

UNIVERZITA KARLOVA V PRAZE

Přírodovědecká fakulta

Katedra aplikované geoinformatiky a kartografie



**INTERAKTIVNÍ POHLEDOVÁ MAPA TRATI ZÁVODU
JIZERSKÉ PADESÁTKY V ROCE 2009**

**INTERACTIVE PANORAMATIC MAP OF JIZERSKA 50
RACE IN 2009**

Bakalářská práce

Filip Skalický

srpen 2009

vedoucí bakalářské práce: RNDr. J. D. Bláha

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně a že jsem všechny použité prameny řádně citoval.

Jsem si vědom toho, že případné použití výsledků, získaných v této práci, mimo Univerzitu Karlovu v Praze je možné pouze po písemném souhlasu této univerzity.

Svoluji k zapůjčení této práce pro studijní účely a souhlasím s tím, aby byla řádně vedena v evidenci vypůjčovatelů.

V Jablonci nad Nisou dne 22. srpna 2009

.....

Filip Skalický

Poděkování

Na tomto místě bych chtěl poděkovat vedoucímu mé práce RNDr. Janovi Danielovi Bláhovi cenné rady a připomínky poskytnuté v průběhu zpracování. Dále bych rád poděkoval společnosti Adobe za poskytnutí potřebného programového vybavení a České informační agentuře spolu s Vojenským geografickým a hydrometeorologickým úřadem za poskytnutí dat pro zpracování. Stejně tak si poděkování zaslouží i rodina a přátelé za neustávající podporu.

Interaktivní pohledová mapa trati závodu Jizerské Padesátky v roce 2009

Abstrakt

Cílem práce je vytvořit interaktivní pohledovou mapu závodu Jizerské Padesátky a její následné umístění na internet. Práce se zabývá problematikou tvorby pohledových map a digitální tvorbou s možností umístění na internet. Zásadním tématem je otázka volby vhodného formátu pro šíření digitální kartografické tvorby na internetu. V rámci práce jsou nastíněny možnosti jednotlivých formátů a jejich vhodnost pro další zpracování. Obsahem práce jsou ukázky vývoje interaktivní webové aplikace pomocí dvou vývojářských platforem a vytvořená cílová aplikace prezentující lyžařský závod.

Klíčová slova: Digitální model, GIS, Interaktivní mapa, Pohledová mapa, Webové aplikace

Interactive panoramic map of Jizerska 50 race in 2009

Abstract

The aim of the project is to create an interactive map of the Jizerska 50 race and its publication on the Internet. The work deals with the question of creation of perspective maps and digital creation, with the possibility of placing on the Internet. The primary issue is the choice of the appropriate method for presentation of digital cartographic work on the Internet. In the context of the work are outlined the possibilities of individual methods and their suitability for further processing. The content of the work are examples of the development of interactive Web applications using the two development platforms and the created target application presenting the mentioned ski race.

Keyword: Digital model, GIS, Interactive map, Perspective map, Web application

OBSAH

Přehled použitých zkratk	6
1 Úvod	7
1.1 Cíle práce	8
1.2 Struktura práce	8
2 Úvod do problematiky	9
2.1 Mapy publikované na webu	9
2.2 Pohledové mapy	11
2.3 Digitální modelování.....	12
2.4 Možnosti tvorby webových aplikací	14
2.4.1 (X)HTML	14
2.4.2 CSS.....	15
2.4.3 JavaScript	16
2.4.4 Adobe Flex	16
2.4.5 Adobe Flash.....	17
2.4.6 SVG.....	18
2.4.7 VRML, X3D.....	19
3 Tvorba interaktivní pohledové mapy	20
3.1 Charakteristika území.....	20
3.2 Data použitá pro tvorbu aplikace.....	20
3.3 Zpracování modelu v GIS	21
3.3.1 Zpracování ve 2D	22
3.3.2 Zpracování ve 3D	23
3.4 Tvorba znakového klíče	24
3.5 Zpracování interaktivní aplikace	26
3.5.1 XHTML, CSS, JavaScript.....	27
3.5.2 Adobe Flex	30
3.6 Tvorba doplňkových objektů	32
4 Zhodnocení prostředí	33
4.1 XHTML, CSS, JavaScript.....	33
4.2 Adobe Flex	35
5 Závěr	36
Seznam zdrojů informací	37
Seznam příloh	41

PŘEHLED POUŽITÝCH ZKRATEK

CAD	C omputer- A ided D esign = počítačové systémy umožňující projektování
CSS	C ascading S tyle S heets = jazyk definující vzhled a rozvržení
DOM	D ocument O bject M odel = rozhraní umožňující přístup a modifikaci obsahu dokumentu
DPZ	D álkový P růzkum Z emě = metoda získávání geografických dat
DSM	D igital S urface M odel = digitální model povrchu viděný z hůry (včetně zástavby a vegetačního pokryvu)
DTM	D igital T errain M odel = digitální model přirozeného povrchu Země
HTML	H yper T ext M arkup L anguage = značkovací jazyk pro tvorbu dokumentů na web
MXML	M inimal e Xtensible M arkup L anguage = značkovací jazyk založený na XML
RIA	R ich I nternet A pliation = druh webových aplikací blízký klasickým aplikacím
S-42	Systém souřadnic vyvinutý pro zpracování vojenských topografických map
S-JTSK	Systém J ednotné T rigonometrické S ítě K atastrální = geodetický systém používaný pro zpracování civilních map
SGML	S tandard G eneralized M arkup L anguage = univerzální značkovací jazyk, jenž posloužil jako základ standardu řady dalších značkovacích jazyků
TIN	T rianguled I rrregular N etworks = nepravidelná trojúhelníková síť pro aproximaci terénu
W3C	W orld W ide W eb C onsortium = konsorcium pro vývoj webových standardů
WGS 84	W orld G eodetic S ystem 1984 = celosvětový geodetický systém souřadnic
WWW	W orld W ide W eb = celosvětová síť propojených hypertextových dokumentů
XHTML	e Xtensible H yper T ext M arkup L anguage = značkovací jazyk založený na XML
XML	e Xtensible M arkup L anguage = rozšiřitelný univerzální značkovací jazyk
ZABAGED	Z Ákladní B Áze G Eografických D at = digitální model území České republiky

KAPITOLA 1

Úvod

Během posledních několika desetiletí došlo k obrovskému překotnému vývoji celé vyspělé společnosti. Již od počátků existence člověka dochází díky kumulativnímu sběru informací k postupnému zrychlování rozvoje technologií, ale i člověka samotného. Nikdy v samotné historii lidstva však nedošlo k tak rychlému a zároveň tak výraznému rozvoji, který by byl zapříčiněn převážně jedním vynálezem.

Tím zásadním přelomem byl vynález výpočetní techniky, posléze počítače. Cyklus rozvoje je samozřejmě postupný, takže by se dalo polemizovat, zda byly spouštěčem vývoje skutečně počítače, či například vynález tranzistoru. Zásadní ovšem je, že právě v této době došlo k obrovskému nárůstu potenciálu pro budoucí rozvoj, jehož využíváme dodnes. Znamenalo to zvýšení rychlosti a efektivity výzkumu, či práce samotné i vznik zcela nových odvětví.

Jedním z projektů, jenž rychlého rozvoje dokonale využil, je i internet. Přestože počátky lze hledat na začátku devadesátých let, právě v posledních letech můžeme pociťovat obrovský potenciál, který nabízí. V podstatě na celém světě dochází k jeho běžnému zapojení do života lidí a zpřístupnění masám obyvatelstva. Internet nabízí téměř nekonečné možnosti výměny informací na libovolné vzdálenosti, komunikaci mezi lidmi, či sdílení dat do světa. Znamená to v podstatě deformaci prostoru z hlediska, jak jsme jej znali dříve. Komunikace probíhá z libovolného místa na Zemi a skutečné vzdálenosti již nejsou podstatné. Právě tato skutečnost je hlavním motorem globalizace a nové revoluce - přerodu v informační společnost.

Samotný přerod přináší samozřejmě i řadu problémů a neznámých k řešení. Mezi takové problémy patří například, jak sdílet informace, které se po celý věk uchovávaly pouze v analogové formě, jak vytvářet informace ve formě digitální, či jak co nejefektivněji převést informace analogové do formy digitální. Názory na tuto problematiku se různí a stále se vyvíjejí. Různé obory mají rozdílné požadavky, byť mohou pracovat s daty stejnými, či v zásadě velmi podobnými. Rozdílnost formátů však může znamenat i neřešitelný problém. Otázky přenositelnosti dat mezi nespočtem formátů, vytvoření mezioborového jednotného formátu jsou dnes velice aktuální.

Do období standardizace jsme stále zdaleka nedospěli a tak často vyvstává otázka, které z obrovského množství technologií a formátů jsou ty pravé, které by splňovaly nejlépe všechny požadavky. Právě tato otázka je tou hlavní při tvorbě nových aplikací a je také hlavním důvodem obrovských nároků na tvůrce samotné, jež dané technologie musí ovládat. Zároveň ale nabízí nepřehledné možnosti výběru a volby.

1.1 Cíle práce

Základním cílem je tvorba plnohodnotné mapové aplikace za účelem prezentace lyžařského závodu Jizerské Padesátky a to primárně na webových stránkách. Kromě reprezentativní funkce by měla uživatelům nabídnout i možnost kvalitní orientace v mapě, což vyžaduje skloubení hledisek kartografické správnosti a estetické a praktické funkce mapy.

Pro vyšší názornost a zkvalitnění orientace v prostoru bude mapa vytvořena jako pohledová. Znamená to vytvoření digitálního modelu terénu pro danou oblast konání závodu. Pro tento účel poslouží vrstevnicový model terénu, spolu s dalšími vrstvami pro orientaci v mapě. Mezi tyto vrstvy patří například říční síť, či síť komunikací v dané oblasti, doplněná o zastavěná území. Dalším obsahem mapy budou samotná data týkající se závodu, jako jsou tratě závodů a další doprovodné informace v podobě fotografií nebo profilů tratí. Mapa samotná bude doplněna o interaktivní prvky, které umožní interakci s uživatelem pro rozšíření možností manipulace s mapou a získu doprovodných informací o celé akci.

Dílčím úkolem je zorientování se v problematice kartografické tvorby umístitelné na internet a zvolení vhodné formy prezentace dat. V první řadě se jedná o zjištění stavu současné kartografické tvorby v oblasti pohledových map a možností využití na internetu. Druhým úkolem je zorientování se v problematice jednotlivých webových vývojových prostředí a formátů, jež jsou pro tento účel použitelné. Na základě nabytých zkušeností pak vybrat vhodné prostředí, v němž bude prezentována samotná mapa.

1.2 Struktura práce

Samotná práce se skládá z několika stěžejních částí. První teoretická část se zabývá hlediskem kartografické prezentace dat v prostoru a možnostmi kartografických děl na internetu. Tato část práce se pokouší o souhrnné vysvětlení problematiky pohledových map, jejich jednotlivých typů a vývoje. Zároveň se zabývá možnostmi tvorby pohledových map pomocí výpočetní techniky. Snaží se shrnout princip digitálních modelů terénu a jejich možné typové rozdělení. Poslední část kapitoly se zabývá možnostmi umístění mapy na internet a funkcemi jednotlivých řešení.

Druhá část týkající se teorie je zaměřena na technologie a postupy, které je možné využít pro prezentaci digitálních map. Jedná se o výčet několika, ať už všeobecně známých, nebo okrajových technologií používaných pro prezentaci na internetu, které mimo jiné umožňují i práci s digitálními mapami. Účelem této části práce je vymezit na základě požadavků na výslednou aplikaci nejvhodnější kandidáty pro praktickou část práce.

Právě tato třetí část práce se zabývá samotnou tvorbou mapové aplikace od začátků až do konce. Jedná se o oblasti tvorby digitálního modelu terénu, tvorby nejvhodnějšího znakového klíče a nastínění možností vývoje interaktivity mapy.

KAPITOLA 2

Úvod do problematiky

2.1 Mapy publikované na webu

V úvodní části byly nastíněny některé aspekty, jimiž se bude tato práce zabývat. Nyní je nutné pro orientaci vysvětlit několik základních pojmů, o nichž byla výše zmínka. Vývoj výpočetní techniky s sebou přinesl nutnost vytvoření principu *digitálního zpracování dat*. Stejná problematika se týká i digitálních map. *Digitální mapa* je digitální záznam konstrukčních prvků a obsahu mapy, který je možné vytvořit buď naskenováním mapy analogové, ručně vytvořené, nebo pomocí k tomu určeného speciálního softwaru. (Pravda, 2001, Cit. In Časarová, 2008). V podstatě tedy platí, že lze digitální mapy dělit na mapová díla vytvořená původně analogovou formou a zdigitalizovaná, či na mapy vytvořené jako původně digitální (Čerba, 2006). Lepším řešením je tvorba map původně digitálních, protože možnosti následné digitalizace jsou přece jen omezené, což může zhoršit čitelnost těchto map oproti původně digitálním. Je možné říci, že libovolnou mapu lze digitalizovat a zároveň libovolnou digitální mapu lze umístit na internet, potažmo web. Limitujícím faktorem je pouze velikost výsledného souboru, či rychlost internetového připojení. Platí také, že většina digitálních map se právě na internetu vyskytuje (Časarová, 2008).

Kromě výhod, které digitální zpracování přináší ve formě jednoduchého šíření dat, obecně rozšířené možnosti zpracování, či aktualizace, přináší i řadu problémů. Hlavním problémem je *výběr vhodných formátů*. Díky rozšířené a různému zaměření zpracování vzniklo i mnoho rozličných formátů dat, které jsou často spojeny právě s jedním softwarem s daným zaměřením (Voženílek, 2007, Cit. In Časarová, 2008). Značně se proto ztěžuje možnost přenositelnosti dat mezi různými obory a programy určenými pro zpracování. Přináší to sebou i zvýšené nároky na zpracování, kdy je nutná koordinace práce mezi oborovým specialistou a specialistou na daný software. Často však bývá tato skutečnost opomíjena, což vede v případě kartografie ke vzniku nekvalitních mapových děl (Monmonier, 2000).

Existuje několik druhů map umístitelných na web. Základní dělení je na *mapy statické a dynamické*, kdy lze obě kategorie dále dělit na *mapy náhledové (only view) a interaktivní* (Kraak, 2001, Cit. In Kozáková, 2005). Pro statické mapy platí, že nabízejí pouze jednu vrstvu náhledu. S mapou můžeme libovolně manipulovat, či používat zoom, avšak obsah mapy se například na základě použitého zoomu nemění, což pro dynamické mapy neplatí. V případě dynamických map dochází ke generování nového obsahu, či jeho generalizaci v určitých vrstvách přiblížení. Pohled na dynamičnost map se ale často liší. Vlastnost interaktivity mapy

spočívá v možnosti aktivace jednotlivých prvků na mapě, které na základě požadavku uživatele nabízejí další doprovodné informace. U map náhledových se s interakcí uživatele nepočítá.

