

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

MOŽNOSTI KONVERZE SOUŘADNÝCH SYSTÉMŮ PRO WEBOVOU APLIKACI

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

MAREK FIALA

BRNO 2015



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

MOŽNOSTI KONVERZE SOUŘADNÝCH SYSTÉMŮ PRO WEBOVOU APLIKACI

COORDINATE SYSTEM CONVERSION OPTIONS FOR WEB APPLICATION

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

MAREK FIALA

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. TOMÁŠ KAŠPÁREK

BRNO 2015

Abstrakt

Cílem této práce je nalézt a porovnat dostupné knihovny a nástroje pro konverzi souřadnic mezi souřadnicovými systémy WGS84, S-JTSK, UTM zone 33N a Pseudo-Mercator, které se dají využít ve webových aplikacích. Nalezené možnosti pro konverzi souřadnic jsou testovány se zaměřením na rychlost a přesnost konverze. Hlavním výsledkem této práce je seznam použitelných variant pro konverzi souřadných systémů včetně porovnání výkonu a přesnosti a zhodnocení jejich výhod a nevýhod.

Abstract

The aim of this work is to find and compare tools and libraries for conversion of coordinates between Coordinate System WGS84, S-JTSK, UTM zone 33N and Pseudo-Mercator, which could be used in Web Application. Found options are then tested with focus on speed and accuracy of conversion. The main result of this work is list of usable possibilities for coordinate system conversion including comparison of performance and accuracy and its advantages and disadvantages.

Klíčová slova

konverze souřadnic, transformace souřadnic, webové aplikace, GIS, S-JTSK, WGS84, Proj4, Proj4js, Mapy API

Keywords

conversion coordinates, transformation coordinates, web application, GIS, S-JTSK, WGS84, Proj.4, Proj4js, Mapy API

Citace

Marek Fiala: Možnosti konverze souřadných systémů pro webovou aplikaci, bakalářská práce, Brno, FIT VUT v Brně, 2015

Možnosti konverze souřadných systémů pro webovou aplikaci

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Tomáše Kašpárka.

Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Marek Fiala
19. května 2015

Poděkování

Zde bych chtěl poděkovat vedoucímu panu Ing. Tomáši Kašpárkovi za poskytnutou odbornou pomoc při řešení této práce.

© Marek Fiala, 2015.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

1 Úvod	3
2 Matematické základy pro tvorbu map	4
2.1 Referenční plochy	4
2.1.1 Referenční elipsoid	4
2.1.2 Referenční koule	5
2.1.3 Referenční rovina	5
2.2 Kartografická zobrazení	5
2.2.1 Dělení podle polohy zobrazovací plochy	5
2.2.2 Dělení podle zobrazovací plochy	6
2.2.3 Dělení podle vlastnosti z hlediska zkreslení	7
2.3 Souřadnicové systémy	7
2.3.1 Zeměpisné souřadnice	7
2.3.2 Prostorové pravoúhlé souřadnice	7
2.3.3 Kartografické (konstrukční) souřadnice	8
2.3.4 Pravoúhlé rovinné souřadnice	9
2.3.5 Rovinné polární souřadnice	9
3 Definice souřadnicových systémů pro účely GIS	10
3.1 Formát pro definování SRS	10
3.1.1 WKT - Well Known Text	10
3.1.2 Formát Proj4	10
3.2 Databáze parametrů EPSG	11
4 Souřadnicové systémy využívané v České republice	12
4.1 World Geodetic System 1984 (WGS84)	12
4.2 Systém Jednotné trigonometrické sítě katastrální (S-JTSK)	13
4.2.1 Křovákovo zobrazení	13
4.3 World Geodetic System 1984 UTM	14
4.4 World Geodetic System 1984 Pseudo-Mercator	15
5 Konverze mapových souřadnic	16
5.1 Transformační metody	16
5.1.1 Podobnostní transformace	16
5.1.2 Metody na bázi gridu	17
5.2 Existující nástroje	18
5.2.1 Proj4	18
5.2.2 GeoTools	19

5.2.3	Mapy API	20
6	Vybrané způsoby řešení	21
6.1	Způsob členění	21
6.1.1	Podle použitého nástroje	21
6.1.2	Podle typu a místa užití nástroje	21
6.1.3	Podle jazykové verze knihovny	21
6.2	Možnost nastavení přesnosti	22
6.2.1	Podpora transformačních metod	22
6.2.2	Transformační klíč	22
6.3	Souhrn nástrojů z předchozí kapitoly po členění	23
6.4	Konkrétní způsoby řešení	24
6.4.1	Proj.4	24
6.4.2	GeoTools	27
6.4.3	Mapy API	27
6.4.4	Vzdálené používání knihoven a API	27
7	Testování	28
7.1	Způsob testování rychlosti převodu	28
7.2	Nástroje v jazyce C/C++ a jazyce Java	29
7.3	Nástroje v jazyce JavaScript	29
7.4	Databázové nástroje	31
7.5	Celkové srovnání	31
7.6	Závislost přesnosti a výkonu	32
8	Demonstrační webová aplikace	34
8.1	Popis aplikace	34
8.2	Implementace	35
9	Závěr	36
A	Obsah CD	39
B	Program v jazyce C využívající knihovnu Proj.4	40

Kapitola 1

Úvod

Cílem této práce je nalézt vhodné knihovny a nástroje na konverzi souřadnic mezi souřadnicovými systémy WGS84, S-JTSK, UTM zone 33N a Pseudo-Mercator se zaměřením na použití ve webových aplikacích.

Tato problematika se stává s rostoucími požadavky na webové aplikace, konkrétně webové klienty pro geografické informační systémy, stále aktuálnější a aplikace, které neumí pracovat s daty v jiném než projektovém souřadnicovém systému, mají značně omezené využití.

Jednou z možností řešení tohoto problému je udržování dat ve všech podporovaných souřadnicových systémech, ale tento způsob je značně nevhodný vzhledem k vyšší paměťové náročnosti. Další nevýhodou je náročná práce s daty, při které je vždy potřeba data vkládat, modifikovat nebo odstraňovat zvláště pro každý souřadnicový systém. Proto je nejvhodnějším řešením právě konverze souřadnic mezi požadovanými souřadnicovými systémy.

Pro tyto účely je potřeba nalézt, nastudovat, uvážit použití jednotlivých řešení pro konverzi souřadnic a dále také zhodnotit jejich výhody a nevýhody. Součástí této práce je rovněž porovnání nalezených nástrojů z hlediska přesnosti konverze a výkonnosti.

Druhá kapitola je určena k přehledu základních informací ohledně tvorby map a nachází se zde také vysvětlení pojmů, které jsou důležité pro pochopení dalších kapitol.

Třetí kapitola obsahuje pojmy, které jsou důležité pro využití souřadnicových systémů a geoinformačních systémů.

Ve čtvrté kapitole jsou uvedeny informace ohledně souřadnicových systémů, kterými se tato práce zabývá.

Pátá kapitola obsahuje přehled transformačních metod, které je možno použít mezi souřadnicovými systémy. Dále se zde nachází seznam nástrojů pro konverzi souřadnic, které byly vyhledány pro účely této práce a které jsou předmětem dalšího zkoumání.

Šestá kapitola popisuje, jakými způsoby lze nalezené nástroje pro konverzi souřadnic rozčlenit pro účely dalšího testování, a také jsou zde popsány bližší informace o jednotlivých nástrojích.

V sedmé kapitole jsou uvedeny výsledky testů a výhody a nevýhody jednotlivých řešení.

Osmá kapitola obsahuje popis demonstrační webové aplikace pro konverzi souřadnic, která byla vytvořena v závislosti na informacích obsažených v kapitole sedm.

Poslední, devátá kapitola obsahuje celkové shrnutí.

Kapitola 2

Matematické základy pro tvorbu map

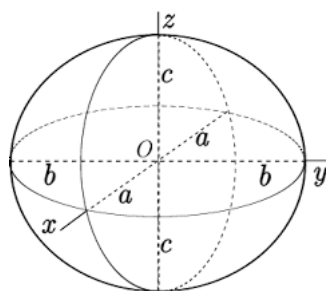
2.1 Referenční plochy

Zemské těleso je velice složité a členité, a proto je jej potřeba pro účely mapování nahradit *referenční plochou*, která je jednodušší a matematicky nebo fyzikálně přesně definovaná.

Geoid je těleso, které nejlépe odpovídá nerušené střední hladině světových moří, potažené i pod kontinenty. Tato plocha je ve všech bodech kolmá na směr tíhové síly Země. Geoid je definován jako fyzikální těleso a jeho matematické vyjádření je značně složité, proto pro potřeby praktické geodézie, mapování, kartografie i celé geoinformatiky je geoid nahrazován referenčním elipsoidem (spheroid), referenční kouli (sphere) nebo referenční rovinou.[26]

2.1.1 Referenční elipsoid

Referenční elipsoid vznikl rotací elipsy kolem vedlejší osy. Jeho velikost i tvar určují dva parametry, a to buď jeho poloosy (a , b) nebo jedna poloosa a excentricita (e^2) nebo jedna poloosa a zploštění (i).[27]



Obrázek 2.1: Referenční elipsoid
Zdroj: GIS StackExchange¹.

¹<http://gis.stackexchange.com/tags/ellipsoid/info>

Referenční elipsoidy jsou používány, pokud je nutné definovat zobrazení s minimálními hodnotami zkreslení. Tento způsob se volí u kartografických zobrazení při definici státních souřadnicových systémů, mezinárodních systémů nebo při tvorbě státních mapových děl.[26]

Do současnosti byla odvozena řada referenčních elipsoidů. Zde jsou dva referenční elipsoidy používané v české geografické a kartografické praxi: [25]

- Besselův elipsoid - používán pro civilní státní mapová díla. Byl určen F. W. Bessellem (1841) na základě výpočtu poledníkových oblouků z deseti různých měření především v evropských zemích. Do roku 1942 byl používán v SSSR, v současnosti je použit jako referenční plocha topografických map ve státech střední Evropy.

Parametry tohoto elipsoidu jsou $a = 6\,377\,397,1550\text{m}$, $b = 6\,356\,078,9633\text{m}$, $f = 1:299,152813$ $= 0,00667437223061$, $e^2 = 0,00334277318158$. [27]

- WGS84 (World Geodetic System 1984) - nejnovější a nejpřesnější referenční elipsoid vypočtený na základě družicového měření. Střed elipsoidu WGS84 je totožný se středem Země. Maximální odchylka jeho povrchu od povrchu geoidu činí 60 m. Tento elipsoid používá ČR pro vojenská mapová díla.

Parametry tohoto elipsoidu jsou $a = 6\,378\,136,0\text{m} \pm 1\text{m}$, $b = 6\,356\,752,0\text{m}$, $f = 1:298,257$ $= 0,00669378$, $e^2 = 0,00335281317$. [27]

2.1.2 Referenční koule

Referenční koule je jako referenční plochu možno použít, pokud není vyžadována vysoká přesnost prostorové lokalizace modelovaných objektů a jevů. Tohoto se využívá zejména při tvorbě map malých měřítek (1:1 000 000 a menší) nebo v případě mapování malého území (do průměru 200 km).

Poloměry referenčních koulí se pohybují od 6 366 743 m do 6 371 291 m². [27]

Zvláštním případem použití referenční koule je při tzv. *dvojitém zobrazení*, kdy je referenční elipsoid nejprve zobrazen na kouli, která se poté zobrazuje do roviny. [26]

2.1.3 Referenční rovina

Referenční rovinu je možné použít jako referenční plochu při tvorbě map z velmi malého území o poloměru zhruba do 20 km. Vodorovné úhly a délky na zakřivené ploše jsou v tomto případě téměř totožné jako v rovině. [26]

2.2 Kartografická zobrazení

Kartografické zobrazení je způsob, jakým lze přiřadit každému bodu na referenční ploše právě jeden bod na zobrazovací ploše.

