov. Obsahujú údaje ako pridelený rozsah IP adries, názov a presnú adresu registrovaného subjektu a kontaktné informácie administratívy a technickej podpory. Tieto údaje sú do databázy vkladané v rámci registračného procesu u registrátora a v prípade zmeny registračných údajov sú dodatočne aktualizované [4].

Za správnosť údajov v týchto databázach zodpovedajú lokálni registrátori, ktorí registráciu a aktualizáciu starých záznamov vykonávajú presne špecifikovaným spôsobom. Napríklad v prípade *RIPE* je to e-mailová komunikácia so špeciálnym robotom, pričom jednotlivé správy musia mať presne stanovený formát a sú zabezpečené a autorizované špeciálnymi mechanizmami.

Pri vyhľadávaní údajov na *Whois* serveri prebieha komunikácia prostredníctvom špeciálneho *Whois* protokolu [9], ktorý na transportnej vrstve referenčného modelu *RMOSI* používa protokol *TCP*, ktorého dátová časť obsahuje čisto textové informácie. Pre jednoduché zisťovanie údajov z *Whois* serverov je dostupný nástroj *Whois*, ktorý býva i súčasťou štandardných Unixových distribúcií. Výpis všetkých dostupných informácií o danej IP adrese je možné vyvolať nasledovne:

```
whois 85.248.20.76
```

Vzhľadom na to, že komunikácia s *Whois* serverom prebieha v podobe obyčajného textu, je možné ľahko sa zaoberať aj bez tohto nástroja, použitím nástroja *telnet* s pripojením na port 43:

```
telnet whois.ripe.net 43
85.248.20.76
```

Jednotliví regionálni registrátori používajú rôzne formáty odpovedí na požiadavku *Whois* serveru. Nasledujúci výpis obsahuje príklad odpovede *Whois* serveru európskeho regionálneho registrátora **RIPE**:

```
% This is the RIPE Whois query server #3.
% The objects are in RPSL format.
%
% Rights restricted by copyright.
% See http://www.ripe.net/db/copyright.html
```

```
% Note: This output has been filtered.
%       To receive output for a database update, use the "-B" flag.
```

```
% Information related to '85.248.20.72 - 85.248.20.79'
```

```
inetnum:      85.248.20.72 - 85.248.20.79
netname:      SK-ENGINE-BB
descr:        ENGINE Slovakia, s.r.o.
descr:        Rudlovska cesta 2, 97401, Banska Bystrica
descr:        POP Banska Bystrica
country:      SK
admin-c:      RS1250-RIPE
tech-c:       GSNH1-RIPE
status:       ASSIGNED PA
mnt-by:       GTSSK-MNT
source:       RIPE # Filtered
```

```
role:         GTS-INEC Slovak Net Hostmaster
address:      GTS Nextra a.s.
address:      Aupark Tower
address:      Einsteinova 24
address:      Bratislava
address:      851 01
address:      Slovak Republic
phone:        +421 2 322 322 32 # Hotline
phone:        +421 2 32487 111
fax-no:       +421 2 32487 222
abuse-mailbox: abuse@gtsnextra.sk
admin-c:      RR857-RIPE
tech-c:       MH6751-RIPE
tech-c:       BP323-RIPE
nic-hdl:      GSNH1-RIPE
mnt-by:       GTSSK-MNT
source:       RIPE # Filtered
```

```
person:       Richard Sudoma
address:      Richard Sudoma
address:      Mladeznicka 17
address:      Banska Bystrica
address:      97404
address:      Slovakia
phone:        +421 903 887411
nic-hdl:      RS1250-RIPE
source:       RIPE # Filtered
```

```
% Information related to '85.248.0.0/16AS5578'
```

```
route:        85.248.0.0/16
descr:        GTS Slovakia NET
origin:       AS5578
mnt-by:       GTSSK-MNT
source:       RIPE # Filtered
```

Jednotlivé časti oddelené samostatným riadkom sa nazývajú objekty. Formáty jednotlivých objektov sú špecifikované na stránkach *RIPE NCC* [10].

Z pohľadu lokalizácie je najvýznamnejší objekt *inetnum*. V tomto objekte sa nachádza rozsah pridelených IP adries a meno používateľa, resp. názov organizácie, ktorej bol tento rozsah IP adries pridelený. Položka *country* tohto objektu obsahuje skratku krajiny a v zázname *descr* je možné nájsť popis, prípadne i adresu tohto používateľa, resp. organizácie. Ide o viaciadkovú položku, ktorá nemá stanovený presný obsah a formát, preto je ťažké jednoznačne určiť reťazec, ktorý obsahuje mesto lokalizácie. Všeobecne býva zvykom, že mesto adresy je uložené na poslednom riadku (analógia písania adresy na klasické poštové listy). V našom príklade však vidieť, že nie je možné posledný riadok záznamu *descr* automaticky považovať za reťazec s mestom (v tomto prípade mu ešte predchádza reťazec „POP“). Pre získanie mesta lokalizácie je preto dobré prechádzať jednotlivé riadky tohto záznamu odspodu a postupne ich porovnávať pomocou regulárnych výrazov s databázou miest.

Zvlášť v prípade IP adries pridelených verejným poskytovateľom Internetu koncovým zákazníkom je dôležité skontrolovať objekt *person*. Sú v ňom uložené kontaktné informácie o technickej

a administratívnej podpore prideleného adresného priestoru. Tieto údaje sú uložené vo viacriadkovom zázname *address*. Pri získavaní informácie o lokalizácii preň platí to isté, čo pre záznam *descr* objektu *inetnum*. Pre určenie lokalizácie by mohli byť použité i kontaktné telefónne čísla (záznam *phone*), a to identifikáciou medzinárodnej a lokálnej telefónnej predvoľby. Pre určenie krajiny lokalizácie je však spoľahlivejšie použitie záznamu *country* objektu *inetnum*. Určenie mesta lokalizácie na základe miestnej telefónnej predvoľby môže byť zavádzajúce, v prípade mobilného telefónneho čísla je to dokonca nemožné. Aplikácia by navyše musela obsahovať databázu lokálnych predvoľieb všetkých krajín.

Pri určovaní mesta lokalizácie IP adresy je tento objekt dôveryhodnejší ako objekt *intenum*, ktorý máva napríklad v prípade veľkých internetových poskytovateľov v položke *descr* uvedenú adresu sídla poskytovateľa, ktorá sa nemusí zhodovať so skutočnou lokalizáciou IP adresy. Objektov *person* môže byť v jednej odpovedi i viacej.

V prípade väčších organizácií (akou je napríklad Slovenská technická univerzita) býva objekt *person* nahradený, prípadne doplnený objektom **organisation**, ktorý rovnako ako tento objekt obsahuje záznam *address*. Tieto údaje možno považovať za veľmi dôveryhodné, pokiaľ však nejde o mimoriadne veľkú organizáciu pôsobiacu na úrovni viacerých miest.

Objekt **role** obsahuje kontaktné údaje osôb, ktoré zastávajú určitú funkciu v organizácii, napríklad funkciu správcu. Z pohľadu testovaných IP adries sa však zdá, že tento objekt zvyčajne popisuje kontaktné údaje poskytovateľa internetového pripojenia vyššej úrovne (v hierarchii nad lokálnym poskytovateľom), preto je jeho použitie vhodné len v prípade, že sa nepodarí určiť lokalizáciu z vyššie uvedených objektov.

Formát odpovede *whois* serveru amerického regionálneho registrátora **ARIN** je o niečo jednoduchší:

```
OrgName:      Google Inc.
OrgID:        GOGL
Address:      1600 Amphitheatre Parkway
City:         Mountain View
StateProv:    CA
PostalCode:   94043
Country:      US

NetRange:     209.85.128.0 - 209.85.255.255
CIDR:         209.85.128.0/17
NetName:      GOOGLE
NetHandle:    NET-209-85-128-0-1
Parent:       NET-209-0-0-0-0
NetType:      Direct Allocation
NameServer:   NS1.GOOGLE.COM
NameServer:   NS2.GOOGLE.COM
NameServer:   NS3.GOOGLE.COM
NameServer:   NS4.GOOGLE.COM
Comment:
RegDate:      2006-01-13
Updated:      2006-06-01

OrgTechHandle: ZG39-ARIN
OrgTechName:   Google Inc.
OrgTechPhone:  +1-650-318-0200
OrgTechEmail:  arin-contact@google.com
```

```
# ARIN WHOIS database, last updated 2008-11-07 19:10
# Enter ? for additional hints on searching ARIN's WHOIS database.
```

Z výstupu je zrejmé, že jednotlivé lokalizačné údaje je možné vyčítať priamo z príslušných záznamov: krajinu zo záznamu *Country*, mesto zo záznamu *City* a prípadnú presnú adresu zo záznamu *Address*.

Latinsko-americký regionálny registrátor **LACNIC** používa veľmi podobný formát odpovede

na *whois* požiadavku ako registrátor *RIPE*:

```
% Joint Whois - whois.lacnic.net
% This server accepts single ASN, IPv4 or IPv6 queries

% LACNIC resource: whois.lacnic.net
% Copyright LACNIC lacnic.net
% The data below is provided for information purposes
% and to assist persons in obtaining information about or
% related to AS and IP numbers registrations
% By submitting a whois query, you agree to use this data
% only for lawful purposes.
% 2008-11-11 09:36:15 (BRST -02:00)

inetnum:      200.16.96/20
status:       assigned
owner:        Ministerio de Relaciones Exteriores, Comercio Inte
ownerid:      AR-MRECL-LACNIC
responsible:  Gustavo Manuel Zazkin
address:      Esmeralda, 1212, 3er Piso
address:      1007 - Buenos Aires -
country:      AR
phone:        +54 11 48197639 []
owner-c:      GMZ
tech-c:       GMZ
abuse-c:      GMZ
inetrev:      200.16.96/20
nserver:      ATHEA.AR
nsstat:       20081110 AA
nslastaa:     20081110
nserver:      CTINA.AR
nsstat:       20081110 AA
nslastaa:     20081110
created:      19950317
changed:      20000530

nic-hdl:      GMZ
person:       Gustavo Manuel Zazkin
e-mail:       gustavo@ATINA.AR
address:      Esmeralda, 1212, 3er Piso
address:      1007 - Buenos Aires -
country:      AR
phone:        +54 11 48197639 []
created:      20050630
changed:      20050704

% whois.lacnic.net accepts only direct match queries.
% Types of queries are: POCs, ownerid, CIDR blocks, IP
% and AS numbers.
```

Oproti formátu *RIPE* sú tu viditeľné malé zmeny, napríklad objektu *person* predchádza identifikátor *nic-hdl*, niektoré objekty majú mierne odlišné názvy záznamov (napríklad objekt *intenum* neobsahuje záznam *descr* ale *address*). Základná štruktúra je však veľmi podobná formátu *RIPE*, a preto je pre získanie lokalizačných údajov možné využiť mierne upravený algoritmus lokalizácie navrhnutý pre *Whois* server registrátora *RIPE*.

Formát regionálneho registrátora ázijsko-pacifickej oblasti **APNIC** je úplne zhodný s formátom **RIPE**:

```
% [whois.apnic.net node-2]
% Whois data copyright terms    http://www.apnic.net/db/dbcopyright.html

inetnum:      202.51.8.0 - 202.51.15.255
netname:      CLARAONLINE-TOKYO
descr:        Clara Online, Inc.
descr:        Web Hosting and Dedicated Server Provider.
descr:        Tokyo Data Center, Japan.
country:      JP
admin-c:      SS649-AP
tech-c:       SS649-AP
mnt-routes:   MAINT-JP-CLARA
remarks:      Send abuse reports to abuse@clara.ad.jp
```



```

mnt-by:          APNIC-HM
mnt-lower:       MAINT-JP-CLARA
status:          ALLOCATED PORTABLE
remarks:         -+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
remarks:         This object can only be updated by APNIC hostmasters.
remarks:         To update this object, please contact APNIC
remarks:         hostmasters and include your organisation's account
remarks:         name in the subject line.
remarks:         -+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
changed:         hm-changed@apnic.net 20040913
source:          APNIC

person:          Shin SHIRAHATA
nic-hdl:         SS649-AP
e-mail:          shin@clara.ad.jp
address:         Ariake Frontier Bld. 6F
address:         3-1-25 Ariake
address:         Koto-ku, Tokyo 135-0063
address:         Japan
phone:           +81-3-3599-7321
fax-no:          +81-3-3528-0028
country:         JP
changed:         shin@clara.ad.jp 20060315
mnt-by:          MAINT-JP-CLARA
source:          APNIC

```

Presne rovnaký formát ako pri *RIPE* je použitý aj v prípade *AfriNIC*:

```

% This is the AfriNIC Whois server.

% Note: this output has been filtered.

% Information related to '163.195.0.0 - 163.195.255.255'

inetnum:         163.195.0.0 - 163.195.255.255
netname:         CPA
descr:           Cape Provincial Administration
descr:           Wale Street
descr:           Cape Town 8000
country:         ZA
org:             ORG-CPA1-AFRINIC
admin-c:         JE2-AFRINIC
tech-c:          JE2-AFRINIC
status:          ASSIGNED PI
mnt-by:          AFRINIC-HM-MNT
mnt-lower:       TF-163-195-MNT
source:          AFRINIC # Filtered
parent:          0.0.0.0 - 255.255.255.255

organisation:    ORG-CPA1-AFRINIC
org-name:        Cape Provincial Administration
org-type:        LIR
country:         ZA
address:         Wale Street
address:         Cape Town 8000
address:         ZA
e-mail:          jengel@gov.za
tech-c:          JE2-AFRINIC
mnt-ref:         AFRINIC-HM-MNT
mnt-by:          AFRINIC-HM-MNT
source:          AFRINIC # Filtered

person:          John Engelbrecht
address:         Cape Provincial Administration
address:         Wale Street
address:         Cape
address:         Town
address:         8000
address:         US
phone:           +27 21 462 2780
e-mail:          jengel@gov.za
nic-hdl:         JE2-AFRINIC
source:          AFRINIC # Filtered

```

1.3.2 Doménové meno

Sieťové aplikácie, ktoré zaisťujú komunikáciu medzi počítačmi v Internete používajú pre identifikáciu komunikujúcich uzlov 32 bitové číslo - IP adresu. Pre ľudí je však táto číselná informácia ťažko zapamätateľná, preto sa namiesto IP adresy často používa doménové meno [8] (napríklad namiesto 147.175.1.12 sa používa *stuba.sk*).

K jednej IP adrese môže byť priradených i niekoľko doménových mien. Väzba medzi doménovým menom a IP adresou je definovaná v DNS databáze. **DNS (Domain Name System)** je celosvetovo distribuovaná databáza, ktorej časti sú umiestnené na jednotlivých *name* serveroch.

Vďaka DNS databáze môžeme k IP adrese priradiť doménové meno uzla (takzvaný spätný DNS preklad [8]), ktoré nám toho o geografickej polohe môže veľa napovedať. Napríklad k IP adrese 85.248.20.76 v čase písania tejto časti dokumentu prislúchalo doménové meno *c4.engine.bb.cust.gts.sk*. Podľa domény prvej úrovne (*sk*) vieme pomocou databázy domén prvej úrovne [11] s dosť veľkou pravdepodobnosťou určiť prislúchajúcu krajinu. V doménovom mene však často nájdeme i názov mesta, prípadne jeho skratku (v tomto konkrétnom prípade išlo o prístup z Banskej Bystrice, ktorú v doménovom mene reprezentuje skratka *bb*).

Jedným z problémov tohto prístupu je to, že niektoré domény, napríklad domény webových serverov, sa môžu nachádzať v úplne inej krajine ako by sme predpokladali na základe domény prvej úrovne (napríklad prevádzkovateľ internetovej stránky si môže zaplatiť lacnejší hosting v inej krajine). Ďalej existujú domény, ktoré nevypovedajú o žiadnej príslušnosti k štátu (*org*, *.com*, *net*...).

Ďalším problémom je existencia prípadov, kedy sa spätný preklad IP adresy na doménové meno vôbec nepodarí vykonať (napríklad nesprávnou konfiguráciou DNS serveru).

V mnohých operačných systémoch je štandardne dostupný nástroj *nslookup*, ktorý predstavuje jednoduché rozhranie pre komunikáciu s DNS servermi. Preklad IP adresy na doménové meno je možné vykonať jednoduchým zadaním IP adresy ako parametra:

```
> nslookup 85.248.20.76
Server:  gatekeeper.dcs.elf.stuba.sk
Address:  147.175.99.1

Name:     c4.engine.bb.cust.gts.sk
Address:  85.248.20.76
```

Preložené doménové meno v tomto prípade predstavuje hodnota položky *Name* výstupu.

1.3.3 DNS záznamy

Informácie o doménových menách a im prislúchajúcich IP adresách, ako aj všetky ostatné informácie distribuované pomocou DNS, sú uložené v pamäti DNS serverov v podobe zdrojových viet (**Resource Records – RR**). V prípade, že DNS klient potrebuje získať informácie z DNS, požaduje od *name serveru* konkrétne vety RR [8].

Existuje niekoľko typov zdrojových viet [12], medzi najvýznamnejšie patria:

- **A** – *host address* – 32 bitová IP adresa (IP verzie 4)
- **AAAA** – *Ipv6 address* – 128 bitová IP adresa (IP verzie 6)
- **NS** – *authoritative name server* – doménové meno serveru, ktorý je autoritou pre danú doménu

- **PTR** – *domain name pointer* – doménové meno
- **HINFO** – *host information* – popis hardvéru a softvéru, ktorý používa daný uzol
- **MX** – *mail exchange* – doménové meno mailového serveru spolu s jeho prioritou
- **TXT** – *text string* – textový reťazec s popisom alebo špecifickými údajmi iného protokolu
- **LOC** – *location information* – údaje o geografickej lokalizácii uzla (súradnice)

PTR veta je zaujímavá z pohľadu predchádzajúcej kapitoly, práve ona sa totiž používa pre spätný DNS preklad IP adresy na doménové meno.

Pre lokalizáciu môže byť zaujímavá i veta **TXT**, ktorá môže obsahovať popis zariadenia a v rámci neho napríklad názov alebo skratku mesta, v ktorom sa dané sieťové zariadenie nachádza. Častejšie sa však v tejto vete nachádzajú špecifické informácie iného protokolu, napríklad protokolu *SPF (Sender Policy Framework)*.

Pre lokalizáciu je však najzaujímavejšia veta **LOC**. Ide o rozšírenie systému DNS [13], ktoré umožňuje správcovi jednotlivých domén uložiť do DNS záznamov uzla priamo súradnice jeho lokalizácie. Túto možnosť lokalizácie možno považovať za vysoko dôveryhodnú, nakoľko tento údaj zvyčajne nastavuje priamo samotný administrátor daného zariadenia. Žiaľ, keďže táto možnosť je dobrovoľná a jej použitie závisí len od rozhodnutia príslušného administrátora, existuje v súčasnosti veľmi málo domén s nastaveným *LOC* DNS záznamom [14].

Pre získanie informácií z DNS serveru je možné použiť napríklad nástroj *dig*, ktorý býva súčasťou štandardných *Unix*-ových distribúcií. Napríklad pre získanie viet *LOC* a *TXT* domény *caida.org* je možné použiť tento nástroj nasledovným spôsobom:

```
> dig caida.org LOC
(výstup skrátene)
;; ANSWER SECTION:
caida.org.                3h59m51s IN LOC   32 53 01.000 N 117 14 25.000 W
107.00m 30.00m 10.00m 10.00m

> dig caida.org TXT
(výstup skrátene)
;; ANSWER SECTION:
caida.org.                4H IN TXT   "Cooperative Association for Internet
Data Analysis, La Jolla CA"
```

Ako je možné vidieť zo skrátenejších výstupov, získali sme presné súradnice lokalizácie skúmanej domény a v rámci textového popisu i mesto lokalizácie (La Jolla v Kalifornii).

1.3.4 Použitie nástroja *traceroute*

V prípade, že nedokážeme zistiť lokalizáciu danej IP adresy popísanými metódami, je možné použiť nástroj *traceroute*, ktorý umožňuje získať IP adresy smerovačov na ceste k danej destinácii. Po ich získaní je možné skúsiť lokalizovať poslednú (prípadne predposlednú atď.) IP adresu na ceste k IP adrese, ktorú nie je možné lokalizovať.