Další možnost dělení map je podle *způsobu vizualizace dat*. Data mohou být šířena ve vektorovém nebo rastrovém modelu, kdy každý z těchto modelů je vhodnější pro různá využití. V případě *rastrového modelu* se jedná o vyjádření ve formě pravidelné matice bodů, většinou tvaru čtverce, kdy každému jednomu bodu je přiřazena hodnota vyjadřující znázorňovanou veličinu. Naproti tomu *vektorový model* se skládá pouze z bodů, spojů a polygonů, které znázorňují jednotlivé předměty. Čistě z hlediska geografických dat je rastrové zpracování nejvhodnější pro znázornění spojitých jevů, zatímco vektorové pro vyjádření tvarů. Ovšem v případě webových map vyvstávají některé další neznámé, ke kterým je třeba při volbě přihlížet. V případě internetu má veliký význam datová náročnost na přenos dat. Při použití *rastrového modelu* roste velikost s kvadrátem rozlišení. U *vektorového modelu* záznamu geodat má vliv na velikost pouze počet objektů, jež soubor vyjadřuje, což je maximálně přínosné například v případě přiblížení. Vektorový formát je proto velice vhodný pro šíření map na internetu. Problémem ale je, že většinu vektorových formátů nelze využít zdarma a podpora integrovaná samotnými webovými prohlížeči není doposud na nejlepší úrovni.

V prostředí internetu lze využít několik způsobů *šíření geodat*. Lze hovořit o možnostech *publikace, distribuce* nebo *sdílení*. V případě *distribuce* jsou data přímo zobrazena pomocí webového prohlížeče. Jedná se proto pouze o formáty dat (rastrové, vektorové) prohlížečem přímo podporované. Možností druhou je *sdílení dat*, kdy je k dispozici ke stažení pouze soubor s daty. Pro otevření těchto dat je zapotřebí vlastnictví aplikace, která je schopna data otevřít navlastním PC. Poslední možností je *distribuce geodat*, kdy jsou data šířena pomocí mapových serverů. Tento typ šíření se týká architektury klient-server, kdy veškerá data jsou uložena na jednom či více serverech a na základě požadavku klienta, který je umístěn na osobním počítači, jsou data odeslána a zobrazena pomocí aplikace generované mapovým serverem (Jedlička, 2008).

Využití *mapových serverů* je budoucností šíření map na internetu. Právě dostupnost veškerých dat z libovolného místa na světě, možnost práce s daty a vyřešení problému nepřenositelnosti, zapříčiněné nekompatibilními formáty, jsou hlavními výhodami takovýchto řešení. V budoucnu nebude nutné na osobním PC instalovat žádný software, ale pouze pomocí internetu zprovoznit služby, jež budou přístupné za pomoci webového prohlížeče. Jak je uvedeno výše, v případě mapového serveru se jedná o polarizovaný vztah složený z klienta, který slouží k zadání požadavku uživatele, a serveru distribuujícího data. Dělení je samozřejmě poněkud složitější. Existuje takzvaný tenký klient, kdy klientovi je odeslán pouze výsledný obraz a veškeré zpracování je prováděno na serveru. Fyzicky se uživatel k datům nedostane. V případě tlustého klienta server veškerá požadovaná data odesílá klientovi a samotné zpracování se provádí na straně klienta. Naskýtá se zde však otázka datové náročnosti na přenos a ztráta ochrany nad daty. Proto se takové spojení provádí především v intranetové síti, kde je nutnost editace dat a s rychlostí přenosu není problém.

2.2 Pohledové mapy

Pohledová mapa je zobrazení patřící do skupiny k tzv. mapám příbuzným zobrazením. Do této skupiny lze zahrnout například blokdiagram, tzv. ptačí pohledy, obrázkové mapy, ale i další (Herzog, 1988, Cit. In Šemberová, 1989).

Všechna tato zobrazení využívají *perspektivy*. Pokud zanedbáme čas, žijí lidé přirozeně ve třírozměrném prostoru, proto jsou schopni lépe porozumět informacím ve třech dimenzích (Cöltehin, 2002, Cit. In Knížová, 2006) a perspektivní zobrazení nám tuto možnost nabízí. *Perspektiva* je způsob, jakým zobrazit trojrozměrný prostor na pouze dvourozměrnou plochu (Šemberová, 1989). Pomocí různých technik převodu se pokouší navodit dojem maximální autentičnosti. Oproti klasickým zobrazením sice dochází ke ztrátě představy o přesné poloze objektu, nicméně tento typ zobrazení se zvýšením názornosti navozuje dojem skutečnosti a umožňuje mnohem jednodušší orientaci ve třetím rozměru. Právě to je důvod, proč poprávka po *perspektivním zobrazení* roste a to nejen v oblasti turismu v podobě například map lyžařských středisek, ale i v odlišných oblastech jako stavebnictví, urbanistické studie, životní prostředí (Hájek, 2008). Je proto vhodné oddělit od sebe dvě odlišné skupiny využití.

Prvním využitím jsou *mapy pro laickou veřejnost*, kdy hlavním záměrem je nabídnout snadnější možnost orientace, ale zároveň vytvořit dílo, které by upoutalo pozornost. Často se tak stává na úkor věrnosti a kartografické přesnosti. V případě druhém však nelze tento postup přijmout. Jedná se o *mapy pro odbornou veřejnost*, kde jde především o přesnost celého projektu, s důrazem na mnohem vyšší názornost, než nabízí zobrazení ve dvou dimenzích.

Perspektivu můžeme vnímat z několika hledisek, kdy maximální autentičnosti lze dosáhnou pouze jejich kombinací. Nejvíce vjem člověka ovlivňuje konstrukce převodu 3D prostoru do dvou rozměrů. *Konstrukční perspektivu* lze ale doplnit i dalšími způsoby, jako je změna tónu barvy, její intenzity, či jasů, díky čemuž lze dosáhnout například vytrácení se terénu v dálce díky vzdušné perspektivě, šednutí terénu, či změnu odstínu terénu díky rozptylu světelných paprsků v atmosféře (Šemberová, 1989).

Z konstrukčního hlediska existuje několik různě dokonalých perspektivních zobrazení. Mezi nejjednodušší patří izometrická projekce, kdy vzniká tzv. *paralelní perspektiva*. Základem je, že všechny promítací přímky mají stejný směr a jsou navzájem rovnoběžné. Při této projekci nedochází ke zkreslení délek ani výšek, což může být v řadě případů důležité. Jako příklady využívající paralelní perspektivy lze uvést vojenskou, či kavalírní perspektivu. Konstrukčně složitější oproti paralelní perspektivě, avšak mnohem lépe odpovídající lidskému vnímání prostoru, je *centrální středová perspektiva*. Centrální perspektiva pracuje s promítáním z jednoho bodu pomocí centrální projekce. Možnosti využití centrální projekce jsou dvě. První je centrální projekce v rovině, kdy dochází pouze ke zkreslení délek. Druhá je centrální projekce v prostoru, kdy veškeré body v prostoru, nehledě na polohu, jsou promítány z jednoho projekčního bodu. U tohoto typu projekce dochází jak ke zkreslení délek, tak i výšek, ale vnímání lidského oka odpovídá nejlépe (Šemberová, 1989). Důsledkem je však nemožnost

využití jednoho z nejdůležitějších prvků kartografického díla, jakým je měřítko. Právě centrální projekce v prostoru bude využito v průběhu vytváření aplikace.

Základním vyobrazením vytvářeným pomocí perspektivního zobrazení je *blokdigram*. Jedná se o výřez zemského povrchu ve tvaru čtyřúhelníka včetně vyobrazených řezných ploch, které mohou sloužit například k zakreslu podloží. V případě konstrukce je nejčastěji využíváno kosoúhlé paralelní perspektivy. Výsledný *blokdigram* proto nabývá zkrusleného tvaru.

Rozdílně je ke konstrukci přistupováno v případě *pohledové mapy*. Výsledkem *pohledové mapy* je většinou pravidelný čtyřúhelník, zatímco původní tvar oblasti může být libovolný. Navíc nebývají zobrazovány řezné plochy terénem. Je to způsobeno faktem, že *pohledová mapa* by měla odpovídat běžnému pohledu lidského oka z výšky. Měla by být proto jak tvarově, tak i velikostně úměrná. Ke konstrukci bývá většinou využíváno centrální perspektivy s šikmým pohledem na terén shora. Dle sklonu pohledu může být *pohledová mapa* ukončena i horizontem. Náplní *pohledové mapy* jsou obdobná data jako v případě map topografických

Vyšším stupněm *pohledových map* jsou pak například *globální pohledy*, kdy je stanovisko pozorovatele natolik vysoko, že přestává být zanedbatelné zakřivení Země nebo *panoramatické mapy*, kdy se jedná o *pohledovou mapu* s širším záběrem, než je přirozené pro lidské oko. Výsledkem mohou být až kruhové záběry z jednoho místa pozorovatele.

V posledních desetiletích dochází díky rozvoji výpočetní techniky k všeobecné digitalizaci prostorových dat. Díky tomuto fenoménu odpadá problém aktualizace prací a podstatně se zvyšuje možnost opakovaného využití již jednou zpracovaných dat. Stejný fenomén potkal i zpracování *pohledových map*. Digitalizace tvorby však znamená i ztrátu jisté jedinečnosti podání informace v případě analogového zpracování. Zatímco v dobách minulých bylo možné *pohledové mapy* chápat na rozhraní kartografie a umělecké tvorby, v době digitální tvorby se již umělecký faktor vytrácí. Dnes naopak dochází k jinému směřování. Digitální tvorba nabírá směr maximální věrohodnosti a co největší reálnosti modelů terénu (Hájek, 2008).

2.3 Digitální modelování

Tradičním, geografy využívaným prostředkem zpracování digitálních prostorových dat, jsou *Geoinformační systémy* (GIS). Jedná se o systém na pomezí několika přístupů zpracování, což zaručuje rozsáhlé možnosti zpracování a jistou komplexnost. Nabízí možnosti CAD systémů, možnost ukládání a správy dat ve formě databází, využití vizualizačních technik z oboru DPZ i zpracování pomocí kartografických nástrojů. (Kolář, 2003).

GIS slouží k uchování a analýze prostorových dat se schopností rozklíčování vztahů mezi geografickými informacemi. Úspěšnost řešení problematiky závisí na schopnosti vytvoření reálného modelu dané problematiky. Jedná se o *modely krajiny* popisující skutečný svět pomocí vlastností jednotlivých krajinných prvků a vztahů mezi nimi, jejichž účelem je zjednodušeně vyjádřit uspořádání okolního světa (Štych, 2008)

Pro vyjádření prostorových dat ve trojrozměrném prostoru jsou vytvářeny *digitální modely terénu* (DMT). Pojem DMT lze však chápat různě a proto je pro zpracování nutné uvést relevantní definice. První možností výkladu je *digitální model reliéfu* (DTM). Základními prvky tvořícími model jsou údolnice a hřbetnice, kdy terénem je myšlen obecně zemský povrch terénu bez vegetačního pokryvu a staveb vytvořených člověkem. Druhým relevantním pohledem je *digitální model povrchu* (DSM), kdy jsou zobrazeny veškeré vrchní plochy objektů, včetně patra vegetace a staveb (Štych, 2008). V případě tvorby pohledových map jsou využívána data DTM, ovšem v konečném stádiu jsou pohledové mapy doplněny i polohovou informací o vegetačním pokryvu a zástavbě.

Existuje řada kartografických metod, které umožňují *znázornit tvary terénu*. Nejvyužívanější možností je vyjádření pomocí *vrstevnic*, linií spojujících body o stejné nadmořské výšce. Dalšími možnostmi jsou například *barevná hypsometrie*, kdy jsou jednotlivé výškové stupně odděleny škálou barev, *stínování reliéfu*, jež dodává dojem plastičnosti, či třeba historická *kopečková metoda*. Pro vyšší názornost se často využívá kombinace více technik.

Mnohem sugestivnější metodou zobrazení prostoru je možnost využití perspektivních zobrazení. Stejně tak dobře lze perspektivní zobrazení kombinovat s předcházejícími technikami, tak typickými pro klasická kartografická zobrazení. Lze uvažovat dělení na *2D modely*, *2,5D* a *3D modely terénu* (Knížová, 2006). V případě *2D modelu* se jedná pouze o rovinnou grafiku. Zaznamenávají se pouze polohopisné údaje, s výškopisnými údaji se v tomto případě nepočítá. V případě *2,5D modelu* je kromě polohy rovinné známá i poloha ve třetím rozměru Z. Každému bodu je formou atributu přiřazena třetí souřadnice. Mezi *2,5D* lze zařadit vyjádření výšky vrstevnicovým modelem, stejně tak lze již uvažovat o perspektivním zobrazení. Omezením je možnost přiřazení pouze jednoho atributu výšky danému bodu. Při vyjádření ve formě *3D modelu* se již jedná o plnohodnotné zobrazení prostoru. Jednotlivým bodům vyjadřujícím polohu je možné přiřadit množinu souřadnic Z. Existuje proto i možnost zobrazení například jeskynních komplexů, či skalních převisů (Štych, 2008).

Vyjádření zemského povrchu pomocí výpočetní techniky je vždy jistou formou generalizace. Reálný povrch je vyjádřen nekonečným množstvím bodů, zatímco v případě zpracování modelů terénu jsme limitováni. Pro reprezentaci terénu lze proto vybrat pouze omezené množství bodů (Hájek, 2008). Na správném výběru bodů a zpracování záleží kvalita výsledného digitálního modelu terénu. Typy modelů terénu lze dělit dle využití konstrukčních ploch. Lze uvažovat *rastrový model*, *plátový model* a *polyedrický model terénu* (Bayer, 2008). V případě *rastrového modelu* jsou výšky vyjádřeny pomocí pravidelné matice bodů. Tradičně se jedná o pravidelnou čtvercovou síť se společnými hranami, tzv. *grid*. Nevýhodou vyjádření pomocí gridu je, že přesnost závisí na hustotě sítě, která určuje schopnost rozlišení jednotlivých tvarů. Tento problém řeší *polyedrický model*. Jedná se o vyjádření pomocí trojúhelníkových plošek, kdy jednotlivé vrcholy trojúhelníku mají stanovenou výšku (TIN). Síť bodů může být na rozdíl od rastrového modelu nepravidelná. To je výhodné vzhledem k možnosti vyjádření bodů libovolné polohy, pokud možno maximálně vystihujících morfologii reliéfu. Stejně tak je možné libovolně měnit hustotu sítě pro vyjádření různě členitého terénu. Tyto vlastnosti

umožňují mnohem kvalitnější přimknutí k terénu, než v případě rastrového modelu. Třetí možnost je vyjádření pomocí *plátového modelu terénu*. Problémem předchozích dvou modelů je lomové vyjádření terénu na spojích jednotlivých plošek. Naproti tomu *plátový model* umožňuje vyhlazení modelu v místech, kde je skutečný terén hladký a zároveň zachování lomových hran v případě, že je tomu tak ve skutečnosti. Nabízí nejuvěrnější možné vyjádření terénu, avšak za cenu složité konstrukce a výpočtů.