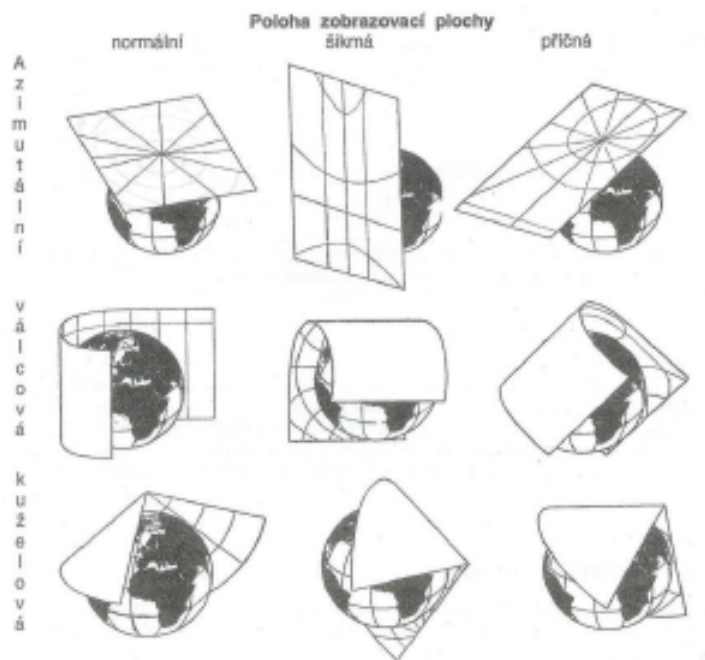
2.2.1 Dělení podle polohy zobrazovací plochy

Podle polohy zobrazovací plochy se kartografická zobrazení dělí na následující: [27]

1. normální (polární) - osa zobrazovací plochy je shodná se zemskou osou, tj. osou referenční plochy

²V české kartografické tvorbě se používá referenční koule s poloměrem odvozeným od elipsoidu Krasovského, který má hodnotu 6 371 110 m. [27]

2. příčná (transversální) - osa zobrazovací plochy leží v rovině rovníku
3. šikmá (obecná) - osa zobrazovací plochy prochází středem referenční plochy, ovšem v jiném směru než u normální a příčné polohy.



Obrázek 2.2: Jednoduchá kartografická zobrazení a poloha zobrazovací plochy.
Zdroj: [27].

2.2.2 Dělení podle zobrazovací plochy

Podle zobrazovací plochy se kartografická zobrazení rozlišují na následující:

1. Jednoduchá zobrazení

Každé kartografické zobrazení je jednoznačně určeno zobrazovacími rovnicemi. Těmito rovnicemi je specifikován vztah polohy bodu na originále a v obraze.[23]

 - (a) Azimutální - zobrazovací plochou je rovina. Vhodné pro mapy území tvaru kulového vrchlíku. Využívá se pro mapy malých měřítek.
 - (b) Válcové - zobrazovací plochou je plášť válce, který je posléze rozvinut do roviny. Vhodné pro mapy území protáhlých podél hlavní kružnice.
 - (c) Kuželové - zobrazovací plochou je plášť kužele, který je poté rozvinut do roviny. Vhodné pro mapy území protáhlých podél vedlejší kružnice.[29]
2. Obecná zobrazení

Popis těchto zobrazení lze nalézt například v [27].

2.2.3 Dělení podle vlastnosti z hlediska zkreslení

Mapa je vždy vůči referenční ploše tvarově zkreslená, protože referenční plocha má elipsoidní nebo kulový tvar. Zkreslení roste v závislosti na zvětšujícím se rozsahu území.

Podle vlastností zkreslení lze rozlišit:

- Ekvidistantní (délkojevná) zobrazení - nezkresluje délky určité soustavy nebo určitou soustavu samotnou (zpravidla bývají touto soustavou zeměpisné poledníky nebo rovnoběžky).[27]
- Ekvivalentní (zplochojevná) zobrazení - nezkresluje plochy, avšak poměrně značně zkresluje úhly. Tato zobrazení se používají v geografii tam, kde je kladen značný důraz na vzájemnou srovnatelnost velikostí území.[27]
- Konformní (úhlojevná) zobrazení - nezkresluje úhly, ale značně zkresluje plochy. Úhlojevná zobrazení se používají nejčastěji v topografii a pro námořní a leteckou navigaci.[27]

2.3 Souřadnicové systémy

Souřadnicový systém je množina matematických pravidel, podle nichž jsou souřadnice přiřazovány k bodům. Souřadnicový systém obsahuje definici souřadnicových os a jednotek.[24]

2.3.1 Zeměpisné souřadnice

Poloha bodu na referenčním elipsoidu nebo na referenční kouli je nejčastěji udávána pomocí zeměpisných souřadnic. Těmi jsou zeměpisná šířka a zeměpisná délka.[23]

Zeměpisná šířka (ϕ) - úhel svíraný normálou ke kuželové referenční ploše s rovinou rovníku referenční koule.[27]

Zeměpisná šířka dosahuje hodnot v rozsahu $\langle -90^\circ, 90^\circ \rangle$. Hodnoty jsou často označovány i jako *jižní* (pro $\langle -90^\circ, 0^\circ \rangle$) a *severní* (pro $\langle 0^\circ, 90^\circ \rangle$) zeměpisná šířka.[26]

Zeměpisná délka (λ) - úhel svíraný rovinou základního poledníku a místního poledníku.

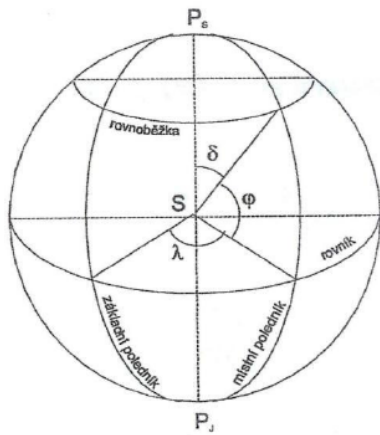
Zeměpisná délka nabývá hodnot $\langle 0^\circ, 360^\circ \rangle$ s počátkem na základním poledníku a přírůstkem ve východním směru. Někdy jsou hodnoty v rozsahu $\langle 0^\circ, 180^\circ \rangle$ označovány jako *východní* (se znaménkem kladným na východ) a *západní* (záporným směrem na západ) zeměpisná délka.[27]

Za *základní (nultý) poledník* je považován ten procházející observatoří v Greenwichi v Londýně. V některých státech je v praktické geodézii používán jako základní poledník i poledník *Ferro*, jehož zeměpisná délka je $17^\circ 40'$ západně od Greenwiche.[26]

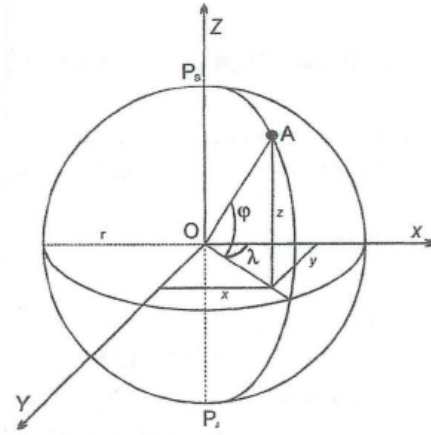
2.3.2 Prostorové pravoúhlé souřadnice

Prostorové pravoúhlé souřadnice jsou definovány počátkem O , ležícím ve středu referenční koule, osou x , danou průsečíkem roviny rovníku se základním poledníkem. Dále osou y , která leží v rovině a svírá s osou x úhel 90° a osou z , která se nachází v zemské ose.

Prostorové pravoúhlé souřadnice se dříve využívaly méně než dnes. Důvodem vzrůstu významu je jejich využití v umělých družicích Země pro geodetické, geofyzikální a kartografické účely.[27]



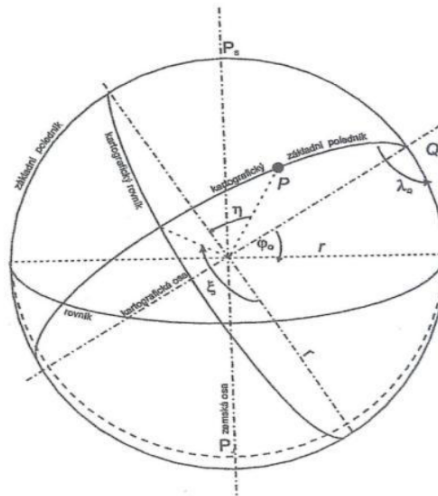
Obrázek 2.3: Zeměpisné souřadnice.
Zdroj: [27].



Obrázek 2.4: Prostorové pravoúhlé souřadnice. Zdroj: [27].

2.3.3 Kartografické (konstrukční) souřadnice

Kartografické souřadnice tvoří *kartografická šířka* η , *kartografická délka* ζ a *kartografický pól* Q . Kartografický pól je bod o zeměpisných souřadnicích ϕ_Q a λ_Q zvolených tak, aby se zobrazovací plocha co nejlépe přimykala³ zobrazovanému území. Kartografická šířka je úhel v nové kartografické síti, který je analogický zeměpisné šířce. Kartografická délka je úhel v nově vytvořené kartografické síti, který odpovídá zeměpisné délce.[27]



Obrázek 2.5: Kartografické souřadnice.
Zdroj: [27].

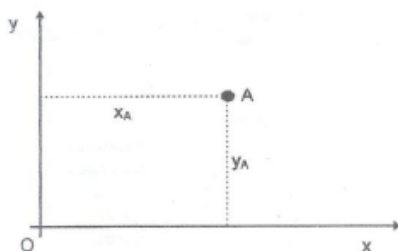
³Přimknutí ovlivňuje věrnost mapového obrazu referenční plochy. Více informací možno nalézt v [23].

2.3.4 Pravoúhlé rovinné souřadnice

Pravoúhlé rovinné souřadnice se zavádí pro určení polohy bodu v rovině. Udávají se kolmé vzdálenosti od souřadných os x a y .

Tyto souřadnice jsou definovány počátkem O , osami x a y . Počátek O je bod, který nejčastěji leží v průsečíku obrazů rovníku a základního poledníku, osa x leží v obraze rovníku a osa y v obraze základního poledníku.

Pravoúhlé souřadnice se používají při válcových a obecných kartografických zobrazeních. Počátek pravoúhlé rovinné souřadnicové soustavy se umísťuje do středu mapy.[27]



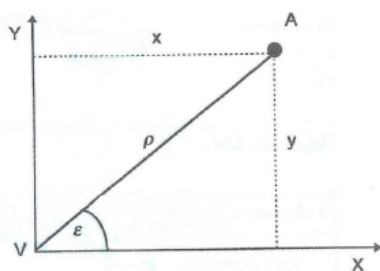
Obrázek 2.6: Pravoúhlé rovinné souřadnice.

Zdroj: [27].

2.3.5 Rovinné polární souřadnice

Rovinné polární souřadnice se vztahují k zeměpisnému, resp. kartografickému pólu a osám. Tyto souřadnice jsou definovány počátkem V a osou y . Počátek V je zvolený bod, obecně konstrukční pól, nejčastěji bod dotyku referenční a zobrazovací plochy nebo obraz zemského pólu. Osa y leží v obraze základního poledníku.

Souřadnice jsou v tomto případě průvodič bodu od počátku ρ a úhlem průvodiče η . Polární souřadnice se používají při azimutálních a kuželových zobrazeních.[27]



Obrázek 2.7: Rovinné polární souřadnice.

Zdroj: [27].

Kapitola 3

Definice souřadnicových systémů pro účely GIS

V této kapitole budou definovány pojmy důležité pro využívání souřadnicových systémů v GIS.

Souřadnicový referenční systém (SRS) - souřadnicový systém spojený se Zemí pomocí geodetického datumu.

Geodetické datum - parametr nebo soubor parametrů definující počátek, měřítko a orientaci SRS.

