

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

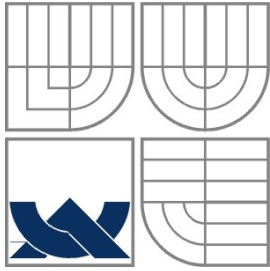
INTERAKTIVNÍ EDITOR 3D TERÉNU

BAKALÁŘSKÁ PRÁCE  
BACHELOR'S THESIS

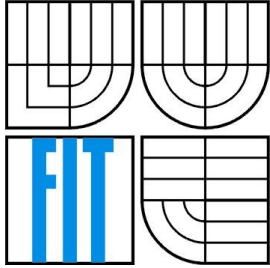
AUTOR PRÁCE  
AUTHOR

JIŘÍ HULVA

BRNO 2011



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF COMPUTER SYSTEMS

# INTERAKTIVNÍ EDITOR 3D TERÉNU

INTERACTIVE 3D TERRAIN EDITOR

BAKALÁŘSKÁ PRÁCE  
BACHELOR'S THESIS

AUTOR PRÁCE  
AUTHOR

JIŘÍ HULVA

VEDOUCÍ PRÁCE  
SUPERVISOR

ING. JAROSLAV PŘIBYL

BRNO 2011

**Vysoké učení technické v Brně - Fakulta informačních technologií**

Ústav počítačové grafiky a multimédií

Akademický rok 2010/2011

**Zadání bakalářské práce**

Řešitel: **Hulva Jiří**

Obor: Informační technologie

Téma: **Interaktivní editor 3D terénu**

**Interactive 3D Terrain Editor**

Kategorie: Počítačová grafika

Pokyny:

1. Prostudujte algoritmy pro renderování terénů z výškových map.
2. Navrhněte funkce interaktivního editoru 3D terénu (např. editace výškové mapy, textura povrchu, generování objektů apod.)
3. Diskutujte objektový návrh editoru z hlediska dalšího rozšiřování funkčnosti jako je generování stromů, zástavby nebo dopravní infrastruktury.
4. Implementujte objektový model editoru a některé vybrané funkce.
5. Zhodnoťte dosažené výsledky a navrhněte možnosti pokračování projektu.

Literatura:

- dle pokynu vedoucího

Při obhajobě semestrální části projektu je požadováno:

- Body 1 až 3.

Podrobné závazné pokyny pro vypracování bakalářské práce naleznete na adrese

<http://www.fit.vutbr.cz/info/szz/>

Technická zpráva bakalářské práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap (20 až 30% celkového rozsahu technické zprávy).

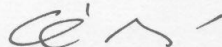
Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVD-R, apod.), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Příbyl Jaroslav, Ing.**, UPGM FIT VUT

Datum zadání: 1. listopadu 2010

Datum odevzdání: 18. května 2011

**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**  
Fakulta informačních technologií  
Ústav počítačové grafiky a multimédií  
60200 Brno, Božetěchova 2



doc. Dr. Ing. Jan Černocký  
vedoucí ústavu

## **Abstrakt**

Cílem této práce je navrhnout a implementovat interaktivní editor 3D terénu. Program by měl být schopen vytvořit 3D model terénu ze vstupních dat, zobrazit jej a umožnit volný pohyb v zobrazené scéně. Uživatelům by měly být poskytnuty nástroje pro úpravu výškových map a povrchových textur. Mělo by být také umožněno načítání modelů objektů, jako jsou budovy nebo vegetace, a jejich umístění na vybrané pozice v terénu.

## **Abstract**

The purpose of this work is to design and implement an interactive 3D terrain editor. The application should be able to create three-dimensional model of a terrain from input data, display it and allow free movement in displayed scene. Users should be provided with tools for editing heightmaps and surface textures. There should be option to load models of objects, like buildings or vegetation and place them on chosen locations.