Nejvhodnějším způsobem zobrazení 3D prostoru pomocí výpočetní techniky je využití možností virtuální reality (Knížová, 2006). *Virtuální realita* umožňuje vizualizaci libovolných předmětů, manipulaci s nimi, pohybu v prostoru a to vše v reálném čase. Vjem prostoru uživatelem je v daném případě nejreálnější.

2.4 Možnosti tvorby webových aplikací

Existuje řada možností, jak umístit produkty na web a často záleží na mnoha faktorech, který ze způsobů je pro tento účel zvolen. Na problematiku lze nahlížet z řady pohledů. Namátkou lze jmenovat například přístupnost, inovativnost, nákladnost, či datovou náročnost řešení. V zásadě nelze nikdy splnit všechny požadavky a tak se často jedná o velice kompromisní řešení.

Praktická část této práce je zaměřena na tvorbu aplikace, jež by měla být umístěna na internetu pro praktické využití. Měla by proto splňovat i požadavky pro reálné využití. Bylo proto stanoveno několik kritérií, které by měla výsledná aplikace splňovat v maximální možné míře.

Požadavky stanovené na výslednou aplikaci:

- ✓ Datová nenáročnost – *nenáročnost na datové přenosy*
- ✓ Dostupnost – *dostupnost pro maximální množství uživatelů – problematika instalace pluginů, plná funkcionalita ve všech webových prohlížečích*
- ✓ Interaktivita – *možnost interakcí uživatele s aplikací*
- ✓ Kartografická správnost
- ✓ Nezávislost na serveru – *nezávislost na běhu konkrétních aplikací na straně serveru*
- ✓ Přehlednost
- ✓ Reprezentativnost

2.4.1 (X)HTML

HTML je značkovací jazyk, založený na původně mnohem rozsáhlejší značkovacím jazyce SGML, umožňující tvorbu hypertextových dokumentů. Původní SGML jazyk byl natolik rozsáhlý, že by práce s ním byla neefektivní (Snížek, 2003). Počátky HTML sahají do období začátku devadesátých let, kdy byl zprovozněn web, a objevila se potřeba jazyka, který umožní tvorbu dokumentů pro web.

Původní první verze byla pouze jednoduchou kostrou dnešního standardu. V průběhu dvou desetiletí prošel internet i web rozsáhlým vývojem. Požadavky na webové stránky se neustále zvyšují, což znamenalo tlak i na *rozvoj HTML*. Bohužel poptávka byla natolik veliká, že rozvoje se formou konkurenčního boje zhostily především webové prohlížeče (Snížek, 2004). Ve skutečnosti to znamenalo v podstatě neexistenci standardu. Jednotlivé prohlížeče přidávaly funkce dle vlastních zájmů, což znemožňovalo tvorbu webové stránky kompatibilní se všemi prohlížeči. Díky dodatečné standardizaci funkcí prohlížečů a jejich zavedení do praxe je dnes situace lepší, nikoliv však ideální. Pozitivum relativní nesystematičnosti vytváření standardu dnešního jazyka HTML je, že webové prohlížeče jsou dnes schopny kvalitně zobrazit i hypertextové dokumenty, které dle standardu nejsou ani zdaleka napsány správně.

Problémy standardu HTML se snaží vyřešit jazyk *XHTML*. Nejedná se o rozšíření, ale spíše naopak. Jedná se o snahu o návrat k původnímu konceptu HTML, pro který byl jazyk stvořen. Původním účelem HTML bylo nabídnout nástroj pro návrh a šíření dokumentů na webu, nikoli však jejich formátování. V průběhu překotného vývoje, však jazyk nabalil kromě formátování i řadu dalších funkcí, které strukturu značně zpřehledňují, a psaní samotného dokumentu se stává neefektivním. Právě tento problém *XHTML* odstraňuje. Jedná se o striktní standard jazyka založeného na XML, jehož úkolem je pouze definice obsahu hypertextového dokumentu, což s sebou přináší jak pevná pravidla převzatá z XML, tak i snadnou propojitelnost (Snížek, 2003). Ostatní vlastnosti jsou již úkolem dalších nástrojů.

Z hlediska šíření mapových produktů, jsou možnosti obou jazyků nedostačující. Umožňují pouze vkládání statických náhledových map formou rastrových obrázků, nicméně slouží jako základ aplikace dalších technik vizualizace jako CSS, JavaScript, SVG, ale i řada dalších. Díky stavbě na jazyku XML je zvláště vhodným základem jazyk XHTML.

2.4.2 CSS

V začátcích webu pravděpodobně nikdo nepředpokládal, že by mohl být v budoucnu důležitý vzhled hypertextových dokumentů. Právě pro tento účel vznikl pod záštitou konsorcia W3C nový jazyk CSS, který byl navržen pro úpravu vzhledu webových stránek. Záměrem *kaskádových stylů* je oddělení vzhledu jazyka od samotného obsahu, zefektivnění práce a podstatné zjednodušení postupů (Snížek, 2004).

Implementace standardu je opět na vývojářích jednotlivých prohlížečů a tak se i v tomto případě často stává, že zvolené řešení není zcela kompatibilní se všemi prohlížeči. Nicméně lze tvrdit, že v případě novějších verzí webových prohlížečů lze většinou nalézt kvalitní řešení.

Je obecně přijímaným faktem, že kaskádové styly slouží k formátování webových stránek vytvořených v jazyce HTML, nicméně zapomíná se na další. Možnost *využití kaskádových stylů* existuje u většiny značkovacích jazyků a u všech jazyků vystavěných na základě jazyka XML.

Z hlediska zpracovávané aplikace i mapových produktů umístovaných na internet nemá CSS zásadní význam, nicméně podílí se významnou měrou na vzhledu aplikací u řady používaných řešení.

2.4.3 JavaScript

JavaScript je v dnešní době na webu asi nepoužívanějším skriptovacím jazykem na straně klienta. Jedná se o objektově orientovaný jazyk bez nutnosti kompilace zdrojového kódu. Znamená to, že jej lze použít přímo při tvorbě webové stránky, kdy interpretaci provádí webový prohlížeč.

Nevýhodou jazyků, jež se spoléhají na *interpretaci webovým prohlížečem*, je různost vykládání standardizací a míra jejich implementace. Právě proto i v případě JavaScriptu vyvstává problém s vytvořením funkčního kódu, jež by byl shodně interpretován ve většině prohlížečů (Suering, 2008). Zatímco v případě HTML a CSS je již tento problém z velké části odstraněn, v případě JavaScriptu se lze s tímto problémem setkat velice často i dnes. Navíc pokud vezmeme v potaz, že většina dnešních webových stránek je tvořena právě kombinací těchto tří jazyků, pak nelze než webovým vývojářům popřát hodně trpělivosti s laděním aplikace.

Úkolem jazyka JavaScript je provádění operací na straně klienta. Původním účelem jazyka bylo pouze ověřování formulářů, nicméně dnes nabízí rozsáhlé možnosti a používá se především pro zatraktivnění webových stránek.

Velice důležitou specifikací je standard *DOM*. Přestože se nejedná přímo o součást specifikace JavaScriptu, značně rozšiřuje funkčnost. Jedná se o model, se kterým pracují snad všechny skriptovací jazyky. *DOM* nahlíží na dokumenty psané značkovacími jazyky jako na hierarchické uspořádání rodičů a potomků. Při přesné specifikaci jednotlivých prvků dokumentu umožňuje pomocí skriptovacího jazyka přístup k jednotlivým elementům dokumentu a jejich dynamickou změnu na základě událostí (Suering, 2008). Této vlastnosti dokáže JavaScript mohutně využít ve prospěch dynamiky a interaktivity webových stránek.

Z hlediska možností umístování map na internet je JavaScript velice důležitým. Umožňuje vytvářet interaktivní a dynamické mapy, což přidává jak na funkčnosti, tak i atraktivnosti webových aplikací. Z hlediska tvorby výsledné aplikace je neoddiskutovatelnou nevýhodou nejednotná interpretace skriptů jednotlivými prohlížeči a omezené možnosti takového řešení. Stejný problém se týká celé vývojářské platformy (CSS, (X)HTML, JavaScript).

2.4.4 Adobe Flex

Současné aplikace na internetu jsou stále nejčastěji vyvíjeny konceptem (X)HTML, CSS, JavaScript. Jak se postupně ukazuje, některým požadavkům již takovéto zpracování není schopné vyhovět. Tento fakt dovedl vývojáře k vytvoření zcela nového konceptu tvorby

webových aplikací. Základními požadavky jsou vysoké nároky na uživatelské prostředí (uživatelská vstřícnost, interakce s uživatelem). Tento koncept programování pro web se ujal teprve v několika posledních letech a znamenal vznik takzvaných *RIA aplikací* (Kunc, 2005). Jedná se o podstatné přiblížení možností internetových aplikací aplikacím pracujícím na osobních počítačích jak z hlediska uživatelského, tak ale i vývojářského. Zaměřuje se na oproštění od striktní komunikace klient-server, kdy při každé akci uživatele musí být znovu načtena/odeslána celá webová stránka. Tento koncept umožňuje obnovení pouze potřebné části aplikace (stránky) a zároveň umožňuje i samostatné zpracování informací na straně klienta (Kunc, 2005).

Během pěti let vývoje se objevilo několik vývojářských platforem. Za předchůdce můžeme považovat i *AJAX*, ale hlavními hráči na trhu se stali především *Microsoft Silverlight*, *OpenLaszlo*, *JavaFX*, ale především *Adobe Flex*, který se snad zatím prosazuje nejvíce. Společným prvkem všech těchto platforem, kromě *JavaFX*, je využití dvou jazyků pro tvorbu aplikací. Jedná se vždy o značkovací jazyk založený na XML a některý ze skriptovacích jazyků založený na ECMAScriptu. *JavaFX* využívá pouze skriptovacího jazyka založeného na Javě. Dalším společným prvkem je i nutnost kompilace zdrojových dokumentů pro získání výsledné aplikace, jež v zásadě vždy potřebuje pro běh na webovém prohlížeči i rozšiřující plugin (běhové prostředí).

Přestože jednotlivá vývojová prostředí se teprve vyvíjí a možnosti uplatnění se teprve utvářejí, jako přední hráč se ukazuje *Adobe Flex*. Koncept vývoje v případě *Flex* velice připomíná vývoj v prostředí HTML a JavaScript. Rozdíl je pouze v mírně odlišné syntaxi použitých jazyků. Pro vývoj uživatelského prostředí, tedy vzhled aplikace, je využíváno značkovacího jazyka *MXML*. Jeho účelem je pouze definice prvků, z nichž je složeno uživatelské prostředí (Bernard, 2008). Řeč je nyní především o prvcích jako tlačítka, formuláře, ale i menu a řada dalších. Druhým jazykem je objektově orientovaný skriptovací jazyk *ActionScript*. Jedná se o jazyk velice podobný jazyku JavaScript využívaný v aplikacích společnosti Adobe. Sémantika i funkčnost obou jazyků je velice podobná, nicméně jazyk *ActionScript* není možné interpretovat přímo pomocí webového prohlížeče. V případě vývojového prostředí *Adobe Flex* slouží pro definici programového kódu, tedy definici dynamiky a interaktivity webové aplikace (Bernard, 2008).

Díky otevřenosti platformy je možné vytvářet aplikace pomocí libovolných editorů, či využít k tomu vytvořeného vývojového prostředí *Flex Builder*, jež je placené. Výslednou aplikaci je nutné před použitím zkompileovat pro *běhové prostředí*. Aplikaci nutnou pro správnou interpretaci zkompileovaného kódu. Pro prostředí internetu se jedná o *Flash Player*. Druhou možností je běhové prostředí *Adobe AIR* umožňující spuštění na osobním počítači bez přístupu na internet.

Pro možnost tvorby cílové aplikace se jedná o velice kvalitní řešení s rozsáhlými možnostmi interaktivity. Nevýhodou je snad nutnost instalace rozšiřujícího pluginu, nicméně v případě *Flash Playeru* platí, že na valné většině počítačů je již nainstalován a jedná se navíc o zavedenou značku. S dostupností pro maximální počet uživatelů proto není problém.

2.4.5 Adobe Flash

Jistou možností tvorby webových aplikací nabízí i vývojová platforma *Adobe Flash*. Jedná se opět o zcela jiný přístup k vývoji webových stránek. Technologie Flash byla vyvinuta jako nástroj pro tvorbu grafických prvků umístitelných na internet. V tomto směru nabízí rozsáhlé možnosti, kdy umožňuje vytvářet interaktivní a dynamické prvky pro webové stránky (Franklin, 2003). V podstatě lze hovořit o grafickém formátu, ve kterém je možné vytvářet například reklamní bannery, animace, ale i celé webové stránky. Jedná se o formát využívající primárně vektorové grafiky, což se pozitivně odráží na velikosti výsledného souboru. Nicméně negativem je, že se jedná o formát komerční s nutností zakoupení vývojových prostředí. To je podmíněno skutečností, že na rozdíl od ostatních zde uvedených vývojových prostředí a jazyků nejsou data spravována formou prostých textových souborů, ale formou binárního kódu, který není možné běžně přečíst.

Tvorba aplikací probíhá zcela odlišnou formou než v případě ostatních přístupů. Základem tvorby je grafický editor s časovou osou, která slouží k vytváření dynamiky prvků. Interaktivitu jednotlivých prvků lze pak dosáhnout pomocí využití scriptovacího jazyka ActionScript, který umožňuje vytvořit programový kód pro jednotlivé akce uživatele. Výsledný produkt je nutné zkompileovat, aby došlo k jeho kompresi. Pro zprovoznění aplikací v prostředí webu je nutná instalace volně dostupného pluginu (běhového prostředí) Flash Player.

