

PRAHA4D

SW NÁSTROJE PRO TVORBU VIRTUÁLNÍHO MODELU MĚSTA

Milan Šimůnek

milan.simunek@praha4D.net

ABSTRAKT:

Článek představuje speciálně vytvořené SW nástroje pro projekt virtuálního modelu Prahy zaměřeného na historii (www.Praha4D.net). Popsány jsou základní etapy předzpracování dat a vytváření vlastního 3D modelu. Pro každou etapu jsou uvedeny vyvinuté SW aplikace, které ji podporují. V rámci detailnějšího popisu jednotlivých etap je naznačena síť navazujících dílčích úloh, které automaticky transformují 2D data a poskytují je jako vstupy dalším dílčím úlohám. Součástí prezentace budou ukázky transformovaných dat i vlastního 3D modelu.

KLÍČOVÁ SLOVA:

virtuální realita, 3D modelování, SW nástroje, implementace

ABSTRACT:

This paper describes tailor-made SW tools for the Praha4D project – history of the city of Prague in virtual reality (www.Praha4D.net). Presented are main phases of data preprocessing, data transformation and building of the 3D model. Supporting applications for each of phases are mentioned. A net of 2D transformations for data preparation is described. Examples of two groups of transformations – buildings and roads are discussed. More examples of data transformations and the final 3D model will be presented at conference.

1. Úvod

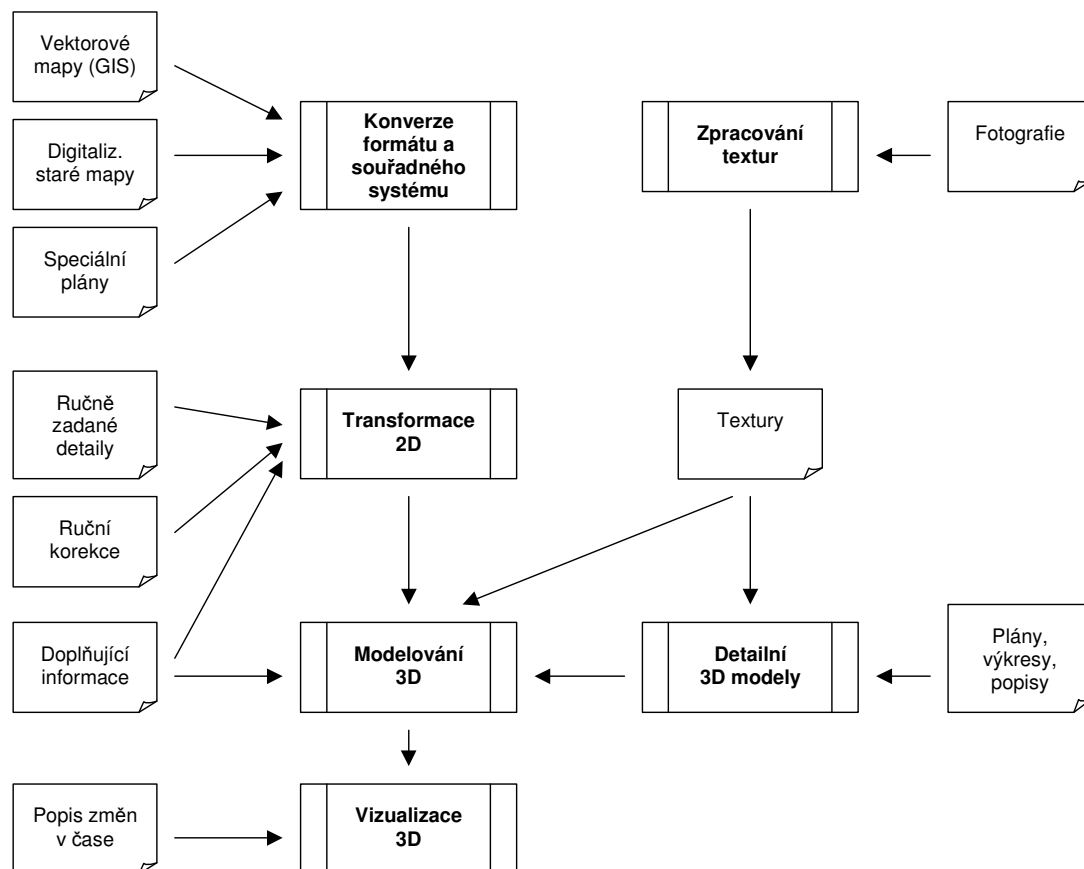
Cílem projektu Praha4D je vytvořit virtuální počítačový model Prahy. Chci zobrazit jak Prahu současnou, tak i její dřívější historické podoby. Velmi odvážnou vizí je zobrazit kompletní historii Prahy od doby Karla IV. po současnost a to včetně možnosti volby ročního období, denní doby a případně i aktuálního počasí. Kromě stavebního vývoje se postupně snažím pokrýt i reálie dané doby – styl oblékání, dobové dopravní prostředky, typické povrchy cest a silnic, osvětlení atp. Ukázky z aktuální verze modelu jsou na adrese www.Praha4D.net.