## **Klíčová slova**

Počítačová grafika, editor, terén, textury, výšková mapa, OpenGL, generování výškových map, COLLADA,

## **Keywords**

Computer graphics, editor, terrain, textures, heightmap, OpenGL, heightmap generating, COLLADA

## **Citace**

Jiří Hulva: Interaktivní editor 3D terénu, bakalářská práce, Brno, FIT VUT v Brně, 2011

# Interaktivní editor 3D terénu

## Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením Ing. Jaroslava Příbyla. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....  
Jiří Hulva  
15. května 2011

## Poděkování

Tímto chci poděkovat Ing. Jaroslavu Příbylovi za ochotu a trpělivost při konzultacích projektu a také za mnoho velmi cenných rad a připomínek, bez nichž by tato práce snad ani nemohla vzniknout.

©Jiří Hulva 2011

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

# Obsah

|  |    |
|--|----|
| Obsah.....                             | 1  |
| 1 Úvod.....                            | 3  |
| 1.1 Analýza zadání.....                | 4  |
| 1.2 dostupné 3D Editory terénu.....    | 5  |
| 1.2.1 Terragen Classic.....            | 5  |
| 1.2.2 Terragen 2 .....                 | 5  |
| 2 Teorie.....                          | 6  |
| 2.1 Paměť hloubky.....                 | 6  |
| 2.2 Display List.....                  | 6  |
| 2.3 Vertex Buffer.....                 | 7  |
| 2.4 Box Filtering.....                 | 7  |
| 2.5 Kartografická data.....            | 8  |
| 2.5.1 Výšková mapa.....                | 8  |
| 2.5.2 Zdroje kartografických dat.....  | 10 |
| 3 Návrh editoru.....                   | 12 |
| 3.1 Grafické uživatelské rozhraní..... | 12 |
| 3.1.1 Mapa terénu.....                 | 12 |
| 3.1.2 Panely nástrojů.....             | 12 |
| 3.1.3 Lišta záložek.....               | 13 |
| 3.1.4 Seznam vrstev.....               | 13 |
| 3.1.5 Stavový panel.....               | 13 |
| 3.2 Vykreslování scény.....            | 13 |
| 3.3 Hlavní funkce.....                 | 14 |
| 3.3.1 Editace výškových dat.....       | 14 |
| 3.3.2 Úprava povrchových textur.....   | 14 |
| 3.3.3 Rozmíst'ování objektů.....       | 15 |
| 3.4 Formát vstupních dat.....          | 15 |
| 3.4.1 Kartografická data.....          | 15 |
| 3.4.2 Modely objektů.....              | 16 |
| 3.5 Ukládání dat.....                  | 16 |
| 3.5.1 Vrstvy.....                      | 17 |
| 4 Rozšiřování editoru.....             | 17 |
| 4.1 Nové nástroje.....                 | 17 |
| 4.1.1 Nový formát 3D modelů.....       | 18 |

|  |    |
|--|----|
| 4.2 Nový formát mapy.....                      | 18 |
| 4.2.1 Nové typy souborů.....                   | 18 |
| 4.2.2 Odlišná struktura mapových podkladů..... | 18 |
| 5 Implementace.....                            | 19 |
| 5.1 Programová struktura.....                  | 19 |
| 5.2 Načítání dlaždic terénu.....               | 20 |
| 5.3 Načítání 3D modelů.....                    | 21 |
| 5.3.1 Specifikace knihovny a její omezení..... | 21 |
| 5.3.2 Průběh načítání modelu.....              | 22 |
| 6 Vzorová scéna.....                           | 23 |
| 7 Závěr.....                                   | 26 |
| 7.1 Další pokračování projektu.....            | 26 |
| 8 Literatura.....                              | 27 |
| Seznam příloh.....                             | 28 |
| 9 Příloha 1.....                               | 29 |
| 9.1 Ovládání aplikace.....                     | 29 |
| 9.1.1 Ovládání kamery.....                     | 29 |
| 9.1.2 Panel nástrojů.....                      | 29 |
| 9.1.3 Záložky.....                             | 30 |
| 9.1.4 Seznam vrstev.....                       | 30 |
| 9.1.5 Klávesové zkratky: .....                 | 30 |

# 1 Úvod

V oblasti počítačové grafiky se často setkáváme s nutností zobrazit terén. Ten bývá většinou velice rozsáhlý a modelovat jej stejně jako ostatní objekty po jednotlivých polygonech by bylo velice časově náročné a ve výsledku neefektivní. Vznikají proto editory zaměřené na tvorbu a editaci terénu.