Z hlediska umístění mapových produktů na web a tvorbu webových stránek se jedná o zajímavý počin, kdy výsledný produkt není nijak svazován po grafické stránce, nicméně přináší to i svá negativa. Znamená to nutnost grafického návrhu veškerých prvků, jež se na stránkách vyskytují a všech jejich jednotlivých stavů. Dalším faktem je, že internet je především informační medium a doposud se k němu tak přistupuje. Je proto logické, že pro veškeré návrhy webových aplikací je typická podřízenost vzhledu na základě obsahu dokumentu. V případě Adobe Flash tomu tak není. Primární je v tomto případě vzhled, což může být přínosné v případě tvorby animací, či reklamních prvků, nicméně v případě webových stránek může takový přístup vést k degradování samotného obsahu. Jedná se o velice kvalitní platformu, avšak je třeba se při tvorbě držet jistých mezí.

2.4.6 SVG

Nový formát SVG byl vytvořen teprve v roce 2001. Projekt byl započat s cílem vytvořit nový, zcela otevřený vektorový formát umožňující vytvoření a umístění dvourozměrné vektorové grafiky na internet. Formát je založen na značkovacím jazyce XML, což umožňuje snadnou propojitelnost formátu s jinými prostředími (Čerba, 2008).

Přestože se mělo jednat o formát vektorový, umožňuje správu jak vektorové, tak i rastrové grafiky a dalším objektem, který je možné spravovat je textový řetězec (Jedlička, 2008). Svou koncepcí tak vznikl formát zcela nového typu. Díky stavbě na jazyce XML je možné s formátem SVG provádět i operace, které nejsou pro vektorové formáty běžné. Podporuje

například tvorbu animací i interaktivitu prvků. Umožňuje změnu vlastností prvků pomocí CSS 2, či přístup k jednotlivým prvkům pomocí technologie DOM a přiřazování skriptů (Jedlička, 2008).

Velkou výhodou formátu SVG zároveň je, že oproti rastrové grafice nabízí podstatně nižší velikost při zachování kvality. Obecně platí, že pro prostředí internetu je vektorová grafika vhodnější a hlavní překážkou byla pouze neexistence otevřeného nekomerčního formátu. Navíc struktura formátu je jednoduchý textový soubor ne nepodobný například HTML. Osvojení si práce s SVG není proto příliš složité. Nevýhodou je zatím stále nedostačující podpora formátu webovými prohlížeči a tudíž nutnost instalace zásuvného modulu.

Vzhledem k vytvářené aplikaci se jedná o velice kvalitní formát, který by bylo možné prakticky využít. Nepřekonatelným omezujícím faktorem je ale špatná implementace jeho vlastností do prohlížečů a nedostatečná rozšířenost pluginu umožňujícího prohlížení.

2.4.7 VRML, X3D

Velice zajímavým počinem jsou formáty *VRML* a *X3D*. Jedná se z velké části identické formáty sloužící k popisu 3D scén. Jazyk *X3D* je pouze další rozvinutější specifikací jazyka *VRML*, která obsahuje řadu rozšíření a především striktnější standardizaci, která v případě *VRML* chyběla.

Zápis formátu je v obou případech zpracován pomocí textového souboru, nicméně *možnosti zápisu* v případě *X3D* existují dvě. První možností je využití deklarativní metody využívané v jazyku *VRML*. Druhý přístup k zápisu je však založen na zcela novém principu. Jedná se o zápis vystavěný na syntaxi značkovacího jazyka *XML*. Možnosti obou zápisů jsou zcela srovnatelné a závisí pouze na preferencích uživatele (Ečer, 2008). Nicméně primárně se u nových aplikací počítá spíše s užitím syntaxe využívající *XML* zápisu. Především z důvodů snadné propojitelnosti s řadou dalších technologií a možností rozšiřitelnosti.

Z pohledu využití aplikace se jedná o zajímavou alternativu především z hlediska možností zobrazení výstupu formou virtuální reality. Pro orientaci v terénu je možnost manipulace s digitálním modelem terénu v reálném čase nejnázornější možnou prezentací. Záporům této technologie je nicméně nutnost instalace rozšiřujícího pluginu, který umožňuje prezentaci projektu. Mezi uživateli neexistuje všeobecné povědomí o tomto formátu, a proto se nemusí jevit příliš důvěryhodně. Navíc výsledná aplikace by měla sloužit k reprezentativním účelům, čemuž nemusí aplikace vytvořená pomocí tohoto formátu zcela odpovídat.

KAPITOLA 3

Tvorba interaktivní pohledové mapy

3.1 Charakteristika území

Tvorba výsledné aplikace je začleněna do území hornatiny Jizerské hory. Jizerské hory jsou nejseverněji položené hraniční pohoří na území České republiky, které dále pokračuje i na území Polska. Nachází se v oblasti Frýdlantského výběžku západně od Krkonoš, na které přímo navazuje. Z ostatních stran je ohraničeno sníženinami.

Nejvyšším vrcholem pohoří je Vysoká kupa (Wysoka kopa) na polské straně státní hranice s nadmořskou výškou 1126 m n. m. Na české straně pohoří dosahuje nejvyšší výšky hora Smrk (1124 m n. m.). Dále se v pohoří nachází dalších 11 vrcholů přesahujících většinou pouze mírně výšku 1000 m n. m. Pro oblast Jizerských hor jsou typická denudovaná území ve formě náhorních plošin s nadmořskou výškou okolo 900 metrů. Této nadmořské výšky dosahují již vrcholy na úbočí Jizerskohorského masivu, jež jsou velice příkré především na severní straně pohoří. Relativní převýšení na této straně dosahuje až 700 výškových metrů. Na straně jižní je to pak okolo 350 metrů.

Oblast konání závodu Jizerské Padesátky se nachází v oblasti náhorní plošiny. Start a cíl jsou umístěny na jihozápadním okraji pohoří v obci Bedřichov. Dále závod pokračuje na východ celým územím Jizerských hor, do oblasti nejvyšších vrcholů pohoří, kde se stáčí nad severní úbočí a zpět do oblasti cíle. Kratší distance závodu jsou vedeny v západní části pohoří.

3.2 Data použitá pro tvorbu aplikace

Data byla poskytnuta pouze pro vymezené území Jizerských hor. Zájmové území bylo vymezeno souřadnicemi Y: 687 000, X: 959 500 a Y: 664 000, X: 978 000.

DMÚ 25 (Digitální model území 1 : 25 000)

Jako základní podkladová data pro tvorbu aplikace byla využita vojenská databáze vektorových dat DMÚ 25. Jedná se o digitální model území vycházející z vojenské topografické mapy měřítka 1 :25 000. Polohová přesnost zakresu objektů je 0,5 m až 20 m, dle třídy přesnosti záznamu jednotlivých objektů. Data jsou zpracována v souřadnicovém systému S-42, poskytována jsou však i v souřadnicových systémech S-JTSK a WGS-84. Data pro zpracování

výsledné aplikace byla obdržena v souřadnicovém systému S-JTSK. Geodata jsou k dispozici i pro omezenou oblast za hranicemi státu, což bylo pro zpracování velice důležité. Obsahem databáze jsou:

<i>Vodstvo</i>	<i>Rostlinný a půdní kryt</i>
<i>Komunikace</i>	<i>Sídla a zástavba</i>
<i>Energetické, telekomunikační a potrubní sítě</i>	<i>Hranice území</i>
	<i>Terénní reliéf</i>

Pro zpracování byly poskytnuty vrstvy obsahující *zástavbu, vegetační pokryv, síť komunikací a vrstevnicový model reliéfu* (po 5 m).

DIBAVOD (*Digitální báze vodohospodářských dat*)

Databáze vodohospodářských objektů vytvořená pro zpracování témat týkajících se vodohospodářské problematiky. Jedná se primárně o nadstavbu pro vrstevnicový model území ZABAGED zpracovanou pro měřítko 1 : 10 000. Databáze je volně dostupná veřejnosti.

Pro zpracování byly využity vrstvy *vodní toky a vodní plochy*.

3.3 Zpracování modelu území v GIS

Jak již bylo zmíněno výše, geodata lze vyjádřit v různě rozměrném prostoru. Pro většinu analýz dostačuje využití dvourozměrného prostoru, avšak vyskytují se i projekty, které vyžadují využití 3D prostoru. Existuje řada aplikací, které jsou přímo na tuto problematiku zaměřeny (např.: Atlas, CADTutor, 3DEM). Zpracování ve 3D však nabízí i řada systémů, které mají mnohem rozsáhlejší využití a pro které je tento druh zpracování pouze nadstavbou. Příkladem, kde lze takovéto systémy využít, jsou architektura, či geodézie s využitím CAD systémů, či právě geografie s využitím Geoinformačních systémů.

Z oblasti geodézie lze jmenovat například hojně využívaný nástroj Microstation s využitím rozšíření DTM, které umožňuje zpracování 3D modelů terénu. Stejně existuje nadstavba Microstation Geoinformatics jako rozšíření na plnohodnotný geoinformační systém. Microstation slouží především pro přesné znázornění méně rozsáhlých oblastí při dosažení vysoké přesnosti. Stejně jako všechny CAD systémy je aplikace zaměřena především na správu vektorových dat.

V oblasti geoinformačních systémů je velice zajímavým počinem systém GRASS. Jedná se o open-source software, která se spoléhá na aktivitu vlastní komunity. Jeho využití není žádným způsobem zpoplatněno a lze se podílet i na jeho dalším vývoji, či jeho funkce jen rozšířit pro vlastní potřeby. Přestože se jedná o řešení šířené zdarma, možnosti, které nabízí, se dají srovnávat s komerčními systémy. Pro účely 3D vizualizace prostorových dat je nutné využít modul NVIZ. Tento modul umožňuje zpracování modelu terénu s využitím rastrových dat a znázornění řady dalších prvků jako například vegetace.

Mezi nejhojněji využívané geoinformační systémy patří komerční řešení ArcGIS. Jedná se o velice kvalitní řešení nabízející rozsáhlé možnosti zpracování jak vektorových, tak i rastrových dat s možností využití zpracování formou 3D modelu terénu. Pro 3D vizualizaci se využívá aplikace ArcScene. Poměrně zásadní je možnost vkládání předpřipravených 3D objektů budov, vegetace a dalších, či import těchto objektů z jiných aplikací.

3.3.1 Zpracování ve 2D

Pro zpracování modelu krajiny bylo využito geoinformačního systému ArcGIS, konkrétně aplikace *ArcMap 9.3* (ArcGIS 9.3, 2008). Aplikace posloužila v daném případě pouze k přípravě jednotlivých vrstev pro jejich další využití. Bylo proto využito nejjednodušších možných řešení, nehledě na omezené možnosti dalšího využití. V této části zpracování bylo provedeno utřídění veškerých dostupných dat nezbytných pro tvorbu výsledné aplikace. Pro zpracování byla dostupná data databáze DIBAVOD, která však obsahovala data celé republiky. Bylo proto nutné vyříznutí pouze potřebné oblasti ve vymezeném území. Jednalo se jak o vrstvy vodních toků, tak i vodních ploch.

Data, která byla poskytnuta, vycházejí z map velkých měřítek – DIBAVOD (1 : 10 000), DMÚ 25 (1 : 25 000) – což zaručuje velice vysokou kvalitu dat, nicméně model krajiny by měl být především přehledný, k čemuž bylo nutné přizpůsobit i obsah. Proto bylo u těchto vrstev (zástavby, komunikací, vodních toků i vodních ploch) přistoupeno k rozsáhlé *generalizaci obsahu*. Ponecháno bylo pouze omezené množství objektů důležitých pro orientaci.

V případě *vodních toků* zůstaly zachovány větší vodní toky, u kterých byl patrný vliv na morfologii terénu. Z důvodů vyšší věrohodnosti modelu byly ponechány i menší toky, kdy by mohly vektory vodních toků osamoceně vypadat nereálně, např. z důvodu nereálného zakřivení toku. Z důvodů přehlednosti byl omezen i počet vodních ploch. Zanechány byly pouze nejrozsáhlejší *vodní plochy*, které jsou nezanedbatelným krajinným prvkem a zároveň jsou veřejností obecně známy. Jedná se o všechny přehradní nádrže a významné rybníční plochy.

Oblast Jizerských hor je protkána velice hustou sítí stezek a cest. Pro zpracování však byly využity pouze *komunikace*, které umožňují přístup k závodu nebo jsou samotné součástí sítě lyžařských tras. Pro tento účel bylo nutné provést klasifikaci jednotlivých komunikací a následně zařazení do jednotlivých vrstev. Silniční síť byla rozřazena na hlavní a vedlejší silnice. Nejedná se ani tak o dělení dle třídy komunikace, jako o oddělení méně významných a místních komunikací od komunikací hlavních, které jsou důležité pro dopravu v okolí závodu. Z dostupné vrstvy komunikací bylo nutné vyčlenit i komunikace, týkající se samotného závodu. Klasifikace byla provedena dle využití na jednotlivé závodní tratě (Jizerská 50, Jizerská 25, Jizerská 30) a další lyžařské tratě Jizerské magistrály, které jsou v průběhu zimní sezony upravovány.

Jako nejefektivnější způsob *klasifikace* obsahu byla zvolena možnost duplikace vrstvy komunikací a následná editace pomocí modulu *Editor*. Byla proto vytvořena řada nových vrstev, které obsahovaly pouze omezené množství původních komunikací. Vzhledem k faktu,

že jednotlivé vrstvy se příliš nepřekrývají, bylo možné vytvořit nižší množství vrstev i s využitím klasifikace dat, pomocí atributů uvnitř vrstvy. Nicméně tento způsob klasifikace se ukázal jako příliš zdlouhavý.

Vrstva zástavby byla zcela přepracována. Poskytnutá data byla použita pouze jako podkladová pro vytvoření nových vrstev zástavby. Jednalo se o pět vrstev zástavby, z nichž první byla využita pro zaznamenání polohy církevních staveb, jako významných orientačních bodů. Do ostatních vrstev byly nahodile přiřazeny lokace budov. Body vyjadřující jednotlivé budovy odpovídají oblasti zástavby obcí, nicméně nejedná se o určení poloh skutečných budov. Zástavba je vyjádřena pouze schematicky.

Další novou vrstvou je vrstva bodů vyjadřující vegetaci. Opět se jedná o schematické vyjádření oblastí s vegetací a bez vegetace. Zároveň jsou oblasti vegetace vymezeny způsobem, aby v případě dalšího zpracování co nejméně kolidovaly s jinými prvky mapy (vodstvo, komunikace), které jsou pro orientaci důležitější. Vzdálenost od vrstev komunikací a vodních toků by měla být minimálně 80 m. Pro zjištění současné situace zalesnění oblasti bylo využito leteckých snímků oblasti.