Obr. 1 – Ukázky z aplikace

2. Vytváření virtuálního modelu

Příprava virtuálního modelu je poměrně rozsáhlý proces. Na obrázku 2 je znázorněno základní funkční schéma.



Obr. 2 – Základní funkční schéma

Dále popsané aplikace jsou vyvíjeny v jazyce C++. Při implementaci byly použity SW knihovny *OpenGL* [4], *OpenSceneGraph* [5] a knihovna *Triangle* [6] pro triangulaci uzavřených 2D oblastí.

3. Vstupy

Základní množinou vstupů jsou **mapová data**:

- parcely, půdorysy domů, hrany chodníků, parkové cesty...
- osy kolejí – železnice, tramvaj, lanovka, koňka
- pozice drobných objektů – lampy, stromy...
- výškopis (TIN)

Data pocházejí z různých zdrojů (včetně vlastního terénního průzkumu a měření), jsou uložena v různých datových formátech (SHP, DGN, VYK, DBF, BMP, JPG, textové soubory...) a kvalita jejich zpracování se liší. Kromě vektorových dat z GISů je třeba pracovat i s rastrovými daty zejména naskenovaných historických plánů. Pro vektorizaci rastrových dat je třeba mít vhodné nástroje umožňující najednou práci s rastrovými obrázky a vektorovými daty ve vrstvách. Jako zvláštní zdroj dat je třeba zpracovávat účelové specializované plány, jako jsou plány kolejíšť nebo opevnění. Mapová data je nutné doplnit o **ručně zadané detaily**, které se v mapách nevyskytují:

- detaily zjišťované z ortofotomap
- pozice drobných objektů nezanesených v mapách – lavičky, drobná zeleň, dopravní značky, popelnice ...

a také o **ruční korekce** opravující chyby a nepřesnosti v převzatých mapových podkladech. Ukazuje se totiž, že sebelepší mapové podklady nejsou úplně přesné a obsahují určité procento chyb vzniklých jak při mapování, tak i při následné digitalizaci. Jde o pochopitelný fakt, protože současné aplikace GIS vznikají zejména za účelem vizuálního zprostředkování informace pro člověka a při jejich tvorbě se obvykle nepočítá s následným automatickým zpracováním. Ruční korekce jsou proto nezbytnou součástí vytváření virtuálního modelu. Jejich důležitou část tvoří i případné úpravy terénu.

Doplňující informace jsou data nemapového charakteru pomáhající při transformacích a hlavně při modelování:

- výšky domů, fasády, typy a výšky střech
- informace o použitém povrch ploch/cest – tráva, asphalt, štěrk, písek...
- výšky a druhy stromů/keřů a řada dalších

Do doplňujících informací patří ale například i popisy tramvajových, autobusových a železničních zastávek, aby bylo později možné sestavit jízdní řády.

Fotografie jsou nutné pro vytváření textur použitých na fasády, povrchy ploch/cest, stromy... a také na textury 3D modelů. Modely jsou vytvářeny (nejlépe na základě **plánů, výkresů a popisů**) pro významné budovy (např. Hlavní nádraží, Národní muzeum), pro dopravní prostředky a pro další objekty – lavičky, čekárny na zastávkách, vstupy do metra apod. Pro modely dopravních prostředků je třeba zanést i další údaje týkající se hmotnosti, výkonu motoru, rozvoru, typu brzd atp. **Popis změn v čase** obsahuje v současné době zejména jízdní řády dopravních prostředků (viz dále) a do budoucna i například změny počasí a střídání ročních období nebo dne a noci.