Nevýhodou tohto prístupu je nepresnosť lokalizácie, keďže fyzická lokalizácia predchádzajúceho smerovača sa od lokalizácie neznámej IP adresy môže značne líšiť, hlavne v prípade, že sa nepodarí lokalizovať ani posledný, či predposledný smerovač na ceste k IP adrese.

Lokalizáciu neznámych „hopov“ na ceste k destinácii by bolo možné odhadnúť napríklad

i na základe sledovania doby odozvy jednotlivých „hopov“ na vzorke paketov a jej porovnania s odozvou „hopov“ so známou lokalizáciou. Na stanovenie presného miesta lokalizácie by táto metóda však nepostačovala, išlo by len o veľmi hrubý odhad lokalizácie.

Istou nevýhodou z pohľadu implementácie je dĺžka behu nástroja *traceroute*, ktorá sa môže pohybovať i v oblastiach niekoľkých desiatok sekúnd, čo v prípade súčasnej lokalizácie väčšieho súboru IP adries spôsobí značné oneskorenie.

1.3.5 Ďalšie možnosti

Jedným zo spôsobov získania informácií o geografickej polohe IP adresy by mohlo byť i priame použitie voľne dostupných databáz a API popísaných v časti 1.2.2. Je to najjednoduchšia, avšak i najmenej spoľahlivá možnosť získania informácie o geografickej polohe IP adresy. Neexistuje prakticky žiadna záruka korektnosti takto získaných údajov – často sú zastarané a nie je možné zistiť, na základe čoho bola lokalizácia určená. Táto možnosť preto v tejto práci ďalej rozoberaná nebude.

Ďalšiu možnosť ponúkajú poskytovatelia internetového pripojenia. Zo zákona musia ukladať informácie o pripojených používateľoch, pri dynamickom pridelení IP adries aj o časoch pridelenia IP adresy konkrétnemu používateľovi. Tieto údaje sú povinní v prípade potreby poskytnúť orgánom činným v trestnom konaní, ktoré takto dokážu úplne presne určiť ich lokalizáciu, bežný používateľ Internetu sa k nim však nedostane. Často sa teda pri lokalizácii IP adries vyššie popísanými postupmi stáva, že je možné určiť maximálne lokalizáciu poskytovateľa internetového pripojenia.

1.4 Voľba implementačného prostredia

Navrhované riešenie je možné navrhnúť ako samostatnú aplikáciu pre PC alebo vo forme internetovej stránky. Medzi výhody samostatnej aplikácie patrí robustnosť samotného riešenia, keďže programové kódy lokalizácie i vizualizácie by bežali na jednej platforme. Ovládanie takejto aplikácie by tiež bolo prirodzenejšie, s takmer žiadnymi oneskoreniami. Takéto riešenie by však používateľ musel stiahnuť do svojho počítača a nainštalovať potrebné knižnice. Ako nevýhodu možno označiť i platformovú závislosť takéhoto riešenia.

Druhou alternatívou je vyvinutie riešenia vo forme webovej aplikácie. K takejto aplikácii by mal používateľ okamžitý prístup na ktoromkoľvek počítači pripojenom k Internetu. Istou nevýhodou je menšie pohodlie ovládania aplikácie (nutnosť čakania na načítavanie stránky) ako i to, že výkonnosť aplikácie klesá s rastúcim počtom súčasne pripojených používateľov.

Pre implementáciu riešenia bola vybraná istá kombinácia týchto dvoch možností. Výsledná aplikácia bude jednak plniť funkciu samostatnej aplikácie (programovej knižnice) určenej na lokalizáciu IP adries, avšak s konzolovým používateľským rozhraním. Nad touto aplikáciou bude vytvorené webové používateľské rozhranie, v ktorom bude zahrnutá vizualizácia lokalizovaných IP adries na prehľadnej mape.

1.4.1 Implementačné prostredie lokalizácie

Pre implementáciu lokalizačnej časti riešenia je na výber veľké množstvo technológií. Jednou z možností by bolo vyvinúť aplikáciu v **jazyku C**, prípadne C++. Hlavnou výhodou takéhoto riešenia by bola efektivita a rýchlosť programu. Jazyk C je veľmi dobre štruktúrovaný a obsahuje množstvo moderných dátových typov. Do istej miery dokáže byť i platformovo nezávislý. V jazyku C by mohli byť vytvorené binárne knižnice určené k lokalizácii IP adries, ktoré by boli ľahko a efektívne použiteľné v iných aplikáciách. Istou nevýhodou jazyka C je ťažkopádna práca s reťazcami, ktorá by práve v tomto prípade mohla byť veľkou príťažou. Doimplementovaním

grafického používateľského rozhrania (vizualizácie) by sa navyše takáto aplikácia stala platformovo závislou. Jazyk C tiež predstavuje prekážku pri implementácii používateľského rozhrania vo forme webovej aplikácie.

Práve z hľadiska implementácie používateľského rozhrania v podobe webovej aplikácie je vhodná technológia **PHP**. PHP je interpretovaný jazyk vyvinutý špeciálne pre tvorbu www stránok. Medzi jeho výhody patrí predovšetkým jeho jednoduchosť a stručnosť, ale i platformová nezávislosť. Silným argumentom pre výber tohto jazyka je fakt, že je používaný na tvorbu väčšiny malých a stredne veľkých internetových portálov, vďaka čomu má silnú vývojársku podporu a disponuje mnohými modulmi určenými na najrôznejšie účely. Nevýhodou tohto jazyka je, že vyvinutá aplikácia je mimo webového prostredia prakticky nepoužiteľná, vyvinuté riešenie by mohlo byť teda ďalej využité len v rámci webových prezentácií.

Jazyk **Java** by mohol byť ďalšou z možností. Ide o široko používaný interpretovaný objektovo orientovaný programovací jazyk s obrovskou podporou a množstvom dostupných knižníc. Umožňuje vytvoriť znovupoužiteľné knižnice, ktoré sú platformovo nezávislé, je možné nad nimi vystavať i webovú aplikáciu. Veľkou nevýhodou jazyka Java je však pomalosť vykonávania jeho programov a tiež pomerne veľká závislosť od verzie interpretéra.

Pre implementáciu lokalizácie IP adries bol nakoniec vybraný jazyk **Perl**. Najväčšou výhodou tohoto jazyka je nepochybne podpora regulárnych výrazov, ktoré predstavujú neskutočne silný nástroj pre prácu s textovými informáciami [15]. Práve spracovávanie textových informácií (odpovede z *Whois* serverov, doménové mená) je kľúčovou zložkou riešenia lokalizácie IP adries.

Ďalšou obrovskou výhodou je dostupnosť mnohých špecializovaných už naprogramovaných modulov prostredníctvom archívu **CPAN** (*Comprehensive Perl Archive Network*). V tomto archíve sú už mnoho rokov zhromažďované dostupné moduly aj s dokumentáciou, pričom ich počet sa pomaly blíži k číslu 15 000.

Perl je tiež známy jednoduchosťou vyjadrovania. Nič, čo programátor napíše, nie je zbytočné, všetko je maximálne jednoduché a programovanie je veľmi rýchle [16]. Z pohľadu teórie programovacích jazykov je však nutné poznamenať, že *Perl* je i značne neprehľadný. Syntax je veľmi premenlivá a veľké množstvo programových konštrukcií je možné zapisovať niekoľkými i úplne odlišnými spôsobmi. Chovanie mnohých prvkov jazyka záleží od kontextu v ktorom sú použité, čo môže spôsobovať problémy najmä pri čítaní cudzieho kódu [15].

Perl je však veľmi vhodný z hľadiska znovupoužiteľnosti programu. Je možné vytvoriť knižnicu určenú na lokalizáciu IP adries, ktorú je možné spustiť na ktoromkoľvek počítači s nainštalovaným interpretérom jazyka Perl (štandarde ho obsahujú všetky distribúcie OS rodiny *Unix*, ale možné je i doinštalovanie do ostatných OS). Takúto knižnicu tiež nie je problém použiť v inom programe. Vďaka podpore *CGI* (*Common Gateway Interface*) skriptov je možné nad touto knižnicou priamo v jazyku *Perl* vystavať i webové používateľské rozhranie.

V mnohých zdrojoch sa uvádza, že dôsledkom toho, že *Perl* je interpretovaný jazyk, nie je jeho efektívnosť dostatočne vysoká. Uskutočnené testy vyhľadávania reťazcov v rôznych programovacích jazykoch [15] však ukazujú, že *Perl* za väčšinou jazykov vôbec nezaostáva. Moderné interpretéry (*Perl* nevynímajúc) spravidla majú fázu predkompilácie do binárneho medzikódu, ktorý následne vykonávajú. Vďaka tomu program beží dostatočne rýchlo. Jedinou daňou, ktorú musia tieto interpretované programy zaplatiť, je úvodná fáza predkompilácie, ktorá prebieha pred každým spustením. Jej podiel na celkovej dobe behu programu však býva zanedbateľný [15].

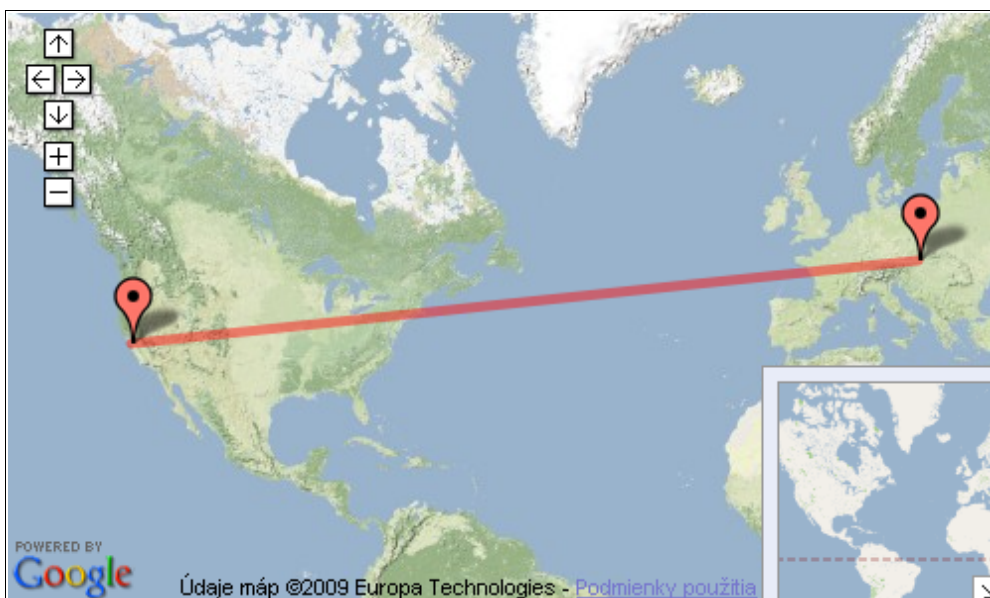
1.4.2 Implementačné prostredie vizualizácie

Keďže vizualizácia lokalizovaných IP adries bude prebiehať v podobe webovej aplikácie, zužuje sa

okruh možných riešení na niekoľko webových technológií. Keďže pre implementáciu lokalizácie IP adres bolo vybrané prostredie jazyka *Perl*, pre generovanie šablón a údajov používateľského rozhrania v *HTML* a *JavaScripte* bude využitá podpora **CGI (The Common Gateway Interface)** tohto jazyka. Pre samotnú vizualizáciu, teda zobrazenie lokalizácie na prehľadnej mape, existuje niekoľko možných riešení.

Jednou z možností by mohlo byť **priame vykresľovanie obrázku s mapou** prostredníctvom jazyka *Perl*, prípadne použitím jednej z voľne dostupných knižníc určených pre vykresľovanie na obrázok, resp. mapu. Takéto riešenie by bolo priamočiare, nakoľko by nedošlo k zmiešavaniu použitých technológií, na druhej strane by však bolo oveľa menej príjemné pre používateľa, nakoľko by neponúkalo dostatočnú interakciu. Implementácia by tiež bola náročnejšia, nakoľko by bolo nutné prepočítavať geografické súradnice na príslušné súradnice v obrázku, pri čom by pri nedokonalých mapových podkladoch mohlo dochádzať k nepresnostiam.

Oveľa prívetivejšie používateľské prostredie ponúka použitie **Google Maps API** [17], vďaka ktorému je možné prostredníctvom *JavaScriptu* vložiť mapy *Google Maps* na ktorúkoľvek internetovú stránku. Súčasťou API (*Application Programming Interface*) je množstvo funkcií pre manipuláciu s týmito mapami a ich obsahom. Výhodou použitia tohto riešenia je jednoduchosť implementácie i ovládania, veľká používateľská prívetivosť (podpora jednoduchého približovania a vzdďalovania, posunu v mape) a vysoká podrobnosť týchto máp. V rámci API je priamo zabudovaná i podpora vykresľovania bodov a priamok na mapu (Obr. 2).



Obr. 2: Príklad zobrazenia mapy *Google Maps* s vykreslenou priamkou

Alternatívou k použitiu *Google Maps API* je knižnica **OpenLayers** [18]. Ide o voľne dostupnú knižnicu založenú na *JavaScripte* ponúkajúcu takmer totožné funkcie ako *Google Maps*. Štandardná podkladová mapa je však oveľa menej podrobná, nezobrazujú sa ani názvy jednotlivých krajín, či miest (Obr. 3), čo však na druhej strane prináša vyššiu rýchlosť vykresľovania mapy. Výhodou tejto knižnice je, že je možné definovať si vlastný mapový podklad vo forme jedného, či viacerých obrázkov, prípadne si nakonfigurovať vlastný server, ktorý bude slúžiť na medziukladanie (*cache*) mapových podkladov, čím je možné výrazne zvýšiť rýchlosť vykresľovania.



Obr. 3: Príklad zobrazenia implicitnej mapy knižnice OpenLayers

1.4.3 Databázové prostredie

Aplikácia pri lokalizácii využíva údaje, ktoré je nutné ukladať v databáze. Ide predovšetkým o veľmi rozsiahlu databázu súradníc jednotlivých miest sveta, v ktorej je potrebné dosahovať veľkú rýchlosť vyhľadávania textových položiek. Databáza ponúka najefektívnejšie riešenie tejto požiadavky.

Medzi najpoužívanejšie *open source* databázové prostredia patria **MySQL** a **PostgreSQL**. O výhodách a nevýhodách týchto databázových systémov sa medzi odborníkmi vedie mnoho diskusií a sporov. Skutočnosť je taká, že oba systémy majú svoje výhody i nevýhody a rozhodnutie o použití jedného, či druhého z nich závisí od potrieb vyvíjanej aplikácie. Databázový systém **MySQL** je zaujímavý predovšetkým pre vysokú rýchlosť pri vyhľadávaní údajov, či jednoduchosť používania a veľmi dobrú technickú podporu. Tento databázový systém sa práve preto stal najpoužívanejším v rámci malých a jednoduchých aplikácií, napríklad pri vývoji internetových stránok. **PostgreSQL** na druhej strane dominuje svojou komplexnosťou a množstvom nadštandardných funkcií, akými sú napríklad vnorené dopyty, referenčná integrita, alebo používateľsky definované funkcie, čo tento systém predurčuje predovšetkým pre použitie v komplexných aplikáciách kladúcich dôraz na stabilitu systému.

Oba spomínané systémy sú implementované ako samostatné serverové procesy. Programy, ktoré chcú pristupovať k databáze s nimi komunikujú niektorým zo spôsobov medziprocesovej komunikácie (zvyčajne prostredníctvom *TCP/IP* komunikácie). To prináša isté nevýhody v prípade, že navrhovaná aplikácia má byť prenositeľná alebo distribuovaná na viacero počítačov. V tomto prípade by bola nutná dodatočná inštalácia a konfigurácia databázového systému a v rámci inštalácie aj import dát do tejto databázy. Inou možnosťou by bolo použitie vonkajšieho centralizovaného databázového servera s aplikačnými dátami, na ktorý by pristupovali všetky inštancie aplikácie prostredníctvom Internetu. Takáto centralizácia však prináša znižovanie výkonnosti aplikácie so stúpajúcim počtom používateľov, ako i možnosť úplnej nefunkčnosti aplikácie pri výpadku databázového serveru, prípadne jeho nedostupnosti.

Všetky vyššie spomínané problémy možno vyriešiť použitím databázového stroja **SQLite** [19]. Proces, ktorý pristupuje k databáze **SQLite** načítava a zapisuje dáta priamo z / do databázového súboru na disku. Neexistuje žiadny sprostredkujúci proces plniaci funkciu serveru, a preto nie je potrebná ani žiadna inštalácia, konfigurácia a správa takejto služby. Každý proces, ktorý má právo na čítanie databázového súboru môže priamo používať túto **SQLite** databázu.

Keďže kompletná **SQLite** databáza obsahujúca všetky tabuľky, indexy, spúšťače (*triggery*) atď. je

uložená v jedinom samostatnom súbore, môže byť ľahko distribuovaná spolu s ostatnými zdrojovými, či binárnymi kódmi aplikácie a môže byť použitá okamžite po spustení aplikácie.

Nakoľko ideou navrhovanej aplikácie je, aby konzolová aplikácia slúžiaci na lokalizáciu IP adres bola prenositeľná a nezávislá od vonkajších systémov, je voľba databázového systému *SQLite* v tomto prípade jednoznačná. Vďaka tomu, že pre implementáciu navrhovanej aplikácie bolo vybrané práve toto databázové prostredie, je možné vytvoriť aplikáciu používajúcu databázu bez toho, aby používateľ musel importovať dáta databázy, konfigurovať databázový server, ba dokonca i bez toho, aby vôbec vedel, že aplikácia nejakú databázu používa.

2 Návrh riešenia

V tejto časti dokumentu sa nachádza podrobný návrh algoritmu pre lokalizáciu IP adres, ktorý bol vytvorený na základe analyzovaných metód lokalizácie popísaných v predchádzajúcej kapitole. Ďalej obsahuje konkrétny návrh konzolovej aplikácie pre lokalizáciu IP adres a webovej aplikácie pre vizualizáciu lokalizácie IP adres.

2.1 Špecifikácia požiadaviek

Navrhnuté riešenie by malo umožňovať určenie čo najpresnejšej geografickej polohy zariadenia so zadanou IP adresou. Podľa možnosti a dostupnosti jednotlivých informácií o IP adrese by malo byť schopné učiť krajinu a mesto lokalizácie tohto zariadenia, prípadne i názov organizácie, ktorej bola táto IP adresa pridelená. Informácia o polohe zariadenia má byť graficky zobrazené na mape.

Riešenie by malo umožniť i zadanie súboru (postupnosti) IP adres a následné zobrazenie tejto postupnosti ako cesty na mape. V prípade vstupu viacerých postupností by malo byť schopné odlišiť menej vyťažené cesty od viacej vyťažných (napríklad odlišnou farbou spojových úsečiek).

Výsledky riešenia by mali byť použiteľné i v praxi, či už vytvorením programovej knižnice poskytujúcej funkciu lokalizácie IP adres, samostatnej aplikácie, či vo forme webovej aplikácie dostupnej širokej verejnosti, prípadne ich kombináciou.

2.2 Návrh lokalizácie IP adres

Lokalizácia IP adres bude prebiehať na základe vybraných metód popísaných v kapitole 1.4. Keďže každá z týchto metód môže vrátiť rôzne výsledky lokalizácie, je nutné rozhodnúť, ktorý z výstupov bude považovaný za výsledný. Toto je zabezpečené prostredníctvom **skóre**, ktoré sa priradí výstupom jednotlivých metód lokalizácie (Tab. 1).

Metóda lokalizácie	Výška skóre	
	Určenie len krajiny	Určenie krajiny i mesta
LOC DNS záznam	4	10
whois	3	8
doménové meno	2	6
TXT DNS záznam	1	6

Tab. 1: Výška skóre pre jednotlivé metódy lokalizácie

Určovanie lokalizácie prebieha sekvenčne od metódy, ktorá má určené najvyššie skóre (*LOC DNS záznam*) po metódu s najnižším skóre (*TXT DNS záznam*). V prípade, že sa podarí určiť krajinu zhodnú s krajinou už určenou inou metódou (prípadne, ak krajina dovtedy ešte nebola určená), k celkovému skóre sa **pripočíta skóre uvedené v tabuľke Tab. 1** v stĺpci „Určenie len krajiny“. Ak sa podarí určiť i mesto lokalizácie, pripočíta sa skóre uvedené v stĺpci „Určenie krajiny i mesta“.