Dalším krokem po dokončení klasifikace obsahu byla tvorba samotného modelu terénu. Pro dosažení maximální přesnosti bylo využito interpolace (nalezení nejpravděpodobnějších hodnot mezi body se známými hodnotami zkoumané veličiny) dostupných dat. Pro vyjádření morfologie skutečného terénu je nejvhodnější využití interpolace *Topo to raster*. Jedná se o interpolaci vyvinutou přímo pro daný účel, jenž bere v potaz i hlavní činitel utvářející terén, vodu (vodní plochy, vodní toky), což model přibližuje skutečnosti a odstraňuje případné chyby. Interpolace byla proto provedena se vstupními vrstvami vodních toků, vodních ploch, vrstevnic a výškových bodů. Tato vrstva byla speciálně vytvořena k účelu zpřesnění vrcholů modelu. Během zpracování bylo zjištěno, že vymezení zájmového území je příliš rozsáhlé, bylo proto využito i možnosti oříznutí modelu v průběhu vytváření interpolace *Topo to Raster*. Důležité při zpracování interpolace je nastavení velikosti výstupní buňky rastru. Pro dosažení dostatečné kvality byla zvolena velikost 5, což pro dané využití zaručuje dostatečnou kvalitu.

Posledním krokem zpracování v dané aplikaci bylo *vytvoření nepravidelné trojúhelníkové sítě* (TIN). Zásadou provedení předešlé interpolace dat dosahuje zpracování sítě vyšší kvality. Především díky zpřesnění zavedením hydrologických procesů, ale také díky zabezpečení možnosti lepšího kopírování terénu než v případě využití samotných vrstevnic. Právě pro kvalitu přimknutí sítě k rastrovému modelu (terénu) je důležité nastavení tolerance výšky mezi modelem a výsledným TINem. Na tomto parametru záleží kvalita výsledného modelu. Pro vytvářený model se ukázala být dostačující hodnota 0,3 m.

3.3.2 Zpracování ve 3D

Pro zpracování 3D modelu byla použita aplikace ArcScene (ArcGIS 9.3, 2008). Jedná se o aplikaci umožňující vizualizaci a manipulaci s modelem v reálném čase. Pro vizualizaci jsou

využity vrstvy vytvořené v předchozí aplikaci ArcMap. Jedná se o veškeré lyžařské tratě, síť komunikací, vodní toky a vodní plochy, vrstvu vegetace, zástavby a státní hranice. Posledním prvkem je trojúhelníkový síťový model TIN.

Základem tvorby 3D modelu krajiny je právě TIN. Jedná se o základní jednotku vyjadřující prostorové uspořádání ve 3D. Na výsledný vzhled má vliv nastavení převýšení modelu oproti skutečnosti, jenž bylo nastaveno trojnásobné, a zvolení způsobu vizualizace modelu. V našem případě bylo provedeno pouze odstranění okrajových linií a byla změněna barva modelu na bílou, která symbolizuje barvu sněhu.

Pro nastavení kopírování terénu (TIN) u ostatních načtených vrstev bylo nutné provést nastavení přimknutí spolu s nastavením shodného převýšení. Dalším nutným nastavením je volba vizualizace jednotlivých vrstev. Nejdůležitější je nastavení vrstvy budov, kde je nutné zvolit typ stavby, která se má na místě bodového znaku zobrazit. Stejně tak je nutné provést nastavení vzhledu pro vrstvu vegetace, kdy je všem bodům přiřazen vzhled vybraného stromu. Právě tyto dvě volby jsou důležité pro autentičnost výsledného modelu. Nastavení ostatních objektů již není natolik důležité. Při následném zpracování budou překryty a tak je nutné nastavit symboly s co nejmenší šířkou, kdy jsou vrstvy stále použitelné.

Cílem zpracování je tvorba pohledových map, proto je nutné provést export dat za aplikace. Pro export dat je nutné nastavit výslednou polohu modelu, jenž odpovídá vzhledu pohledové mapy. Lze nastavit sklon pohledu, vzdálenost i výšku nad terénem (poloha pozorovatele). Bohužel možnosti exportu z aplikace jsou velice omezené. Nabízí sice možnost exportu jak ve 3D (VRML), tak i ve 2D, avšak značně nekvalitní. Export ve 2D, který je pro naše účely důležitý, je natolik nekvalitní, že je jeho další využití nemožné. Jako alternativní řešení proto bylo nutné využít snímání obrazovky. Toto řešení také není zdaleka ideální. Nabízí pouze omezené rozlišení výstupu, které odpovídá nastavenému rozlišení vlastní obrazovky. Platí proto, že čím vyšší rozlišení obrazovky, tím více detailů je patrných na výsledné pohledové mapě. Maximální rozlišení obrazovky, kterého se podařilo dosáhnout je 2048x1536. Výsledná exportovaná oblast pak může dosahovat rozlišení okolo 2000x900. Na základě tohoto faktu lze uvažovat pouze omezené přiblížení výsledného výstupu při tvorbě výsledné aplikace.

3.4 Tvorba znakového klíče

Stejně jako v případě analogových mapových výstupů je nutné i v případě prezentace v digitální podobě vytvořit znakový klíč. V případě webových interaktivních aplikací je situace ještě složitější. Je nutné uvažovat, že obsah interaktivní aplikace může nabývat řady stavů a proto může být znakových klíčů několik. Stejně tak je nutné, aby byla interaktivita jednotlivých prvků webové mapové aplikace na první pohled patrná a uživateli bylo zřejmé, že v případě interakce mu bude nabídnuta další informace. Většinou je proto interaktivita prvků znázorněna nepatrnou, nicméně na první pohled viditelnou změnou znaku. Je proto zřejmé, že znakový klíč je v případě interaktivních mapových aplikací mnohonásobně rozsáhlejší než v případě

analogových map. Vystává zde proto problém tvorby legendy. V běžné analogové mapě má každý znak přesně definovaný vzhled, který se v mapovém díle opakuje ve stejné podobě. U interaktivních aplikací není takové provedení praktické vzhledem k rozsahu legendy. Nicméně znakový klíč by měl být navržen natolik kvalitně, aby byla správná interpretace uživatelem možná i bez využití legendy. Pokud je tento stav splněn, postačí pouze legenda s vyjádřením typického stavu znaku, kdy jednotlivé stavy se od uvedeného v legendě příliš neliší.

Vzhledem k faktu, že v prostředí aplikace ArcScene nelze vytvářet kvalitní znakový klíč, bylo nutné pro tvorbu znakového klíče zvolit vhodnou aplikaci. Aplikací vhodných k tomuto účelu existuje celá řada. Je možné využít dostupných grafických editorů. Určujícím faktorem může být praktičnost pro navrhování znakového klíče a změn jednotlivých znaků, či podpora formátů vhodných pro další využití. Hlavní volbou je, zda využít vektorové nebo rastrové grafiky. Pro tvorbu výsledné aplikace byl použit rastrový formát dat, nicméně využití vektorové grafiky se v jistých fázích zpracování ukázalo být velice účelné. Pro návrh znakového klíče byla zvolena aplikace OCAD (OCAD 10 Profesional, 2009), jež je vyvíjena jako editor přímo zaměřený na tvorbu mapových produktů. Jeho výhodou je snadná editace znakového klíče a zpracování ve vektorovém formátu s možností exportu do řady dalších. Pro další úpravy rastrové grafiky byl použit program GIMP (GIMP, 2008), zaměřený na práci s rastrovými formáty.

Pro aplikaci OCAD byla z ArcScene exportovaná pohledová mapa použita především jako podklad. Z ArcScene byly důležité pouze vrstvy zástavby, vegetace a samotná 3D modelace



Obr. 1 Nekvalitní snímání obrazovky – 200% přiblížení

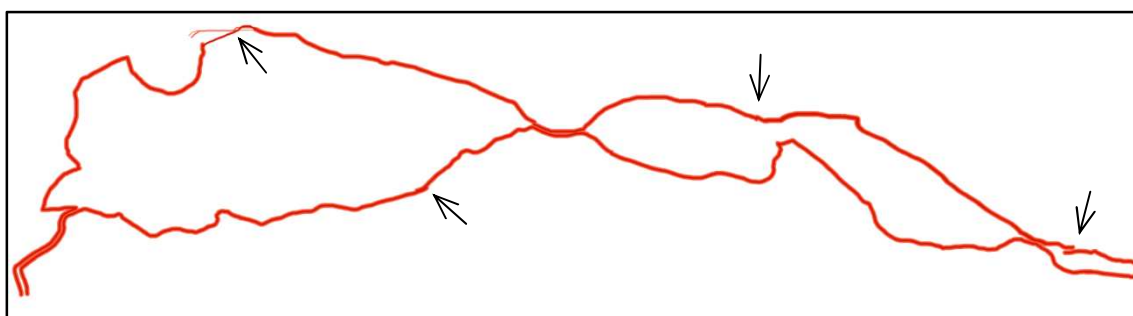
terénu. Ostatní vrstvy byly nepoužitelné jak z důvodů nemožnosti vytvořit ucházející znakový klíč v prostředí aplikace ArcScene, tak i díky nekvalitnímu snímání obrazovky (Obr. 1). Byly proto využity jako podklad pro novou vektorizaci v prostředí OCAD. Vektorizovány byly vrstvy komunikací, závodních tratí, vodních toků, vodních ploch a ostatních tratí Jizerské magistrály. Tento úkol s sebou nese problém vizualizace vrstev díky perspektivnímu zobrazení. Zaprvé v případě perspektivního zobrazení dochází se zvyšováním vzdálenosti od pozorovatele ke zmenšení velikosti objektů. Proto bylo v případě komunikací nutné přistoupit k využití dvojí tloušťky liniového znaku pro různé vzdálenosti. Druhým problémem je samotné využití 3D vizualizace terénu, kdy se jednotlivé vrstvy ztrácí za reliéfem nebo kdy jsou přerýty zástavbou, případně jinými vrstvami. Zde bylo nutné provést přesné zakreslení v co nejskutečnější podobě pro zachování maximální autentičnosti výstupu. V případě velkého přiblížení však může být tento jev patrný. Právě to je také další důvod pro využití pouze omezené možnosti přiblížení.

Pro novou vektorizaci v prostředí OCAD. Vektorizovány byly vrstvy komunikací, závodních tratí, vodních toků, vodních ploch a ostatních tratí Jizerské magistrály. Tento úkol s sebou nese problém vizualizace vrstev díky perspektivnímu zobrazení. Zaprvé v případě perspektivního zobrazení dochází se zvyšováním vzdálenosti od pozorovatele ke zmenšení velikosti objektů. Proto bylo v případě komunikací nutné přistoupit k využití dvojí tloušťky liniového znaku pro různé vzdálenosti. Druhým problémem je samotné využití 3D vizualizace terénu, kdy se jednotlivé vrstvy ztrácí za reliéfem nebo kdy jsou přerýty zástavbou, případně jinými vrstvami. Zde bylo nutné provést přesné zakreslení v co nejskutečnější podobě pro zachování maximální autentičnosti výstupu. V případě velkého přiblížení však může být tento jev patrný. Právě to je také další důvod pro využití pouze omezené možnosti přiblížení.

Dalšími prvky tvořenými v OCADu jsou bodové znaky. Pro vyjádření důležitých bodů na mapě bylo využito symbolických znaků, které by měly kvalitně vystihovat zaměření daného

symbolu. Znakový klíč by měl vždy splňovat podmínky přehlednosti, snadné čitelnosti a vyváženosti celého obsahu mapového pole. Vzhledem k využití při různém přiblížení obsahu bylo přistoupeno k mírným obměnám, především velikostním a v některých případech i grafickým. Ke každé z těchto verzí byla vytvořena druhá, jež byla využita pro znázornění interaktivních prvků v obsahu mapy. Zde bylo přistoupeno pouze k nepatrné změně jasu barvy znaku a u některých prvků změně velikosti. Stejný fakt se kromě bodových znaků týkal i závodních tratí, které byly stěžejním prvkem mapy a změna byla proto výraznější.

Z důvodu nevhodnosti výchozího formátu bylo nutné přistoupit k exportu do jiného formátu. Samotné mapy byly exportovány do bezztrátového rastrového formátu BMP a v prostředí aplikace GIMP dále upraveny. Jednalo se především o ořez na mapová pole s požadovanou velikostí a konverzi do formátu JPEG. Při použitím rozlišení je ztráta kvality výstupu v daném formátu v podstatě nezatelná oproti jiným, zato nabízí několikanásobně nižší výslednou velikost souboru, což je pro prezentaci na internetu důležité. V případě interaktivních prvků bylo nutné provést export jednotlivě pro každý jedinečný znak. Přestože v případě OCAD se jedná o export poměrně nekvalitní, jako nejlepší řešení se ukázalo využití formátu SVG. Export do SVG byl zásadní především v případě dalšího použití v prostředí Adobe Flex, kde je možné s tímto formátem dále pracovat. Bohužel nepříliš kvalitní export a zároveň málo tolerantní modul starající se o interpretaci v Adobe Flex (Obr. 2) zapříčinil, že bylo nutné provést manuální úpravy textového souboru formátu SVG. V případě dalšího zpracování v HTML se SVG ukázalo jako též velice přínosné z důvodu nejvyšší kvality výstupu a úpravy při dalším zpracování v aplikaci GIMP.



Obr. 2 Chybná interpretace SVG pomocí Adobe Flex při exportu z programu OCAD

3.5 Zpracování interaktivní aplikace

Pro zpracování výsledné aplikace a umístění na internet byly vybrány dvě poměrně rozdílné vývojářské platformy. První již tradiční a stále nejhojněji využívaná kombinace třech jazyků (X)HTML, CSS a JavaScript. A druhá velice aktuální a teprve se zabydlující v prostředí internetu, Adobe Flex. Cílem této kapitoly je nastínit způsob tvorby aplikací v jednotlivých prostředích a předvedení možností a úskalí jednotlivých platform.