Aplikace GisConv

Pro základní transformace vstupních (geografických) dat byla vyvinuta aplikace *GisConv*. Ta obsahuje kromě jiného i funkce pro převod mezi různými souřadnými systémy (implementovány na základě informací v [3]).

Pro další výpočty nad vektorovými daty se ukázalo jako nezbytné zavést vlastní počátek souřadného systému. Číselné datové typy *float* a *double* mají pevný počet platných míst a z toho vyplývající zaokrouhlovací chyby. Při výpočtech nad souřadnicemi pracujeme pouze s relativními vzdálenostmi mezi dvěma body. Přitom hraje podstatnou roli, pohybují-li se absolutní hodnoty v rozmezí 5557324,320 až 5547957,255 nebo v rozmezí 0 až 368. Zavedením vhodného počátku souřadného systému můžeme ušetřit několik počátečních míst na začátku čísel a zvýšit tím i přesnost výpočtů. Pro účely projektu *Praha4D* byl počátek zvolen na Staroměstském náměstí.

Aplikace Vec – zobrazení 2D dat

Pro vizuální kontrolu vstupů a také pro sledování průběhů transformací dat (viz dále) je nezbytný jednoduchý prohlížeč 2D vektorových dat, který umožňuje i základní funkce překrývání bitmapových vrstev. Po zvážení možností existujících prohlížečů jsem se rozhodl přistoupit k implementaci vlastního prohlížeče, do kterého byly implementovány i editační funkce podle potřeb jednotlivých kroků přípravy dat.

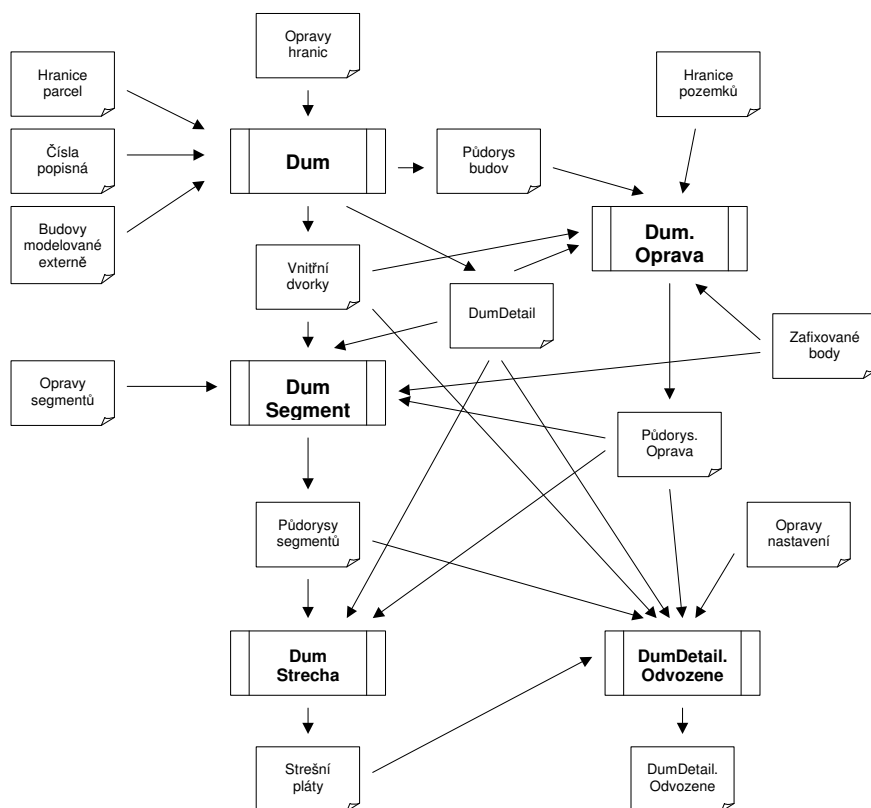
Aplikace *Vec* zobrazuje vektorová data ve vrstvách (odlišených pomocí barev). Speciální (needitovatelnou) vrstvou mohou být rastrová data, která slouží zejména při ruční vektorizaci starých map, ale i při zakreslování objektů menších rozměrů z ortofotomap.

Základní editační funkcí je přidání nové lomené čáry. Na základě získaných zkušeností byly implementovány některé speciální funkce: při přidávání segmentu je možné využít již existující bod z jiné lomené čáry, přidat segment v požadovaném úhlu vzhledem k předchozímu segmentu (0°, 45°, 90°...) nebo požadovat, aby nový bod ležel na již existujícím segmentu. Tyto funkce zjednodušují vektorizaci půdorysu budov (např. kolmé zdi) nebo napojování parcel na sousední (první bod leží na již existující hranici).