Ak sa však určená lokalizácia (mesto či krajina) nezhoduje s dovtedy určenou lokalizáciou, od celkového skóre sa **odpočíta polovica príslušného skóre** uvedeného v tabuľke *Tab. 1* (v prípade neprirodzeného čísla sa desatinná časť výrazu oreže).

Po aplikovaní všetkých metód lokalizácie sa na základe celkového skóre vyhodnotí i relatívna presnosť určenia správnej lokalizácie stanovená na základe údajov uvedených v tabuľke *Tab. 2*.

Celkové skóre	Relatívna presnosť
0 - 5	veľmi nízka
6 - 13	nízka
14 - 17	vysoká
18 - 30	veľmi vysoká

Tab. 2: Relatívna presnosť určenia správnej lokalizácie podľa výšky skóre

Výnimkou z popísaného postupu je prípad, kedy je analýzou doménového mena nájdené mesto lokalizácie a dovtedy určené mesto je lokalizované len na základe údajov z Whois. V prípade, že sú tieto dve mestá rôzne, uprednostní sa mesto získané analýzou doménového mena. Táto výnimka bola zavedená na základe výsledkov testovania - takáto metóda je presnejšia pri lokalizácii IP adres smerovačov chrbtice Internetu (prejavuje sa pri lokalizácii jednotlivých „hopov“ získaných nástrojom *traceroute*).

Aplikácia neurčuje presnú adresu lokalizácie IP adresy (ulicu a číslo). Presnú adresu je možné získať len dvomi z implementovaných spôsobov lokalizácie. Pri lokalizácii na základe *LOC DNS* záznamu však nie je možné previesť súradnice na presnú adresu (kvôli obmedzeniu používanej databázy, v ktorej sa nachádzajú lokalizačné údaje len po úroveň miest - vyžadovalo by to oveľa rozsiahlejšiu databázu alebo použitie externej služby tretej strany). Údaje o presnej adrese, získané na základe údajov z *Whois* serverov zase takmer nikdy nekorešponujú so skutočnou lokalizáciou zariadenia s danou IP adresou (ide predovšetkým o kontaktné adresy poskytovateľov internetového pripojenia a správcov daného adresného priestoru). Nakoľko tieto adresy by používateľa mohli skôr zmiasť ako poskytnúť relevantné údaje, prebieha lokalizácia IP adres len po úroveň miest. Pre vytvorenie lepšieho obrazu o príslušnosti IP adresy však bude k IP adrese priradený textový popis, ktorý bude v prípade dostupnosti takejto informácie obsahovať napr. názov organizácie, ktorej bola IP adresa pridelená.

V rámci štandardného správania výslednej aplikácie budú aplikované všetky uvedené metódy lokalizácie v takom poradí, ako sú uvedené v tabuľke *Tab. 1*. Explicitným zadaním je však možné špecifikovať iba určené metódy lokalizácie, prípadne zapnúť tzv. rýchly mód - ukončenie určovania lokalizácie hneď po prvom úspešnom určení presnej lokalizácie (lokalizácie mesta).

2.2.1 Lokalizácia na základe LOC DNS záznamu

Pri tomto spôsobe lokalizácie sa program pokúsi získať z DNS príslušnú vetu *LOC* (*LOC DNS* záznam) [14], ktorá obsahuje údaje o geografickej lokalizácii uzla – súradnice a nadmorskú výšku.

Po získaní geografických súradníc je nutné zistiť informácie o mieste tejto polohy - krajinu a mesto lokalizácie. Pre tento účel je možné využiť databázu súradníc miest sveta. Získané súradnice sa však takmer nikdy nebudú priamo zhodovať so súradnicami jedného z miest z databázy. Je preto nutné krajinu a mesto nejakým spôsobom odhadnúť.

Odhad mesta a krajiny polohy danej geografickými súradnicami je možné vykonať jediným *SQL* dopytom. Získané súradnice, ako i porovnávané súradnice v databáze sa zaokrúhľia na jedno desatinné miesto, čím sa dosiahne určitý rozptyl súradníc (plocha s obsahom cca 100 km²). Vybrané mestá spadajúce do tejto oblasti sa zoradia podľa vzdialenosti ich súradníc od určovaného bodu (*distance*). Záznam s najmenšou vzdialenosťou od určovaného bodu sa považuje za určené miesto lokalizácie. Celý *SQL* dopyt pre získanie informácií o mieste zadanej

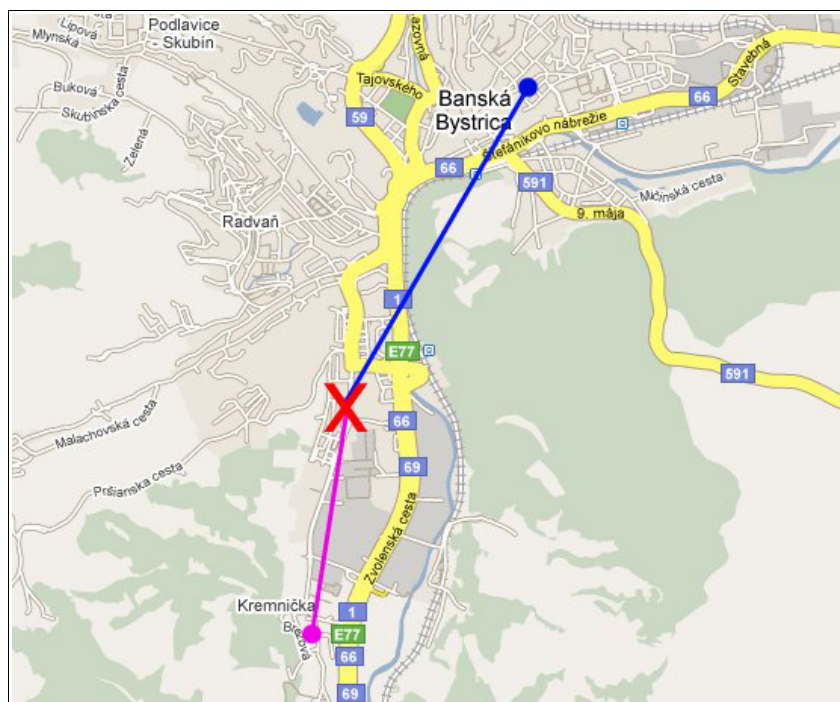
geografickej polohy je uvedený v nasledujúcom výpise (obsah premenných: $\$lat$ – geografická šírka, $\$lon$ – geografická dĺžka).

```
SELECT country, city, sqrt((latitude- $\$lat$ )*(latitude- $\$lat$ )+(longitude- $\$lon$ )*(longitude- $\$lon$ )) AS distance FROM geo_city WHERE ROUND(latitude, 1) = ROUND( $\$lat$ , 1) AND ROUND(longitude, 1) = ROUND( $\$lon$ , 1) ORDER BY distance LIMIT 1
```

Je nutné poznamenať, že vypočítaná vzdialenosť medzi dvoma bodmi zadanými v geografických súradniciach použitím Pytagorovej vety je dosť odlišná od skutočnej vzdialenosti, nakoľko sa pri tomto výpočte neberie do úvahy zakrivenie Zeme. Pre výpočet skutočnej vzdialenosti je potrebné použiť oveľa zložitejší výpočet využívajúci goniometrické funkcie (popísané v kapitole 2.3). V tomto prípade však nie je dôležité poznať skutočnú vzdialenosť, hlavným cieľom je nájsť najlepšie vyhovujúci záznam v databáze s veľkým množstvom údajov. Preto je oveľa výhodnejšie použiť tento jednoduchší a efektívnejší algoritmus, ktorý na tento účel úplne postačuje.

Pri tomto algoritme určovania názvu mesta zadanej geografickej polohy dochádza k istým nepresnostiam, hlavne v prípade lokalizácie okrajových častí väčších miest. V takomto prípade môže dôjsť k situácii, že vzdialenosť k inej obci za mestom je menšia, ako vzdialenosť k bodu, ktorý predstavujú súradnice mesta uložené v databáze.

Túto situáciu ilustruje obrázok Obr. 4. Miesto označené červeným krížikom sa nachádza v okrajovej časti mesta *Banská Bystrica*. Súradnice tohto mesta, ktoré sú uložené v databáze však popisujú bod v centre mesta, ku ktorému je vzdialenosť od referenčného bodu (modrá úsečka) väčšia, ako vzdialenosť k obci *Kremnička* (ružová úsečka) za mestom. Preto je v tomto prípade popísaným algoritmom nesprávne určený názov obce *Kremnička* namiesto obce *Banská Bystrica*.



Obr. 4: Problém nesprávneho určenia názvu obce na okraji väčšieho mesta

2.2.2 Lokalizácia na základe údajov z Whois databáz

Tento postup lokalizácie využíva podrobné textové údaje o IP adrese načítané z *Whois* databáz. Po načítaní obsahu odpovede na *Whois* požiadavku je nutné identifikovať, od ktorého

regionálneho registrátora táto odpoveď prišla. Je to dôležité hlavne z toho hľadiska, že rôzni regionálni registrátori majú rôzne formáty odpovedí (kapitola 1.4.1).

V prípade, že odpoveď obsahuje reťazec `arin whois database`, je táto odpoveď spracovávaná ako odpoveď amerického regionálneho registrátora **ARIN**. V tomto prípade sa určí krajina a mesto lokalizácie priamo z jednotlivých záznamov *Country* a *City* odpovede. Po identifikovaní krajiny a mesta sa z databázy súradníc jednoduchým výberom získajú súradnice tohto miesta. V prípade, že súradnice mesta nie sú v databáze (ak ide o veľmi malé mesto), prípadne ak z nejakého dôvodu nedôjde k spárovaniu mesta získaného z Whois odpovede s názvom mesta v databáze (napr. *St. Louis* vo Whois odpovedi a *Siant Louis* v databáze), použije sa aj záznam *StateProv* odpovede a súradnice sa určia len na základe krajiny (napr. USA) a štátu (napr. CA - California).

Odpovede ostatných regionálnych registrátorov sa spracovávajú iným spôsobom. Krajinu je v každom prípade možné získať zo záznamu *country* objektu *inetnum*. Keďže však neexistuje žiadne špecifické miesto, kde by bolo možné v každom prípade nájsť mesto lokalizácie, ju tu nutné použiť istý druh heuristickej analýzy.

Zostaví sa vyhľadávacie pole, obsahujúce reťazce, v ktorých by sa mohlo nachádzať slovo alebo slovné spojenie tvoriace názov mesta. Priorita jednotlivých reťazcov klesá s rastúcim indexom poľa (čiže reťazec s indexom 0 sa považuje za významnejší ako reťazec s indexom 1).

Ak odpoveď Whois serveru obsahuje reťazec `whois.lacnic.net`, znamená to, že odpoveď prišla od latinsko-amerického regionálneho registrátora **LACNIC**. V tomto prípade sa do spomínaného poľa uložia v poradí podľa priority (od indexu 0) tieto údaje:

1. záznam *owner* objektu *inetnum*
2. záznam *address* objektu *inetnum*
3. záznam *address* objektu *organisation*
4. záznam *address* objektu *person*
5. záznam *address* objektu *role*

V prípade regionálnych registrátorov **RIPE**, **AfriNIC** a **APNIC**, ktorí používajú rovnaké formáty odpovedí sa do vyhľadávacieho poľa v poradí podľa priority uložia nasledovné údaje:

1. záznam *owner* objektu *inetnum*
2. záznam *address* objektu *inetnum*
3. záznam *address* objektu *organisation*
4. záznam *address* objektu *person*
5. záznam *address* objektu *role*

Viacriadkové záznamy sú rozdelené po jednotlivých riadkoch a vložené do poľa v poradí od posledného riadku k prvému (keďže ide o záznamy obsahujúce adresu, mesto zvyčajne býva uvedené na poslednom riadku, analogicky s klasickými poštovými listami). Ak niektorý z uvedených záznamov alebo objektov vo Whois odpovedi chýba, pokračuje sa ďalším záznamom.

Napríklad pre príklad odpovede Whois serveru regionálneho registrátora **RIPE** uvedeného v kapitole 1.4.1 by obsah vyhľadávacieho poľa zostavený podľa vyššie popísaného algoritmu vyzeral nasledovne:

```
0          SK-ENGINE-BB
1          POP Banska Bystrica
2          Rudlovska cesta 2, 97401, Banska Bystrica
3          ENGINE Slovakia, s.r.o.
4          Slovakia
5          97404
6          Banska Bystrica
```

```
7           Mladeznicka 17
8           Richard Sudoma
9           Slovak Republic
10          851 01
11          Bratislava
12          Einsteinova 24
13          Aupark Tower
14          GTS Nextra a.s.
```

Obsah tohto poľa je sekvenčne prechádzaný a každý reťazec je porovnávaný s databázou súradníc miest uložených v tabuľke `geo_city`. Z tabuľky sú vybrané len záznamy tých miest, ktoré patria do už zistenej krajiny. Stĺpec databázovej tabuľky obsahujúci názov mesta (`city`) je použitý ako regulárny výraz pre vyhľadávanie v skúmanom reťazci. Pre odfiltrovanie náhodných výskytov názvu mesta v reťazcoch, ktoré názov mesta vôbec neobsahujú sú vyhľadávané len mestá, ktorých názov je dlhší ako 3 znaky a pri nájdení viacerých názvov miest v jednom reťazci je uprednostnené to mesto, ktoré je v reťazci na pozícii viac vľavo. Mesto tiež nemôže byť súčasťou väčšieho slova, musí ísť o samostatné slovo, v prípade viacslovných názvov miest skupinu slov, ktoré sú od ostatného textu v prehľadávanom reťazci oddelené prázdny znakom. Zápis tohto algoritmu do jazyka SQL je uvedený v nasledujúcom výpise (premenná `$country` obsahuje určenú krajinu a premenná `$line` obsahuje práve skúmaný reťazec vyhľadávacieho poľa):

```
SELECT city, latitude, longitude FROM geo_city WHERE country LIKE
'$country' AND length(city) > 3 AND '$line' REGEXP city ORDER BY
length(city) DESC LIMIT 1
```

Ak sa v skúmanom reťazci poľa nájde zhoda s mestom v databáze, je toto mesto spolu s jeho súradnicami získanými z databázy považované za lokalizáciu IP adresy a sekvenčne prechádzanie vyhľadávacieho poľa končí. Ak sa zhoda nenašla, pokračuje vyhľadávanie nasledujúcim prvkom vyhľadávacieho poľa.

Údaje z databázy *Whois* sú použité aj na získanie textového popisu IP adresy – organizácie, ktorej bola táto IP adresa pridelená. V odpovedi *Whois* serveru regionálneho registrátora **ARIN** je priamo použitá hodnota záznamu *OrgName*. V prípade **RR RIPE**, **AfrINIC** a **APNIC** je vybraný prvý riadok záznamu *descr* objektu *inetnum* a v prípade **RR LACNIC** záznam *owner* objektu *inetnum*.

K nesprávnemu alebo žiadnemu určeniu polohy dôjde v prípade, že z *Whois* odpovede nie je možné vyčítať žiadne lokalizačné údaje (v niektorých prípadoch odpoveď vôbec neobsahuje bližšie lokalizačné údaje, prípadne nastávajú problémy s kódovaním reťazca).

2.2.3 Lokalizácia na základe analýzy doménového mena

Pri tomto druhu lokalizácie je vstupným údajom doménové meno (*hostname*) prislúchajúce analyzovanej IP adrese (ak danú IP adresu je možné na doménové meno preložiť).

Podľa domény prvej úrovne (napr. v prípade Slovenska *sk*) je možné odhadnúť krajinu lokalizácie zariadenia s danou IP adresou. Existujú však i také domény prvej úrovne, podľa ktorých sa krajina odhadnúť nedá (napr. *net*, *eu*, *com*). Proces lokalizácie IP adresy na základe analýzy doménového mena sa preto delí na dve hlavné vetvy – keď krajinu z doménového mena je možné určiť, a keď jej určenie možné nie je. Celý tento proces je znázornený sekvenčným diagramom na obrázku *Obr. 5*.

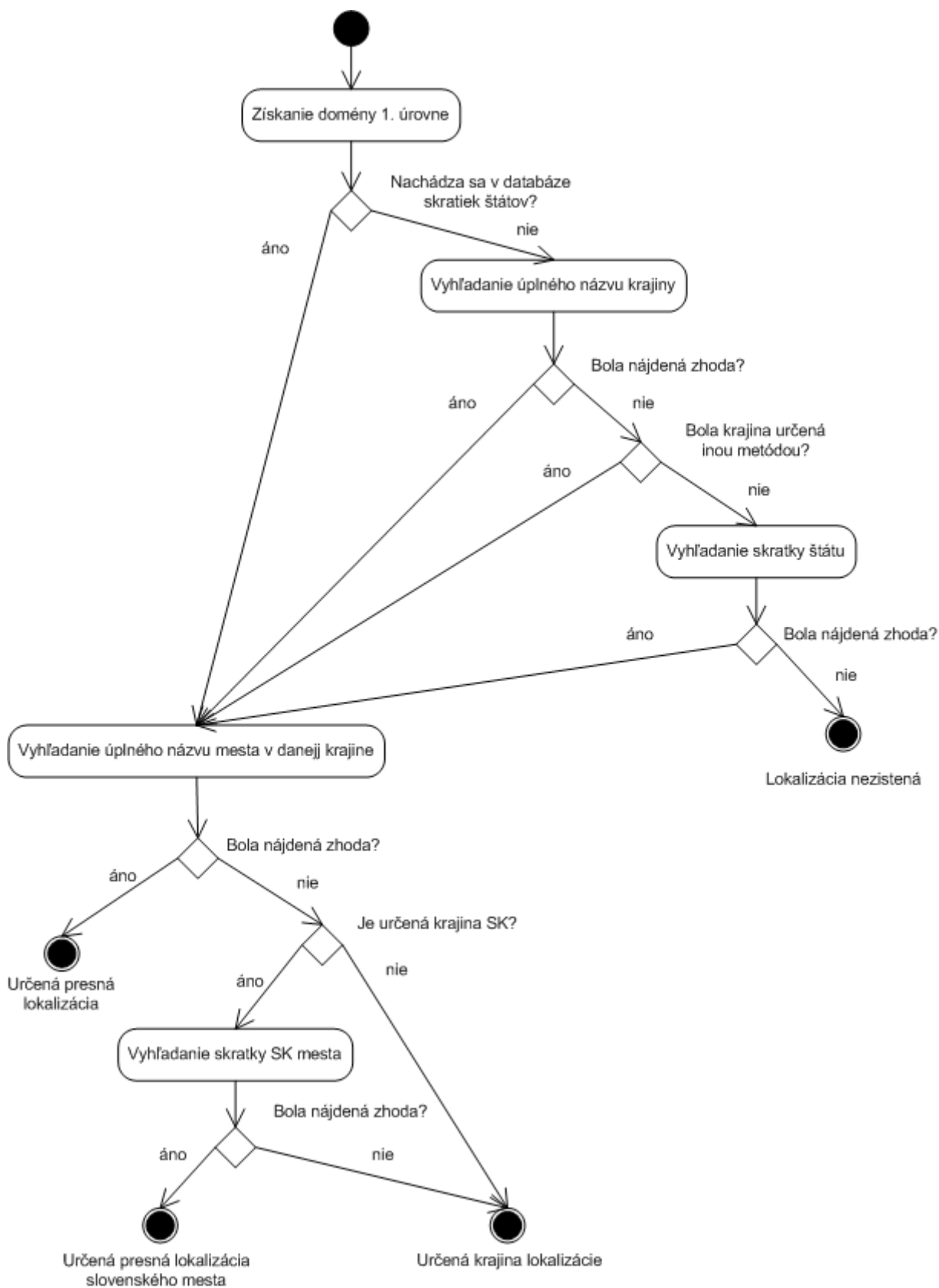
V prípade, že je známa krajina lokalizácie IP adresy, je v doménovom mene vyhľadávaný úplný názov mesta patriaceho do príslušnej krajiny. Na toto vyhľadávanie je použité porovnanie reťazca na základe databázy súradníc miest uložených v databázovej tabuľke `geo_city`, kde stĺpec `city` tejto tabuľky, obsahujúci názov mesta, je použitý ako regulárny výraz pre vyhľadávanie v doménovom mene (toto vyhľadávanie je podrobne popísané v časti 2.2.2). Doménové meno sa predtým rozanalyzuje na postupnosť slov, pričom platí, že každé slovo je postupnosť písmen

oddelená nepísmenovým znakom alebo začínajúca veľkým písmenom. Rovnaký princíp je použitý i v rámci ostatných vyhľadávaní v doménovom mene, popísaných ďalej.