Konverze souřadnic - tímto pojmem se rozumí operace, ve které nedochází ke změně *datumu*. Vztah mezi souřadnicovými systémy je jednoznačně definován. Příkladem konverze je kartografické zobrazení nebo přepočítání ze zeměpisných souřadnic na geodetické.[24]

Transformace souřadnic - změna souřadnic z jednoho souřadnicového referenčního systému do jiného souřadnicového referenčního systému založeného na odlišném *datumu*. Transformace souřadnic používá parametry (transformační klíč), které jsou empiricky odvozeny z množiny bodů o souřadnicích známých v obou souřadnicových referenčních systémech.[24]

3.1 Formát pro definování SRS

3.1.1 WKT - Well Known Text

WKT je značkovací jazyk pro popis geometrie vektorových prvků a také pro definici referenčního souřadnicového systému, použitého kartografického zobrazení a případně i transformace. Binární obdobou je formát Well-known binary (WKB).[24]

3.1.2 Formát Proj4

Formát Proj4 je způsob zadávání vlastních parametrů do patrně nejrozšířenější open source knihovny pro práci se SRS a kartografickými zobrazeními. Vzhledem k vysoké rozšířenosti se tento formát stal de facto standardem pro definování SRS v GIS produktech.[24]

3.2 Databáze parametrů EPSG

V roce 1985 European Petroleum Survey Group (EPSG) vytvořila databázi geodetických parametrů určenou pro definici referenčních souřadnicových systémů a jejich převodů.

V roce 2005 se EPSG změnila na Surveying and Positioning Committee spadající pod International Association of Oil and Gas Producers (OGP). Dnes se používá označení EPSG databáze.

EPSG databáze je určena především pro: [\[18\]](#)

- Jednoznačné definování souřadnicových referenčních systémů.
- Definování transformací a konverzí z jednoho souřadnicového referenčního systému do druhého.

Databáze EPSG je distribuována jako sada SQL (Structured Query Language) příkazů pro relační databáze anebo jako databáze ve formátu MS Access.

Kapitola 4

Souřadnicové systémy využívané v České republice

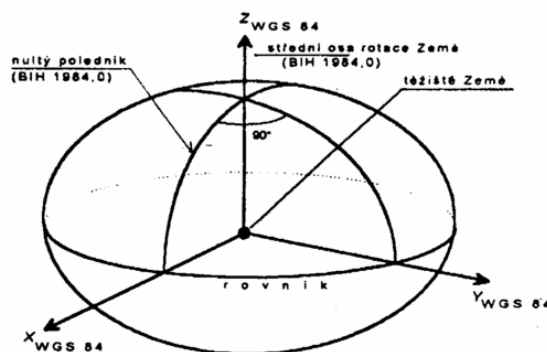
Dle [22] jsou v České republice nejvíce používané souřadnicové systémy WGS 84, S-JTSK, WGS 84/UTM zone 33N a WGS 84/Pseudo-Mercator. Vzhledem k tomu je tato práce zaměřena právě na tyto souřadnicové systémy. V této kapitole jsou uvedeny základní informace o těchto souřadnicových systémech.

4.1 World Geodetic System 1984 (WGS84)

World Geodetic System 1984 (WGS84) je geodetický geocentrický systém používaný globálním systémem určování polohy GPS a zároveň je standardizovaným geodetickým systémem armád NATO. Od 1. 1. 1998 je WGS84 zaveden ve vojenském a civilním letectvu a v Armádě České republiky.

WGS84 používá zeměpisné souřadnice, jednotlivé body jsou definované zeměpisnou délkou, šířkou a výškou. Systém má počátek v hmotném středu Země. Referenční plochou je elipsoid WGS84.[4]

EPSG kód tohoto systému je 4326.

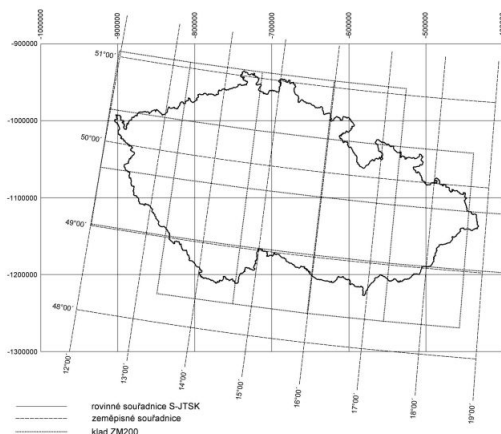


Obrázek 4.1: Světový geodetický systém 1984.

Zdroj: [20].

4.2 Systém Jednotné trigonometrické sítě katastrální (S-JTSK)

Souřadnicový systém Jednotné trigonometrické sítě katastrální (S-JTSK) je definován Besselovým elipsoidem. Jako referenční bod byl zvolen Hermannskogel. Pro jeho zobrazení do rovinných souřadnic bylo použito Křovákovo dvojí konformní kuželové zobrazení v obecné poloze.



Obrázek 4.2: Zobrazení ČR v souřadnicovém systému S-JTSK
Zdroj: Tvorba map pro OB¹.

S-JTSK se využívá v státních mapových dílech určených pro státní správu a samosprávu v České republice [1]. Do roku 2011 bylo zákonem závazné i pro Slovensko.[21]
EPSG kód tohoto systému je 5514.

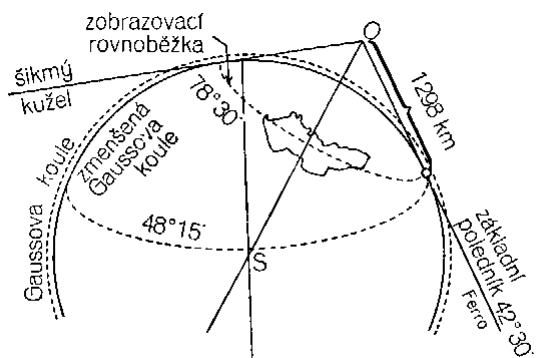
4.2.1 Křovákovo zobrazení

Křovákovo zobrazení je konformní kuželové zobrazení v obecné poloze. Bylo v roce 1922 navrženo Ing. Josefem Křovákem tak, aby co nejlépe vystihovalo tvar Československé republiky (tehdy včetně Podkarpatské Rusi) a poskytovalo co nejmenší délkové zkreslení na celém území státu.

Prvním krokem Křovákova zobrazení je zobrazení výchozí referenční plochy (Besselův elipsoid) na referenční kouli. Na referenční kouli jsou definovány kartografické souřadnice, pomocí kterých je povrch koule transformován do zobrazovací roviny konformním kuželovým zobrazením. Poslední fází je transformace z polárních rovinných souřadnic na pravoúhlé.[26]

Osa X je tvořena obrazem základního poledníku ($\lambda = 42^{\circ}30'$ východně od Ferra) a má kladný směr orientovaný k jihu. Osa Y je kolmá k ose X a směřuje na západ. Ve výsledku to znamená, že se celé území republiky dostává do 1. kvadrantu a všechny souřadnice jsou kladné. Pro libovolný bod na území bývalého ČSR platí $Y < X$. [20]

¹<http://tvorbamap.shocart.cz>



Obrázek 4.3: Křovákovo zobrazení.
Zdroj: [20].

4.3 World Geodetic System 1984 UTM

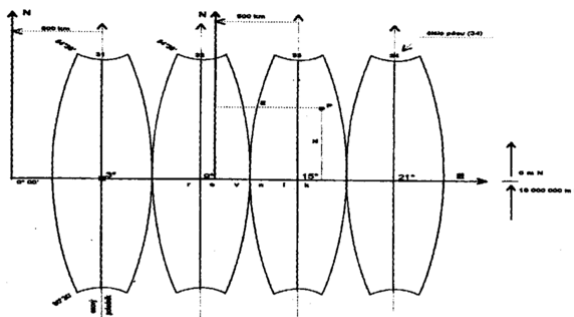
Pod tímto názvem se rozumí geodetický systém World Geodetic System 1984, který používá kartografické zobrazení UTM (Univerzální transversální Mercatorovo). Kartografické zobrazení UTM se používá kvůli svému přesnějšímu zobrazení a menšímu zkreslení.

Zobrazení UTM je příčné konformní válcové Mercatorovo zobrazení poledníkových pásů. Každý pás má vlastní souřadnicovou soustavu tvořenou osou N , která je vložena do obrazu osového poledníku, a osou E , které je vložena do obrazu rovníku. Každý pás je šestistupňový a pro určení polohy bodu je nutné doplnit, ve kterém z těchto pásů se bod nachází. K přesné lokalizaci polohy bodu se používá hlásná síť UTM.[20]

Souřadnicový systém UTM pokrývá povrch Země mezi 80° jižní šířky a 84° severní šířky. Na území ČR se nacházejí dva pásy WGS84/UTM, poledníkové pásy UTM 33N a 34N.

WGS84/UTM se používá pro topografické mapování velkých a středních měřítek.

EPSG kód tohoto systém v zóně 33N je 32633.



Obrázek 4.4: Zobrazení UTM.
Zdroj: [20].

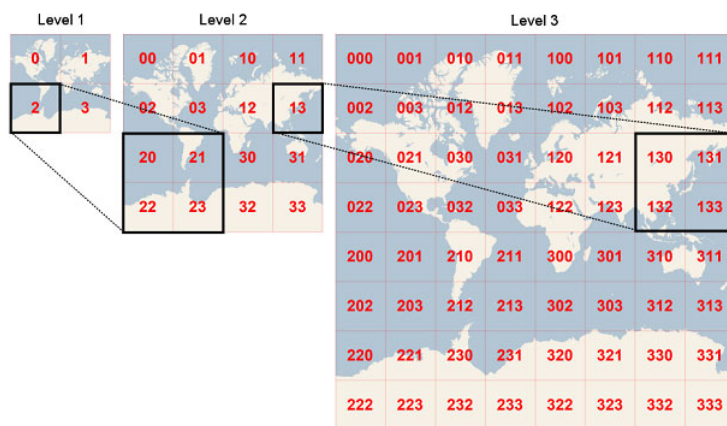
4.4 World Geodetic System 1984 Pseudo-Mercator

Jedná se opět o variantu World Geodetic System 1984 tentokrát používající Mercatorovu projekci. Velkou výhodou tohoto zobrazení je jeho schopnost přibližování a oddalování pomocí dlaždicového schématu. Při každé úrovni přiblížení se mapa rozdělí na čtyřnásobný počet dlaždic (pyramida obrázků pro více rozlišení).

Pro použití je WGS84/Pseudo-Mercator (někdy také označovaný jako Web Mercator) vhodný pro povrch Země mezi 85° jižní šířky a 85° severní šířky.

Využití tento souřadnicový systém nachází v centrálním webovém mapování a vizuálních aplikacích. Je použit například v Google Maps, Bing Maps, Mapquest a OpenStreetMap.

EPSG kód tohoto systému je 3857.



Obrázek 4.5: Dlaždicové schéma.

Zdroj: Microsoft Developer Network článek Bing Maps Tile System².

²<https://msdn.microsoft.com/en-us/library/bb259689.aspx>

Kapitola 5

Konverze mapových souřadnic

Tato kapitola obsahuje teorii ohledně transformačních metod (viz 5.1) a stěžejní část této práce - nalezení vhodných nástrojů pro konverzi souřadnic (viz 5.2).