Aplikace *Vec* je dále využívána pro přesné umístění půdorysu 3D modelů budov do mapy. Půdorys je možné posunovat a natáčet až přesně zapadá do hranice stavební parcely.

4. Transformace 2D

V této fázi jsou vykonávány všechny nezbytné automatické transformace vstupních dat, které probíhají pouze v rovině (nebo výška vystupuje pouze jako doprovodný údaj). Takových transformací je v současné době kolem 200. Každá může mít několik vstupů a také výstupů. Některé z výstupů jsou následně vstupem pro další transformace. Výsledkem transformací jsou nakonec datové podklady pro co nejnázší vytváření 3D modelu v následující fázi.



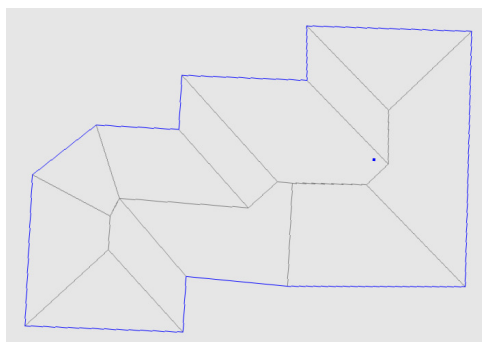
Obr. 3 – Ukázka části sítě transformací nad zástavbou

Graf závislostí je poměrně složitý a bylo nutné vytvořit automatické mechanismy kontrolující aktuálnost výstupů jednotlivých transformací. Není-li některý z výstupů aktuální (došlo například ke změně doplňujících informací používaných v transformaci nebo ke změně zdrojového kódu transformace), je transformace automaticky vyvolána. To může vést k řetězové reakci, kdy je třeba vyvolat i všechny další transformace, které používají výstupy z první transformace jako své vstupy. Zjednodušená ukázka vybrané části transformací – transformace týkající se zástavby (podrobněji viz dále) je na obrázku 3.

Při návrhu algoritmů transformací musel být brán ohled i na povahu vstupních dat. Základní vstupní data – mapy z GISů nejsou obvykle vytvářeny pro snadné vytváření 3D modelů. Neobsahují (a ani to není jejich smyslem) celou řadou drobných detailů, které jsou naopak pro virtuální model nezbytné. Velmi časté jsou také drobné nepřesnosti a zaokrouhlovací chyby v souřadnicích, které vedou k tomu, že místo jednoho bodu jsou v mapě vlastně body dva a některé zamýšlené rovnoběžky mají průsečík rozhodně dříve než v nekonečnu. V případě prohlížení map na obrazovce počítače je jedno, zda je v souřadnici rozdíl 1mm, ale pro následné automatické zpracování to představuje problém. Celá řada transformací proto za pomoci ručně zadaných detailů a korekcí (viz výše) vyhledává a opravuje tyto případy, aby navazující algoritmy nezhavarovaly.

Transformace jsou vytvářeny také s ohledem na zamýšlený přesun do historie. Přestože jsou v současné době k dispozici přesné digitální mapy Prahy, nejsou rozhodně pro Prahu v roce 1850, ale ani například pro kolejiště Hlavního nádraží v roce 1980. Některá data nejsou k dispozici ani pro současnost a jiná nejsou dostupná cenově (pro nekomerční povahu projektu). Z těchto důvodů je celá posloupnost transformací vytvářena tak, aby jako minimální vstup mohly být použity pouze hranice parcel. Ty lze získat (v rastrové podobě a s různou mírou přesnosti) pro libovolný historický okamžik minimálně od doby Karla IV. a následně i vlastními prostředky vektorizovat.

Jednotlivé transformace lze rozdělit do skupiny podle povahy vstupů, kterými se zabývají. Příkladem takové skupiny transformací jsou transformace týkající se zástavby nebo transformace nad komunikacemi.