Ak v doménovom mene nebol nájdený výskyt plného názvu mesta príslušnej krajiny, je v prípade IP adres, ktorých krajina lokalizácie bola určená mimo územia Slovenska vyhľadávanie ukončené. V prípade slovenských IP adres sa program pokúsi o fulltextové vyhľadanie skratky mesta na základe dvojnakových skratiek názvov slovenských miest uložených v tabuľke *sk_spz*. Podobné vyhľadávanie by bolo možné vytvoriť aj pre ostatné krajiny, vyžadovalo by to však zostavenie kompletnej databázy skratiek všetkých miest sveta. Keďže takúto databázu nemáme k dispozícii, bolo na ukážku implementované aspoň vyhľadávanie v rámci slovenských miest.

Ak krajinu na základe domény prvej úrovne nie je možné určiť (napr. doména *.net*), program sa pokúsi v doménovom mene vyhľadať úplný názov krajiny. Ak sa nenájde zhoda, použije sa pre vyhľadávanie mesta krajina, ktorá už bola určená inými metódami lokalizácie, popísanými v predchádzajúcich kapitolách.

Ak ani na základe metód popísaných v predchádzajúcich častiach dokumentu nebola určená krajina lokalizácie IP adresy, program sa pokúsi určiť krajinu na základe výskytu samostatne stojacej skratky štátu (t.j. dvojnaková skratka oddelená od okolitého textu nealfanumerickým znakom) na ktorejkoľvek pozícii v doménovom mene.



Obr. 5: Proces lokalizácie IP adresy na základe analýzy doménového mena

2.2.4 Lokalizácia na základe TXT DNS záznamu

Pri tomto spôsobe lokalizácie sa program pokúsi získať z DNS príslušnú vetu *TXT* (*TXT DNS* záznam), ktorá obsahuje textový údaj, v ktorom sa môže vyskytovať napríklad krajina alebo mesto lokalizácie daného sieťového uzla.

V získanom texte je najprv vyhľadávaný úplný názov krajiny. V prípade, že sa krajinu nepodarilo nájsť, použije sa krajina zistená predchádzajúcimi metódami lokalizácie. Potom sa v reťazci vyhľadáva úplný názov mesta príslušnej krajiny.

Pre fulltextové vyhľadávanie je opäť použité porovnávanie reťazca na základe databázy súradníc miest uloženej v databázovej tabuľke *geo_city*, kde stĺpec *city* tejto tabuľky, obsahujúci názov mesta, je použitý ako regulárny výraz pre vyhľadávanie v reťazci získanom z *TXT DNS* záznamu (systém vyhľadávania je podrobne popísaný v časti 2.2.2).

V prípade, že sa v texte *TXT DNS* záznamu podarilo nájsť mesto lokalizácie, použije sa *TXT DNS* záznam aj ako textový popis IP adresy (namiesto popisu získaného z *Whois*).

2.2.5 Lokalizácia s použitím nástroja traceroute

V kapitole 1.4.4 bolo popísané použitie nástroja *traceroute* ako poslednej záchrany v prípade, že sa nám nepodarí zistiť žiadne lokalizačné údaje o skúmanej IP adrese. V praxi je však vo valnej väčšine prípadov na základe vyššie popísaných postupov možné určiť aspoň krajinu lokalizácie. Výnimočné prípady, kedy sa o IP adrese nedajú získať naozaj žiadne lokalizačné údaje väčšinou aj tak zastupujú smerovače chrbticových častí Internetu, kde predchádzajúci smerovač na ceste k nemu zvyčajne býva v celkom inej krajine.

Z týchto dôvodov nemá veľký význam používať nástroj *traceroute* v samotnom procese určovania lokalizácie IP adresy. Trasovanie k zadanej IP adrese, spojené s lokalizáciou každého známeho smerovača na ceste k nej je však v navrhnutej aplikácii možné explicitne spustiť prostredníctvom používateľského rozhrania.

2.3 Výpočet vzdialenosti medzi IP adresami

Aplikácia umožňuje výpočet vzdialenosti medzi dvoma lokalizovanými IP adresami. Vzdialenosť je možné vypočítať na základe známych súradníc dvoch bodov, pre výpočet je použitý tzv. **Haversineho vzťah** [20]. Ide o výpočet najkratšej vzdialenosti medzi dvoma bodmi na povrchu gule, čo predstavuje malé zanedbanie vzhľadom na to, že povrch Zeme je mierne elipsoidný, pre účely našej aplikácie je však tento vzorec plne postačujúci.

R	- polomer Zeme (6371 km),
φ_1, φ_2	- zemepisné šírky bodu 1 a 2 (v radiánoch),
$\Delta\varphi = \varphi_1 - \varphi_2$	- rozdiel zemepisných šírok bodov 1 a 2 (v radiánoch)
$\Delta\lambda$	- rozdiel zemepisných dĺžok bodov 1 a 2 (v radiánoch)

$$a = \sin\left(\frac{\Delta\varphi}{2}\right)^2 \cdot \cos(\varphi_1) \cdot \cos(\varphi_2) \cdot \sin\left(\frac{\Delta\lambda}{2}\right)^2$$

$$d = R \cdot 2 \cdot \operatorname{atan2}(\sqrt{a} \cdot \sqrt{1-a})$$

Funkcia $\operatorname{atan2}(x, y)$ počíta *arkustangens* (inverzný tangens) podľa zadaného bodu so súradnicami x a y - uhlu, ktorý zvierá os x s priamkou, ktorá prechádza počiatkom súradnicovej sústavy a bodom so súradnicami (x, y) .

2.4 Návrh konzolovej aplikácie pre lokalizáciu IP adries

Výsledkom prvej časti riešenia projektu je konzolová aplikácia určená na lokalizáciu IP adries. IP adresy sú programu zadávané v podobe parametrov príkazového riadku, zo súboru, alebo štandardného vstupu. Výstup programu obsahuje výpis IP adries s im prislúchajúcimi lokalizačnými údajmi – krajina, mesto a súradnice lokalizácie, textový popis a skóre (relatívna presnosť určenia lokalizácie). Namiesto IP adresy je možné zadať i doménové meno uzla, program ho automaticky prevedie na príslušnú IP adresu.

Prepínačmi je možné špecifikovať len niektorú z metód lokalizácie, prípadne zvoliť rýchly režim, ktorý ukončí vyhľadávanie lokalizácie okamžite po prvom úspešnom pokuse určenia polohy IP adresy.

Aplikácia disponuje možnosťou zobrazenia vzdialeností medzi jednotlivými IP adresami, ako i možnosťou vyhodnotenia dvoch zadaných IP adries takým spôsobom, že na výstupe je zobrazená len vzdialenosť medzi nimi.

Použitím špeciálneho prepínača je tiež možné spustiť trasovanie (*traceroute*) k zadanej IP adrese, ktorého výstupom sú lokalizácie jednotlivých smerovačov na ceste k tejto IP adrese.

2.5 Návrh webovej aplikácie pre vizualizáciu lokalizácie IP adries

Vizualizáciu lokalizácie IP adresy zabezpečujú API funkcie *Google Maps* [17] popísané v časti 1.5.2.

Rozhraním medzi modulmi lokalizácie naprogramovanými v jazyku *Perl* a webovou aplikáciou poskytujúcou používateľské rozhranie pre vizualizáciu sú *CGI* skripty napísané v jazyku *Perl*, ktoré spracovávajú údaje vložené používateľom a generujú príslušné kódy *HTML* a *JavaScriptu*, ktorých kombinácia vytvorí vizuálne zobrazenie polohy IP adresy, prípadne ich postupností.

Aplikácia poskytuje možnosť zobrazenia lokalizácie jednej IP adresy, ako i postupnosti adries (cesty) v podobe bodov spojených úsečkami. V prípade, že používateľ zadá viacej postupností IP adries, budú farebne odlišené najmenej vyťaženej linky (zelená farba) od najvyťaženejších (červená farba). Vyťaženie liniek medzi týmito dvoma okrajovými hodnotami je znázornené príslušnými odtieňmi zelenej, resp. červenej farby.

Okrem zobrazenia polohy IP adries na mape aplikácia ponúka i zobrazenie lokalizačných údajov - krajiny, mesta a súradníc lokalizácie, textového popisu a skóre (relatívnej presnosti určenia lokalizácie).

Jednou z možností aplikácie je i trasovanie – vizuálny *traceroute*. Ide o nástroj, ktorý umožňuje vizualizovať cestu sieťových paketov Internetom. Vizualizácia spočíva v lokalizácii IP adries jednotlivých sieťových smerovačov na ceste k zadanej destinácii. Tento nástroj umožňuje vizualizáciu cesty sieťovej prevádzky Internetom od webserveru umiestneného v sídle Fakulty informatiky a informačných technológií STU v Bratislave k destinácii, ktorú zadá používateľ.

Pri používaní webovej aplikácie pre vizualizáciu lokalizácie IP adries dochádza často k tomu, že počas jedného sedenia (*session*) používateľa dochádza viackrát k lokalizácii tej istej IP adresy. Ide predovšetkým o samotnú IP adresu používateľa, ktorá sa automaticky lokalizuje pri každom prístupe na hlavnú stránku webu. K viacnásobnej lokalizácii však dochádza aj pri vizualizácii lokalizácie postupností IP adries a vizuálnom *traceroute* (niekoľko prvých „hopov“ je do väčšiny destinácií vždy rovnakých).

Pre zefektívnenie a urýchlenie práce aplikácie boli tieto prípady vyriešené zavedením **dočasnej pamäte lokalizovaných IP adries**.

Po každej úspešnej lokalizácii IP adresy program uloží zistené údaje do databázy. Pred každým ďalším pokusom o lokalizáciu tejto IP adresy je najprv skontrolovaná databáza dočasnej pamäte, k lokalizácii dochádza len v tom prípade, že daná IP adresa v dočasnej pamäti ešte, alebo už nie je uložená.

3 Implementácia

Navrhnuté riešenie bolo implementované v skriptovacom jazyku *Perl*. Program okrem vlastného programového kódu využíva i niekoľko voľne šíriteľných modulov, dostupných vo verejnom archíve modulov jazyka *Perl*, *CPAN*. Implementácia webového rozhrania aplikácie je verejne dostupná na internetovej adrese <http://tracker.fiit.stuba.sk>.

3.1 Použité externé moduly

Pre prácu s *DNS* a *Whois* servermi boli pri implementácii použité externé moduly z archívu voľne dostupných modulov jazyka *Perl*, *CPAN*.

3.1.1 *Net::DNS*

Tento modul poskytuje funkcionality *DNS resolveru* a umožňuje programátorovi zostrojiť takmer každý typ *DNS* požiadavky priamo zo skriptu jazyka *Perl*.

Metódou `query` objektu `Resolver` je možné získať v parametri zadaný typ *DNS* záznamu (v našom prípade sú podstatné vety *LOC* a *TXT*). Metóda vráti objekty typu `RR` (*Resource Record*) obsahujúce výsledky požiadavky.

Ak ide o *LOC* záznam, aplikovaním metódy `latlon` na objekt typu `RR` je možné získať číselnú reprezentáciu geografickej šírky a dĺžky (*latitude* a *longitude*) výsledku. V prípade záznamu *TXT* je potrebné použiť metódu `txtdata`.

3.1.2 *Net::Whois:Proxy*

Pre získavanie údajov z *Whois* databáz je použitý voľne dostupný modul jazyka *Perl* s názvom ***Net::Whois::Proxy***. Ide o knižnicu poskytujúcu rekurzívne vyhľadávanie na *Whois* serveroch. Rekurzívne vyhľadávanie zabezpečí, že odpoveď z príslušného serveru jedného z regionálnych internetových registrátorov dostaneme vždy bez toho, aby sme sa starali, u ktorého z nich je daná IP adresa registrovaná.

Nevýhodou tohto modulu je jeho relatívna pomalosť (vybavenie jednej požiadavky trvá v niektorých prípadoch až 1-2 sekundy). Jedným z alternatívnych riešení by mohlo byť spúšťanie externého príkazu *Whois*, čo by však pri lokalizácii mnohých IP adries súčasne znamenalo vytvorenie mnohých nových procesov. Testované boli i mnohé ďalšie podobné moduly dostupné v archíve *CPAN*, žiadny z nich však dostatočne nevyhovoval. Ďalšou alternatívou je použitie *telnetu*, či priameho *socketového* spojenia na port 43 daného *Whois* serveru. Toto riešenie však neposkytuje možnosť rekurzívneho vyhľadávania, bolo by teda nutné pripájať sa vždy na ten *Whois* server, ktorý má o danej IP adrese záznam. Zvolené riešenie je však dostatočne funkčné a prípadná optimalizácia rýchlosti je ponechaná na ďalšie dopracovanie.

Hlavná metóda modulu – `whois`, ktorého parametrom je skúmaná IP adresa, zabezpečí načítanie celého obsahu odpovede z *Whois* serveru do reťazca.

Na základe testovania bola v module odhalená drobná chyba – modul nasledoval i *rwhois* odkazy (*Referral Whois*), ktoré však majú odlišný formát ako štandardné *Whois* záznamy a modul ich nedokáže spracovať. Úpravou regulárneho výrazu na riadku 347 tohto modulu bol problém odstránený a IP adresy, ku ktorým modul pôvodne nedokázal vrátiť žiadne výsledky sú aspoň čiastočne lokalizovateľné.

3.1.3 Sys::HostIP

Špecifickou funkciou dostupnou len v konzolovej aplikácii je možnosť lokalizácie IP adresy aktívneho sieťového zariadenia počítača, na ktorom bola aplikácia spustená. Táto funkcia je v aplikácii zavedená hlavne z dôvodu rýchleho otestovania funkčnosti aplikácie na danom zariadení. Pre získanie tejto IP adresy je použitá jediná metóda modulu **Sys::HostIP** s názvom `ip`.

3.2 Viacvláknovosť

Keďže aplikácia ponúka možnosť súčasnej lokalizácie viacerých IP adries a lokalizácia každej IP adresy istú dobu trvá (z dôvodu čakania na odpovede z *Whois* a *DNS* serverov, a nie kvôli výpočtovej náročnosti algoritmu), bola aplikácia navrhnutá tak, aby súčasne mohla prebiehať lokalizácia viacerých IP adries.

Hlavná funkcia modulu lokalizácie IP adresy `localize_host` bola implementovaná tak, aby mohla byť spustená v hlavnom vlákne programu, ale aj ako samostatné vlákno. V tomto prípade zapíše získané lokalizačné údaje do rúry (radu), ktorej deskriptor dostane ako voliteľný vstupný parameter.

Prácu s rúrou v hlavnom vlákne programu a dcérskych vláknach lokalizácie IP adresy popisuje nasledovný výňatok z programového kódu:

```
# vytvorenie objektu rury v hlavnom vlakne programu:
my $queue = Thread::Queue->new();

# zapísanie lokalizačných údajov do rury v samostatnom vlakne:
lock($queue);
$queue->enqueue($ip, $hostname, $country, $city, $lat, $lon,
               $score, $info, $success_method);

# nacistanie údajov z rury v hlavnom vlakne programu:
$hostinf{$varname} = $queue->dequeue();
```

Hlavná funkcia aplikácie volá funkciu lokalizácie IP adresy takým spôsobom, aby bolo naraz aktívnych vždy **maximálne 10 vlákien**.

Pre zabezpečenie bezpečnosti aplikácie (bližšie popísané v časti 3.7) pristupuje webová aplikácia ku konkrétnej položke v databázovej tabuľke z viacerých vlákien súčasne. Nakoľko databázové prostredie *SQLite* neponúka možnosť používateľského uzamknutia tabuľky pre zápis, je nutné prístup k týmto zdieľaným údajom zabezpečiť pomocou semafora, ktorý v našom prípade nadobúda len hodnoty 0 alebo 1 (binárny semafor - *mutex*):

```
# vytvorenie binarneho semafora pre riadenie prístupu k databaze:
my $db_semaphore = new Thread::Semaphore; # inicializacia hodnoty s. na 1

$db_semaphore->down; # uzamknutie semafora

# ... kriticka oblast...

$db_semaphore->up; # odomknutie semafora
```

3.3 Traceroute

Pre vykonanie externého príkazu *traceroute* je použitá konštrukcia, ktorá umožní čítať výstup tohto externého programu ako súbor. Slúži na to funkcia `open`, ktorá v tomto prípade namiesto mena

súboru dostane externý príkaz ukončený znakom „|“, ktorý reprezentuje rúru (*pipe*):

```
open TR, "traceroute $host |";
```

Výstup príkazu *traceroute* je potom možné čítať ako štandardný súbor:

```
while (my $line = <TR>) { }
```

V prípade, že je aplikácia spustená v prostredí *OS Windows*, je namiesto príkazu *traceroute* použitý príkaz *tracert*.

Vzhľadom na to, že každý paket *traceroute*-u môže byť smerovaný inou cestou, môže byť na každom očíslovanom riadku výstupu (jeden „hop“) viacej rôznych IP adries. Môže nastať napríklad aj taká situácia, že pri 15. „hope“ sa nachádza aj taká IP adresa, ktorá sa nachádzala aj pri 14. „hope“ (k danému smerovaču teda vedie viacej ciest, s rôznym počtom „hopov“).

Implementovaná aplikácia túto situáciu rieši takým spôsobom, **že z daného riadku vždy vezme prvú IP adresu**. V prípade, že daná IP adresa už bola lokalizovaná a na riadku sa nachádza iná IP adresa, ktorá ešte lokalizovaná nebola, použije sa táto neznáma adresa.

Medzi funkcionalitou príkazov *traceroute* v prostredí *OS Unix* a *tracert* v prostredí *OS Windows* je fundamentálny rozdiel. Kým príkaz *tracert* v prostredí *OS Windows* používa ako odosielané skúšobné pakety štandardné *ICMP echo request* požiadavky, príkaz *traceroute* v prostredí *OS Unix* odosiela *UDP pakety s cieľovým portom 33434*, pričom cieľový port je pri každom odoslanom pakete inkrementovaný. V oboch prípadoch je v paketoch nastavovaná hodnota *TTL (Time To Live)* od 1 postupne vyššie a od jednotlivých smerovačov na ceste k destinácii je prijatá odpoveď vo forme *ICMP* paketu typu 11 – *Time-to-live exceeded*. Cieľová destinácia odpovedá v prípade skúšobných paketov *ICMP* prostredníctvom *ICMP echo reply*, v prípade skúšobných paketov *UDP ICMP* paketom typu 3 (*Destination unreachable*) s kódom 3 (*Port unreachable*).

Medzi výslednými cestami získanými použitím nástroja *traceroute* (a teda aj použitím implementovanej aplikácie) v prostredí *OS Windows* a *OS Unix* môžu preto nastať značné odlišnosti, závislé na nastavení jednotlivých zariadení na ceste k destinácii (napríklad niektoré *firewally* môžu zahadzovať *ICMP* pakety a *UDP* pakety preposielať ďalej, alebo opačne). V prípade volania externého príkazu *traceroute* v **prostredí OS Unix** je možné parametrom vynútiť i odosielanie *ICMP* alebo *TCP* skúšobných paketov, sú na to však potrebné práva administrátora, a preto **implementovaná aplikácia v tomto prípade používa UDP pakety**.