5.1 Transformační metody

5.1.1 Podobnostní transformace

Jedná se o metody, které jsou matematicky nejvýstižnějším vyjádřením mezi jednotlivými systémy a jasně vyjadřují rozdíly mezi nimi.[24]

Moloděnského metoda umožňuje přímou transformaci zeměpisných souřadnic definovaných v souřadnicových systémech bez toho, aniž by byl nutný jejich převod do prostorových pravoúhlých souřadnic.[26]

K této transformaci je nutná znalost parametrů původního elipsoidu velikost poloos a , b , lineárních posunů dx , dy a dz a rozdílů parametrů použitých referenčních elipsoidů (původního a nového) – velké poloosy Δa a zploštění Δf . [26]

Moloděnského metoda se používá především tam, kde je důležitá rychlost přepočtu.[24]

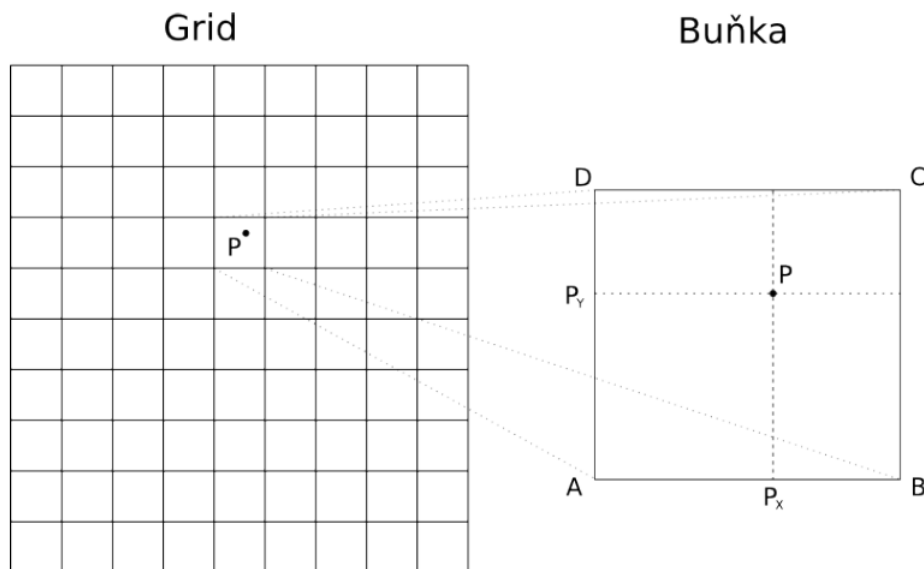
Metoda Burša-Wolf je jedním z nejužívanějších algoritmů, jež slouží pro převod souřadnicových referenčních systémů. Někdy je v GIS také nazývána Position Vector, Coordinate Frame Rotation nebo nejčastěji Burša-Wolf. V geodézii se často označuje jako Helmertova sedmiprvková 3D transformace.[24]

Při použití této transformace je nutné nejprve rovinné souřadnice X , Y převést pomocí kartografického zobrazení na geografické ϕ , λ a ty následně na geocentrické X , Y , Z . Metoda využívá těchto 7 parametrů: [24]

- ΔX - posun počátku cílové souřadnicové soustavy ve směru osy X
- ΔY - posun počátku cílové souřadnicové soustavy ve směru osy Y
- ΔZ - posun počátku cílové souřadnicové soustavy ve směru osy Z
- α - rotace cílové soustavy okolo osy X
- β - rotace cílové soustavy okolo osy Y
- γ - rotace cílové soustavy okolo osy Z
- q - měřítko (též zkreslení)

5.1.2 Metody na bázi gridu

Transformace pomocí gridu jsou transformační metody, které lze využít především v případě, kdy nelze definovat matematicky přesný vztah mezi výchozím a cílovým systémem. Metoda je založena na principu známých (resp. odhadnutelných) hodnot posunů v dostatečně husté a pravidelné síti (gridu). Pro konkrétní bod se vypočte hodnota posunu z příslušné mřížky na základě známých vektorů posunu ve vrcholech buňky gridu. Po určení buňky gridu, ve které se bod určený k transformaci nachází, se získá hodnota posunu ve směru osy X a Y díky metodě zvané lineární interpolace.[24]



Obrázek 5.1: Transformace pomocí gridu.

Zdroj: [24].

Metody na tomto principu jsou poměrně rozšířené, ale samotné konfigurační soubory jsou k dispozici pouze pro některé SRS.[24]

Mezi metody pracující na tomto principu jsou v GIS používány tyto:

Metoda NADCON (North American Datum Conversion) je metoda používaná především v USA. Metoda využívá dvou transformačních gridů. Jedná se o dva soubory, jeden s příponou *.laa* (soubor obsahuje posuny v zeměpisné šířce) a druhý *.loa* (soubor obsahuje posuny v zeměpisné délce).[5]

Metoda NTV2 (The National Transformation version 2) je velice podobná metodě NADCON. I ona využívá binárních souborů obsahujících posuny v zeměpisné šířce a zeměpisné délce.

Rozdílem metody NTV2 a NADCON je to, že metoda NTV2 umožňuje používat i více gridů najednou. Tímto je možno zvýšit přesnost.[5]

5.2 Existující nástroje

V této podkapitole jsou uvedeny nástroje, které odpovídají požadavkům na podporované souřadnicové referenční systémy a jsou volně dostupné.

5.2.1 Proj.4

Proj.4 je knihovna napsaná v jazyce C určená pro práci s kartografickými zobrazeními. Tato knihovna vznikla na počátku devadesátých let a jejím autorem byl Gerald Evenden pod záštitou USGS¹, posléze byl vývoj převzat Frankem Warmerdamem a OSGeo².

Knihovna je v současné verzi pod X/MIT licencí.

Domovskou stránkou projektu je <http://trac.osgeo.org/proj/>.

Nyní budou uvedeny možné varianty knihovny Proj.4, jedná se buď o přepis do jiného programovacího jazyka nebo pouze o využití této knihovny.

Proj4js je JavaScriptová knihovna, která vznikla jako přepis knihovny Proj.4.

Licence, pod kterou je tato knihovna vydávána je X/MIT.

Domovskou stránkou projektu je <http://proj4js.org/>.

Java Map Projections Library (JMPL) je knihovna napsaná v jazyce Java a je určena pro práci s kartografickými zobrazeními. Představuje částečný port do knihovny Proj.4. Jedná se o čistý přepis do jazyka Java a doposud jsou implementována nejpoužívanější zobrazení. Každé zobrazení je implementováno jako samostatná třída.

Knihovna je vydávána pod Apache licencí.

Domovská stránka projektu je <http://www.jhlab.com/>.

PostGIS je rozšíření pro objektově-relační databázový systém PostgreSQL umožňující uložení geoprostorových objektů. Pro konverzi mezi referenčními souřadnicovými systémy je použita Proj.4. PostGIS je napsán v jazyce C/Java.

Licence, pod kterou je PostGIS vydáván, je GNU GPL.

Domovskou stránkou je <http://postgis.net/>.

OpenLayers je JavaScriptová knihovna určená k zobrazování map. Tato knihovna je použita například pro zobrazování map projektu OpenStreetMap.

OpenLayers bez jakýchkoliv přídatných knihoven podporuje konverzi mezi WGS84 a WGS84/Pseudo-Mercator. Po přidání knihovny Proj4js umožňuje transformaci mezi referenčními souřadnicovými systémy podporovanými touto knihovnou.

Licence, pod kterou je tato knihovna vydávána, je FreeBSD.

Domovská stránka je <http://openlayers.org/>.

¹<http://www.usgs.gov/>

²<http://www.osgeo.org/>

Google Maps API je API vytvořené firmou Google, které je využito v Google Maps. Pro transformaci používá různých podpůrných knihoven (např. Proj4js).

Toto API má omezený počet přístupů z jedné IP adresy za den. Pokud aplikace překonává 25 000 mapových načtení za den po více než 90 dní, musí být zaplacená dodatečná vyšší licence.

Celé API je vydáváno pod Apache Licencí.

Domovskou stránkou API je <https://developers.google.com/maps/>.

Geospatial Data Abstraction Library (GDAL) je knihovna napsaná v jazyce C++ pro čtení a zápis různých GIS rastrových formátů. Pro transformaci interně používá knihovnu Proj.4.

Knihovna je vydávána pod licencí X/MIT.

Domovskou stránkou projektu je <http://gdal.org/>.

SpatiaLite je prostorové rozšíření pro databázi SQLite napsaná v jazyce C/C++, umožňující uložení geoprostorových dat.

SpatiaLite umožňuje vnitřně používat knihovnu GDAL. Vzhledem k tomuto je pro transformaci možno použít knihovnu Proj.4.

SpatialLite je dostupná pod licencí Mozilla Public License.

Domovská stránka projektu je <http://www.gaia-gis.it/gaia-sins/>.

DotSpatial je knihovna napsaná pro .NET 4, která umožňuje vývojářům pracovat s prostorovými daty a používat mapové funkce.

Konverze souřadnic je v této knihovně řešena použitím knihovny Proj.4, která je přepsána do C#.

Knihovna je dostupná pod licencí LGPL.

Domovská stránka je <https://dotspatial.codeplex.com/>.

5.2.2 GeoTools

GeoTools je Java GIS toolkit, který implementuje řadu OGC specifikací. Tento toolkit vznikl v roce 1996 na univerzitě v Leedsu (GeoTools 1) a později (2002) byl přepsán mezinárodní skupinou vývojářů (GeoTools 2). Od roku 2007 se tato verze označuje jako „GeoTools“.

Transformace souřadnic je implementována jako OGC Coordinate Transformation Service³. Jedná se o projekt OSGeo⁴.

Projekt je pod Licence GNU Lesser General Public License (LGPL).

Domovskou stránkou projektu je <http://www.geotools.org/>.

³<http://www.opengeospatial.org/standards/ct>

⁴<http://www.osgeo.org/>

5.2.3 Mapy API

Mapy API jsou součástí služeb, které poskytuje největší český portál *Seznam.cz*. API je celé napsáno v programovacím jazyce JavaScript. Obsahuje funkce pro transformaci mezi souřadnicovými systémy S-JTSK, UTM, Pseudo-Mercator a WGS84.

Použití je zcela zdarma a to i pro komerční účely. Jedinou podmínkou pro použití je nutnost zobrazování loga Mapy.cz v nezměněné podobě a texty copyrightů dodavatelů mapových podkladů.

Domovská stránka projektu je <https://api.mapy.cz/view?page=instruction>.

Kapitola 6

Vybrané způsoby řešení

Tato kapitola je rozdělena do čtyř podkapitol. Podkapitola 6.1 obsahuje možnosti členění nalezených nástrojů na konverzi souřadnic podle podobných znaků, které by se daly využít pro porovnávání v další části práce. V části 6.2 je uvedeno rozdělení podle možnosti nastavení přesnosti. Nalezené nástroje z předchozí kapitoly a možnosti členění shrnuje podkapitola 6.3. Poslední část 6.4 obsahuje popis instalace, použití a případně také klady nebo zápory jednotlivých řešení.

6.1 Způsob členění

6.1.1 Podle použitého nástroje

Prvním možným způsobem, jak členit nalezená existující řešení, je podle použité knihovny pro konverzi souřadnic. Z toho dělení vyplývají tři možné skupiny - řešení založená na knihovně Proj.4, na knihovně GeoTools a řešení založená na dalších knihovnách.

Jako zástupce skupiny založené na dalších knihovnách bylo vybráno pouze Maps API, jelikož jako jediná z nalezených podporuje všechny požadované referenční souřadnicové systémy.

6.1.2 Podle typu a místa užití nástroje

Dalším možným členěním pro další práci je členění podle typu a místa užití nástroje.

Nalezená řešení se takto dají členit podle skupin použitelných na straně klienta (typickým zástupcem je například JavaScriptový přepis knihovny Proj.4 – Proj4js) či na straně serveru (například PostGIS nebo GeoTools).

Podle typu se dají nástroje rozčlenit na knihovnu, API, SQL databáze a SQLite databáze.

6.1.3 Podle jazykové verze knihovny

Posledním zde uvedeným členěním je podle použitého programovacího jazyka.

Tímto členěním lze spojit OpenLayers, Google Maps API a Proj4js, jelikož první dvě využívají třetího jmenovaného (knihovnu Proj4js).

6.2 Možnost nastavení přesnosti

6.2.1 Podpora transformačních metod

Pro možnost nastavení přesnosti je důležitá podpora transformačních metod.