Obr. 4 – Návrh střešních plátů štítové střechy pro budovu složitějšího půdorysu



Obr. 5 – Vyhledání středové osy (červeně) pro cesty dané krajnicí (modře)

Transformace nad zástavbou

Od 2D půdorysu budovy je k jejímu 3D modelu poměrně dlouhá cesta. Sled nutných kroků je přibližně následující (viz také obr. 3):

- identifikace oblastí tvořících půdorys budov (transformace **Dum**)
- oprava půdorysu (transformace **Dum.Oprava**)
- identifikace dílčích „křídel“ budovy (**DumSegment**)
- návrh střešních plátů (**DumStrecha**)
- zjištění výšky budovy a typu/výšky střechy
- odvození pomocných informací (např. které stěny bude nutné modelovat)
- nalezení vhodných textur pro stěny (čelní fasáda, boční stěny, fasáda do dvora) a pro střechu (podle jejího typu) (vše v **DumDetail.Odvozene**)

Půdorysy domů jsou obvykle dány hranicemi parcel jako lomenými čarami. Objekt „dům“ jako uzavřenou oblast je z nich nutné nejprve vytvořit. Ve změní čar bývá někdy problém určit, která oblast je skutečně dům a která pouze dvorek ve vnitrobloku. Podle kvality vstupních dat je potom nutné opravovat drobné nepřesnosti v půdorysu (odstranění nevýznamných výstupků, napravení ne úplně pravých úhlů stěn apod.).

Z doplňujících údajů je určen typ střechy a v případě štítové musí být z půdorysu odvozen tvar jednotlivých střešních plátů. To je velmi zajímavá úloha, která názorně ukáže vliv zaokrouhlovacích chyb a nepřesnosti v počítání s čísly s plovoucí desetinnou čárkou na číslicovém počítači. Ty vedou například k tomu, že průsečík dvou mimoběžek je posunut o několik metrů a ocitne se mimo půdorys domu. Na obrázku 4 je předvedeno řešení štítové střechy pro budovu nepravidelného půdorysu.

Jako poslední přípravný krok jsou vybrány vhodné textury a provedeny doplňující výpočty, které například ze známých výšek okolních domů určí výšku aktuálně zpracovávaného domu nebo zda a které boční stěny domu je vůbec nutné vykreslovat (když nejsou zcela zakryty sousedním domem).

Transformace nad komunikacemi

Další skupina transformací se zabývá ulicemi a cestami, jejich rozdělením do segmentů a také vyhledáním středové osy – viz. obrázek 5. Ta je nezbytná pro pozdější mapování textur (aby bylo přirozeně ve směru cesty a navazovaly na sebe) a také pro určení trajektorie, po které se budou pohybovat dopravní prostředky. Přestože bylo vyzkoušeno několik různých algoritmů, není nalezená středová osa vždy uspokojivá. Je otázkou, jestli je hledání obecného algoritmu rozumné a jestli by zvláště v místech složitějších křižovatek nebylo jednodušší zadat průběh středové osy ručně.

Aplikace Convert

Všechny 2D transformace jsou implementovány v aplikaci *Convert*. Ta zajišťuje i automatické spuštění transformací, jejichž výstupu nejsou aktuální. O veškeré činnosti aplikace je vypisován podrobný log-soubor, který informuje o úspěšných transformacích, ale i lokalizuje případné problémy. Výstupy všech transformací jsou ukládány do souborů, které je možné načíst v aplikaci *Vec* a jednotlivé kroky vizuálně kontrolovat. Graficky jsou označeny i případné chyby. Příkladem může být přerušená čára v definici stavební parcely, která zapříčiní nemožnost uzavření oblasti stěn budovy.

5. Modelování 3D

Fází *Modelování 3D* se rozumí vytváření 3D objektů pomocí vhodné grafické knihovny, např. *OpenGL* resp. (v případě projektu Praha4D) její nadstavby *OpenSceneGraph*. Pomocí tzv. geometrických primitivů je nutné poskládat všechny domy, povrchy silnic, dopravní prostředky, ale i například oblohu. Je nutné podotknout, že pod „geometrickým primitivem“ se v zásadě skrývá pouze trojúhelník – se vším, co z toho plyne, zejména pro modelování kol a zaoblených tvarů. Při modelování tak jde v zásadě o neustálý souboj mezi kvalitou (více trojúhelníků) a limity výkonnosti použitého HW (méně trojúhelníků).