3.5.1 XHTML, CSS, JavaScript

Přestože, jak se zdá, jazyk XHTML nebude v budoucnu dále vyvíjen, bylo jeho využití shledáno jako přínosnější oproti jazyku HTML. Jedná se o poměrně striktní definici značkovacího jazyka s pevně danými pravidly pro interpretaci, což může zaručit lépe správnou funkcionalitu ve více prohlížečích. Nicméně jazyky jsou téměř identické a jejich volba nemá na výsledky dané aplikace větší význam. Druhým použitým je stylový jazyk CSS2, sloužící především k zavedení vzhledu výsledné aplikace. Posledním do trojice je skriptovací jazyk JavaScript umožňující rozšíření jinak velice omezených funkcí předchozích dvou jazyků. Tyto tři jazyky vytváří poměrně mocnou platformu sloužící k vývoji webových stránek při interpretaci kódu na straně klienta. Velice důležitým faktem je spojení trojice jazyků. Existují tři možnosti propojení a jako nejpřínosnější bylo zvoleno to nejméně pevné s využitím připojení externích souborů. Je to vysoce účelné z hlediska možnosti změny nastavení ve všech html dokumentech naráz. Možnosti zápisu přímo k jednotlivým atributům bylo využito pouze v případě jedinečnosti informace pro daný html dokument.

Prvním krokem při tvorbě webových stránek je nutná volba *velikosti*. Ideální webové stránky umožňují automatickou změnu šíře dle rozlišení zobrazovacího zařízení na straně uživatele. V případě této aplikace je však využíváno mapových polí o přesně daných rozměrech a není proto možné takovou změnu uskutečnit. Byla proto zvolena šíře 1000 pixelů. Důvodem je možnost využití i na zařízeních s nižším rozlišením bez nutnosti rolování obsahu. Tato šíře odpovídá rozlišení 1024 při využití rolovacího sloupce na straně. S nižším rozlišením se dnes na osobních počítačích v podstatě nesetkáváme.

Mapová aplikace

Základním stavebním kamenem tvorby mapové aplikace s využitím HTML jsou *imagemapy*. Jedná se o využití speciálního elementu, který umožňuje v rámci obrázku vytvořit oblasti, které odkazují na další dokumenty. Na základě tohoto elementu je možné vytvořit princip přiblížení. Aplikace přiblížení je při využití tohoto řešení skoková. Jinou možností by bylo plynulé přiblížení zpracované pomocí skriptovacího jazyka JavaScript, nicméně vytvoření takové aplikace je velice náročné. Jak již bylo předesláno výše, není v případě vytvářené aplikace možné použít rozsáhlého přiblížení. Jako ideální bylo shledáno řešení tří úrovní přiblížení s malým krokem. Obsahem mapového pole v první úrovni je celé zájmové území. Jedná se tedy o stoprocentní přiblížení. V druhé úrovni bylo využito přibližně stopadesátiprocentního přiblížení. Zájmová oblast byla rozdělena na matici čtyř polí, kdy bylo dosaženo asi třetinového překrytu mapových výřezů. Praktické je to především z hlediska obsazení celých tratí kratších distancí v jednom mapovém poli. Poslední úroveň byla zvolena s přiblížením více než 300 %, kdy je zájmové území rozděleno na matici 3x3. Jednotlivé mapové výřezy se naopak téměř nepřekrývají a plynule na sebe navazují, což umožňuje plynulé prohlížení. Tematický obsah mapových polí je u všech úrovní přiblížení téměř totožný. Rozdíly lze nalézt pouze ve velikosti

některých znaků a v případě druhého a třetího stupně je obsah obohacen o místopisné názvy významných vrcholů, které by měly sloužit k lepší orientaci.

Z hlediska ovládání přiblížení je využití samotných *imagemap* nedostačující. *Ovládání zoomu* (přiblížení/oddálení) je řízeno pomocí dvou funkcí v jazyce JavaScript, jenž pomocí standardu DOM odkazují na elementy html dokumentu. Funkce jsou sestaveny tak, aby na základě číslování jednotlivých mapových výřezů byly schopné vyhodnotit, který následující výřez je požadován. Výsledný mapový výřez je pak předán jako relativní odkaz webovému prohlížeči, který výsledek zobrazí. Ovládání je připojeno ke dvojici tlačítek plus a mínus, pomocí nichž jsou obě funkce volány.

Podobným způsobem je zaručeno *ovládání polohy*, tedy výběr z jednotlivých mapových polí matice definující danou úroveň přiblížení. Funkčnost je zajištěna voláním funkcí posunu ve zvoleném směru pomocí čtyř směrových tlačítek v mapovém rámu. Zajištěno je i omezení posunu mimo zájmové území, kde neexistují mapové podklady.

```
function levo() {
    var lid = document.getElementById("pole");
    var la = lid.name;

    if (la=="1211" || la=="1221" || la=="1311" || la=="1321" || la=="1331" || la=="1111") {
        lb = la;
    } else {
        lb = --la;
    }
    var lhref = lb+".html";
    var levo = document.getElementById("levo");
    levo.setAttribute("href",lhref);
}
```

Obr. 3 Ukázka zdrojového kódu funkce volané pro posun vlevo

Důležitou součástí mapy jsou samotné *interaktivní znaky* v mapovém poli. Původním záměrem bylo zpracovat tyto prvky s využitím *imagemap*, jako je tomu v případě přiblížení. V takovém případě by bylo postačující vymezení aktivní oblasti, které by reagovaly na interakci uživatele. Toto řešení však skýtá řadu omezení. Aby byl uživatel schopen interaktivní prvky rozeznat, mělo by na základě interakce uživatele docházet ke změně stavu. Vzhledem k tomu, že možnosti této vývojové platformy jsou v tomto případě omezené, jediným řešením je načtení nového, mírně pozměněného překryvného obrázku. V takovém případě však dojde k překrytí aktivní oblasti a nedochází k žádané reakci na interakci uživatele. Bylo proto nutné přistoupit k umístění jednotlivých obrázků aktivních prvků na správné souřadnice v mapovém poli. Změny obsahu obrázku na základě interakce je dosaženo pomocí jednoduchého skriptu, který reaguje na najetí ukazatelem na aktivní prvek a následné opuštění aktivní oblasti. Dává tak jasně najevo, že se jedná o nestandardní prvek v mapě.

```

```

Obr. 4 Ukázka zdrojového zajišťujícího změnu symbolu

Všechny takto definované prvky v mapovém poli jsou definovány jako hypertextové odkazy odkazující na rozličná umístění. Dané řešení je využito u bodových znaků definujících důležitá místa na trati, jež se relativně odkazují na doplňující informace umístěné v rámci vlastní webové stránky. Dalšími interaktivními prvky jsou popisy obcí, které pomocí absolutního odkazování směřují uživatele na webové stránky obce. Stránky jsou v tomto případě odkazovány pomocí atributu *target*, který zajišťuje otevření v novém mapovém listu se zachováním stávající webové stránky. Posledním aktivním prvkem jsou samotné tratě závodu, které mění svůj vzhled ze všech prvků nejvýrazněji a odkazují na doplňující informace k celé trati.

Změna vzhledu závodních tratí je definována odlišně. Důvodem je, že interakce uživatele má zapříčinit hned dvě změny stavu. Jedná se o výše zmíněný vzhled trati, ale druhou funkcí je zobrazení profilu trati, který definuje, na kterém místě trati se v danou chvíli ukazatelem uživatel nachází. Efektu je dosaženo pomocí využití *imagemap*. Celá trať byla rozdělena na přesně definované úseky, které jsou ve všech mapových polích týkajících se dané tratě zvoleny shodně. Charakter daného úseku musí být vždy něčím typický. Dáno to může být buď morfologií terénu, anebo definováním úseků mezi důležitými body na trati závodu. K interakci dochází stejně jako v případě bodových znaků najetím, či odjetím z aktivní oblasti. Při nastalé události je volána jedna z příslušných funkcí, jež má nastarosti zobrazení příslušného překryvného obrázku. Pomocí standardu DOM je přistoupeno k souboru, který definuje pomocí kaskádových stylů vzhled a je nastavena vlastnost viditelnosti obrázku trati. Druhou funkcí je zobrazení profilu. Na základě umístění ukazatele na jeden z úseků trati je předána informace definující přesně daný úsek. Funkce pak na základě této definice zobrazí obrázek, jenž odpovídá zvýraznění stejného úseku trati v doprovodném profilu.

```
function trat(id) {
    profilcislo = id;

    var trat = document.getElementById("trat50");
    trat.style.display = "inline";

    var profil = document.getElementById("profil");
    profil.src = "../profily/" + profilcislo + ".png";
    profil.style.display = "inline";
}

function trat2() {
    var trat = document.getElementById("trat50");
    trat.style.display = "none";
}
```

Obr. 5 Ukázka zdrojového kódu funkce volané ovládající interaktivní profil

Doprovodné informace o tratích

Další stránkou, kterou webová aplikace obsahuje, jsou informace o jednotlivých tratích závodu. Jak bylo zmíněno výše, z jednotlivých mapových polí je odkazováno na informace důležité pro závodníky. Dochází k odkazu na přesné umístění právě v dané webové stránce. Veškeré informace jsou souhrnně sepsány pro jednotlivé tratě všech tří distancí. Na konkrétní umístění je odkazováno pomocí takzvaného *hash* znaku (#), který spolu se definicí atributu *id* v obsahu html dokumentu odkazuje na požadované umístění.

Stránka obsahuje jak popis jednotlivých míst, tak i služby, které jsou na daném místě poskytovány a které jsou znázorněny pomocí piktogramů. Pokud by symboly nebyly uživatelem jednoznačně rozeznány, zobrazí se nad nimi popisek se slovní definicí. Popřípadě lze využít tlačítka legendy, pro její vyvolání. Dalším obsahem je 3D profil znázorňující průběh dané trati, včetně přesného popisu závodu.

Fotogalerie

Fotogalerie je sestavena z matice o čtyřech sloupcích plovoucích elementů *div*. V rámci tohoto elementu je definován obsah ve formě popisu a samotné zmenšeniny fotografie. Daná fotografie je odkazem, takže po kliknutí odkazuje na svou originální velikost.

Samotné prohlížení fotografií je zpracováno pomocí speciálního skriptu. V tomto případě bylo využito volně dostupného skriptu GreyBox. Pro správnou funkci je nutné do obsahu stránky definovat umístění zdrojových skriptů a css souboru ovládajícího vzhled. Při definici samotného odkazu na fotografii je nutné využít atributu *rel*, čímž je definováno využití GreyBoxu, a při použití definice sjednocujícího názvu galerie vytvoří GreyBox slideshow ze všech fotografií. Pomocí atributu *title* je možné definovat i popis zobrazený u obrázku.

Legenda

Využití legendy je v rámci aplikace dvojí. Při zobrazení mapy jsou obsahem legendy piktogramy odpovídající symbolům použitým v mapovém poli. Podruhé je legenda využita při zpracování informací o trati. Jak bylo zmíněno výše, obsahem jsou piktogramy definující symboly obsažené na dané stránce. Zpracování je provedeno podobným způsobem jako v případě fotogalerie. V tomto případě je využito jiného volně dostupného skriptu LightBox. Princip zpracování do webové stránky je zcela shodný s předchozím případem. Výsledkem je však pouze jeden obrázek.

3.5.2 Adobe Flex

Druhým vybraným prostředím, ve kterém by mohla být aplikace kvalitně zpracována, je Adobe Flex. Základní princip zpracování je velice podobný zpracování předchozímu. Jedná se o spojení značkovacího jazyka MXML a skriptovacího jazyka ActionScript s možností přidání

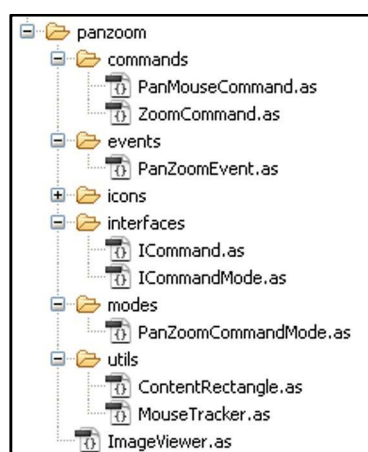
stylů pomocí nepatrně pozměněného jazyka CSS2. Na první pohled by se mohlo zdát spojení stejné, nicméně zde se jedná o speciální ucelenou vývojářskou platformu, nikoliv o spojení funkčnosti tří oddělených jazyků, jako je tomu v případě předchozím. Využití jednotlivých jazyků je navíc mírně odlišné. Celá filosofie prostředí je postavena na faktu, že moderní webová aplikace by se měla co nejvíce podobat a fungovat jako aplikace běžně spustitelná na osobním počítači.

Značkovací jazyk MXML slouží v daném případě pouze pro stanovení vzhledu stránky. Využívá k tomu dostupných předpřipravených komponent. Lze jmenovat například popis, tlačítko, textové pole, ale i panel, se kterým se lze běžně setkat v aplikacích a řada dalších i velice specifických a složitých prvků. K jednotlivým komponentám lze nastavit nepřehledné množství vlastností, ať již se jedná o běžné rozměry, či pozici, až po různé efekty jako rotace a vytrácení se v pozadí.

Důležitým jazykem je ale především ActionScript. Vždyť i veškeré prvky kódu vytvořené pomocí MXML by bylo možné přepsat pomocí jazyka ActionScript, nicméně jazyk MXML byl přidán právě z důvodu větší uživatelské přívětivosti. Veškeré stanovení funkcionality a akce, které se mají provádět, jsou definovány právě pomocí skriptovacího jazyka. Aplikaci je též možné obohatit o řadu efektů, čímž se stává mnohem zajímavější a přívětivější pro uživatele.

Adobe Flex je otevřená vývojářská platforma s možností zápisu v libovolném textovém editoru. Mnohem praktičtější zpracování se však nabízí pomocí speciálně vyvinuté vývojářské aplikace Flex Builder (Adobe Flex Builder 3, 2008), která již však není běžně volně dostupná. Aplikace obsahuje řadu panelů usnadňujících práci. Jedná se například o panel stromu složek obsahující všechny soubory otevřeného projektu, panel oznamující chyby v zápisu kódu, či panel s dostupnými předpřipravenými komponentami. Velice příjemným prvkem, z hlediska uživatelské přívětivosti, je možnost volby vývoje pomocí zdrojového kódu aplikace, či graficky, v designérském modu, který nabízí srovnatelnou funkčnost.

V tomto vývojovém prostředí byl proveden pouze nástin vývoje celé aplikace. Základním problémem bylo zpracování přiblížení jednotlivých mapových podkladů a následné umístění vrstev tratí a bodových znaků do mapového pole. Jako řešení bylo vybráno převzetí rozsáhlého zdrojového kódu volně dostupné aplikace (FIG|PAN Zoom)



Obr. 6 Třídy aplikace FIG|PAN Zoom

a jeho následná úprava a rozšíření podle potřeb vytvářené aplikace. Obsahem samotné převzaté aplikace jsou funkce plynulého přiblížení jak pomocí posuvníku, tak i pomocí kolečka myši. Dalšími obsaženými funkcemi jsou možnost plynulého posunu mapového pole a přizpůsobení pozice obsahu výřezu. Veškeré funkce vzorové aplikace byly rozděleny do několika tříd (Obr. 6), z nichž každá provádí zpracování rozdílných vlastností aplikace. Záměrem bylo na základě získaného zdrojového kódu vytvořit rozšíření pro načítání většího množství vrstev a spojení stávající aplikace pracující pouze s jedním podkladem pro práci se všemi.