Knihovna Proj.4 a nástroje od ní odvozené podporují transformační metody:

- Moloděnského metoda
- metoda Burša-Wolf
- metody na bázi gridu (NADCON, NTv2)

Knihovna GeoTools využívá pro transformaci mezi SRS metody:

- Moloděnského metoda
- metoda Burša-Wolf
- metody na bázi gridu (NADCON)

Mapy API v současné verzi nepodporují žádné nastavení přesnosti.

6.2.2 Transformační klíč

Požadovanou přesnost je možno v knihovnách založených na Proj.4 a GeoTools ovlivnit vhodně zvoleným transformačním klíčem. Velice často používaným transformačním klíčem je sedmiprvkový pro Helmertovu transformaci.


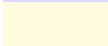



Globální transformační klíč jsou parametry určené pro celé území ČR, resp. SR. V případě menšího zájmového území je možné vypočítat lokální transformační klíč, nebo rovnou použít metodu používající grid a případná chyba se tak až řádově zmenší.[\[21\]](#)

Na stránkách zdroje [\[21\]](#) je možno nalézt přesnost. V případě nepoužití transformačního klíče dochází ke zhoršení přesnosti o dva řády (přibližně <100 metrů do S-JTSK, 3 vteřiny do WGS84 v horizontálních souřadnicích, výška není převedena vůbec). Pro oficiální klíč ČÚZK pro ČR je uváděna maximální odchylka v poloze $_{x,y}$ <1m a výšce $_z$ <2m. V případě použití gridu nacházejícího se rovněž na těchto stránkách je možno docílit uváděné přesnosti maximálně 17 cm.

6.3 Souhrn nástrojů z předchozí kapitoly po členění

Název	Jazyk	Typ	Domovská stránka
Proj.4	C	knihovna	http://trac.osgeo.org/proj/
Proj4js	JavaScript	knihovna	http://proj4js.org/
OpenLayers	JavaScript	knihovna	http://openlayers.org/
Google Maps API	JavaScript	API	https://developers.google.com/maps/
JMPL	Java	knihovna	http://www.jhlab.com/
GDAL	C++	knihovna	http://gdal.org/
DotSpatial	C#	knihovna	http://dotspatial.codeplex.com/
PostGIS	C/Java	SQL databáze	http://postgis.net/
SpatiaLite	C/C++	SQLite databáze	http://www.gaia-gis.it/gaia-sins/
GeoTools	Java	knihovna	http://www.geotools.org/
Mapy API	JavaScript	API	http://api.mapy.cz/

Tabulka 6.1: Souhrn všech nalezených řešení pro konverzi souřadnic.

Barva	Význam
	Konverze souřadnic založena na knihovně Proj.4
	Konverze souřadnic založena na knihovně GeoTools
	Nástroje využívající další knihovny.
	Nástroje založené na Proj4js.
	Nástroje použitelné pouze na serveru.

Tabulka 6.2: Legenda k tabulce 6.1.

6.4 Konkrétní způsoby řešení

Popis instalace je určen pro prostředí Ubuntu 14.04 64-bit.

6.4.1 Proj4

Zdrojové kódy a dokumentaci je možno nalézt na domovských stránkách knihovny [2]. Popis instalace knihovny je možno nalézt na stránce [7].

Po nainstalování knihovny je možno používat konzolový program *cs2cs*, který umožňuje převod souřadnic z jednoho systému na druhý.

Projekce mohou být do této knihovny zadávány pomocí parametrů řetězce ve formátu *Proj4*.

Proj4js

Návod na použití je možno nalézt na domovské stránce knihovny [3].

Vzhledem ke skutečnosti, že knihovna je napsaná v jazyce JavaScript, stačí ji pouze začlenit do HTML stránky. Žádná další instalace již není nutná.

Knihovnu je možno začlenit dvěma způsoby. Buď lokálně staženou kopií z repozitáře z webové služby GitHub¹, nebo je možné používat knihovnu bez jejího stažení pomocí hostingu na CDN (viz ukázka 6.1).

```
<script
src="http://cdnjs.cloudflare.com/ajax/libs/proj4js/2.2.1/proj4.js">
</script>
```

Ukázka kódu 6.1: Použití knihovny Proj4js pomocí hostingu CDN.

Projekce mohou být do této knihovny zadány pomocí řetězce *WKT* nebo ve formátu *Proj4*.

OpenLayers

Návod, jak knihovnu používat, i velké množství příkladů je možno nalézt na domovských stránkách projektu [10].

Jedná se opět o JavaScriptovou knihovnu a tudíž stačí pouze její začlenění do HTML stránky.

Knihovnu je možno stáhnout například z repozitáře z webové služby GitHub². Druhou možností použití knihovny je pomocí nasměrování programu na servery OpenLayers (viz ukázka 6.2).

```
<script src="http://openlayers.org/en/v3.3.0/build/ol.js"></script>
```

Ukázka kódu 6.2: Použití knihovny OpenLayers pomocí hostingu na serverech OpenLayers.

Po zahrnutí knihovny Proj4js je možno v OpenLayers používat funkce, které jsou popsány ve vývojové dokumentaci knihovny OpenLayers³.

¹<https://github.com/proj4js/proj4js/releases>

²<https://github.com/openlayers/ol3/releases>

³<http://openlayers.org/en/v3.1.0/apidoc/ol.proj.html>

Knihovna umožňuje přidat vlastní transformace.

Google Maps API

Jednoduchý tutoriál, jak toto API používat, a příklady použití je možné nalézt na domovských stránkách tohoto projektu [8].

Za zmínku stojí i tutoriál na stránce *w3school.com*⁴, který je velice užitečný.

API je možno používat pouze pomocí nasměrování programu na servery google (viz ukázka 6.3). API je psáno v jazyce JavaScript, a proto stačí pouze začlenit do HTML stránky.

```
<script src="http://maps.googleapis.com/maps/api/js?key=YOUR_KEY">
</script>
```

Ukázka kódu 6.3: Použití Google Maps API pomocí hostingu na stránkách Google [8].

Místo `YOUR_KEY` v ukázce se očekává klíč od Google⁵, který umožňuje vyšší množství přístupů.

API bez zahrnutí knihovny Proj4js nebo vlastní implementace nepodporuje konverze souřadnic.

Pro definování vlastní projekce obsahuje API rozhraní `google.maps.Projection`. Současně s definováním vlastní projekce je potřeba implementovat metodu pro převod mezi systémem WGS84 a nově vzniklým souřadnicovým systémem (metodu `fromLatLngToPoint` pro převod z WGS84 a metodu `fromPointToLatLng` pro převod do WGS84).

JMPL

Odkazy na stažení knihovny jsou dostupné na domovské stránce projektu [6]. Zdrojové kódy je možno získat také z repozitáře webové služby GitHub⁶.

Stažitelný archiv obsahuje zdrojové kódy a `jar` soubor pro sestavení.

Knihovna má velkou nevýhodu v tom, že je stále ještě ve fázi vývoje a nemá plně funkční implementaci konverze mezi geodetickými *daty*. Rovněž chybí implementace některých referenčních souřadnicových systémů.

I přes tyto nedostatky by se však mohlo v budoucnu jednat o zajímavého kandidáta.

GDAL

Zdrojové kódy a základní informace je možno nalézt na domovské stránce projektu [14]. Instalace knihovny GDAL je možno nalézt na stránce [7].

Součástí knihovny je celá řada konzolových programů, mimo jiné i například *gdaltransform*, který slouží k převodu mezi dvěma SRS.

Výhodou této knihovny je, že podporuje velké množství formátů na zadání projekcí, například *WKT*, *Proj4* nebo také soubory Esri *.PRJ*.

DotSpatial

Knihovnu lze stáhnout na domovských stránkách projektu [11].

DotSpatial v současné verzi není plně funkční a dále neběží pod jiným operačním systémem než Windows a proto nebude dále rozváděna.

⁴<http://www.w3schools.com/googleapi>

⁵Klíč je možno vygenerovat na stránce <https://code.google.com/apis/console/>.

⁶<https://github.com/OSUCartography/JMapProjLib>

PostGIS

Zdrojové kódy, velice rozsáhlou dokumentaci a příklady lze nalézt na domovské stránce projektu [15]. Instalace a spuštění PostgresSQL a následně také rozšíření PostGIS je možno nalézt například na stránce Jon Saint⁷.

Další krokem před samotnou transformací souřadnic pomocí PostGIS je potřeba definovat SRS vložením požadovaných parametrů do tabulky `spatial_ref_sys` (příklad vložení viz 6.4).

```
1 INSERT into spatial_ref_sys
2 (srid, auth_name, auth_srid, proj4text, srttext)
3 values
4 ( 5514, 'EPSG', 5514,
5 '+proj=krovak +lat_0=49.5 +lon_0=24.833333333333333
6 +alpha=30.288139722222222 +k=0.9999 +x_0=0
7 +y_0=0 +ellps=bessel
8 +towgs84=570.8,85.7,462.8,4.998,1.587,5.261,3.56 +units=m +no_defs ',
9 'PROJCS["S-JTSK / Krovak East North",
10   GEOGCS["S-JTSK",
11     DATUM["System_Jednotne_Trigonometricke_Site_Katastralni",
12       SPHEROID["Bessel 1841",6377397.155,299.1528128,
13         AUTHORITY["EPSG","7004"]],
14       TOWGS84[570.8,85.7,462.8,4.998,1.587,5.261,3.56],
15       AUTHORITY["EPSG","6156"]],
16     PRIMEM["Greenwich",0, AUTHORITY["EPSG","8901"]],
17     UNIT["degree",0.0174532925199433, AUTHORITY["EPSG","9122"]],
18     AUTHORITY["EPSG","4156"]],
19     PROJECTION["Krovak"],
20     PARAMETER["latitude_of_center",49.5],
21     PARAMETER["longitude_of_center",24.833333333333333],
22     PARAMETER["azimuth",30.288139722222222],
23     PARAMETER["pseudo_standard_parallel_1",78.5],
24     PARAMETER["scale_factor",0.9999],
25     PARAMETER["false_easting",0],
26     PARAMETER["false_northing",0],
27     UNIT["metre",1, AUTHORITY["EPSG","9001"]],
28     AXIS["X",EAST], AXIS["Y",NORTH],
29     AUTHORITY["EPSG","5514"]]' );
```

Ukázka kódu 6.4: Vložení SRS S-JTSK do tabulky `spatial_ref_sys`. Zdroj: [13].

Pro transformace se v PostGIS používá `ST_Transform`, tento příkaz lze použít během získávání, modifikace i zapisování dat v databázi.

Před samotným použitím nebo v případě nefungující transformace PostGIS je dobré použít příkazu `PostGIS_Full_Version` pro zjištění, zda se knihovna Proj.4 správně přeložila a je možné ji v databázi použít.

Spatialite

Zdrojové kódy je možno nalézt na domovské stránce projektu [16]. Instalace Spatialite je popsána například na stránce [7].

Stejně jako v případě PostGIS je před samotnou transformací souřadnic potřeba definovat SRS vložením požadovaných parametrů do tabulky `spatial_ref_sys`. Rozdílem oproti vložení do PostGIS je, že Spatialite nevyžaduje vložení poslední hodnoty, kterým je definice referenčního sou-

⁷<http://www.saintsjd.com/2014/08/13/howto-install-postgis-on-ubuntu-trusty.html>

řadnicového systému pomocí WKT (v případě PostGIS se jedná o `srttext` a v případě SpatiaLite o `srs_wkt`).

Spatialite, stejně jako PostGIS, obsahuje příkaz `ST.Transform`, který lze použít pro transformaci během získávání, modifikaci i zapisování dat v databázi.

6.4.2 GeoTools

Na domovských stránkách projektu [12] se nachází tutoriály pro instalaci a základní použití této knihovny.

Velký rozdíl této knihovny oproti ostatním zde uvedeným nástrojům je její rozsáhlost. Rozsah knihovny GeoTools je evidentní i v složitosti spuštění, které není možné bez nástroje *maven* a skládá se z několika kroků.