Na základě finálních výstupů 2D transformací, dalších pomocných vstupů a také připravených textur je v tomto kroku skládán celý model. Každý bod se musí umístit do trojrozměrného prostoru a všechny trojúhelníky pokrýt texturou. V pomocných údajích tak musí být například výšky domů a střech, ale zejména co nejpřesnější popis terénu. Popis terénu ve městě opět tvoří samostatnou kapitolu a přesahuje rámec tohoto článku. Jako hlavní problém lze uvést, že tradiční popis terénu pomocí vrstevnic nebo dokonce čtvercové mřížky výšek je pro město (zvláště tak členité jako Praha) naprosto nevyhovující. Dokonce ani v současné době dostupné nejpřesnější měření bodů a povinných spojnic (TIN) nemůže zahrnovat všechny detaily. Dodatečně je nutné řešit například „zařiznutí“ silnic do svahu nebo vlnění parkových cest a jejich křížení.

Důležitou součástí přípravy modelu města je vhodně navržená stromová struktura 3D dat, do které patří i definice úrovně detailu (LOD). Kvůli (v každé době) nedostatečnému výkonu HW je nutné velmi pečlivě zvažovat, který detail a do jaké vzdálenosti od pozorovatele je skutečně nutné zobrazovat. Pro budovy nebo dopravní prostředky je připraveno několik verzí modelů v různé úrovni podrobnosti (tedy s různým počtem trojúhelníků). Verze jsou postupně nahrazovány od nejjednoduššího po nejsložitější, jak se pozorovatel přibližuje. Menší objekty, jako jsou lavičky, lampy, výhybky, nemusí být vůbec vykreslovány, pokud je pozorovatel od nich dále než např. jeden kilometr.

Protože je 3D model budován po částech, je dále nutné zajistit, že stejná textura (např. tráva nebo povrch vozovky) nebude načítána a hlavně ukládána do video-paměti dvakrát. To samé platí i pro modely dopravních prostředků a další objekty (lavičky, lampy...). Způsob modelování musí být tedy uzpůsoben pozdějšímu snadnému sdílení při vizualizaci.

Modely významných budov a objektů jsou vytvářeny externě ve specializovaných aplikacích pro 3D modelování. Pro jejich snazší umístění do modelu města je součástí aplikace *GisConv* i procedura, která z 2D hranic stavební parcely vytvoří jednoduchý 3D

model půdorysu budovy. Zároveň model transformuje do počátku souřadné soustavy a natočí, aby hlavní stěna byla rovnoběžná s osou X. Bez této transformace by se 3D model vytvářel velmi špatně. Opačným postupem je pak výsledný 3D model přesunut a otočen na správnou pozici v celkovém modelu města.

Aplikace Make3D

Make3D vytváří vlastní 3D model města. Podobně jako *Convert* obsahuje i *Make3D* sít transformací, které však v tomto případě převádějí 2D data na 3D objekty. Objekty nejnižší úrovně detailu jsou následně i agregovány do skupin vyšší a vyšší úrovně a celý model tak dostává stromovou strukturu.

6. Komunikace mezi aplikacemi Convert a Make3D

Předávání výstupů z 2D transformací do aplikace *Make3D* je řešeno následovně. Výstupy 2D transformací, které mají být přístupné z aplikace *Make3D*, musí být označeny jako „exportované“. Takto označené výstupy jsou vyjmenovány v automaticky vytvářeném textovém souboru, který je vlastně zdrojový text v C++ a je používán při kompilaci aplikace *Make3D*. V aplikaci *Make3D* je možné načítat pouze výstupy uvedené v tomto zdrojovém textu (kontrolováno automaticky po spuštění aplikace *Make3D*). Zároveň je zajištěno, že po úspěšném běhu aplikace *Convert* budou všechny výstupy označené pro export aktuální (vyvoláním odpovídajících transformací v případech, kdy je to nutné).

7. Vizualizace 3D

Vlastní vizualizace je pak už jen třešničkou na dortu, ve které se „pouze“ zobrazuje model pracně vybudovaný v předchozích krocích.

Aplikace Vis3D

Hlavní aplikace *Vis3D* umožňuje pohyb po městě (jak po zemi, tak ve vzduchu), inkrementální načítání dat pouze pro oblast, ve které se uživatel právě nachází, a několik pomocných funkcí jako jsou přednastavené pozice, průlety po předem nadefinované trajektorii a ukládání „fotografií“ a „filmů“ na disk.

Hlavní část aplikace však tvoří řízení změn v čase – volba ročního období, změna počasí a také pohyb dopravních prostředků. Roční období, denní doba a aktuální počasí se nastavují variantou použitých textur (např. zasněžené střechy v zimě) a dále vhodně zvolenými parametry 3D scény – osvětlení, viditelnost (mlha) apod.