3.4 Databázové prostredie

Pre implementáciu aplikácie bolo zvolené databázové prostredie *SQLite* [19]. Vykonávanie operácií nad databázami je zabezpečené štandardným modulom databázového rozhrania jazyka *Perl*, *DBI*. Pre pripojenie databázového rozhrania na túto databázu je použitý nasledovný príkaz:

```
my $dbh = DBI->connect("dbi:SQLite:dbname=geo_data.db","","");
```

Keďže databázové prostredie *SQLite* v sebe neobsahuje operátor porovnania reťazca regulárnym výrazom (známy napríklad ako *RLIKE* alebo *REGEXP*), na ktorom stojí návrh vyhľadávania miest v reťazcoch (kapitola 2.2.2), bol problém vyriešený vytvorením používateľskej funkcie *regexp* priamo v jazyku *Perl*:

```
$dbh->func('regexp', 2, sub {  
    my ($regex, $string) = @_;  
    return $string =~ /\b$regex\b/i;  
}, 'create_function');
```

V našom prípade bola klasická funkcionálnosť príkazu *regex* rozšírená a vlastnosť vyhľadávania zhody len v tom prípade, ak sa v skúmanom reťazci vyskytuje vzor ako celé slovo alebo postupnosť slov (*\b* v regulárnom výraze).

Podobne bola vytvorená i používateľská funkcia počítajúca druhú odmocninu čísla, *sqrt*.

Databázové tabuľky aplikácie sú rozdelené do dvoch databáz (kapitola 6.2) – databázových súborov. Súbor *geo_data.db* obsahuje všetky tabuľky určené pre lokalizáciu. Obsah tejto databázy aplikácia nemení a je spoločná pre konzolovú i webovú aplikáciu. Databáza uložená v súbore *tracker_web.db* obsahuje údaje o už lokalizovaných IP adresách (tabuľka *cache*) a údaje umožňujúce limitovať počet vykonaných vizualizácií a *traceroute*-ov jednotlivými používateľmi (tabuľka *limiter*). Túto databázu využíva len webová aplikácia pre vizualizáciu lokalizácie IP adres.

3.5 Vymazávanie dočasných údajov

Ako bolo popísané v návrhu webovej aplikácie pre vizualizáciu lokalizácie IP adres (kapitola 2.5), webová aplikácia obsahuje dočasnú pamäť lokalizovaných IP adres, ako i tabuľku s počtom lokalizácií a *traceroute*-ov jednotlivých používateľov, pre účely ochrany pred zahŕňaním (kapitola 3.9.2).

Platnosť záznamov v dočasnej pamäti IP adres bola stanovená na **24 hodín**, aktuálnosť záznamov zabezpečuje ukladanie časovej značky vytvorenia daného záznamu. Záznamy staršie ako 24 hodín sú automaticky vymazávané spustením SQL dopytu:

```
DELETE FROM cache WHERE added < datetime('now', '-1 day');
```

Platnosť záznamov v tabuľke s počtom lokalizácií a *traceroute*-ov jednotlivých používateľov bola stanovená na **3 hodiny**, aktuálnosť záznamov zabezpečuje ukladanie časovej značky vytvorenia daného záznamu. Záznamy staršie ako 3 hodiny sú automaticky vymazávané spustením SQL dopytu:

```
DELETE FROM limiter WHERE added < datetime('now', '-3 hour');
```

Automatické spúšťanie skriptu s týmito príkazmi zabezpečuje služba *cron*, spustenie skriptu každých 20 minút bolo zabezpečené pridaním nasledovného príkazu do *cron* tabuľky operačného systému webservera:

```
*/20 * * * * /home/www/tracker/html/cache_cleanup.pl
```

3.6 CGI a šablónovací systém

Rozhraním medzi modulmi lokalizácie IP adres naprogramovanými v jazyku *Perl* a webovou aplikáciou poskytujúcou používateľské rozhranie pre ich vizualizáciu sú *CGI* skripty napísané v jazyku *Perl*. Pre zjednodušenie práce s *CGI* skriptami sú použité dva moduly, ktoré sú súčasťou štandardnej distribúcie interpretéra jazyka *Perl* (takzvané štandardné knižničné moduly):

Modul *CGI* obsahuje množstvo funkcií pre uľahčenie práce s *CGI* skriptami. V našom riešení je používaný na spracovávanie *HTTP* premenných *GET* a *POST*, ktoré sú pri použití tohto modulu automaticky dostupné v asociatívnom poli (*hash-i*) *param*, kde sú ako kľúče použité názvy jednotlivých *GET* alebo *POST* premenných.

Modul *CGI::Carp* je použitý na zefektívnenie ladenia aplikácie. V prípade výskytu chyby v kóde *CGI* skriptu je do log súboru webservera uložená len ťažko identifikovateľná chyba (varovanie), v internetovom prehliadači klienta je zobrazená len chybová správa *500 Internal Server Error*.

Použitím tohto modulu príkazom:

```
use CGI::Carp qw(warningsToBrowser fatalsToBrowser);
```

zabezpečíme výpis podrobných chybových hlášok, ako do log súboru webservera, tak i priamo do internetového prehliadača používateľa. Po ukončení práce na projekte a odladení všetkých chýb je možné použitie tohto modulu z aplikácie odstrániť, jeho ponechanie však môže byť užitočné pri aktualizácií a dopĺňaní funkcionality stránky.

Template Toolkit [21] je rýchly, flexibilný a ľahko rozširiteľný šablónovací systém napísaný v jazyku *Perl*. Hlavnou ideou šablónovacích systémov je oddelenie programových kódov internetových stránok od ich dizajnu – *HTML* kódu. Jazyk *Perl* má už sám o sebe pomerne dosť komplikovanú syntax (kapitola 1.5.1), keď sa navyše v tom istom zdrojovom kóde pomedzi príkazy jazyka *Perl* vkladajú i konštrukcie jazyka *HTML*, prehľadnosť a čitateľnosť kódu ešte viac poklesne.

Pri použití šablónovacieho systému je v oddelených skriptoch jazyka *Perl* uložená samotná logika systému, zobrazenie výsledkov spracovania skriptu je zabezpečené aplikovaním šablóny obsahujúcej príslušný *HTML* kód, ktorá je uložená v ďalšom samostatnom súbore.

Výňatok programového kódu uloženého v skripte obsahujúcom logiku aplikácie - zabezpečuje dostupnosť obsahu premennej *\$hostinf* v šablóne a aplikovanie (zobrazenie) šablóny *skeleton.tpl*:

```
my $tt = Template->new();
my $vars = { hostinf => \%hostinf };
$tt->process('skeleton.tpl', $vars);
```

Výňatok *HTML* šablóny uloženej v samostatnom súbore (*skeleton.tpl*) používajúci premennú *\$hostinf*:

```
[% IF hostinf.hostname %]
<tr>
  <td>Doménové meno:</td>
  <td>[% hostinf.hostname %]</td>
</tr>
[% END %]
```

Použitie šablónovacieho systému prináša okrem zlepšenia čitateľnosti kódu i mnohé iné pozitíva, napríklad veľmi ľahkú **možnosť zmeny vzhľadu webu bez nutnosti zásahu do programovej časti jeho logiky** a naopak, úpravu funkčnosti webu bez nutnosti úprav v *HTML* kóde. Šablónovací systém umožňuje i **dekompozíciu výsledného HTML kódu** do menších celkov, ktoré môžu byť modulárne, takže napríklad v prípade požiadavky na pridanie novej položky do hlavného menu na všetkých podstránkach webu postačí upraviť jednu jedinú šablónu.

Súčasne je veľmi jednoduché vytvoriť stránku vo viacerých jazykových mutáciách – postačí vytvoriť kópie existujúcich šablón a ich texty preložiť do iného jazyka, pričom nie je nutný žiadny zásah do aplikačnej logiky stránky.

3.7 Google Maps API

Pre vizualizáciu lokalizovaných IP adries je použité rozhranie *Google Maps API*, vďaka ktorému je možné prostredníctvom *JavaScriptu* vložiť mapy *Google Maps* na ktorúkoľvek internetovú stránku. Súčasťou *API* (*Application Programming Interface*) je množstvo funkcií pre manipuláciu s týmito mapami a ich obsahom.

Pre vloženie mapy do internetovej stránky je nutné vytvoriť funkciu v jazyku *JavaScript*

obsluhujúcu udalosť *onLoad* stránky a do jej tela umiestniť inicializačný kód mapy:

```
var map = new GMap2(document.getElementById("map_canvas"));
map.setMapType(G_PHYSICAL_MAP);
map.setCenter(new GLatLng(lat, lon), 5);
```

map_canvas je identifikátor elementu *DIV* v *HTML* kóde stránky, do ktorého bude mapa umiestnená. Premenné *lat* a *lon* obsahujú súradnice miesta, ktoré má byť stredom aktuálneho zobrazenia mapy, nasledujúci číselný argument funkcie *setCenter* predstavuje úroveň priblíženia mapy (*zoom*).

Pre vyznačenie jednotlivých bodov na mape sú použité štandardné značky (*markers*), ktorých pridanie do vytvorenej mapy je veľmi jednoduché:

```
var point = new GLatLng(lat, lon);
var marker = new GMarker(point);
marker.bindInfoWindowHtml("textový popis");
map.addOverlay(marker);
```

lat a *lon* sú zemepisné súradnice označovaného bodu.

Vizualizácia postupnosti IP adries vyžaduje vykreslenie úsečiek medzi jednotlivými lokalizovanými bodmi. Toto umožňuje objekt *Gpolyline*:

```
var polyline = new GPolyline(coordinates, "#ff0000", 5);
map.addOverlay(polyline);
```

Premenná *coordinates* predstavuje pole bodov – objektov *GLatLng*, ktoré boli použité aj v predchádzajúcich ukážkach. Nasledujúci parameter konštruktoru *Gpolyline* špecifikuje farbu úsečky (v hexadecimálnom *HTML* formáte) a posledným parametrom je číselný údaj, špecifikujúci šírku úsečky.

Po vykreslení všetkých bodov a úsečiek je potrebné zabezpečiť také zobrazenie mapy (centrovanie a *zoom*), aby používateľ videl všetky vykreslené body v čo najväčšej presnosti. Napríklad pri vizualizácii bodov, ktoré ležia len na území Slovenska je potrebné zacieliť mapu tak, aby väčšou časťou zobrazovala len územie Slovenska, no pri vizualizácii bodov z USA a súčasne z Ázie je potrebný pohľad na takmer celú Zem.

Pri bodoch a úsečkách, ktoré sa rozprestierajú medzi kontinentami vzniká problém pri výpočte okrajových zemepisných dĺžok, nakoľko vďaka sféricosti Zeme je možné dostať sa z bodu A do bodu B jednak cestou na východ, ako i cestou na západ. Kým napríklad medzi Amerikou a Európou vedie najkratšia cesta cez nultý poludník, medzi Severnou Amerikou a Japonskom cez poludník 180°.

Popísané problémy a zložité výpočty je však možné obísť použitím API objektu *GLatLngBounds*. Tento objekt predstavuje obdĺžnik, ktorý pokrýva všetky body, ktoré tomuto objektu priradíme.

```
var latlngbounds = new GLatLngBounds( );
for (var i = 0; i < coordinates.length; i++) {
    latlngbounds.extend(coordinates[i]);
}
```

Premenná *coordinates* opäť predstavuje pole bodov – objektov *GLatLng*, ktoré majú v obdĺžniku ležať. Objekt poskytuje metódu *getCenter*, ktorá vráti súradnice stredu oblasti

obdĺžnika a metódu `getBoundsZoomLevel`, ktorá vráti optimálne priblíženie (*zoom*) – také, aby bola viditeľná celá plocha obdĺžnika so zachovaním čo najväčšej presnosti.

```
map.setCenter(  
    latlngbounds.getCenter(),  
    map.getBoundsZoomLevel(latlngbounds)  
);
```

3.8 Jazykové mutácie webovej aplikácie

Webová aplikácia pre vizualizáciu lokalizácie IP adres bola implementovaná v dvoch jazykových mutáciách – anglickej a slovenskej. Po príchode používateľa na stránku aplikácie je jeho IP adresa lokalizovaná, a na základe krajiny tejto lokalizácie je automaticky zvolený buď slovenský (Slovenská republika, Česká republika), alebo anglický jazyk (ostatné krajiny).

Používateľ má kedykoľvek možnosť zmeniť jazyk aplikácie. Zvolený jazyk (automaticky, či manuálne) je uložený v *session* premennej `language`.

3.9 Bezpečnosť webovej aplikácie

Keďže webová aplikácia pre vizualizáciu lokalizácie IP adres je verejne prístupná na Internete, je nutné zabezpečiť ju pred potencionálnymi ohrozeniami škodným používateľom.

3.9.1 Overovanie platnosti vstupných údajov

Najdôležitejšou zásadou zabezpečenia aplikácie je overovanie platnosti používateľom zadaných údajov. V prípade, že je od používateľa očakávaná IP adresa alebo doménové meno (resp. ich postupnosť), sú údaje overené príslušným regulárnym výrazom:

```
$ip =~ /^\\d{1,3}\\.\\d{1,3}\\.\\d{1,3}\\.\\d{1,3}$/; # ip adresa  
$hostname =~ /^(\\w|\\.|-)+\\. (\\w|\\.|-)+$/; # doménové meno
```

Takýmto spôsobom je možné **zamedziť podstrčeniu škodných príkazov** do SQL príkazu (**SQL injection**) alebo *shellu* (pri spúšťaní externého príkazu *traceroute* s parametrom IP adresy / doménového mena).

Základným ochranným mechanizmom pred *SQL injection* je i používanie parametrizovaných príkazov pre prácu s databázou namiesto priameho vloženia obsahu premennej do reťazca s SQL príkazom:

```
my $sel = $dbh_web->prepare_cached("SELECT * FROM cache WHERE ip =  
                                     ? OR hostname = ? LIMIT 1");  
$sel->execute($host, $host);
```

3.9.2 Ochrana pred zahltením

Vzhľadom na to, že pri lokalizácii IP adres dochádza ku komunikácii s *DNS* servermi a *Whois* databázami, je nutné ošetriť zneužitie aplikácie nadmerným počtom lokalizácií. *Whois* servery totiž obmedzujú počet požiadaviek z každej IP adresy a po prekročení istej miery môžu požiadavky z IP adresy nášho webservera na istú dobu ignorovať. Z tohto dôvodu bolo zavedené **obmedzenie počtu lokalizácií z jednej IP adresy na 30 za tri hodiny**.

Aj po obmedzení počtu lokalizácií však môže dôjsť k zahlteniu množstvom požiadaviek

na *traceroute*, čo môže spôsobiť neprimerane veľký počet prenášaných paketov. Pre zamedzenie tejto situácii bolo zavedené **obmedzenie počtu *traceroute*-ov z jednej IP adresy na 3 za tri hodiny**.

Výnimku z týchto obmedzení majú používatelia, ktorí na webovú stránku prístupujú z IP adres pridelených Fakulte informatiky a informačných technológií STU v Bratislave, prípadne majú pridelenú zvláštnu výnimku.

Počet povolených lokalizácií, *traceroute*-ov, dĺžka časového intervalu obmedzenia a IP adresy s úplným prístupom sú nastaviteľné, spôsob nastavenia je popísaný v technickej dokumentácii.

3.9.3 Zakrytie citlivých údajov

Ďalším potencionálnym bezpečnostným ohrozením môže byť vykonávanie príkazu *traceroute* a výpis IP adres jeho výstupu používateľovi. Takýmto spôsobom má používateľ prehľad o IP adresách smerovačov v sieti, v ktorej sa nachádza webserver s aplikáciou. Z tohto dôvodu sú prvé dva „hopy“ výstupu *traceroute* vynechané, ak IP adresa používateľa nepatrí do zoznamu privilegovaných IP adres s úplným prístupom, popísaného v predchádzajúcej časti.

4 Overenie riešenia

Implementované riešenie bolo overené z hľadiska posúdenia správnosti lokalizácie IP adries, ako i overením funkčnosti vizualizácie lokalizácie IP adries a používateľského rozhrania webovej i konzolovej aplikácie.

4.1 Porovnanie výsledkov lokalizácie s externou databázou

V tejto fáze testovania bol výstup lokalizácie IP adries implementovanej aplikácie porovnaný s voľne prístupnou databázou **GeoLite City** od spoločnosti *MaxMind, Inc.* [6] (popísaná v kapitole 1.3.2). Cieľom bolo porovnať výsledky na úrovni čo najväčšieho počtu rôznych krajín sveta.

Na stránke *MaxMind, Inc.* [6] je tiež uvedená správnosť údajov uložených v databáze *GeoLite City* vyhodnotená na základe údajov zo stránok, ktoré od používateľa požadujú zadanie jeho lokalizácie. Úspešnosť správneho určenia krajiny lokalizácie je 99.5%, úspešnosť správneho určenia mesta lokalizácie (s odchýlkou 25 míľ) je v každej krajine rôzna - pre Slovensko je to 53%, pre USA 79%, priemerne 70.11%.

4.1.1 Vstupné dáta

Popisované výsledky pochádzajú z testovacej vzorky **500 IP adries**, ktoré boli vybrané náhodným spôsobom. Na získanie tohto počtu lokalizovaných IP adries bolo vykonaných 596 lokalizácií náhodných IP adries verejného adresného priestoru.

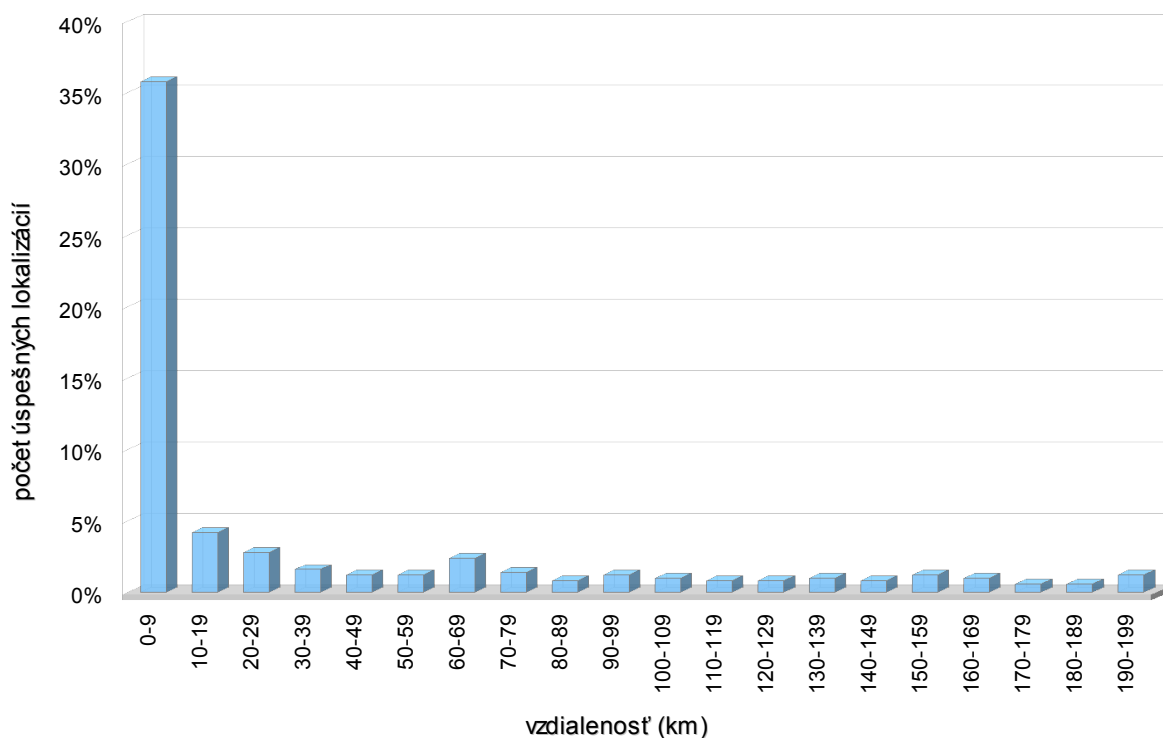
Z databázy *GeoLite City* bola ku každej z týchto IP adries priradená krajina a mesto. Testovanou aplikáciou **sa v 436 (87.2 %) prípadoch podarilo odhadnúť mesto lokalizácie**, v ostatných prípadoch bola odhadnutá iba krajina.