Referenční souřadnicové systémy můžou být v této knihovně specifikovány pomocí řetězce ve formátu *WKT*, který je většinou uložen v souboru `.prj` (někdy se používá označení *shapefile*) nebo v databázi PostGIS či Oracle. Jako alternativní způsob pro specifikaci SRS je použití *EPSG* kódu daného systému.

6.4.3 Mapy API

Návod na spuštění s řadou příkladů je možno nalézt na domovských stránkách API [9].

API je napsáno v programovacím jazyce JavaScript a lze tedy začít využívat pouhým začleněním do HTML stránky.

API je možno používat pouze pomocí připojení k serverům společnosti *Seznam.cz* (viz ukázka 6.5).

```
1 <script src="http://api4.mapy.cz/loader.js"></script>
2 <script>Loader.load()</script>
```

Ukázka kódu 6.5: Připojení Mapy API do uživatelské aplikace. Zdroj: [9].

Nestandardní výhodou tohoto API je rychlá komunikace s vývojáři, kteří jsou ochotní přidat požadovanou funčnost.

Nevýhodou tohoto API je absence možnosti nastavení přesnosti pro transformaci, omezený počet podporovaných souřadnicových systémů a dále to, že v současné verzi nepodporuje konverzi jednotlivých souřadnic do a z SRS WGS84/Pseudo-Mercator.

6.4.4 Vzdálené používání knihoven a API

U nástrojů napsaných v programovacím jazyce JavaScript⁸ se lze v předchozí kapitole setkat s možností jejich použití na vzdálených serverech.

Dle článku [28] je vzdálené použití knihoven a API výhodné v tom, že může snížit latency, zvýšit paralelizmus a zlepšit využití mezipaměti (cache). Snížení latence je způsobeno tím, že uživatel umožňuje používat servery v jeho blízkosti, které mohou být uživateli fyzicky blíže než server s běžící aplikací. Ke zvýšení paralelizmu může docházet v důsledku toho, že server s aplikací dostává o jeden dotaz méně a zasílá tedy o jednu odpověď méně. Zlepšení využití mezipaměti může nastat, pokud už jednou uživatel knihovnu použil a celá knihovna je stále ve vyrovnávací paměti. To vede k tomu, že knihovna nemusí být opakovaně stahována.

Nevýhodou tohoto způsobu je nemožnost použití bez přístupu k internetu, potenciální výpadky serverů a závislost na třetí straně.

⁸Konkrétně nástrojů Proj4js, OpenLayers, Google Maps API a Mapy API.

Kapitola 7

Testování

Následující kapitola je věnována samotnému testování nástrojů. Nejvíce jsou testy zaměřeny na výkonnost (počet přetransformovaných bodů za jednotku času).

Nejprve je provedeno srovnání nástrojů založených na podobném principu a posléze celkové srovnání. Podkapitola 7.2 obsahuje porovnání nástrojů Proj.4, GDAL a GeoTools. V části 7.3 jsou porovnávány nástroje používající knihovnu Proj4js a Mapy API. Podkapitola 7.4 obsahuje porovnání databázových nástrojů PostGIS a SpatiaLite. V podkapitole 7.5 jsou srovnány všechny nástroje jako celek.

7.1 Způsob testování rychlosti převodu

Testování rychlosti převodu souřadnic probíhalo na náhodně vygenerovaných souřadnicích v rozsahu daného souřadnicového systému.

Každý nástroj byl spouštěn na převod 1 000, 10 000, 100 000 a 1 000 000 souřadnic. V jednom běhu programu byl daný počet bodů převeden vždy patnáctkrát, tento počet se ukázal jako vhodný pro ustálení počáteční odchylky. Nástroj byl vždy spouštěn dvakrát, aby se předešlo případným odchylkám způsobeným vedlejšími vlivy.

U každého testu byl brán medián, maximální odchylka mezi mediánem a nejrychlejší nebo nejpomalejší dobou převodu.

Testy probíhaly na konverzi souřadnic z **WGS84 do S-JTSK**.

Prvním důvodem tohoto rozhodnutí byla skutečnost, že souřadnice v WGS84 je možné zadávat do každého z nástrojů ve stejném formátu. Druhým důvodem byl, že tato transformace měla po testech v nástrojích Proj.4, GeoTools, Proj4js a SpatiaLite nejvyšší časovou náročnost a tedy nejlépe dokázala ukázat potenciální rozdíly mezi nástroji bez nutnosti zvýšení počtu vygenerovaných bodů.

Pokud bylo možno nastavit transformační klíč, byl použit oficiální klíč ČÚZK pro ČR (viz ukázka 7.1). Udávaná maximální odchylka tohoto klíče je do 1 metru v poloze_{xy} [21].

```
+towgs84=570.8,85.7,462.8,4.998,1.587,5.261,3.56
```

Ukázka kódu 7.1: Oficiální klíč ČÚZK pro ČR. Zdroj: [21].

7.2 Nástroje v jazyce C/C++ a jazyce Java

V této sekci budou testovány nástroje Proj.4 ve verzi 4.8.0, GeoTools verze 14 a GDAL verze 1.10.1.

Doba převodu je měřena rozdílem časů před a po zavolání funkce na transformaci souřadnic. Pro nástroje v jazyce C/C++ je použita funkce `clock()` ze standardní knihovny `time`. U nástroje v jazyce Java je použita funkce `nanoTime()` objektu `System`.

Před testováním je důležité dodat, že GeoTools verze 14 pro S-JTSK používá EPSG kód 5513. Toto EPSG obsahuje jiný transformační klíč než který je napsán výše, ale jeho přesnost je dle [17] stejná (tedy maximálně 1 metr v poloze_{xy}) a tudíž neovlivňuje výsledky testování.

Počet souřadnic	Doba převodu [ms]					
	GeoTools		Proj.4		GDAL	
	Medián	Max. σ	Medián	Max. σ	Medián	Max. σ
1 000	6,0973	4,8506	1,9900	1,0812	1,4700	1,2810
10 000	34,3457	33,7140	14,6040	5,8340	15,1325	1,9005
100 000	218,1181	143,6606	142,0900	7,5050	143,1655	4,1315
1 000 000	2120,5341	476,0084	1396,9425	32,8655	1408,2205	8,2545

Tabulka 7.1: Výsledky porovnání rychlosti transformace nástrojů GeoTools, Proj.4 a GDAL.

Poznámka: Symbol σ je použit místo slova odchylka pro přehlednější zobrazení tabulky.

7.3 Nástroje v jazyce JavaScript

Nástroje napsané v jazyce JavaScript, jak již bylo zmíněno výše, jsou Mapy API (verze 4.8), Proj4js (verze 2.3.3), OpenLayers (verze 3.4.0) a Google Maps API v3. Vzhledem k tomu, že poslední jmenovaný nástroj používá pro transformace a konverze knihovnu Proj4js a její funkčnost nijak nerozšiřuje, bude z testů vynechán.

Aby byly podmínky pro testování rychlosti transformace souřadnic v této skupině nástrojů stejné, byla vytvořena jednotná HTML stránka s JavaScriptovým kódem pro generování souřadnic. Dle aktuálně testovaného nástroje bylo třeba pouze změnit funkci na převod souřadnic.

Pro měření doby převodu je použita metoda `performance.now()`, která je použita před a po volání funkce na transformaci souřadnic. Následně je doba převodu určena jako rozdíl těchto časů.

Jak již bylo uvedeno v předchozí kapitole, Mapy API nepodporuje nastavení přesnosti, avšak pro transformaci mezi WGS84 a S-JTSK používá téměř identický transformační klíč (s drobnou změnou v desetinné části čísla) jako ten uvedený na začátku této sekce a tedy rozdíl přesnosti API a ostatních nástrojů je zanedbatelný.

Mozilla Firefox 37						
Počet souřadnic	Doba převodu [ms]					
	Mapy API		OpenLayers		Proj4js	
	Medián	Max. σ	Medián	Max. σ	Medián	Max. σ
1 000	2,7094	10,3854	4,5698	13,0828	4,0985	15,6273
10 000	27,6047	8,14173	47,0521	28,1336	40,6219	44,6893
100 000	287,7559	64,62170	452,1842	53,6367	420,4243	45,5572
1 000 000	3037,7793	391,24138	4649,2848	222,7737	4096,0519	223,1063

Chromium verze 41						
Počet souřadnic	Doba převodu [ms]					
	Mapy API		OpenLayers		Proj4js	
	Medián	Max. σ	Medián	Max. σ	Medián	Max. σ
1 000	0,9495	14,1725	3,3950	30,3970	3,3820	20,2190
10 000	10,5985	10,8365	27,9730	39,9060	22,4620	33,3660
100 000	122,9170	20,9370	357,4425	75,8645	329,1770	58,8270
1 000 000	1119,4820	1721,4310	3726,7545	3405,8145	2880,5055	347,3135

Internet Explorer 11						
Počet souřadnic	Doba převodu [ms]					
	Mapy API		OpenLayers		Proj4js	
	Medián	Max. σ	Medián	Max. σ	Medián	Max. σ
1 000	23,2890	7,8265	24,3236	15,2368	23,9525	15,3827
10 000	145,3532	22,5756	220,5854	46,1165	248,0463	20,7797
100 000	1447,9799	39,3607	2242,4473	403,2119	2563,8953	136,4431
1 000 000	15720,5458	449,3998	24114,4114	404,6638	27933,9049	700,3691

Tabulka 7.2: Výsledky porovnání rychlosti transformace nástrojů Mapy API, OpenLayers a Proj4js ve webovém prohlížeči Mozilla Firefox, Chromium a Internet Explorer.

Poznámka: Symbol σ je použit místo slova odchylka pro přehlednější zobrazení tabulky.

7.4 Databázové nástroje

Databázové nástroje zahrnují PostGIS (ve verzi 2.1.2) a SpatiaLite (verze 4.1.1). Před samotným testováním rychlosti transformace bylo potřeba nějakým způsobem vytvořit a vložit do databázi náhodně vygenerované souřadnice. K tomuto účelu byl vytvořen skript v jazyce Python, který za pomoci knihoven *psycopy2* pro nástroj PostGIS a *pyspatialite* pro nástroj SpatiaLite umožňuje vkládat, modifikovat a prohlížet data v daném typu databáze.

Databáze SQLite, kterou SpatiaLite rozšiřuje, umožňuje maximálně 500 komponent příkazu INSERT [19], a proto jsou souřadnice při testech SpatiaLite vkládány právě po tomto počtu.

Samotné testování rychlosti transformace souřadnic probíhalo na měření časové náročnosti příkazu SELECT, který obsahoval příkaz na provedení transformace (ukázka viz 7.2).

```
select ST_AsText(ST_Transform(the_geom,5514)) from points;
```

Ukázka kódu 7.2: Získání souřadnic v S-JTSK z tabulky points.

Pro měření doby převodu byla použita funkce `time()` knihovny `time` před a po vykonání příkazu SELECT. Výsledná doba převodu byla určena jako rozdíl dvou takto získaných časů. Výsledky tohoto způsobu testování byly dále porovnány s výsledky měření rychlosti příkazu SELECT pomocí unixového příkazu `time`. Doby převodu získané pomocí obou měření byly téměř totožné.

Počet souřadnic	Doba převodu [ms]			
	PostGIS		SpatiaLite	
	Medián	Maximální odchylka	Medián	Maximální odchylka
1 000	12,387991	4,6401024	54,246068	6,855130
10 000	74,056506	5,3333044	464,338899	13,570189
100 000	668,316126	15,4857636	4658,497095	239,368916
1 000 000	6556,247592	481,7804098	46979,788420	1678,430438

Tabulka 7.3: Výsledky porovnání rychlosti transformace nástrojů PostGIS a SpatiaLite.

7.5 Celkové srovnání

U všech porovnávaných nástrojů lze po zakreslení hodnot do grafu (viz 7.1) vidět lineární růst doby převodu v závislosti na počtu souřadnic.