Podařilo se však pouze nalézt dílčí řešení pro načtení jednotlivých vrstev na mapový podklad. Propojení s funkcemi přiblížení a pohybu se však ukázalo jako příliš náročné a pro zpracování by bylo nutné přepracovat celou koncepci stávající aplikace.

Proto byla v tomto prostředí vytvořena pouze ukázka demonstrující možnosti tvorby vzhledu pomocí jazyka MXML a funkcí převzaté aplikace. Základem koncepce úvodní obrazovky je panel s možností přepínání mezi jednotlivými tratěmi. Obsahem první záložky panelu je předpřipravená aplikace s možnostmi manipulace s mapou. Pro vytvoření ukázky (Příloha 1) zdrojového kódu pro změnu stavu (změna načtených kompot) celé aplikace byl vytvořen nástin fotogalerie. Mezi stavem základní obrazovky a fotogalerie je možné přepínat pomocí jednoduchého tlačítka.

3.6 Tvorba doplňkových objektů

Interaktivní profil

Jako základní podkladová data po zpracování profilu byla zvolena mapa Jizerských hor (Petr, 2005). Jedná se o kvalitní mapu obsahující nadmořské výšky množství bodů i znázornění kilometráže turistických, cyklistických i lyžařských tras. Data získaná z mapy byla zpracována pomocí tabulkového editoru Microsoft Office Excel a na jejich základě byl vytvořen graf odpovídající profilu trati. Po potřebných úpravách byl profil exportován pomocí snímače obrazovky. Další editace již byla prováděna v grafickém editoru GIMP. Profil byl rozmnožen na stejné množství snímků, jako bylo vytvořeno úseků trati na mapě a následně byl barevně upraven, aby vždy zvýrazňoval úsek odpovídající úseku na trati. Výsledný obrázek byl uložen pod specifickým názvem, jenž je volán pomocí funkce v jazyku JavaScript. Aby bylo možné umístit profil do prostoru mapového pole, bylo využito efektu průhlednosti a výsledek byl uložen ve formátu PNG, který alfa kanál průhlednosti podporuje a umožňuje umístění na webové stránky.

3D Profil

V případě 3D profilu probíhalo 3D zpracování obdobně. Data exportovaná z tabulkového editoru však byla pouze podkladem pro další tvorbu. Pro nabytí dojmu 3D profilu bylo nutné obrázek pozměnit. V aplikaci GIMP bylo proto provedeno otočení a transformace profilu. Výsledek byl použit jako podklad pro zpracování v aplikaci OCAD. V této aplikaci byl pak manuálně vytvořen celý 3D profil (Příloha 2). Jedná se o posunutí dvou profilů vycházejících z předešlého podkladu a spojení do tělesa. Z podkladu byly převzaty i kilometráž a převýšení profilu. Pro další zpřehlednění byly do 3D profilu jako orientační body přidány význačné body na trati, k nimž byla dodána i kilometráž a nadmořská výška. Pro účely znázornění služeb poskytovaných na trati během závodu byly v aplikaci vytvořeny charakteristické piktogramy. Shodné piktogramy jsou použity i na informační stránce týkající se jednotlivých tratí.

KAPITOLA 4

Zhodnocení prostředí

Tato kapitola by měla být objektivním zhodnocením jednotlivých vybraných vývojářských platforem na základě práce v prostředí během vývoje vlastní aplikace a zkušeností nabytých během zpracování, či z dalších doprovodných materiálů.

3.7 XHTML, CSS, JavaScript

Obecně platí, že zmíněná trojice je nejpoužívanější vývojářskou platformou v prostředí internetu. Příčinu lze hledat v několika faktorech. Především všechny tyto jazyky se během vývoje internetu vykrytalizovaly jako reakce na potřeby uživatelů. Znamená to, že byly u samotného zrodu internetu v dnešní podobě a právě díky těmto třem jazykům vypadá dnes internet právě takto. Řada uživatelů se právě v počátcích s těmito jazyky učila a vzhledem k tomu, že jejich potřebám dostačují, jsou stále využívány. Zásadní je také fakt, že ke všem třem jazykům je dostupné rozsáhlé množství literatury.

Obrovskou výhodou tohoto řešení je naprostá jednoduchost koncepce. Je to způsobeno právě historií, kdy celkový koncept HTML i CSS vznikl před již poměrně dlouhou dobou, v čase, kdy jednoduchý jazyk postačoval. Všechny značkovací jazyky, včetně (X)HTML jsou velice snadno pochopitelné a stejně tak je to se strukturou CSS, jazyka který najde využití ve stylizaci celé řady dalších jazyků. Právě to zaručuje kombinaci těchto dvou jazyků takovou oblibu i u laické veřejnosti. V případě JavaScriptu je již situace zcela jiná. Jazyk patří do skupiny objektově orientovaných jazyků, což samo o sobě znamená, že nedosahuje zdaleka takové jednoduchosti, jako je tomu u předchozích dvou. Právě proto není jeho obliba laickou veřejností tak rozsáhlá a není tak často používán. V řadě webových stránek není využití ani potřebné a může to být často i na škodu. Samotná koncepce není v zásadě složitá, a proto jednoduché skripty nejsou problémem, ale rozsáhlejší aplikace již vyžadují jiný typ uvažování než v případě předchozích dvou jazyků. Často se proto lze setkat s již předpřipravenými řešeními, jež nabízí požadovanou funkcionalitu, takzvanými API. Právě takovýchto řešení bylo využito v případě tvorby fotogalerie vytvářené aplikace. Stejně tak by bylo možné využít takové řešení i v případě zoomu pohledové mapy. Nicméně zcela jistě by byla nutná řada úprav dle vlastních potřeb, a jak se ukázalo i v případě zpracování v prostředí Flex, jsou takové úpravy velice náročná na programátorské schopnosti vývojáře.

U rozsáhlejších a složitějších webových aplikací bývá často vývoj značně ztížen. Právě díky dlouhodobému vývoji dochází často k problémům s kompatibilitou kódu. Webové prohlížeče

často dosahují rozdílné míry implementace funkcionality jednotlivých jazyků a navíc často interpretují kód rozdílně. Proto nemusí využití správně napsaného kódu znamenat vždy vytvoření funkční aplikace. Nelze proto využívat striktně nejnovějších a nejkvalitnějších řešení, ale je nutné tvořit kompromisy ve prospěch všech uživatelů, nehledě na fakt, jaký z webových prohlížečů používají. To však značně brzdí rozvoj samotné platformy a navíc ztěžuje práci vývojářům aplikací. Vytvoření kódu kompatibilního se všemi prohlížeči není objektivně možné. Otázkou však často bývá, kde stanovit hranici, kdy přestává být vývoj pro daný prohlížeč zajímavý. Například v případě vytvořené aplikace by se dalo polemizovat, zda je odladění dostatečné. Aplikace je funkční ve všech nových verzích webových prohlížečů, nicméně v dávno překonaném, ale stále hojně využívaném prohlížeči Internet Explorer 6 nevypadá příliš dobře. Je pak otázkou, zda je to u aplikace, jež by měla být přístupná široké veřejnosti chybou, či ne. Na funkčnosti aplikace ani v tomto případě neztrácí, ale problém je zde především s omezenou podporou vlastností formátu PNG a pozicováním prvků webové aplikace pomocí jazyka CSS2.

Naštěstí díky spojení těchto tří jazyků lze problematiku zpracovat pomocí řady různých řešení. Často však bývá problematické nalézt řešení, které by bylo funkční v naprosté většině prohlížečů.

```
//verze 1
var trat = document.getElementById("trat50");
    trat.style.display = "inline";

var profil = document.getElementById("profil");
    profil.src = "../profily/" + profilcislo + ".png";
    profil.style.display = "inline";

//verze 2      nefunkční v Internet Exploreru
window.document.trat50.style.display = 'inline';
window.document.profil.style.display = 'inline';
window.document.profil.src = "../profily/" + profilcislo + ".png";
```

Obr. 7 Ukázka rozdílného řešení skriptu téže funkce

Přestože jsou možnosti této platformy rozsáhlé, zdaleka nejsou nekonečné a lze narazit na řadu omezení. Řešením může být přibrání dalšího jazyka, například skriptovací jazyk na straně serveru. Nejlépe na tom platforma není ani v případě efektivity zápisu kódu. Externí připojení souborů s kaskádovými styly a skripty značně zjednodušuje práci, avšak i tak není efektivita zpracování na nejvyšší úrovni. Příkladem může být i vytvářená aplikace, kdy bylo nutné vytvářet z velké části naprosto identické html dokumenty. Naprosto nevhodné se z tohoto hlediska ukázalo pozicování jednotlivých interaktivních prvků v rámci mapových polí. Řešením by snad mohlo být předání rozsáhlejší části funkcí JavaScriptu, či vhodného využití skriptovacího jazyka na straně serveru, například formou generování pomocí PHP skriptů. Právě hlediska efektivity tvorby, nároků na vývojáře i uživatele, či vytížení serveru zpracováním informací by měly být důvodem pro hledání alternativních způsobů řešení, nejvhodnějších pro danou aplikaci.

3.8 Adobe Flex

Jako alternativní možnost zpracování byla vybrána právě platforma Adobe Flex. Jedná se o zcela novou vývojářskou platformu, která je zaměřena trochu odlišným směrem, než je tomu v případě předchozím. Snaží se o zcela nové nasměrování vývoje aplikací na internetu. Jednoznačným kladem je naprostá svoboda rozvoje, nehledě na starší vývojové verze. Nabízí proto plně funkční, zcela nové a neotřelé postupy řešení, což zásadně rozšiřuje možnosti aplikací na internetu i trh nabízených služeb.

Hlavním stavebním kamenem všech aplikací vytvářených v tomto prostředí je interakce uživatele. Nejedná se pouze o rozšíření funkcionality, jako je tomu v předchozím případě, ale interaktivita je podstatou fungování aplikací vyvíjených ve Flexu. Nabízí proto nové možnosti, jichž by se dalo pomocí starých technik jen těžko dosáhnout. Pro představu právě díky tomuto novému konceptu lze nyní na internetu nalézt aplikace podobné kancelářským balíčkům, či grafickým editorům. V takovém případě se již v podstatě chce říci programy, vždyť funkcemi se aplikacím běžně spustitelným na osobním PC téměř rovnají.

K vytváření aplikací v podstatě využívá pouze jednoho jazyka a to scriptovacího nebo chcete-li programovacího. Jazyk MXML se v tomto případě zdá být spíše úlitbou uživatelům pro zjednodušení tvorby rozhraní, byť se při tvorbě aplikací jedná pouze o okrajovou záležitost. Pozitivem je, že se tato platforma neomezuje pouze na tyto dva jazyky. V rámci tvorby aplikací lze například využít i data zpracovaná ve formátu XML, propojení se skriptovacími jazyky na straně serveru, či rozšíření o programovací jazyky jako C#.

Jak již bylo řečeno, jedná se o velice progresivní novou platformu s neotřelými postupy. Zde lze nalézt nezanedbatelný rozdíl oproti předcházející platformě. Mohlo by se zdát, že se jedná o pouhé pozitivum, avšak jak se ukázalo i v případě tvorby této práce, zdaleka tomu tak není. Postupný vývoj znamená postupné rozšiřování funkcionality dle vzrůstajících požadavků uživatelů internetu, což s sebou přináší i postupný nárůst náročnosti vývoje aplikací. Zatím co v případě postupného vývoje zůstávají i již překonané postupy v rámci vývoje zachovány, v případě platformy zcela nové tomu tak není. Vývoj proto již není zdaleka tak jednoduchý a dostupný laické veřejnosti, jako je tomu v případě kombinace jazyků okolo HTML. Dobře patrné je to při řešení implementací funkcí v rámci vytvářené aplikace. Zatím co aplikace vystavěná na HTML a JavaScriptu nabízí volbu mezi jednoduchým, byť omezeným řešením v rámci HTML nebo složitější a efektnější verzí v rámci JavaScriptu, Adobe Flex takovou alternativu nenabízí. I zde se proto ukazuje pravda, že v dnešní době je již nutná mezioborová spolupráce odborníků na danou problematiku pro společné dosažení výsledku. Zpracování je pak mnohem efektivnější a výsledek může být podstatně hodnotnější, než v případě zpracování jednou osobou s omezenými možnostmi znalostí.

Jak se bude situace v případě této vývojářské platformy vyvíjet, je otázkou. Stále se jedná o nový počín. Zatím se však zdá, že se jedná o rok správným směrem. K přílišnému přiblížení laické veřejnosti například formou jednoduchých příruček, jako tomu je v případě například HTML, zřejmě nedojde. Zatím je ale problém i s dostupností knižní odborné literatury, což se, doufejme, v budoucnu změní.

KAPITOLA 5

Závěr

Prvotním účelem práce bylo vytvoření funkční pohledové mapy prezentované na internetu široké veřejnosti. Práce se zabývá problematikou tvorby pohledových map. Shrnuje principy, na kterých je tvorba pohledových map vystavěna a zabývá se možnostmi tvorby pohledových map pomocí výpočetní techniky. Dalším řešeným tématem jsou možnosti šíření map na internetu, především co se týká možného využití formátů a jejich funkčních možností.

Velice důležitým tématem je možnost volby platform, které slouží k vývoji webových aplikací. Možnosti jsou v tomto případě velice široké, tato práce se však zaměřuje především na platformy vhodné pro využití k tvorbě cílové mapové aplikace. Právě zde se ukazuje, jak důležité jsou požadavky na výslednou aplikaci a jak vysoce selektivní mohou tyto požadavky být. Nelze totiž jednoznačně definovat nejkvalitnější prostředí. Důležitými faktory ovlivňujícími výběr jsou kromě schopností daného prostředí i schopnosti vývojáře a pokud budeme uvažovat tržní prostředí, velkým tlakem je i maximální dostupnost uživatelům. Výsledné požadavky tak často mohou diskvalifikovat i technicky nejkvalitnější řešení.