4.1.2 Výsledky testovania

Z počtu 500 testovaných IP adries sa v 490 prípadoch podarilo určiť rovnakú krajinu lokalizácie. Z toho vyplýva, že **zhoda lokalizácie na úrovni krajiny nastala v 98.0%**.

Ostrá rovnosť názvov mesta alebo súradníc lokalizácie nastala v 172 prípadoch (34.4%). Vzhľadom na to, že názvy mesta sa môžu líšiť diakritikou a súradnice toho istého mesta môžu byť mierne odlišné, je vhodnejšie správnosť posudzovať s určitou vzdialenostnou odchýlkou. Keďže databáza *GeoLite City* má pri špecifikácii správnosti uvedenú odchýlku 25 míľ, bola zhoda medzi touto databázou a testovanou aplikáciou posúdená **s odchýlkou 40 km** (≈ 25 míľ). Zhoda medzi údajmi nastala v 227 prípadoch. Z toho vyplýva, že **zhoda lokalizácie na úrovni mesta nastala v 45.4%**.

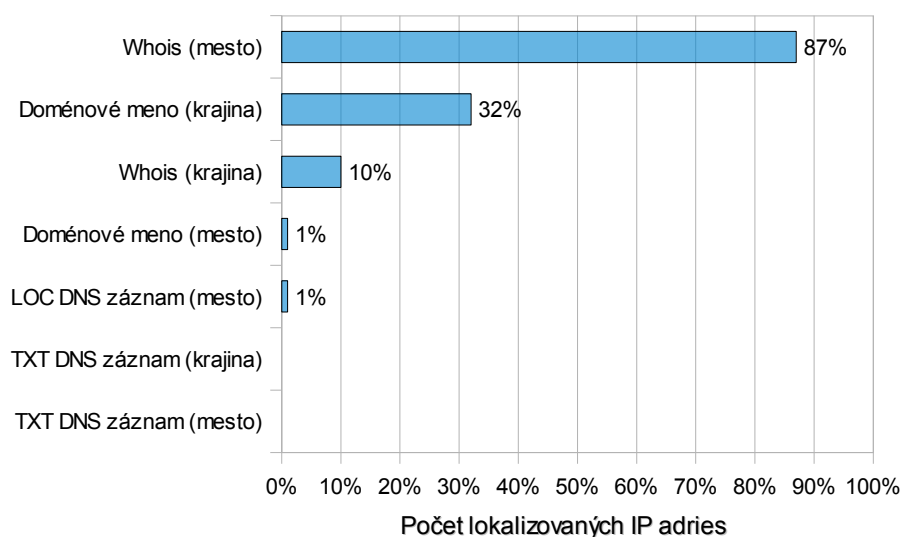
Ako je zrejmé z grafu na obrázku *Obr. 6*, počet úspešných lokalizácií so zvyšovaním vzdialenostnej odchýlky nad 40 km stúpa už len minimálne, preto možno zvolenú odchýlku označiť za vhodne stanovenú.



Obr. 6: Závislosť počtu úspešných lokalizácií od veľkosti vzdialenostnej odchýlky

Vzhľadom na špecifikovanú správnosť údajov uložených v databáze *GeoLite City* možno dosiahnuté výsledky zhody medzi touto databázou a výstupom implementovanej aplikácie považovať za postačujúce.

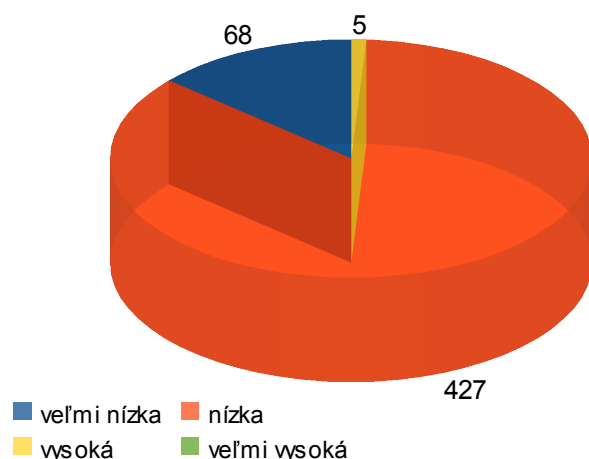
Na základe výstupu testovania bolo vyhodnotené i zastúpenie jednotlivých metód lokalizácie pri určovaní výslednej lokalizácie IP adres (Obr. 7).



Obr. 7: Zastúpenie jednotlivých metód lokalizácie pri určovaní výslednej lokalizácie IP adres (testovanie na základe porovnania výsledkov s externou databázou)

Najúspešnejšou metódou lokalizácie je podľa výsledkov testovania analýza dát z *Whois* (436 lokalizácií mesta, 49 lokalizácií krajiny) a analýza doménového mena (159 lokalizácií krajiny, 7 lokalizácií mesta). Najmenej úspešnou metódou je analýza *TXT DNS* záznamu, ktorou sa na testovanej vzorke nepodarilo lokalizovať ani jednu IP adresu.

Priemerné skóre úspešne lokalizovaných IP adries je rovné hodnote **7.82**, čo predstavuje úroveň nízkej presnosti lokalizácie (podľa *Tab. 2* v kapitole 2.2). Rozloženie jednotlivých úrovní presnosti lokalizácie je zobrazené na obrázku *Obr. 8*.



Obr. 8: Úroveň presnosti lokalizácie podľa výšky skóre pri testovaní na základe porovnania výsledkov s externou databázou

4.2 Používateľské testovanie lokalizácie

Cieľom tejto fázy testovania bolo overiť funkčnosť lokalizácie IP adries na úrovni slovenských miest. Motiváciou bola implementácia špeciálneho spôsobu analýzy doménového mena pre slovenské mestá (kapitola 2.2.3).

4.2.1 Vstupné dáta

Prostredníctvom e-mailu a IM (*Instant Messaging*) správ bolo oslovených niekoľko subjektov (pre zamedzenie vloženia nepravdivých údajov predovšetkým priateľov), ktorým sa po kliknutí na odkaz v správe zobrazil webový formulár. Formulár obsahoval odhadovanú lokalizáciu používateľa, ktorý mal prostredníctvom prepínacích tlačidiel označiť, či je táto lokalizácia správna alebo nie. Ako doplnkové informácie mohol uviesť jeho skutočnú lokalizáciu a typ internetového pripojenia.

4.2.2 Výsledky testovania

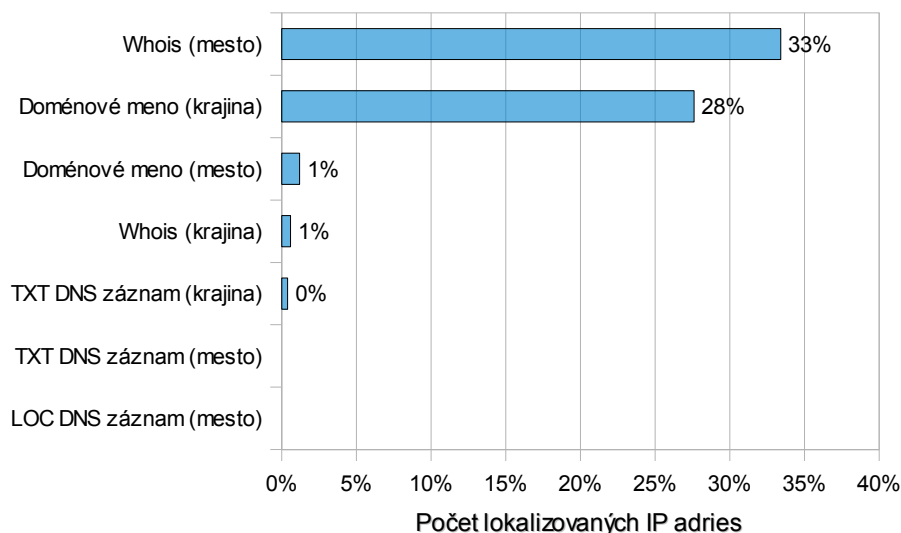
Z celkového počtu **170 otestovaných IP adries** sa v každom prípade podarilo odhadnúť krajinu lokalizácie, a v 167 prípadoch (98.2 %) mesto lokalizácie.

Z počtu 170 testovaných IP adries sa v 164 prípadoch podarilo určiť správnu krajinu lokalizácie. Z toho vyplýva, že **správnosť lokalizácie na úrovni krajiny nastala v 96.5%**. V 92 prípadoch podarilo určiť správne mesto lokalizácie. Z toho vyplýva, že **správnosť lokalizácie na úrovni mesta nastala v 54.1%**.

Z doplnkových informácií používateľov vyplýva, že lokalizáciu na úrovni mesta sa podarilo určiť správne v prípade, ak sa nachádzali na pôde nejakej školy, univerzity, alebo internátu, boli

pripojení prostredníctvom rozvodu káblovej televízie, alebo malým miestnym poskytovateľom internetového pripojenia. Naopak, neúspešná lokalizácia na úrovni mesta nastala v prípade, že používateľ bol do Internetu pripojený prostredníctvom *DSL* alebo mobilného pripojenia. V týchto prípadoch bolo za mesto lokalizácie určené hlavné mesto, Bratislava.

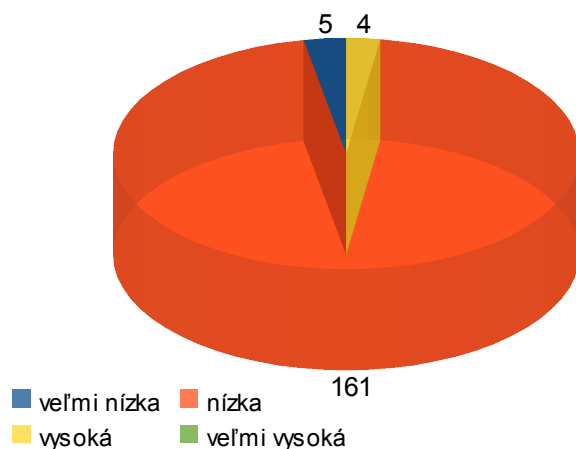
Na základe výstupu testovania bolo vyhodnotené i zastúpenie jednotlivých metód lokalizácie pri určovaní výslednej lokalizácie IP adres (Obr. 9).



Obr. 9: Zastúpenie jednotlivých metód lokalizácie pri určovaní výslednej lokalizácie IP adres (používateľské testovanie)

Najúspešnejšou metódou lokalizácie je podľa výsledkov testovania analýza dát z *Whois* (167 lokalizácií mesta, 3 lokalizácie krajiny) a analýza doménového mena (138 lokalizácií krajiny, 6 lokalizácií mesta). Najmenej úspešnou metódou v tomto prípade bola analýza *LOC DNS* záznamu, ktorou sa na testovacej vzorke nepodarilo lokalizovať ani jednu IP adresu.

Priemerné skóre úspešne lokalizovaných IP adres je rovné hodnote **9.6**, čo predstavuje úroveň nízkej presnosti lokalizácie (podľa Tab. 2 v kapitole 2.2). Rozloženie jednotlivých úrovní presnosti lokalizácie je zobrazené na obrázku Obr. 10.



Obr. 10: Úroveň presnosti lokalizácie podľa výšky skóre pri používateľskom testovaní

Z výšky priemerného skóre, ktoré je v prípade tohto testovania na slovenských IP adresách takmer o dva body vyššie ako pri testovaní svetových IP adries v časti 4.1, ako i z vyššej percentuálnej úspešnosti lokalizácie na základe analýzy doménového mena možno usúdiť, že implementácia špeciálneho spôsobu analýzy doménového mena pre slovenské mestá zvýšila presnosť lokalizácie slovenských miest, i keď len nepatrne.

4.3 Beta testovanie

V rámci beta testovania bolo niekoľkými používateľmi otestované základné správanie programu, a to konzolovej i webovej aplikácie. Pri testovaní webovej aplikácie veľmi napomohlo i používateľské testovanie (4.2), kedy mnohí zúčastnení vyskúšali i ostatné funkcionality webovej aplikácie. Testerí prispeli niekoľkými pripomienkami, žiadna zásadná chyba však nebola ohlásená.

5 Zhodnotenie

V práci sa podarilo analyzovať prípady použitia lokalizácie IP adries v praxi, identifikovať a podrobne rozobrať jednotlivé postupy a metódy, ktorými je možné odhadnúť lokalizáciu zariadenia s danou IP adresou.

Následne bol podrobne navrhnutý proces lokalizácie IP adries použitím štyroch metód rozobraných v analýze problematiky, ako i proces vizualizácie lokalizácie IP adries.

Návrh bol implementovaný do modulu umožňujúceho lokalizáciu IP adries, ktorý bol využitý v konzolovej aplikácii pre lokalizáciu IP adries a webovej aplikácii pre vizualizáciu lokalizácie IP adries.

Konzolová aplikácia pre lokalizáciu IP adries umožňuje používateľovi lokalizáciu zadaných IP adries a výpis agregovaných údajov o IP adrese, lokalizáciu cesty z *traceroute*-u, ako i výpočet vzdialenosti medzi zadanými IP adresami.

Webová aplikácia poskytuje možnosť zobrazenia lokalizácie jednej IP adresy, postupnosti adries (cesty) v podobe bodov spojených úsečkami, ako i viacerých postupností s farebným vyznačením vyťaženia uzlov a vizuálny *traceroute*.

Implementácia webovej aplikácie bola umiestnená na webový server umiestnený v sídle Fakulty informatiky a informačných technológií STU a je verejne prístupná na adrese <http://tracker.fiiit.stuba.sk/>. Z tejto stránky je možné stiahnuť i konzolovú aplikáciu pre lokalizáciu IP adries a elektronickú verziu tohto dokumentu.

Implementácia bola následne otestovaná a výsledky lokalizácie IP adries overené s externou databázou a údajmi používateľov. Výsledky možno považovať za postačujúce, čo oprávňuje aplikáciu na použitie v praxi, ako nástroj na odhad lokalizácie IP adries. Odchýlky od skutočnosti zapríčiňujú predovšetkým nesprávne alebo neaktuálne údaje v databázach *Whois* a *DNS*, ako i nedokonalosť algoritmu, ktorý nedokáže rozanalyzovať a vyhodnotiť také množstvo údajov tak dokonale, ako ľudský mozog. Je tiež nutné uvedomiť si, že presnú lokalizáciu IP adries často poznajú len jednotliví poskytovatelia internetového pripojenia, ktorí daný adresný priestor spravujú a tieto informácie nie sú verejne známe.

Za hlavný prínos aplikácie oproti ostatným dostupným riešeniam možno označiť lokalizáciu IP adresy v reálnom čase, bez využitia externých databáz lokalizácií IP adries. Používateľovi je vždy transparentne oznámené, na základe akých metód bola lokalizácia IP adresy vyhodnotená, a to priamo uvedením príslušnej metódy, ako i výškou skóre (relatívnej presnosti lokalizácie). Oproti podobným dostupným riešeniam ponúka aplikácia i analýzu *DNS* záznamov (*LOC* a *TXT DNS* záznamy), ako i rozšírené vyhľadávanie slovenských miest v doménovom mene.

Aplikáciu by bolo možné ďalej rozvíjať smerom zefektívnenia rýchlosti lokalizácie IP adries (napríklad implementovaním viacvláknovosti i pri aplikovaní jednotlivých metód lokalizácie alebo uskutočňovaním lokalizácie jednotlivých „hopov“ *traceroute*-u priebežne, ešte pred jeho úplným skončením).

Tiež by bolo možné zdokonaľiť rozoznávanie lokalizácie z výstupu *Whois* alebo doménového mena (napríklad aplikovaním postupu, ktorý bol použitý pre rozoznávanie skratiek slovenských miest v doménovom mene (kapitola 2.2.3), i na iné krajiny).

Rozvinúť by bolo možné i lokalizáciu použitím nástroja *traceroute* - lokalizáciu neznámych „hopov“ na ceste k destinácii by bolo možné odhadnúť napríklad i na základe sledovania doby odozvy jednotlivých „hopov“ na vzorke paketov a jej porovnania s odozvou „hopov“ so známou lokalizáciou.

6 Technická dokumentácia

Táto časť dokumentu obsahuje prehľad zdrojových kódov riešenia, fyzický dátový model použitý v aplikácii, inštaláciu a používateľskú príručku.

6.1 Prehľad zdrojových kódov riešenia

Zdrojové kódy riešenia tvoria tri časti - skupina modulov vykonávajúcich lokalizáciu IP adres a dve aplikácie používajúce tieto moduly - konzolová aplikácia pre lokalizáciu IP adres a webová aplikácia pre vizualizáciu lokalizácie IP adres.

6.1.1 Zdrojové kódy modulov lokalizácie IP adres

Zdrojové kódy zabezpečujúce lokalizáciu IP adres sú umiestnené vo vlastných adresároch:

/LocIP/ - obsahuje vlastný programový kód lokalizácie

/Net/ - obsahuje voľne šíriteľné moduly z archívu CPAN pre prácu s DNS (*Net::DNS*) a Whois (*Net::Whois::Proxy*)

/Sys/ - obsahuje voľne šíriteľný modul z archívu CPAN umožňujúci získanie IP adresy aktívneho sieťového rozhrania počítača (*Sys::HostIP*)

V koreňovom priečinku sa nachádza databázový súbor **geo_data.db** obsahujúci *SQLite* databázu s geografickými údajmi (popísanú v časti 6.2).

Ďalej je popísaný obsah adresára */LocIP/*, ktorý obsahuje jednotlivé moduly lokalizácie IP adres:

Localize.pm

Hlavný modul lokalizácie IP adres. Tento modul je nutné zahrnúť do aplikácie, ktorá má využívať naprogramovanú knižnicu lokalizácie IP adres (`use LocIP::Localize;`).

Exportuje funkciu `localize_host`, ktorá vykonáva samotnú lokalizáciu IP adresy. Funkcia preberá tri parametre, pričom povinný je len prvý z nich:

`$host` – IP adresa alebo doménové meno, ktoré chceme lokalizovať

`$queue` – ukazovateľ na rad (rúru), do ktorého je vložený výsledok lokalizácie v prípade, že funkcia beží vo vlastnom vlákne

`$methods` – metódy, ktoré majú byť použité pri lokalizácii - reťazec obsahujúci maximálne 4 znaky: l (*LOC DNS* záznam), w (*Whois*), d (doménové meno), x (*TXT DNS* záznam)

Funkcia vracia zoznam údajov o lokalizácii IP adresy:

`$ip` – IP adresa lokalizovaného uzla

`$hostname` – doménové meno lokalizovaného uzla

`$country` – krajina lokalizácie

`$city` – mesto lokalizácie

`$lat` – zemepisná šírka lokalizácie (desatinné číslo)

`$lon` – zemepisná dĺžka lokalizácie (desatinné číslo)

`$score` – dosiahnuté skóre lokalizácie (celé číslo)

`$info` – textový popis lokalizácie

`$success_method` – metódy určenia lokalizácie (reťazec, formát rovnaký ako pri vstupnom parametri `$methods`)

Hlavnou úlohou tejto funkcie je volať lokalizačné funkcie jednotlivých metód lokalizácie IP adresy, umiestnených v samostatných moduloch, spracovať ich a vyhodnotiť výslednú lokalizáciu IP adresy s príslušným skóre.

Modul tiež obsahuje definíciu konštánt skóre pre jednotlivé metódy lokalizácie podľa tabuľky *Tab. 1* v časti 2.2.

Whois.pm

Modul zabezpečuje lokalizáciu IP adresy použitím údajov uložených na *Whois* serveroch. Exportuje funkciu `localize_by_whois`, ktorá sa pokúša lokalizovať IP adresu na základe týchto údajov. Funkcia preberá dva povinné parametre:

`$host` - IP adresa lokalizovaného uzla

`$dbh` - deskriptor databázového pripojenia na databázu s geografickými údajmi

Funkcia vracia zoznam údajov o lokalizácii IP adresy, ktorého obsah je zhodný s funkciou `localize_host` hlavného modulu *Localize.pm*, bez premennej `$success_method`.

Funkcia spracováva textové odpovede z *Whois* serverov, ktoré získava použitím externého modulu *Net::Whois::Proxy*. Na základe ich analýzy sa pokúša určiť odhad lokalizácie IP adresy.