Obr. 6 – Pohyb dopravních prostředků

Pohyb dopravních prostředků výrazně umocňuje dojem z modelu města. Pro každý dopravní prostředek jsou zadány jeho základní fyzikální parametry (hmotnost, výkon motoru, adhézní hmotnost...), ze kterých je pak pomocí fyzikálních rovnic odvozována odpovídající schopnost akcelerace nebo naopak brzdění. Implementovány jsou funkce pro zabránění srážkám, které

obsahují i jistý prvek náhodnosti (zejména v reakčních dobách virtuálních řidičů) pro vyšší přirozenost pohybu (např. automobilů v kolonách). Dopravní prostředky hromadné dopravy se pohybují stejně jako v realitě po předem definovaných trasách a zastavují v zastávkách. Podrobný popis implementace je uveden v [2].

Aplikace Rovinka3D

Pro odladění implementace pohybu dopravních prostředků byla vyvinuta speciální vizualizace, která znázorňuje závod (kolejových) dopravních prostředků na rovné trati s častými zastávkami (viz <http://www.Praha4D.net/vmovie.html>). Názorně tak bylo možné otestovat schopnost akcelerace a brzdění lokomotiv i tramvají s různě velkou zátěží v podobě připojených vagonů.

Aplikace Kol3D

Tato aplikace předvádí v detailu všechny vytvořené 3D modely budov, dopravních prostředků a dalších objektů. Jako na kolotoči se modely natáčejí ze všech stran a je tak možné detailně zkontrolovat jejich vzhled před umístěním do modelu města. To pomáhá v odhalení drobných nepřesností, které nebyly patrné v modelovacím nástroji, a také k posouzení vzhledu modelu při nastavení různých parametrů 3D scény (zejména intenzity a směru osvětlení).

8. Plány dalšího rozvoje

Možnosti dalšího rozvoje jsou nepřehledné. Asi nejzajímavějším problémem k řešení je přesun do historie. Netýká se jen shromáždění dostatečného množství podkladů, ale zejména navržení vhodného způsobu reprezentace modelu města s ohledem na v něm probíhající změny. Bude nutné vytvořit aparát vhodný pro popis všech možných změn, které v průběhu historického vývoje nastaly. Základem budou údaje o jednotlivých domech od jejich postavení (tvar, fasáda, střecha), přes různé stavební úpravy (zvýšení/snížení, změna tvaru střechy) až k okamžiku zbourání. Dále bude nutné popsat změny vedení cest (ulic i cest v parcích), růstu vegetace (zejména stromů), povrchu silnic a dalších reálií dané doby (dopravní prostředky, móda...), ale také terénní změny způsobené jak přirozenými procesy (změna tvaru ostrovů), tak zejména činností člověk (vyrovnávání terénu, úprava pro dopravní cesty, nábřeží apod.). První zkušenosti s popisem změn v čase byly učiněny už při tvorbě aplikace Dějepisný atlas [1].

Literatura

- [1] Šimůnek, M. *Atlas – GIS s časovou dimenzí*. GIS Ostrava 1999, ISSN: 1211-4855, strana 289 až 298
- [2] Šimůnek, M. *Autonomous Movement of Vehicles in Virtual City*. In: Novák, M.; Goulias, K. (Eds.) *TRANSTEC 2007*. FD ČVUT Praha and University of California Santa Barbara, 2007, p. 471 - 480. ISBN 978-80-01-03782-9
- [3] Ježek, J.: Diplomová práce, ČVUT 2003 – Převody mezi zeměpisnými souřadnými systémy
- [4] OpenGL, <http://www.opengl.org>
- [5] OpenSceneGraph, <http://www.openscenegraph.org>
- [6] Shewchuk, J. R.: *Triangle: Engineering a 2D Quality Mesh Generator and Delaunay Triangulator*, in *Applied Computational Geometry: Towards Geometric Engineering* (Ming C. Lin and Dinesh Manocha, editors), volume 1148 of *Lecture Notes in Computer Science*, pages 203-222, Springer-Verlag, Berlin, May 1996. (From the First ACM Workshop on Applied Computational Geometry.)

Aktuální seznam použitých zdrojů je uveden na adrese www.Praha4D.net/zdroje.html.