Své uplatnění naleznou například při tvorbě prostředí pro mnoho různých druhů simulací a v oblasti vývoje počítačových her. Také všude tam, kde se využívají fotorealistické modely krajiny, například ve filmovém průmyslu, je potřeba daný terén nejprve vytvořit.

Další důležitou oblastí, kde najdou editory terénu využití, je kartografie. Začaly vznikat stále detailnější mapy složené z milionů leteckých nebo družicových snímků. Před tím, než taková mapa vznikne, musí vstoupit data projít mnoha úpravami.

Už jen samotné slícování snímků je velice náročné. Povrch bývá často zkreslen perspektivou, jednotlivé navazující díly se mohou od sebe barevně velmi lišit z důvodů pořízení snímků v různou denní dobu, nemluvě o nežádoucích anomáliích, jako jsou mraky či světelné odlesky.

U prostorových map navíc přibývá problém s chybami ve výškových datech, kdy snímací zařízení nedokáže dostatečně přesně zachytit složitý tvar snímaných objektů, jako jsou například budovy, mosty nebo zdi. Přínejlepším jsou výsledkem hodně zdeformované a realitě velmi vzdálené modely, v naprosté většině případů jsou však zredukovány na drobnou terénní nerovnost nebo doslova „srovnány se zemí“.

Kromě toho se ve výškových datech objevují také nepřesnosti způsobené dočasnými rozměrnými objekty, jako mohou být třeba lodě na vodní hladině.

Následné zpracovávání získaných mapových podkladů je dosti zdlouhavý proces a vzhledem k obrovskému množství vstupních dat je snaha nechat co největší díl práce na specializovaných algoritmech. Mnoho chyb však software nedokáže opravit, nebo dokonce ani rozpoznat, a musí se odstranit manuálně.

Hlavním cílem této bakalářské práce je vytvořit sadu nástrojů, pomocí kterých by právě takovéto úpravy bylo možné provádět. Aplikace musí být schopná zobrazit vstupní data v podobě trojrozměrného terénu v dostatečném detailu. V případě nalezení chyb nebo nepřesností pak poslouží při jejich efektivní korekci. Dále umožní vylepšit vzhled mapy umístěním trojrozměrných modelů budov, vegetace a jiných objektů.

Pro zobrazení terénu v takovémto editoru je důležitá především přehlednost a rychlost. Není proto nutné se snažit dosáhnout co nejfotorealističtějšího zobrazení, například pomocí implementací náročných grafických efektů.

Také si tento projekt neklade za cíl vyvinout program, pomocí kterého by bylo možné vytvářet úplně nový terén. Problematika náhodného generování kartografických dat je v práci diskutována, ale ve výsledné aplikaci nebyly tyto nástroje implementovány, neboť pro hlavní zaměření práce (úpravy reálných mapových podkladů) jsou relativně nadbytečné.

# 1.1 Analýza zadání

Tato část se zabývá detailním pohledem na jednotlivé body zadání práce s popisem, jak bylo k jejich splnění přistoupeno. Zadání bakalářské práce je následující:

**1. Prostudujte algoritmy pro renderování terénů z výškových dat.**

Byl nastudován princip několika algoritmů, které by bylo možné použít pro optimalizaci vykreslování rozsáhlých částí terénu: ROAM, Geometrické MipMapy a chunked LOD. Ovšem vzhledem k zaměření vyvíjené aplikace nebylo jejich využití nutné. Editor pracuje s terénem uloženým ve vysokém detailu ve velkém množství souborů. Proto by byl při zobrazení rozsáhlejší plochy stejně omezen rychlostí načítání z datového úložiště. Navíc navržené nástroje pro editaci terénu je možné efektivně používat spíše při detailnějším pohledu.