DNS.pm

Modul zabezpečuje lokalizáciu IP adresy na základe *DNS* záznamov – použitím analýzy doménového mena, *LOC DNS* záznamu a *TXT DNS* záznamu. Na tieto účely exportuje nasledovné funkcie:

`localize_by_hostname` - funkcia lokalizuje IP adresu na základe analýzy doménového mena.

`localize_by_loc_dns` - funkcia lokalizuje IP adresu na základe analýzy *LOC DNS* záznamu

`localize_by_txt_dns` - funkcia lokalizuje IP adresu na základe analýzy *TXT DNS* záznamu

Všetky tri funkcie majú rovnaké vstupné parametre:

`$hostname` – doménové meno lokalizovaného uzla

`$dbh` - deskriptor databázového pripojenia na databázu s geografickými údajmi

`$default_country` - krajina, ktorá už bola určené inými metódami lokalizácie (okrem funkcie `localize_by_txt_dns`)

Funkcia vracia zoznam údajov o lokalizácii IP adresy:

`$country` – krajina lokalizácie

`$city` – mesto lokalizácie

`$lat` – zemepisná šírka lokalizácie (desatinné číslo)

`$lon` – zemepisná dĺžka lokalizácie (desatinné číslo)

Database.pm

Modul obsahuje jedinú verejnú funkciu `database_connect`, ktorej úlohou je vytvoriť databázové pripojenie na databázu s geografickými údajmi a vrátiť deskriptor tohto databázového spojenia. V rámci tohto procesu funkcia pridá do databázového stroja *SQLite* definíciu vlastného operátora **REGEXP**, ktorý slúži na porovnávanie reťazca regulárnym výrazom a používateľskej funkcii `sqrt`, ktorá slúži na výpočet druhej odmocniny zadaného čísla (popísané v kapitole 3.4).

Utils.pm

Tento modul obsahuje niekoľko jednoduchých podporných funkcií, ktoré používané vo vyššie popísaných moduloch:

`is_valid_ip` - funkcia overí platnosť formátu IP adresy

`is_valid_hostname` - funkcia overí platnosť formátu doménového mena

`is_private_ip` - funkcia zistí, či je zadaná IP adresa špeciálna (*loopback, multicast*)

`get_distance` - funkcia vráti vzdialenosť medzi dvoma bodmi zadanými v súradniciach

`decimal2dms` - funkcia premení uhol v stupňoch (desatinné číslo) na stupne, minúty a sekundy

`latlon2readable` - funkcia premení súradnice (vo formáte desatinných čísel) na súradnice vo formáte stupne, minúty, sekundy, pologuľa (severná/južná, východná/západná)

`trim` - funkcia odstráni biele znaky zo začiatku a konca reťazca

`round` - funkcia zaokrúhli desatinne číslo na zadaný počet desatinných miest

`get_precision` - funkcia premení skóre na úroveň presnosti lokalizácie (reťazec dĺžky 1 - 2 znaky)

Popis parametrov a návratových hodnôt týchto funkcií je možné nájsť v zdrojovom kóde - priamo v hlavičkách týchto funkcií.

6.1.2 Zdrojové kódy konzolovej aplikácie

Konzolová aplikácia pre lokalizáciu IP adres využíva moduly lokalizácie popísané v časti 6.1.1. Celú aplikáciu tvorí jediný skript, nachádzajúci sa v súbore **locip.pl**. Skript spracuje vstup používateľa a na základe neho volá hlavnú funkciu lokalizačného modulu `localize_host`.

Na začiatku zdrojového súboru sa nachádza konštanta **MAX_THREADS**, ktorá určuje maximálny počet vlákien, v ktorých môže prebiehať lokalizácia IP adresy (štandardne nastavené na **10**).

6.1.3 Zdrojové kódy webovej aplikácie

Webová aplikácia pre vizualizáciu lokalizácie IP adres využíva moduly lokalizácie popísané v časti 6.1.1. Súbory webovej aplikácie pre vizualizáciu lokalizácie IP adres sú umiestnené v niekoľkých adresároch:

/db/ - adresár obsahuje databázový súbor formátu *SQLite* `tracker_web.db`, v ktorom sú uložené údaje o už lokalizovaných IP adresách a údaje umožňujúce limitovať počet vykonaných vizualizácií a `traceroute`-ov jednotlivými používateľmi

/images/ - adresár obsahuje obrázky používateľského rozhrania aplikácie. Podadresár **/images/markers/** obsahuje obrázky značiek pre mapy *Google Maps*

/styles/ - obsahuje kaskádové štýly (*CSS - Cascading Style Sheets*) dizajnu webovej aplikácie

/Template/ - obsahuje zdrojové kódy externého modulu *Template Toolkit*

/templates/ - obsahuje *HTML* šablóny jednotlivých podstránok webu. Hlavnou šablónou je súbor *skeleton.tpl*, ktorý obsahuje hlavnú kosť stránky. V podadresári **/templates/javascripts/** sa nachádzajú šablóny využívané na generovanie funkcií v jazyku *JavaScript*

Ďalej je popísaný obsah koreňového adresára webovej aplikácie, ktorý obsahuje zdrojové kódy skriptov obsluhujúcich lokalizáciu IP adres a zobrazenie jednotlivých podstránok webu:

index.pl – zabezpečuje vizualizáciu lokalizácie jednej IP adresy (úvodná stránka webu)

traceroute.pl – zabezpečuje funkcionálnosť vizuálneho trasovania IP adres (*traceroute*)

sequence.pl – zabezpečuje vizualizáciu lokalizácie jednej postupnosti IP adres

sequences.pl - zabezpečuje vizualizáciu lokalizácie viacerých postupností IP adres

cache_cleanup.pl – zabezpečuje periodické vymazávanie dočasnej pamäte údajov v databáze, automatické spúšťanie skriptu zabezpečuje služba *cron* (popísané v kapitole 3.5)

Ostatné skripty koreňového adresára obsluhujú zobrazenie statických podstránok webu.

6.2 Fyzický dátový model

Implementovaná aplikácia využíva databázové úložisko dát. Jednotlivé databázové tabuľky predstavujú samostatné dátové entity, nie sú medzi nimi definované žiadne vzťahy.

Tabuľka *geo_city*

Táto tabuľka predstavuje zoznam miest sveta s ich geografickými súradnicami. Obsah tabuľky je získaný z voľne dostupnej databázy spoločnosti *MaxMind, Inc.* [6]. Tabuľka obsahuje vyše 218 000 záznamov, slovenských miest a obcí sa v tabuľke nachádza takmer 900.

Názov poľa	Typ poľa	Vlastnosti / obmedzenia	Poznámka
locId	INTEGER	PRIMARY KEY NOT NULL	identifikátor položky tabuľky
country	VARCHAR(2)	NULL	skratka krajiny
region	VARCHAR(50)	NULL	skratka / kód oblasti
city	VARCHAR(50)	NULL	názov mesta
latitude	FLOAT	NULL	geografická šírka
longitude	FLOAT	NULL	geografická dĺžka

Tab. 3: Štruktúra databázovej tabuľky *geo_city*

Nad tabuľkou boli pre urýchlenie vyhľadávania vytvorené 4 indexy:

IDX_GEO_CITY_COUNTRY (stĺpec *country*)

IDX_GEO_CITY_REGION (stĺpec *region*)

IDX_GEO_CITY_LATITUDE (stĺpec *latitude*)

IDX_GEO_CITY_LONGITUDE (stĺpec *longitude*)

Tabuľka *country_codes*

Tabuľka obsahuje mapovanie dvojnakových skratiek štátov na ich úplný názov (použitá je databáza domén prvej úrovne [11]). Tabuľka by teoreticky mohla byť prepojená s tabuľkou *geo_city*, v navrhnutej aplikácii sa však pre zníženie počtu databázových požiadaviek toto prepojenie nevyužíva a prevod zo skratky na úplný názov krajiny sa uskutočňuje až po definitívnom stanovení výslednej lokalizácie.

Názov poľa	Typ poľa	Vlastnosti / obmedzenia	Poznámka
code	VARCHAR(2)	PRIMARY KEY NOT NULL	skratka krajiny
country	VARCHAR(50)	NULL	názov krajiny

Tab. 4: Štruktúra databázovej tabuľky *country_codes*

Tabuľka *sk_spz*

Tabuľka obsahuje mapovanie dvojnakových skratiek názvov slovenských miest (identifikátorov okresu používaných na štátnych poznávacích značkách v SR) na celý názov slovenského okresného mesta.

Názov poľa	Typ poľa	Vlastnosti / obmedzenia	Poznámka
spz	VARCHAR(3)	PRIMARY KEY NOT NULL	ŠPZ okresu
city	VARCHAR(50)	NULL	názov mesta

Tab. 5: Štruktúra databázovej tabuľky *sk_spz*

Tabuľka *cache*

Tabuľka obsahuje údaje o už lokalizovaných IP adresách. Platnosť záznamov je obmedzená na 24 hodín, staré záznamy sú odstraňované automaticky spúšťaným skriptom (kapitola 3.4). Tabuľku využíva len webová aplikácia pre vizualizáciu lokalizácie IP adries.

Názov poľa	Typ poľa	Vlastnosti / obmedzenia	Poznámka
ip	VARCHAR(50)	PRIMARY KEY NOT NULL	IP adresa
hostname	VARCHAR(50)	NULL	doménové meno
country	VARCHAR(50)	NULL	krajina
city	VARCHAR(50)	NULL	mesto
lat	FLOAT	NULL	zemepisná šírka
lon	FLOAT	NULL	zemepisná dĺžka
score	INTEGER	NULL	skóre presnosti lokalizácie
info	VARCHAR(50)	NULL	textový popis
method	VARCHAR(50)	NULL	metódy vyhľadania
added	TIMESTAMP	CURRENT_TIMESTAMP	dátum pridania záznamu

Tab. 6: Štruktúra databázovej tabuľky *cache*

Nad tabuľkou boli pre urýchlenie vyhľadávania vytvorené 2 indexy:

IDX_CACHE_HOSTNAME (stĺpec *hostname*)

IDX_CACHE_ADDED (stĺpec *added*)

Tabuľka *limiter*

Tabuľka obsahuje údaje o používateľoch webovej aplikácie – uchováva počet lokalizácií a *traceroute-ov*, ktoré používateľ s danou IP adresou vykonal. Slúži pre účely ochrany pred zahľtením (kapitola 3.8.2). Staré záznamy sú periodicky odstraňované automaticky spúšťaným skriptom (kapitola 3.4).

Názov poľa	Typ poľa	Vlastnosti / obmedzenia	Poznámka
ip	VARCHAR(50)	PRIMARY KEY NOT NULL	IP adresa
localizations	VARCHAR(50)	NULL	počet lokalizácií
traceroutes	VARCHAR(50)	NULL	počet <i>traceroute-ov</i>
added	TIMESTAMP	CURRENT_TIMESTAMP	dátum pridania záznamu

Tab. 7: Štruktúra databázovej tabuľky *limiter*

Nad tabuľkou bol pre urýchlenie vyhľadávania vytvorený index:

IDX_CACHE_ADDED (stĺpec *added*)

6.3 Inštalačná príručka

Aplikáciu je možné použiť na platformách **OS Unix/Linux, Windows, Mac OS X, či Solaris**. Pre spustenie aplikácie je potrebný interpretér jazyka *Perl*, konkrétny popis inštalácie sa nachádza v nasledujúcich častiach tejto kapitoly.

6.3.1 Inštalačná príručka konzolovej aplikácie

V tejto časti je popísaný postup inštalácie konzolovej aplikácie pre lokalizáciu IP adres. Príkazy popísané v tejto príručke je nutné vykonať z konta administrátora (*root*), alebo pred samotný príkaz vložiť príkaz `sudo`.

Pre inštaláciu na distribúciách **OS Debian GNU/Linux** a distribúciách od nej odvodených (napr. **Ubuntu**) je dostupný **inštalačný balíček**, ktorý celý postup inštalácie zjednodušuje. Inštalácia spočíva v nainštalovaní balíka `locip_0.1.3_i386.deb` cez správcu balíčkov. V grafickom prostredí pre jeho spustenie väčšinou postačí dvojklik na súbor s balíčkom, alternatívou je použitie príkazu `dpkg`:

```
dpkg -i locip_0.1.3_i386.deb
```

Takto nainštalovanú aplikáciu je možné spustiť z ktoréhokoľvek adresára použitím príkazu `locip`.

V prípade inštalácie na inú platformu je nutné použiť **manuálnu inštaláciu**. Konkrétne príkazy popisujú inštaláciu v prostredí operačného systému *Ubuntu 9.04 Desktop Edition*, inštalácia v prostredí iných operačných systémov sa môže viac či menej líšiť.

Postup manuálnej inštalácie:

- 1) Rozbalenie archívu so zdrojovými kódmi aplikácie:
`tar -xzf locip.tar.gz`
- 2) Inštalácia interpretéra jazyka *Perl* (v prostredí *OS Unix/Linux* je vo väčšine prípadov nainštalovaný spolu s operačným systémom):
`apt-get install perl`
- 3) Inštalácia rozhrania pre prístup k databázam - ovládača *Perl DBI*:
`apt-get install libclass-dbi-perl`
- 4) Inštalácia modulu podpory databáz *SQLite* v *Perl DBI*:
`apt-get install libdbd-sqlite3-perl`
- 5) Nastavenie práv pre spustenie aplikácie:
`chmod 755 locip.pl`
- 6) Spustenie aplikácie
`./locip.pl`

6.3.2 Inštalačná príručka webovej aplikácie

V tejto časti je popísaný postup inštalácie webovej aplikácie pre vizualizáciu lokalizácie IP adres. Konkrétne príkazy popisujú inštaláciu v prostredí operačného systému *Ubuntu 9.04 Desktop Edition*, inštalácia v prostredí iných operačných systémov sa môže viac či menej líšiť. Príkazy je nutné vykonať z konta administrátora (*root*), alebo pred samotný príkaz vložiť príkaz `sudo`.

Postup inštalácie:

- 1) Inštalácia webového servera *Apache*:
`apt-get install apache2`
- 2) Integrácia jazyka *Perl* do prostredia webového servera *Apache*:
`apt-get install libapache2-mod-perl2`
- 3) Úprava konfiguračného súboru webového servera *Apache* – zabezpečenie interpretovania súborov s príponou *.pl* ako skriptov v jazyku *Perl*:
`gedit /etc/apache2/apache2.conf`

Do súboru pridať nasledovné riadky:

```
DirectoryIndex index.pl index.html
AddHandler cgi-script .pl
<Files ~ "\.pl$" >
    Options +ExecCGI
</Files>
```

- 4) Reštartovanie procesu webového servera:
`sudo /etc/init.d/apache2 restart`
- 5) Presun zdrojových kódov aplikácie do koreňového adresára webového serveru a ich rozbalenie :
`mv locip-web.tar.gz /var/www/`
`cd /var/www/`
`tar -xzf locip-web.tar.gz`
- 6) Nastavenie práv pre spustenie všetkých skriptov webu:
`sudo chmod 755 *.pl`
- 7) Spustenie aplikácie – otvorenie URL vo webovom prehliadači:
`http://localhost/`

6.4 Používateľská príručka

Aplikáciu je možné ovládať pomocou štandardných prvkov používateľského rozhrania – príkazového riadku alebo internetového prehliadača. Spôsob ovládania je popísaný v nasledujúcich častiach kapitoly.

6.4.1 Používateľská príručka konzolovej aplikácie

Konzolovú aplikáciu pre lokalizáciu IP adres je možné ovládať prostredníctvom prepínačov a parametrov príkazového riadku.

Základný vstupný formát aplikácie ma nasledovný tvar:

```
locip.pl [-options] [-f [filename1 filename2 ...]] [host1 host2 ...]
```

Z formátu je zrejmé, že IP adresy pre lokalizáciu je možné zadať ako vo forme parametrov programu, tak prostredníctvom súborov zadanych ako parametre prepínača `-f`. V prípade použitia tohto prepínača a nezadania názvu vstupného súboru program číta vstupné IP adresy

zo štandardného vstupu.

Vstup a výstup programu možno ďalej ovplyvniť nasledovnými prepínačmi:

- h : nápoveda k používaniu programu
- m : lokalizuje IP adresu aktívneho sieťového zariadenia počítača, na ktorom je aplikácia spustená
- t : lokalizuje IP adresy jednotlivých „hopov“ na ceste k zadanej destinácii (*traceroute*)

- v : vypočíta vzdialenosť medzi každými dvoma po sebe zadanými IP adresami (v kilometroch)
- b : vypočíta a vypíše vzdialenosť medzi prvými dvoma zadanými IP adresami (v kilometroch)

- q : rýchly režim lokalizácie – po určení mesta lokalizácie neaplikuje ďalšie metódy lokalizácie
- l : lokalizácia analýzou *LOC DNS* záznamu
- w : lokalizácia analýzou údajov z *Whois*
- d : lokalizácia analýzu doménového mena
- x : lokalizácia analýzou *TXT DNS* záznamu

- a : podrobný výpis informácií o lokalizovaných uzloch
- s : skrátenejší výpis informácií o lokalizovaných údajoch
- p : výpis geografických súradníc vo formáte desiatinných čísiel (negatívne číslo značí južnú, resp. západnú pologuľu)

V zozname prepínačov si možno všimnúť, že aplikácia umožňuje používateľovi špecifikovať, ktoré metódy sa pre lokalizáciu IP adresy majú použiť. Napríklad pri zvolení prepínačov *-wd* bude pre lokalizáciu použitá len analýza údajov z *Whois* a analýza doménového mena.

Vy výpise informácií o lokalizovaných uzloch je pre sprehľadnenie výpisu použitých niekoľko pojmov a skratiek:

Formát geografických súradníc

N – severná zemepisná šírka, **S** – južná zemepisná šírka, **W** – západná zemepisná dĺžka, **E** – východná zemepisná dĺžka

Odhad presnosti lokalizácie

Vyjadruje odhadovanú úroveň presnosti stanovenej lokalizácie, relatívnu pravdepodobnosť toho, že určená lokalizácia sa zhoduje so skutočnou lokalizáciou. Určené podľa výšky skóre - číselné ohodnotenie 0 - 30.

VL – veľmi nízka (*very low*), **L** – nízka (*low*), **H** – vysoká (*high*), **VH** – veľmi vysoká (*very high*)

Metóda lokalizácie

Označuje metódu, ktorou sa podarilo stanoviť nejakú (i nesprávnu) lokalizáciu. V prípade určenia mesta lokalizácie je táto skratka uvedená veľkým písmenom, v prípade učenia len krajiny lokalizácie malým písmenom.

L - LOC DNS záznam, **W** - Whois, **D** – doménové meno, **X** - TXT DNS záznam

6.4.2 Používateľská príručka webovej aplikácie

Webovú aplikáciu pre vizualizáciu lokalizácie IP adres je možné spustiť v ľubovoľnom internetovom prehliadači, ideálne zobrazenie používateľského rozhrania je však možné dosiahnuť

použitím prehliadača *Mozilla Firefox 3*.

Hlavnú podstránku webu tvorí rozhranie pre **lokalizáciu jednej IP adresy** (Obr. 11). Po príchode na túto podstránku prebehne automatická lokalizácia používateľovej IP adresy. IP adresu alebo doménové meno uzla na lokalizáciu je možné zadať do príslušného textového poľa (1), novú lokalizáciu je možné vyvolať použitím tlačidla *Lokalizovať*. Lokalizácia zadaného uzla je na mape znázornená značkou (2), po kliknutí na značku sa zobrazia súhrnné informácie v bubline nad značkou. Podrobné informácie o lokalizácii IP adresy sa nachádzajú v pravej časti stránky (3). Pohyb po mape, približovanie a vzdiaľovanie umožňujú ovládacie prvky v ľavej hornej časti mapy, tlačidlá a koliesko myši.

V pravej časti hlavného menu stránky zobrazeného na každej podstránke webu (4) sa nachádza vstupné pole rýchlej lokalizácie. Po vložení IP adresy, resp. doménového mena uzla do tohto poľa a kliknutí na tlačidlo s obrázkom Zeme je táto adresa okamžite lokalizovaná a výsledky lokalizácie sú zobrazené práve popísaným spôsobom.