Překvapivě nejrychlejším nástrojem jsou Mapy API spouštěné ve webovém prohlížeči Chromium, ale vzhledem k tomu, že jsou závislé na prohlížeči (v případě prohlížeče Mozilla Firefox jsou na pomyslném 3. místě, pro větší množství souřadnic než 10 000 na 4. místě a v I až na 5. místě) je nejrychlejším nástrojem knihovna Proj.4 a také knihovna GDAL, která tuto knihovnu interně používá.

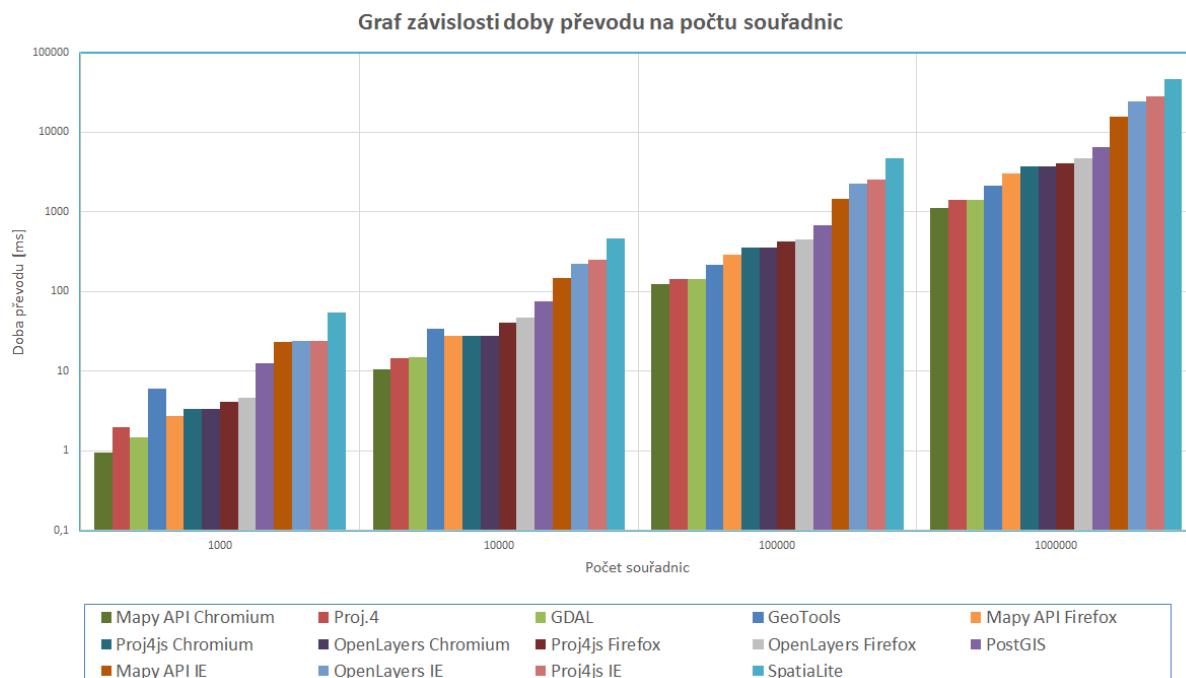
Knihovny Proj.4 a GDAL jsou nejrychlejší, bez ohledu na množství souřadnic. Tyto nástroje jsou velice vhodné, pokud je po cílové aplikaci požadován rychlý převod velkého množství souřadnic. Nevýhodou může být zatížení serveru, které může způsobit při vyšším počtu uživatelů používající webovou aplikaci přetížení serveru. GDAL má dále nevýhodu v tom, že se jedná o poměrně velký a komplexní nástroj a pokud již není používán v aplikaci, nemusí být jeho začlenění do aplikace jednoduché.

Knihovna GeoTools se při konverzi vyššího množství souřadnic umístila na 3. místě (oproti nejrychlejšímu času je pomalejší přibližně o 50%) a její nevýhodou je opět potenciální přetížení

serveru podobně jako u Proj.4 a GDAL. Další nevýhodou je podobně jako u knihovny GDAL, že se jedná o poměrně velký a komplexní nástroj a jeho dodatečné začlenění do aplikace může být obtížné.

Nástroje Proj4js, OpenLayers a Mapy API jsou závislé na rychlosti interpretace JavaScriptu webového prohlížeče. Nástroje se například po testech ve webovém prohlížeči Chromium umístily mezi prvními pěti, avšak výsledkem testů ve webovém prohlížeči Internet Explorer (IE) byly téměř nejpomalejší doby převodu. Pro všechny tyto nástroje platí, že jsou velice jednoduché na začlenění do webové aplikace a při jejich použití není zatížen server. Nevýhodou může být počáteční zpoždění způsobené stahováním, nebo v případě Mapy API používáním nástroje, a v případě použití na pomalejším zařízení (například mobilní telefony) může výrazně zatížit klientské zařízení.

Nejpomalejšími nástroji se ukázaly databázové nástroje SpatiaLite (ve všech testech jednoznačně nejpomalejší) a PostGIS (druhý nejpomalejší vyjma JavaScriptových nástrojů spouštěných v IE). Tyto nástroje jsou pomalé i pro menší množství souřadnic a pro aplikace, které tyto nástroje dosud nepoužívají velice náročné na zapojení. Nejvhodnější použití SpatiaLite a PostGIS pro konverzi souřadnic nabývají, když jsou souřadnice už do databáze vloženy a dále se s těmito souřadnicemi pracuje. V tomto případě je další výhodou, že není potřebná žádná dodatečná režie na získání bodů.



Obrázek 7.1: Graf závislosti doby převodu na počtu souřadnic.

Poznámka: Obě osy jsou v logaritmickeém měřítku (o základu 10).

7.6 Závislost přesnosti a výkonu

Během testů v předchozích částech této kapitoly byla pro transformaci z WGS84 do S-JTSK použita metoda Burša-Wolf. Přesnost této metody lze ovlivnit použitím vhodného transformačního klíče, ale vzhledem k tomu, že transformační klíč je pouze seznam parametrů, jeho změna neovlivňuje rychlost konverze.

Moloděnského metoda podporuje většina nástrojů. Tato metoda je sice rychlá, ale pro transformaci mezi WGS84 a S-JTSK se nepoužívá, jelikož je velice nepřesná.

Některé z nástrojů, konkrétně GeoTools a nástroje používající Proj.4, umožňují transformaci pomocí gridu. Transformace pomocí gridu dosahuje oproti metodě Burša-Wolf vyšší přesnosti a může být i rychlejší¹. Nevýhodou metody na bázi gridu může být samotný grid, který není dostupný pro všechny souřadnicové systémy a jeho vytvoření je značně složité.

¹Například u knihovny Proj.4 bylo změřeno, že metoda využívající grid je až o 12% rychlejší než metoda Burša-Wolf.

Kapitola 8

Demonstrační webová aplikace

Cílem této kapitoly je popsat demonstrační webovou aplikaci pro konverzi souřadnic, které vznikla jako součást této práce. Demonstrační webová aplikace má za úkol ukázat případným uživatelům rozdíly mezi vybranými nalezenými nástroji.

8.1 Popis aplikace

Pro účely této aplikace byly pro konverze souřadnic použity tyto nástroje:

- Všechny výše zmiňované nástroje v jazyce JavaScript umožňující konverzi souřadnic - Mapy API, Proj4js a OpenLayers.
- Nejrychlejší nástroj ze skupiny nástrojů napsaných v jazyce C/C++ nebo Java - Proj.4.
- Databázový nástroj s nejrychlejší konverzí souřadnic - PostGIS.

Zvolení nástroje PostGIS a Proj.4 bylo závislé na výsledcích testů z předchozí kapitoly.

Vytvořená aplikace je určena pro převod souřadnic mezi všemi požadovanými souřadnicovými systémy. Souřadnice je možno zadat vložení souboru ve formátu `souřadnice1_x;souřadnice1_y;...;souřadniceN_x;souřadniceN_y` nebo lze použít testovací soubory, které obsahují souřadnice v daném souřadnicovém systému. Souřadnice souřadnicového systému S-JTSK aplikace očekává záporné a s prohozenou složkou X a Y.

Výsledkem převodu jsou dvě tabulky:

- První tabulka obsahuje dobu převodu (konkrétně čas strávený ve funkci pro transformaci souřadnic) v milisekundách.
- Druhá tabulka obsahuje vstupní souřadnice a převedené souřadnice.

Souřadnice převedené nástroji, které používají nebo přepisují knihovnu Proj.4, jsou na zobrazených 6 desetinných míst shodné. Jedinou výjimkou je převod z WGS84 do WGS84/UTM33n, při kterém se souřadnice převedené pomocí Proj.4 a PostGIS (využívající Proj.4) liší od souřadnic převedených pomocí nástrojů Proj4js a OpenLayers (využívající Proj4js) v řádech tisícín.

Souřadnice převedené pomocí nástroje Mapy API se od souřadnic převedených ostatními nástroji liší v desetinné části, v některých případech i v jednotkách.

Mapy API očekává souřadnice v souřadnicovém systému S-JTSK kladné a v opačném pořadí, výsledkem jsou opět kladné souřadnice v opačném pořadí. Ostatní nástroje očekávají souřadnice v souřadnicovém systému S-JTSK záporné s prohozenou složkou X a Y, výsledné souřadnice jsou také v tomto formátu.

Aby byly vstupní souřadnice zadávány do aplikace jednotným způsobem, jsou při použití v Mapy API upraveny do požadovaného formátu. Obdobně jsou pak upraveny výsledné souřadnice tak, aby odpovídaly formátu výstupních souřadnic ostatních nástrojů.

8.2 Implementace

Aplikace je implementována v jazyce PHP, HTML, JavaScript a C.

Pomocí PHP je aplikace propojena s databází PostGIS, a dále také s programem v jazyce C.

Program v jazyce C (zdrojový kód programu viz [B.1](#)) využívá pro konverzi souřadnic knihovnu Proj.4. Po spuštění program očekává na standardním vstupu EPSG kód zdrojového a cílového souřadnicového systému a počet souřadnic pro konverzi. Následně je potřeba do programu postupně zadat jednotlivé souřadnice ve formátu `souřadnice_x;souřadnice_y`; , počet souřadnic musí odpovídat počtu zadanému v předchozím kroku. Po vložení poslední souřadnice program provede konverzi a výsledky vytiskne na standardní výstup. Poslední řádek výpisu výsledků je čas (v milisekundách) strávený ve funkci pro konverzi.

Spojení programu v jazyce C a části aplikace v PHP je zajištěno pomocí funkce `proc_open`, kterou je možno spustit program, a dále umožňuje komunikaci pomocí standardního vstupu a výstupu.

Jazyk JavaScript slouží pro propojení aplikace s nástroji Proj4js, OpenLayers a Mapy API, a dále je také využíván pro výpis výsledků.

Výsledná webová stránka je vytvořena pomocí HTML a vzhled je upraven pomocí CSS.

Kapitola 9

Závěr

Tato práce měla za cíl nalézt vhodné nástroje a knihovny na konverzi souřadnic mezi souřadnými systémy WGS84, S-JTSK, UTM zone 33N a Pseudo-Mercator se zaměřením na použití ve webových aplikacích.

Vhodných nástrojů či knihoven na konverzi souřadnic mezi systémy WGS84, UTM33n a Pseudo-Mercator existuje vzhledem k rozsahu a intenzitě používání těchto souřadnicových systémů velké množství. Naopak nalézt nástroje či knihovny podporující konverze souřadnic mezi S-JTSK a dalšími systémy se ukázalo jako značně problematické. Důvodem toho je malý rozsah území, výhradně Česká republika a Slovensko, na kterém se S-JTSK používá. Velkou roli v tom, že S-JTSK je v mnoha nástrojích nepodporovaný souřadnicový systém může hrát i fakt, že konverze souřadnic mezi souřadnicovými systémy WGS84, UTM33n a Pseudo-Mercator je značně jednodušší na implementaci než konverze mezi těmito souřadnicovými systémy a S-JTSK.