Pro zpracování aplikace byly v rámci práce vybrány dvě rozdílné vývojářské platformy, na kterých se projevilo, jak důležité stanovení požadavků může být. První volba padla na kombinaci jazyků (X)HTML, CSS a JavaScript, jež jsou základními stavebními kameny dnešního internetu. Druhým vybraným kandidátem se stal Adobe Flex. Zcela nové prostředí, na něhož budoucnost stále čeká. První prostředí lze definovat jako jednoduše ovladatelné s možností volby jak jednoduchých, tak i složitých řešení, ovšem za cenu řady problémů se správnou interpretací kódu prohlížeči. Prostedí Adobe Flex lze naopak definovat jako vyspělé konsistentní prostředí značně rozšiřující možnosti dnešního internetu s bezproblémovou interpretací výsledku prohlížeči, ale vykoupené vyššími nároky na schopnosti vývojáře a nutností instalace rozšiřujícího pluginu. Zde je dobře patrné, jak každá volba má svá pro i proti a výběr záleží především na schopnostech a potřebách pro danou aplikaci.

V případě tvorby výsledné aplikace se za pomoci platformy vystavěné okolo jazyka XHTML podařilo vytvořit komplexní aplikaci prezentovatelnou v prostředí internetu. Otázkou však je, nakolik by bylo takové řešení reprezentativní a v komerční sféře. Pravděpodobně příliš velký tržní potenciál nenabízí. V případě Adobe Flex je tržní potenciál naopak obrovský, nicméně i v průběhu této práce se ukázalo, že kvalitní zpracování na profesionální úrovni již dnes nemůže být dílem jednoho člověka a je nutné využít schopností více odborníků.

Výsledná aplikace je dostupná na webové adrese <http://www.jizerska50.eu>

SEZNAM ZDROJŮ INFORMACÍ

TIŠTĚNÉ ZDROJE

- BELKOVÁ, E. 1994. *Tvorba pohledových map pomocí počítače* [rukopis]. Praha, 1994. 113 s. Diplomová práce na PřF UK.
- BERNARD, B. 2008. Adobe Flex – co je a co není. *Interval.cz* [online]. 2008 [cit. 2009-07-12]. Dostupné z URL: < <http://interval.cz/clanky/adobe-flex-co-je-a-co-neni/>>.
- ČAPEK, R. ; MIŠKOVSKÝ, M. ; MUCHA, L. 1992. *Geografická kartografie*. 1. vyd. Praha : Státní pedagogické nakladatelství, 1992. 373 s. ISBN: 80-04-25153-6.
- ČASAROVÁ, E. 2008. *Multimediální možnosti digitálních dopravních map* [rukopis]. Praha, 2008. 72 s. Diplomová práce na PřF UK.
- EČER, P. 2007. *Jazyk X3D a jeho aplikace v geoinformaticce* [rukopis]. Praha, 2007. 63 s. Bakalářská práce na PřF UK.
- FRANKLIN, D. 2003. *Macromedia Flash MX : kompletní průvodce*. 1. vyd. Brno : Computer Press, 2003. 846 s. ISBN 80-7226-831-7.
- HÁJEK, F. 2008. *Tvorba digitálních dynamických pohledových map* [rukopis]. Praha, 2008. 40 s. Bakalářská práce na PřF UK.
- JEDLIČKA, J. 2008. *Využití SVG pro tvorbu interaktivních internetových map* [rukopis]. Praha, 2008. 116s. Diplomová práce na PřF UK.
- KNÍŽOVÁ, L. 2006. *Využití VRML pro tvorbu webového Informačního systému Jizerské magistrály* [rukopis]. Praha, 2006. 105 s. Diplomová práce na PřF UK.
- KOLÁŘ, J. 2003. *Geografické informační systémy*. 2. přeprac. vyd. Praha : Vydavatelství ČVUT, 2003. 161 s. ISBN: 80-01-02687-6.
- MONMONIER, M. 2000. *Proč mapy lžou*. 1. vyd. Praha : Computer Press, 2000. 221 s. ISBN: 80-7226-238-6.
- RAUCH, T. 2008. *Interaktivní mapa golfových hřišť v ČR* [rukopis]. Praha, 2008. 36 s. Bakalářská práce na PřF UK.
- SNÍŽEK, M. 2004. *CSS pro zelenáče*. 1. vyd. Praha : Neocortex, 2004. 295 s. ISBN 80-86330-14-1.

SUERING, S. 2008. *JavaScript krok za krokem*. 1. vyd. Brno : Computer Press, 2008. 336 s. ISBN: 978-80-251-2241-9.

ŠEMBEROVÁ, P. 1989. *Pohledové mapy* [rukopis]. Praha, 1989. 96 s. Diplomová práce na PřF UK.

ŠTYCH, P. a kol. 2008. *Vybrané funkce geoinformačních systémů*. Praha, 2008. 177 s. PřF UK, Katedra aplikované geoinformatiky a kartografie.

VOŽENÍLEK, V. 2001. *Aplikovaná kartografie I : tematické mapy*. 2. vyd. Olomouc : Univerzita Palackého, 2001, 187 s. ISBN: 80-244-0270-X.

INTERNETOVÉ ZDROJE

ADOBE DEVELOPER CONNECTION: THE PAN/ZOOM COMPONENT [online]. Adobe Systems Incorporated, © 2009. [cit. 2009. 06. 20]. Dostupné z URL: <http://www.adobe.com/devnet/flex/samples/fig_panzoom/>.

ADOBE DEVELOPER CONNECTION: FLEX IN WEEK [online]. Adobe Systems Incorporated, © 2009. [cit. 2009. 06. 20]. Dostupné z URL: <<http://www.adobe.com/devnet/flex/videotraining/>>.

ADOBE FLEX 3 HELP [online]. Adobe Systems Incorporated, © 2009. [cit. 2009. 06. 20]. Dostupné z URL: <<http://livedocs.adobe.com/flex/3/html/index.html>>.

BAYER, T. 2008. *Algoritmy v digitální kartografii : Digitální modely terénu* [online]. Praha, 2008. [cit. 2009-07-03]. Dostupné z URL: <<http://www.natur.cuni.cz/~bayertom/IM/idm5.pdf>>.

ČERBA, O. 2006. *Mapy na internetu* [online]. Plzeň, 2006. [cit. 2009-06-24]. 10 s. Dostupné z URL: <www.gis.zcu.cz/studium/pok/Materialy/dig_mapy.pdf>.

ČERBA, O. 2006. *SVG v kartografii* [online]. Plzeň, 2006. [cit. 2009-07-08]. Dostupné z URL: <http://geoinformatics.fsv.cvut.cz/wiki/index.php/SVG_v_kartografii>.

ČERBA, O. 2007. *Úvod do počítačové kartografie* [online]. Plzeň, 2007. [cit. 2009-06-24]. 12 s. Dostupné z URL: <<http://www.gis.zcu.cz/studium/pok/Materialy/uvod.pdf>>.

DIBAVOD 2007. [database online]. Praha : Výzkumný ústav vodohospodářský T.G.Masaryka, 2007 [cit. 2008-12-10]. Dostupné z URL: <<http://www.vuv.cz/oddeleni-gis/>>.

GOOGLE MAPS. 2009. [online]. © 2009. [cit. 2009-08-18]. Dostupné z URL: <<http://maps.google.com/>>.

GRIMMICH, Š. 2008. *XHTML. Tvorba-webu.cz* [online]. 2008. [cit. 2009-04-12]. Dostupné z URL: <<http://www.tvorba-webu.cz/xhtml/>>.

JIZERSKÁ 50 [online]. [cit. 2009-06-24]. Dostupné z URL: <<http://www.jiz50.cz/>>.

- KNÍŽOVÁ, L. 2003. *Animace nad digitálním modelem terénu* [online]. Praha, 2003 [cit. 2009-06-24]. 22 s. Ročníková práce na PřF UK. Dostupné z URL: <<http://kgi.wz.cz/GIS/dmt.pdf>>.
- KOLLINGER, M. 2004. *Návrh a implementace finančně nenáročného způsobu publikace geografických dat v síti Internet* [online]. Plzeň, 2004 [cit. 2009-06-24]. 85 s. Diplomová práce na ZČU v Plzni. Dostupné z URL:<http://www.kma.zcu.cz/DATA/zaverecne_prace/2004/Kollinger__Navrh_a_implementatione_financene_nenarocneho_zpusobu_publicace_geografickych_dat_v_siti_internet__DP.pdf>.
- KOZÁKOVÁ, M. 2005. *Kartografické hodnocení webových map* [online]. Olomouc, 2005 [cit. 2009-06-24]. 8 s. Dostupné z URL: <http://gis.vsb.cz/GIS_Ostrava/GIS_Ova_2005/Sbornik/cz/Referaty/kozakova.pdf>.
- KREJČÍ, R. 2004. Encyklopedie publikačních formátů : SVG. *Grafika on-line* [online]. 2004 [cit. 2009-07-08]. Dostupné z URL: <<http://www.grafika.cz/art/polygrafie/encsvg.html>>.
- KRÁTKÝ, M. 2004. *Mapy na internetu* [online]. Praha, 2004 [2009-06-24]. 51 s. Ročníková práce na PřF UK. Dostupné z URL:<<http://kratas.borec.cz/down/mapy.pdf>>.
- KUNC, P. 2005. *Rich Media Applications : Představení technologií* [online]. 2005 [2009-07-12]. Dostupné z URL:<<http://www.beneta.cz/cs/publikujeme/clanky-a-prezentace/Rich%20Internet%20Applications.pdf>>.
- MICROSTATION DIGITAL TERRAIN MODELING* [online]. [cit. 2009-07-26]. Dostupné z URL: <<http://www.arch.virginia.edu/arch541/Handouts/dtmdoc/dtm.html>>.
- PETR, J. 2005. *Jizerské hory 1 : 25000 : turistická, cykloturistická, lyžařská mapa*. 1. vyd. Praha, 2005 [2008-12-01]. Diplomová práce na PřF UK. Dostupné z URL: <http://www.horydoly.cz/files/jizerky_petr_mapa.jpg>.
- POLÁČKOVÁ, J. 2009. *Podoba a struktura kvalifikačních prací na katedře* [online]. Praha, 2008 [cit. 2009-03-15]. 55 s. Dostupné z URL: <<http://www.natur.cuni.cz/gis>>. Materiál vytvořený J. D. Bláhou pro studenty, kteří píšou svou kvalifikační práci na katedře aplikované geoinformatiky a kartografie na Přírodovědecké fakultě Univerzity Karlovy.
- PROGRAMMING ACTIONSCRIPT 3.0* [online]. Adobe Systems Incorporated, © 2009. [cit. 2009. 06. 20]. Dostupné z URL: <http://livedocs.adobe.com/flex/3/html/help.html?content=Part6_ProgAS_1.html>.
- ŘÍHA, J. 2007. *Distribuce dat pomocí webových služeb* [online]. Praha, 2007. Bakalářská práce na FSV ČVUT. Dostupné na URL: <<http://geo2.fsv.cvut.cz/~soukup/bkl/riha/wms2.html>>.
- SNÍŽEK, M. 2003. XHTML - seriál. *Interval.cz* [online]. © 2002 - 2003 [cit. 2009-04-12]. Dostupné z URL: <<http://interval.cz/webdesign/xhtml/>>.

TIŠNOVSKÝ, P. 2008. Seriál grafické formáty. *Root.cz* [online]. © 2006 - 2008 [cit. 2009-07-08]. Dostupné z URL: <<http://www.root.cz/serialy/graficke-formaty/>>. Relevantní články týkající se SVG, VRML a X3D.

WIKIPEDIA. *World Wide Web* [online]. [cit. 2009-07-02]. Dostupné z URL: <http://cs.wikipedia.org/wiki/World_Wide_Web>.

PROGRAMOVÉ VYBAVENÍ

ARCGIS DESKTOP [CD-ROM]. Ver. 9.3. ESRI Inc., © 2008.

KIMBALL, S. ; MATTIS, P. 2008. GIMP [online]. Ver. 2.6.4. © 2008. Aplikace je šířena pod licencí GNU jako svobodný software.

ADOBE FLEX BUILDER 3 [online]. Ver. 3.0.2. Adobe Systems Incorporated, © 2008.

Aplikace je k dispozici na 30 dní zdarma. Studentům je poskytována neomezená licence.

OCAD 10 PROFESIONAL [online]. Ver. 10.0. OCAD AG., © 2008. Aplikace je na 21 dní poskytována jako trial verze.

SEZNAM PŘÍLOH

Příloha 1 Ukázka zdrojového kódu v Adobe Flex

Příloha 2 Ukázka 3D profilu trati Jizerské 30

Příloha 3 DVD s elektronickou verzí práce

PŘÍLOHA 1 : Ukázka zdrojového kódu v Adobe Flex

```
<?xml version="1.0" encoding="utf-8"?>
<mx:Application
  xmlns:mx="http://www.adobe.com/2006/mxml"
  xmlns:ns1="com.adobe.wheelerstreet.fig.panzoom.*"
  layout="absolute" horizontalAlign="center"
  >
  <mx:Script>
    <![CDATA[
      import com.adobe.wheelerstreet.fig.panzoom.ImageViewer;
      import com.adobe.wheelerstreet.fig.panzoom.utils.ContentRectangle;
      [Bindable] public var _loadMap:Boolean = false;
      [Bindable] private var _imageUrl:String = "images/pohled2.jpg";
      [Embed (source="images/502.svg")]
      [Bindable] private var j50:Class;
    ]]>
  </mx:Script>
  <mx:Canvas label="Jizerská 30" verticalScrollPolicy="off" horizontalScrollPolicy="off"
    left="5" right="5" top="5" bottom="5" width="2000" height="1000" id="canvas1">
    <ns1:ImageViewer
      id="imageView"
      imageUrl="{ _imageUrl }"
      bitmapScaleFactorMax="2" bitmapScaleFactorMin="1"
      width="100%" height="100%" x="0" y="0"
    />
    <mx:VBox left="10" top="10" width="20">
      <mx:Button
        styleName="ZoomOutButton" tooltip="Zoom In"
        click="{imageView.zoom('in');}"
      />
      <mx:HBox width="20">
        <mx:VSlider
          minimum="{ imageView.bitmapScaleFactorMin }"
          maximum="{ imageView.bitmapScaleFactorMax }"
          value ="{ imageView.bitmapScaleFactor }"
          change ="{ imageView.setZoom(VSlider(event.currentTarget).value) }"
          snapInterval=".02"
          liveDragging="true"
        />
      </mx:HBox>
      <mx:Button
        styleName="ZoomInButton" tooltip="Zoom Out"
        click="{ imageView.zoom('out'); }"
      />
    </mx:VBox>
  </mx:Canvas>
</mx:Application>
```

PŘÍLOHA 2 : Ukázka 3D profilu trati Jizerské 30