The screenshot shows the IP Tracker website interface. At the top, there is a navigation menu with options: "Lokalizácia IP", "Traceroute", "Postupnosť IP", and "Viac postupností IP". The current IP address being tracked is "92.52.12.141". Below the navigation menu, there is a search bar with the IP address "147.175.187.62" entered. A red "1" is next to the search bar, and a "Lokalizovať" button is to its right. To the right of the search bar, it says "Vaša IP adresa: 92.52.12.141". Below the search bar, there is a map of Europe with a red location marker (2) in Bratislava, Slovakia. A tooltip above the marker displays the IP address "147.175.187.62", the name "Slovak Technical University", the location "Bratislava, Slovakia", and the coordinates "48° 08' 59" N, 17° 07' 00" E". To the right of the map, there is a table of information for "Slovak Technical University":

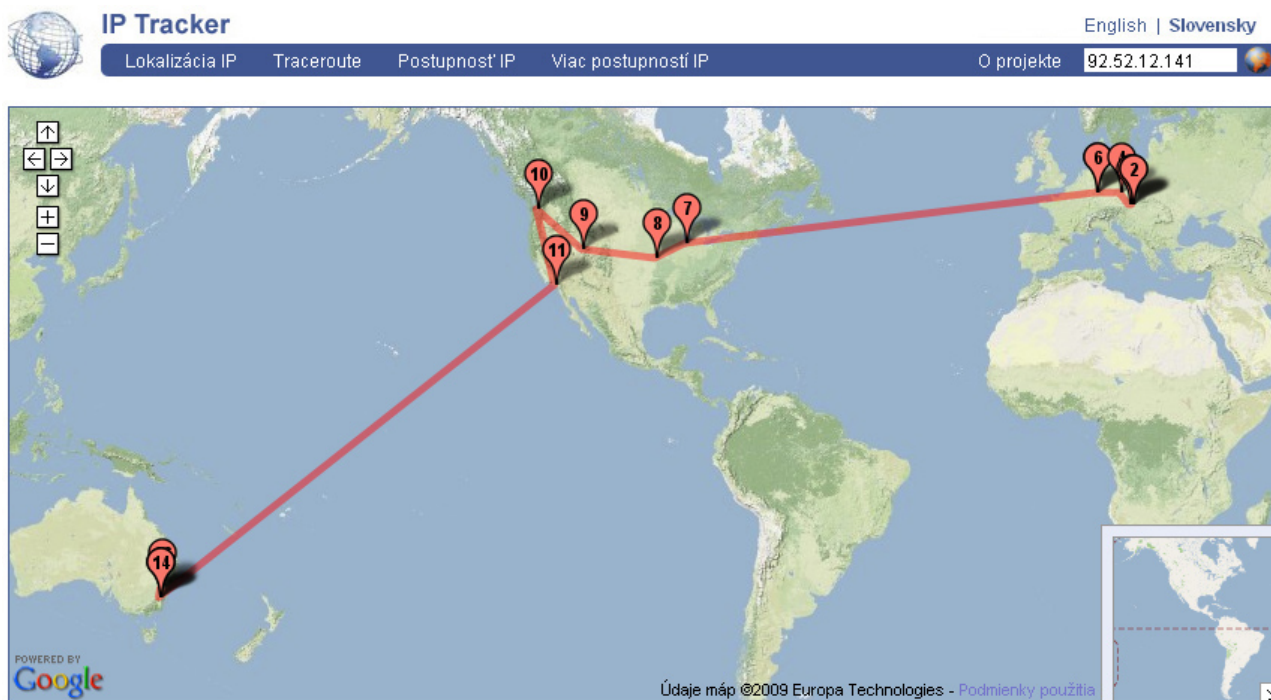
Slovak Technical University	
IP adresa:	147.175.187.62
Doménové meno:	creas.ynet.sk
Lokalizácia:	Bratislava, Slovakia
Súradnice:	48° 08' 59" N 17° 07' 00" E 3
Metóda lokalizácie:	whois (mesto) doménové meno (krajina)
Presnosť lokalizácie:	nízka (10)

Obr. 11: Používateľské rozhranie pre lokalizáciu jednej IP adresy

Druhou položkou v hlavnom menu stránky je **traceroute**. Tento nástroj umožňuje vizualizovať cestu sieťových paketov Internetom. IP adresu alebo doménové meno destinácie je možné zadať do príslušného textového poľa (rovnaké ako na Obr. 11), lokalizáciu jednotlivých „hopov“ na ceste k destinácii je možné spustiť použitím tlačidla *Lokalizovať*.

Príklad výsledku požiadavky na vizuálny *traceroute* je zobrazený na obrázku Obr. 12. Jednotlivé „hopy“ *traceroute*-u sú znázornené značkami s poradovým číslom „hopu“ (v prípade, že do tej istej lokalizácie je namapovaných viacej „hopov“, zobrazené je číslo posledného „hopu“ v danej lokalizácii). Pri podržaní kurzora myši nad textom s prerušovaným podčiarknutím (1) sa zobrazia podrobné informácie o danom objekte, v tomto prípade doménové meno prislúchajúce IP adrese.

Kliknutím na riadok tabuľky s informáciami o lokalizácii jednotlivých „hopov“ sa lokalizácia zvoleného „hopu“ zobrazí v strede mapy. Pôvodné zobrazenie pohľadu na všetky lokalizované uzly je možné vyvolať tlačidlom *Pôvodné zobrazenie* (2). Tlačidlom *Nová postupnosť* (3) je umožnený návrat k zadaniu destinácie *traceroute-u*.



N.	IP addressa	Info	Lokalizácia	Metóda	Presnosť
1	193.87.2.9	Slovak Technical University	Bratislava, Slovakia	W D	vyšoká (14)
2	193.168.0.150	Slovak Academic Network (SANET)	Bratislava, Slovakia	W D	vyšoká (14)
3	62.40.125.5	DANTE Ltd.	Innere Stadt, Austria	L W	nízka (6)
4	62.40.112.6	IP allocation for GEANT network	Bubenec, Czech Republic	L W	nízka (6)
5	62.40.112.38	IP allocation for GEANT network	Frankfurt Am Main, Germany	L W	nízka (6)
6	62.40.125.18	DANTE Ltd.	Frankfurt Am Main, Germany	L W	nízka (6)
7	64.57.28.12	Internet2	Chicago, United States	L W	nízka (6)
8	64.57.28.36	Internet2	Kansas City, United States	L W	nízka (6)
9	64.57.28.24	Internet2	Salt Lake City, United States	L W	nízka (6)
10	64.57.28.26	Internet2	Seattle, United States	L W	nízka (6)
11	207.231.240.2	Pacific Wave	Los Angeles, United States	W	nízka (8)
12	202.158.194.93	Australian Academic and Research Network	Canberra, Australia	W d	nízka (10)
13	202.158.202.26	Australian Academic and Research Network	Canberra, Australia	W d	nízka (10)
14	202.158.206.10	Australian Academic and Research Network	Canberra, Australia	W d	nízka (10)
16	137.166.4.200	imported inetnum object for CSU-1	Bathurst, Australia	W d	nízka (10)

Nová postupnosť 3

2 Pôvodné zobrazenie

Obr. 12: Používateľské rozhranie výsledku vizuálneho traceroute-u k destinácii *csu.edu.au*

Ako je možné si všimnúť na obrázku *Obr. 12*, tabuľka neobsahuje riadok s 15. „hopom“ (riadok s číslom 14 je nasledovaný riadkom s číslom 16). Znamená to, že z 15. „hopu“ neprišla odpoveď *Time-to-live exceeded* (kapitola 3.3).

Ďalšou možnosťou aplikácie je **vizualizácia postupnosti IP adres** (*Obr. 13*). Postupnosť IP adres je možné zadať buď do textového poľa na stránke (1), alebo vložení textového súboru s postupnosťou IP adres (2). Používateľ má na výber medzi dvoma formátmi vloženia postupnosti IP adres – striktného a voľného. Pri striktnom formáte (3) musí byť každá adresa na samostatnom riadku, čo však umožní zadávať adresy uzlov nielen v podobe IP adres, ale i v podobe

doménového mena uzlov. Voľný formát (4) umožňuje zadanie akéhokoľvek textu, z ktorého aplikácia automaticky načíta všetky IP adresy. Je možné použiť napríklad výstup programu *traceroute* vykonaného z používateľovho počítača. V tomto prípade však nie je možné zadávať adresy uzlov v podobe doménových mien.

The screenshot shows the IP Tracker application interface. At the top, there is a navigation bar with the application name 'IP Tracker' and language options 'English | Slovensky'. Below the navigation bar, there is a search bar with the IP address '147.175.187.62' and a 'Lokalizovať' button. The main content area displays a list of IP addresses and their ping times, numbered 3 through 18. The list is as follows:

```
3 FEI-Bratislava.sanet2.sk (193.87.2.9) 2.548 ms 2.584 ms 2.604 ms
4 CVT-Bratislava.sanet2.sk (194.160.8.150) 2.634 ms 2.669 ms 2.819 ms
5 sanet.rt1.vie.at.geant2.net (62.40.125.5) 3.877 ms 4.120 ms 4.270 ms
6 so-7-0-0.rt1.pra.cz.geant2.net (62.40.112.6) 10.441 ms 9.240 ms 9.403 ms
7 so-6-3-0.rt1.fra.de.geant2.net (62.40.112.38) 17.206 ms 17.049 ms 17.238 ms
8 abilene-wash-gw.rt1.fra.de.geant2.net (62.40.125.18) 118.161 ms 118.278 ms 118.343 ms
9 so-0-2-0-0.rtr.chic.net.internet2.edu (64.57.28.12) 127.233 ms 127.313 ms 127.376 ms
10 so-4-3-0-0.rtr.kans.net.internet2.edu (64.57.28.36) 137.971 ms 138.005 ms 138.040 ms
11 so-0-0-0-0.rtr.salt.net.internet2.edu (64.57.28.24) 187.283 ms 181.272 ms 181.308 ms
12 so-0-0-0-0.rtr.seat.net.internet2.edu (64.57.28.26) 178.456 ms 178.487 ms 178.079 ms
13 aarnet-1-lo-jmb-706.stlwa.pacificwave.net (207.231.240.2) 178.229 ms 178.211 ms 178.302 ms
14 so-3-3-0.bb1.b.syd.aarnet.net.au (202.158.194.93) 333.212 ms 333.024 ms 333.041 ms
15 gigabithernet0.er1.a.syd.aarnet.net.au (202.158.202.26) 333.252 ms 334.126 ms 334.095 ms
16 csu.gw.aarnet.net.au (202.158.206.10) 348.547 ms 348.492 ms 348.566 ms
17 ***
18 www.csu.edu.au (137.166.4.200) 351.669 ms 349.023 ms 345.722 ms
```

Below the list, there is a 'Prehľadávať...' button and a 'Lokalizovať' button. At the bottom, there is a 'Formát vstupu:' section with two radio buttons: 'striktný (každá adresa na samostatnom riadku)' (selected) and 'voľný (len IP adresy)'. The selected option is marked with a red '3' and the unselected option with a red '4'.

Obr. 13: Používateľské rozhranie vstupu postupnosti IP adries vo voľnom formáte

Výsledné zobrazenie postupnosti IP adries je zhodné ako v prípade *traceroute* (Obr. 12).

Poslednou voľbou v hlavnom menu aplikácie je **vizualizácia viacerých postupností IP adries** (Obr. 14). V tomto prípade je možné vložiť i kombináciu IP adries a doménových mien, pričom každá adresa musí byť na samostatnom riadku. Jednotlivé postupnosti sú oddelené prázdny riadkom. Používateľ má možnosť voľby medzi relatívnym (1) a absolútnym (2) zobrazením vyřaženia liniek.



Zadajte **postupnosti IP adries** (každá adresa na samostatnom riadku, prázdny riadok pre oddelenie postupností):

```
caida.org
123.193.64.146
124.101.248.42

123.193.64.146
caida.org
124.170.89.241
141.48.223.1

137.224.10.105
123.193.64.146
caida.org
124.170.89.241
```

Prehľadávať...

Zobrazenie vyťaženia liniek: relatívne absolútne

1

2

Lokalizovať

Obr. 14: Používateľské rozhranie vstupu viacerých postupností IP adries

V prípade relatívneho zobrazenia vyťaženia liniek sú jednotlivé linky farebne odlišené podľa počtu prechodov danou linkou od najmenej vyťaženej (0% - zelená farba) po najviac vyťaženej (100% - červená farba).

Pri absolútnom zobrazení vyťaženia liniek sú jednotlivé linky farebne odlišené podľa počtu prechodov danou linkou od 1 (zelená farba) po 20 (červená farba).



Relatívne vyťaženie linky: ■ 0% ■ 50% ■ 100%

Postupnosť A:

N.	IP address	Info	Lokalizácia	Metóda	Presnosť
1	192.172.226.78	Cooperative Association for Internet Data Analysis, La Jolla CA	La Jolla, United States	L W X	veľmi vysoká (24)
2	123.193.64.146	TUNG HO MULTIMEDIA CO. Ltd.	Taipei, Taiwan	W d	nízka (10)
3	124.101.248.42	NTT Communications Corporation	Marunouchi, Japan	W D	veľmi nízka (2)

Postupnosť B:

N.	IP address	Info	Lokalizácia	Metóda	Presnosť
1	123.193.64.146	TUNG HO MULTIMEDIA CO. Ltd.	Taipei, Taiwan	W d	nízka (10)
2	192.172.226.78	Cooperative Association for Internet Data Analysis, La Jolla CA	La Jolla, United States	L W X	veľmi vysoká (24)
3	124.170.89.241	iiNet Limited	Perth, Australia	W d	nízka (10)
4	141.48.223.1	Martin-Luther-Universitaet Halle-Wittenberg	Halle, Germany	W	nízka (8)

Postupnosť C:

N.	IP address	Info	Lokalizácia	Metóda	Presnosť
1	137.224.10.105	Wageningen University and Research Centre	Wageningen, Netherlands	W d	nízka (10)
2	123.193.64.146	TUNG HO MULTIMEDIA CO. Ltd.	Taipei, Taiwan	W d	nízka (10)
3	192.172.226.78	Cooperative Association for Internet Data Analysis, La Jolla CA	La Jolla, United States	L W X	veľmi vysoká (24)
4	124.170.89.241	iiNet Limited	Perth, Australia	W d	nízka (10)

Nová postupnosť

Pôvodné zobrazenie

Obr. 15: Používateľské rozhranie výsledku vizualizácie viacerých postupností IP adres

7 Zoznam použitej literatúry

- [1] HOLZHAUER, F.: IP geolocation. Berlin: Technische Universität. 2007. 11 s.
- [2] Google Adwords: Regionálne a lokálne zacielenie [online]. 2008 [cit. 2008-09-26]. Dostupné na Internete: <<https://adwords.google.com/select/targeting.html>>.
- [3] DataStorm Information Systems: What is my IP address? [online]. 2008 [cit. 2008-10-05]. Dostupné na Internete: <<http://whatismyipaddress.com>>.
- [4] Hongfeng, Sun: Find the geographical location given IP address [online]. 2004 [cit. 2008-10-11]. Dostupné na Internete: <http://blogs.warwick.ac.uk/java/entry/find_the_geographical>.
- [5] Direct Information Pvt. Ltd.: IP Address Lookup [online]. 2008 [cit. 2008-10-13]. Dostupné na Internete: <<http://ip-lookup.net>>.
- [6] MaxMind, Inc: GeolIP database [online]. 2008 [cit. 2008-10-05]. Dostupné na Internete: <<http://www.maxmind.com/app/ip-location>>.
- [7] Hostip.info: Community Geotarget IP Project [online]. 2008 [cit. 2008-10-06]. Dostupné na Internete: <<http://www.hostip.info>>.
- [8] DOSTÁLEK, L., KABELOVÁ, A.: Velký průvodce protokoly TCP/IP a systémem DNS. 2. vyd. Praha: Computer Press, 2000. 426 s. ISBN: 80-7226-323-4
- [9] The Internet Engineering Task Force: Request for Comments 3912 [online]. 2004 [cit. 2008-10-26]. Dostupné na Internete: <<http://www.ietf.org/rfc/rfc3912.txt>>.
- [10] RIPE NCC: RIPE Database Query Reference Manual [online]. 2008 [cit. 2008-11-16]. Dostupné na Internete: <<http://www.ripe.net/db/support/query-reference-manual.pdf>>.
- [11] Internet Assigned Numbers Authority: Root Zone Database [online]. 2008 [cit. 2008-10-13]. Dostupné na Internete: <<http://www.iana.org/domains/root/db/index.html>>.
- [12] Salamon, András: DNS Resource Records [online]. 2004 [cit. 2008-10-26]. Dostupné na Internete: <<http://www.dns.net/dnsrd/rr.html>>.
- [13] The Internet Engineering Task Force: Request for Comments 1876 [online]. 1996 [cit. 2008-10-13]. Dostupné na Internete: <<http://www.ietf.org/rfc/rfc1876.txt>>.
- [14] Davis, Christopher: Geo-enabling the Domain Name System [online]. 2001 [cit. 2008-10-26]. Dostupné na Internete: <<http://www.ckdhr.com/dns-loc>>.
- [15] SATRAPA, P.: Perl pro zelenáče. 2. vyd. Praha: Neocortex, 2001. 224 s. ISBN: 80-86330-02-8
- [16] Václavík, Jiří: Perl [online]. 2005 [cit. 2008-11-05]. Dostupné na Internete: <http://www.linuxsoft.cz/article.php?id_article=675>.
- [17] Google, Inc: Google Maps API [online]. 2008 [cit. 2008-11-05]. Dostupné na Internete: <<http://code.google.com/apis/maps/>>.
- [18] MetaCarta, Inc: Open Layers [online]. 2008 [cit. 2008-11-05]. Dostupné na Internete: <<http://www.openlayers.org/>>.

- [19] Hipp, Wyrick & Company, Inc: About SQLite [online]. 2008 [cit. 2008-11-23]. Dostupné na Internetu: <<http://www.sqlite.org/about.html>>.
- [20] Movable Type Ltd: Calculate distance, bearing and more between two Latitude/Longitude points [online]. 2009 [cit. 2009-04-23]. Dostupné na Internetu: <<http://www.movable-type.co.uk/scripts/latlong.html>>.
- [21] Wardley, Andy: Template Toolkit [online]. 2008 [cit. 2009-04-11]. Dostupné na Internetu: <<http://template-toolkit.org>>.

Zoznam skratiek

- AfriNIC** - African Network Information Center
- API** - Application Programming Interface
- APNIC** - Asia Pacific Network Information Centre
- ARIN** - American Registry for Internet Numbers
- CGI** - Common Gateway Interface
- CPAN** - Comprehensive Perl Archive Network
- CSS** - Cascading Style Sheets
- DNS** – Domain Name System
- IANA** – Internet Assigned Numbers Authority
- ICMP** - Internet Control Message Protocol
- IM** - Instant Messaging
- IP** – Internet Protocol
- LACNIC** - Latin American and Caribbean IP Address Regional Registry
- OS** – Operating System
- OSI** – Open Systems Interconnection
- RIPE NCC** - Reseaux IP Europeens Network Coordination Centre
- RIR** - Regional International Registries
- RR** - Resource Record
- SPF** - Sender Policy Framework
- TCP** - Transmission Control Protocol
- TTL** - Time To Live
- UDP** - User Datagram Protocol
- VoIP** – Voice over IP

Obsah elektronického média

Obsah priloženého elektronického média je rozdelený do nasledovných adresárov:

/application/ - balíčky s konzolovou aplikáciou pre lokalizáciu IP adres (*.deb, .tar.gz, .zip*)

/documentation/ - elektronická verzia tohto dokumentu

/install/ - inštalačné súbory interpretéra jazyka *Perl* pre *OS Linux, Mac OS X, Windows*

/sources/module/ - zdrojové kódy modulu pre lokalizáciu IP adres

/sources/console/ - zdrojové kódy konzolovej aplikácie pre lokalizáciu IP adres

/sources/web/ - zdrojové kódy webovej aplikácie pre vizualizáciu lokalizácie IP adres

/sources/deb/ - zdrojové kódy potrebné pre vytvorenie *Debian* balíčka

/test/inputs/ - niekoľko textových súborov so zaujímavými testovacími vstupmi

/test/data/ - databázové súbory výstupu testovania (formátu *SQLite 3*)