Jediné nalezené nástroje a knihovny, které podporovaly všechny požadované souřadnicové systémy byly knihovna Proj.4 a nástroje od ní odvozené, GeoTools a Mapy API.

Asi nejpoužívanějším řešením je knihovna Proj.4 a nástroje, které ji využívají nebo přepisují do jiného programovacího jazyka. Tato varianta získala svojí popularitu díky tomu, že podporuje nejvyšší počet souřadnicových systémů. Knihovnu Proj.4 je možno používat například v databázi PostGIS nebo v knihovně GDAL. Pro účely webových aplikací je asi nevhodnější variantou Proj4js, což je knihovna Proj.4 přepsaná do jazyka JavaScript.

Knihovna GeoTools nemá oproti Proj.4 žádné přepisy do jiných programovacích jazyků a je dostupná pouze v jazyce Java. GeoTools je nejrozsáhlejší projekt ze všech nalezených z hlediska podporovaných funkcí, tato knihovna má velice různorodé použití a pokud je již v projektu použita za jiným účelem, je vhodné ji použít i pro konverzi souřadnic.

Zajímavým nástrojem jsou Mapy API, které překvapily svojí rychlostí, avšak oproti ostatním nástrojům mají velice omezený počet podporovaných souřadnicových systémů. V aktuální verzi nepodporují ani převod jednotlivých souřadnic mezi požadovaným Pseudo-Mercator a jinými souřadnicovými systémy, ale vzhledem k tomu, že v současnosti Mapy API prochází velmi aktivním vývojem, je možné že se bude počet podporovaných souřadnicových systémů dále zvyšovat.

Co se porovnání přesnosti týče, tak všechny v této práci uvedené nástroje kromě Mapy API podporují vyšší množství transformačních metod, kterými je možno docílit požadované přesnosti.

Při výkonnostním srovnání se ukázalo, že nejrychlejšími jsou nástroje využívající knihovnu Proj.4 a GeoTools. S mírným odstupem na druhém místě jsou nástroje Proj4js, Mapy API a OpenLayers, rychlost těchto nástrojů však závisí na rychlosti interpretace JavaScriptu webového prohlížeče. Databázové nástroje PostGIS a SpatiaLite vyšly z testů jako nejpomalejší vzhledem ke konverzi souřadnic. Rozdíl rychlosti ostatních a PostGIS není tak znatelný, zatímto u SpatiaLite je výrazný.

Výsledky této práce mohou pomoci dalším programátorům vybrat nástroj na konverzi souřadnic, který by odpovídal jejich potřebám a požadavkům v daném problému.

Další možné rozšíření této práce je například zaměření na více souřadnicových systémů, měření rychlosti konverze v závislosti na různé požadované přesnosti u jednotlivých nástrojů nebo měření výkonu na mobilních či vestavěných zařízeních.

Literatura

- [1] Na řízení vlády ČR č. 430/2006 Sb. O stanovení geodetických systémů a státních mapových děl závazných na území státu a zásadách jejich používání.
- [2] PROJ.4 - Cartographic Projections Library [online]. Dostupné z: <http://trac.osgeo.org/proj/>, 2009 [cit. 2015-03-27].
- [3] PROJ4JS [online]. Dostupné z: <http://proj4js.org/>, 2009 [cit. 2015-03-27].
- [4] WGS-84 [online]. Dostupné z: <http://wiki.geocaching.cz/wiki/WGS-84>, 2010 [cit. 2015-02-19].
- [5] Grid-based methods - ArcGIS 10.1 Desktop Help [online]. Dostupné z: <http://resources.arcgis.com/en/help/main/10.1/index.html#//003r00000013000000>, 2013 [cit. 2015-03-22].
- [6] Java Map Projection Library [online]. Dostupné z: <http://www.jhllabs.com/java/maps/proj/>, 2014 [cit. 2015-03-27].
- [7] Installing Geospatial libraries [online]. Dostupné z: <https://docs.djangoproject.com/en/1.7/ref/contrib/gis/install/geolib/>, 2015 [cit. 2015-03-18].
- [8] Google Maps JavaScript API v3 [online]. Dostupné z: <https://developers.google.com/maps/>, 2015 [cit. 2015-03-24].
- [9] Mapy API verze 4.8 - Hanzelka a Zikmund [online]. Dostupné z: <https://api.mapy.cz/view?page=instruction>, 2015 [cit. 2015-03-24].
- [10] OpenLayers 3 [online]. Dostupné z: <http://openlayers.org/>, 2015 [cit. 2015-03-24].
- [11] DotSpatial [online]. Dostupné z: <https://dotspatial.codeplex.com/>, 2015 [cit. 2015-03-25].
- [12] GeoTools [online]. Dostupné z: <http://www.geotools.org/>, 2015 [cit. 2015-03-25].
- [13] Spatial Reference [online]. Dostupné z: <http://spatialreference.org/>, 2015 [cit. 2015-03-25].
- [14] GDAL - Geospatial Data Abstraction Library [online]. Dostupné z: <http://www.gdal.org/>, 2015 [cit. 2015-03-28].
- [15] PostGIS [online]. Dostupné z: <http://postgis.net/>, 2015 [cit. 2015-03-28].
- [16] SpatiaLite [online]. Dostupné z: <https://www.gaia-gis.it/fossil/libspatialite/index/>, 2015 [cit. 2015-03-29].
- [17] EPSG.io Coordinate Systems Worldwide [online]. Dostupné z: <http://epsg.io/>, 2015 [cit. 2015-04-21].
- [18] EPSG Dataset [online]. Dostupné z: <http://www.epsg.org/>, 2015 [cit. 2015-04-30].
- [19] Limits In SQLite [online]. Dostupné z: <http://www.sqlite.org/limits.html>, 2015 [cit. 2015-04-30].

- [20] Geoinformatika [online]. Dostupné z: <http://gis.zcu.cz/>, [cit. 2015-02-19].
- [21] Portál FreeGIS [online]. Dostupné z: <http://freegis.fsv.cvut.cz/>, [cit. 2015-02-19].
- [22] Geoportál ČÚZK [online]. Dostupné z: <http://geoportal.cuzk.cz/>, ČÚZK: 2010 [cit. 2015-02-19].
- [23] Buchar, P.: *Matematická kartografie*. ČVUT, 1996, ISBN 80-010-1491-6.
- [24] Ježek, J.: *Studijní materiály k předmětu MK2*. ZČU, 2011.
- [25] Krtička, L.: *Úvod do kartografie studijní materiál pro distanční kurz*. Ostravská univerzita, 2007, ISBN 978-807-3683-443.
- [26] Talhofer, V.: *Základy matematické kartografie*. Univerzita obrany, 2007, ISBN 978-807-2312-979.
- [27] Voženílek, V.: *Aplikovaná kartografie I: tematické mapy*. Univerzita Palackého, 2001, ISBN 978-802-4402-703.
- [28] Ward, D.: Encosia - ASP.NET, AJAX, jQuery, and more [online]. Dostupné z: <http://encosia.com/3-reasons-why-you-should-let-google-host-jquery-for-you/>, 2015 [cit. 2015-03-31].
- [29] Čapek, R.: *Geografická kartografie*. Státní pedagogické nakladatelství, 1992, ISBN 80-04-25153-6.

Příloha A

Obsah CD

Příložené CD obsahuje následující adresáře a soubory:

- Adresář src
 - Adresář demo - zdrojové kódy demonstrační webové aplikace.
 - Adresář tools_test - zdrojové kódy programů použitých při testování přesnosti a výkonnosti jednotlivých nástrojů.
- Adresář doc - zdrojové kódy tohoto dokumentu.
- Soubor xfiala46_BP.pdf - elektronická verze bakalářské práce.

Příloha B

Program v jazyce C využívající knihovnu Proj.4

```
1 #include <proj_api.h>
2 #include <stdio.h>
3 #include <math.h>
4 #include <time.h>
5 #define ERROR 5
6 #define EOK 0
7
8 /* Decode CRS from EPSG. */
9 int getCRSfromEPSG (int EPSG, projPJ * proj) {
10     if(EPSG == 4326) {
11         if (!(*proj = pj_init_plus("+proj=longlat +datum=WGS84 +no_defs"))){
12             fprintf(stderr, "pj_init_plus error: longlat\n");
13             return ERROR;
14         }
15     }
16     else if(EPSG == 5514) {
17         if (!(*proj = pj_init_plus("+proj=krovak +lat_0=49.5 +lon_0=24.833333333333333
18             +alpha=30.288139722222222 +k=0.9999 +x_0=0 +y_0=0 +ellps=bessel
19             +towgs84=570.8,85.7,462.8,4.998,1.587,5.261,3.56 +units=m +no_defs "))) {
20             fprintf(stderr, "pj_init_plus error: krovak\n");
21             return ERROR;
22         }
23     }
24     else if(EPSG == 3857) {
25         if (!(*proj = pj_init_plus("+proj=merc +a=6378137 +b=6378137
26             +lat_ts=0.0 +lon_0=0.0 +x_0=0.0 +y_0=0 +k=1.0 +units=m +nadgrids=@null
27             +wktext +no_defs")) ){
28             fprintf(stderr, "pj_init_plus error: mercator\n");
29             return ERROR;
30         }
31     }
32     else if(EPSG == 32633) {
33         if (!(*proj = pj_init_plus("+proj=utm +zone=33 +ellps=WGS84 +datum=WGS84 +units=m
34             +no_defs")) ){
35             fprintf(stderr, "pj_init_plus error: utm\n");
36             return ERROR;
37         }
38     }
39     else {
40         fprintf(stderr, "Unknown EPSG of from CRS\n");
41         return ERROR;
42     }
43     return EOK;
44 }
45
46
```

```

47 /* Function for conversion of coordinates. */
48 int convert(int fromCRS, int toCRS, double* points, double* points2, long count) {
49     projPJ pj_from, pj_to;
50     clock_t begin, end;
51     double time_spent;
52
53     if(getCRSfromEPSG(fromCRS,&pj_from) == ERROR || getCRSfromEPSG(toCRS,&pj_to) == ERROR) {
54         return ERROR;
55     }
56     // To radians if form WGS84
57     if(fromCRS == 4326) {
58         for(int i = 0; i < count; i++) {
59             points[i] *= DEG_TO_RAD;
60             points2[i] *= DEG_TO_RAD;
61         }
62     }
63     // Time measure functions and transform function of Proj.4
64     begin = clock();
65     pj_transform(pj_from, pj_to, count, 1, points, points2, NULL);
66     end = clock();
67     // To degrees if to WGS84
68     if(toCRS == 4326) {
69         for(int i = 0; i < count; i++) {
70             points[i] /= DEG_TO_RAD;
71             points2[i] /= DEG_TO_RAD;
72         }
73     }
74     for(int i = 0; i < count; i++) {
75         printf("%lf %lf\n", points[i], points2[i]);
76     }
77     time_spent = (double)(end - begin) / CLOCKS_PER_SEC * 1000;
78     printf("%lf",time_spent);
79     fflush(stdout);
80
81     return EOK;
82 }
83
84 int main() {
85     int fromCRS = 0, toCRS = 0;
86     long count = 0;
87     double *coords, *coords2;
88
89     if(scanf("%d %d %ld",&fromCRS, &toCRS, &count) != 3) {
90         fprintf(stderr, "Unable to find EPSG or count of coords\n");
91         return ERROR;
92     }
93
94     coords = (double*)malloc(count * sizeof(double));
95     coords2 = (double*)malloc(count * sizeof(double));
96
97     if(coords == NULL || coords2 == NULL) {
98         fprintf(stderr, "Unable to malloc\n");
99         return ERROR;
100    }
101
102    for(int i = 0; i < count; i++) {
103        scanf("%lf;%lf",&coords[i], &coords2[i]);
104    }
105
106    int error = convert(fromCRS,toCRS,coords,coords2,count);
107
108    free(coords);
109    free(coords2);
110
111    return error;
112 }

```

Ukázka kódu B.1: Program v jazyce C, součást demo aplikace, který využívá knihovnu Proj.4.