**2. Navrhněte funkce interaktivního editoru 3D terénu (např. Editace výškové mapy, textura povrchu, generování objektů apod. ).**

Návrhem aplikace se zabývá kapitola 3 této zprávy. Mezi navržené funkce editoru patří nástroje pro úpravu výškové mapy terénu, výběr a editaci povrchových textur a také nahrávání 3D modelů objektů a jejich rozmístování na mapě. Editor musí být interaktivní. To znamená, že změny terénu by měly být ihned viditelné. Součástí návrhu je také přehledné uživatelské rozhraní a návrh několika funkcí pro usnadnění práce.

**3. Diskutujte objektový návrh editoru z hlediska dalšího rozšiřování funkčnosti, jako je generování stromů, zástavby nebo dopravní infrastruktury.**

Editor by měl být snadno rozšiřitelný a postup při přidávání jeho funkcí pokud možno co nejjednodušší. Příklady několika možných rozšiřujících funkcí jsou uvedeny v Závěru, v podkapitole 7.1. Postupu, jak nové nástroje integrovat do stávající implementace, se věnuje podkapitola 4.1.

**4. Implementujte objektový model editoru a některé vybrané funkce.**

Podle vypracovaného návrhu byla vyvinuta aplikace TerraEdit, umožňující zobrazení a editaci vstupních kartografických dat. Společně s demonstračními daty je umístěna na příloženém DVD.

**5. Zhodnoťte dosažené výsledky a navrhněte možnosti pokračování projektu.**

Zhodnocení dosažených výsledků a možným směrům jeho budoucího vývoje se věnuje kapitola 7.

## 1.2 dostupné 3D Editory terénu

Zde budou představeno několik již existujících nástrojů pro vytváření a editaci terénu.

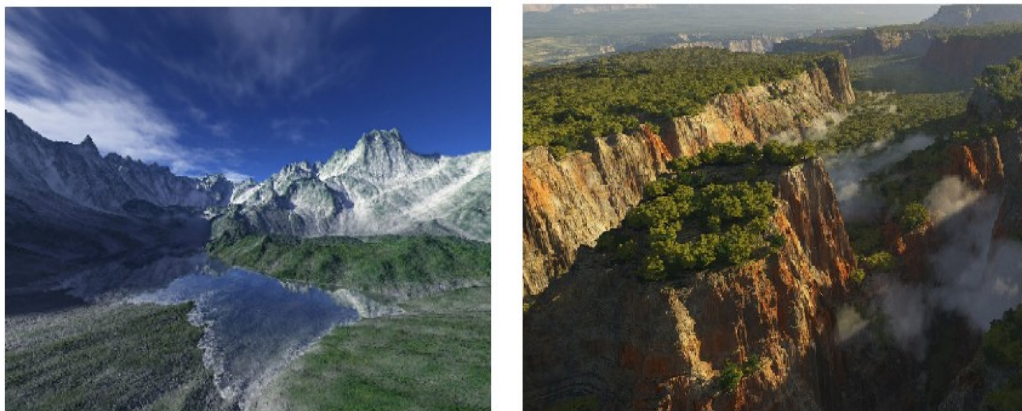
### 1.2.1 Terragen Classic

Volně dostupný program pro vytváření a editaci snímků nebo animací velice realistických scenérií, včetně efektů jako je mlha, stíny a podobně. Umožňuje generování a úpravu terénu, vodní hladiny a oblohy. Jeho ovládání je poměrně jednoduché. [1]

### 1.2.2 Terragen 2

Nástupce Terragenu Classic. Odstraňuje spoustu omezení, které obsahoval jeho předchůdce. Za nejzásadnější změny lze označit možnost vykreslování povrchů celých planet a pokročilou manipulaci s vodní hladinou. Oblaka již nejsou reprezentována plochou texturou, ale jsou generována trojrozměrně. Je také konečně možné do scény vkládat objekty.

Terragen 2 dává uživateli mnohem větší kontrolu nad samotným procesem renderování scény s možností definování vlastních shaderů pro tvar, rozmístění a vlastnosti jednotlivých prvků. Ačkoliv je proces tvorby krajiny složitější než v Terragenu Classic, kvalita výsledných snímků je na mnohem vyšší úrovni a jsou mnohdy těžko rozeznatelné od skutečnosti [1].



*Obrázek 1: Vlevo snímek vytvořený v Terragenu Classic; vpravo pomocí Terragenu 2. (zdroj obrázků [1])*

## 2 Teorie

Tato kapitola se bude zabývat teoretickými základy, potřebnými při následném vývoji editoru.

### 2.1 Paměť hloubky

Při vykreslování scény s více objekty dochází často k tomu, že jeden z objektů překrývá jiný. Aby GPU mohla rozeznat, které plochy jsou budou zobrazeny a které jsou překryty jinými objekty, používá mechanismus zvaný paměť hloubky, nebo také z-buffer.

Většinou se jedná o dvojrozměrné pole hodnot, jehož výška a šířka odpovídá rozměrům vykreslované plochy. Když je vykreslen objekt, do z-bufferu se pro každý z vykreslených pixelů uloží hodnota udávající jejich vzdálenost od kamery. Při vykreslování dalších objektů se nejprve provede kontrola, zda v z-bufferu už nejsou na příslušných pozicích uloženy nižší hodnoty vzdálenosti. To by znamenalo, že právě zpracovávané pixely jsou dál od pozorovatele a jsou zakryty těmi už vykreslenými. V opačném případě jsou nové hodnoty vzdáleností zapsány do z-bufferu a příslušné pixely mohou být vykresleny.

Výsledkem je běžná podoba 3D zobrazení, kdy bližší objekty zakrývají ty vzdálenější[2].

### 2.2 Display List

Při základním způsobu vykreslování objektů v OpenGL se na grafickou kartu zasílají souřadnice bodů společně s dalšími daty a pomocí nich je pak vykreslena celá scéna. Při opakovaném vykreslování stejné scény a minimálními změnami jsou tak stejná data generována a zaslána pořád znova. To představuje zcela zbytečný datový přenos vytváří zátěž na CPU.

Jedním ze způsobů, jak tento problém eliminovat, je použití display listů. Tento mechanismus umožní jednou zasláná data na grafické kartě uložit a poté jediným příkazem zadat jejich provedení kdykoliv je potřeba. Odpadá tak nutnost zasílat stejná data znova.

Display listy však mají i své nevýhody, kvůli kterým se v dnešní době od jejich používání upouští. Především je po vytvoření již nelze modifikovat a jedinou možností je jejich smazání a vytvoření nového display listu [3].

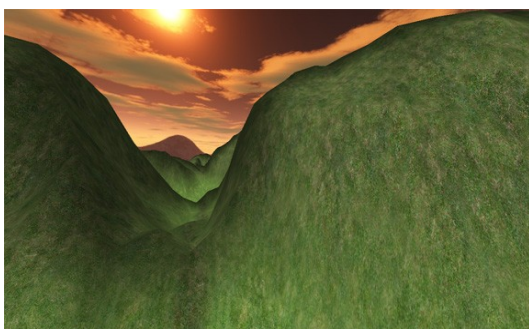
### 2.3 Vertex Buffer

O něco modernější přístup než display listy představuje použití vertex bufferu. Umožňuje uložit na paměť grafické karty posloupnosti souřadnic bodů ale i jiných druhů dat v takzvaných vertex buffer objektech, které se později využijí při vykreslování grafických primitiv. Je možné vybírat, která data se použijí a v jakém pořadí, což s sebou přináší úsporu paměti a větší flexibilitu. K přednostem také patří možnost jednoduše upravovat a nahrazovat vybrané části uložených dat a jednoduše a rychle tak měnit tvar výsledných modelů[3].

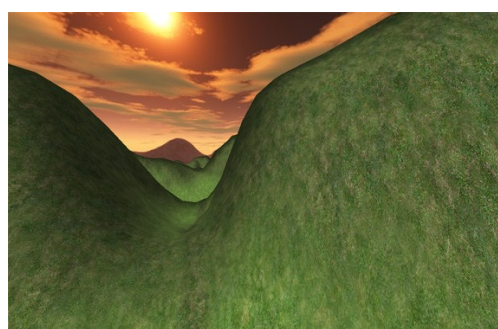
## 2.4 Box Filtering

Při vykreslování trojrozměrného modelu terénu z výškových map bývá často výsledkem nerovný terén plný ostrých hran. Mohou za to nepřesnosti nebo šum na výškové mapě. Jedním ze způsobů, jak tento problém vyřešit, je takzvaný Box Filtering.

Jedná se o metodu, která sníží rozdíly v hodnotách sousedních bodů. Běžně se používá k redukci šumu. Celý její princip spočívá v tom, že novou hodnotu každého bodu nastaví tak, aby odpovídala průměru původní hodnoty a hodnot všech 8 sousedních bodů. Tím se vyhladí prudké skoky v hodnotách jednotlivých bodů a výsledkem je vyhlazená výšková mapa s plynulými přechody [4]. Na obrázcích 2 a 3, převzatých ze zdroje [4] je možné vidět příklad použití této metody.



*Obrázek 2: Terén před vyhlazením.*



*Obrázek 3: Terén po vyhlazení metodou Box Filtering.*

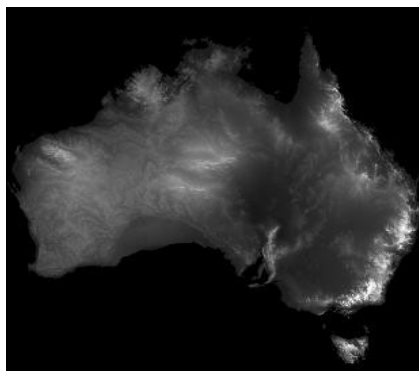
## 2.5 Kartografická data

Tato kapitola se bude zabývat druhy dat popisující terén. A způsobu jejich získání.

### 2.5.1 Výšková mapa

Nejběžnějším způsobem ukládání dat o tvaru terénu je tzv. výšková mapa. Jedná se o dvourozměrné pole hodnot, které určují výšku na příslušné pozici. Často se používá rastrový obrázek v barvách šedi, kde barva jednotlivých pixelů reprezentuje výškovou hodnotu v rozmezí černá=nejnižší, bílá=nejvyšší.

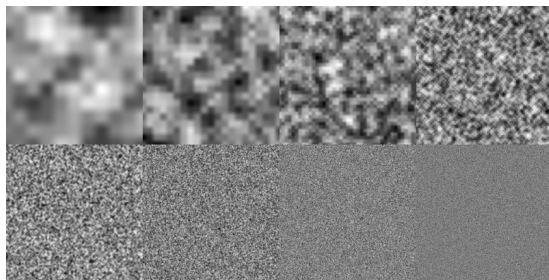
Výškové mapy jsou jednoduchým a efektivním způsobem, jak zachytit složitý a nepravidelný tvar povrchu. Nejen v oblasti počítačové grafiky se staly pro tento účel takřka standardem. Umožňují snadný a rychlý přístup k údajům o výšce konkrétního bodu a je velice snadné z nich vytvořit pravidelnou trojrozměrnou síť, kterou je následně možné zobrazit [5].



Obrázek 4: Výšková mapa Austrálie [6]

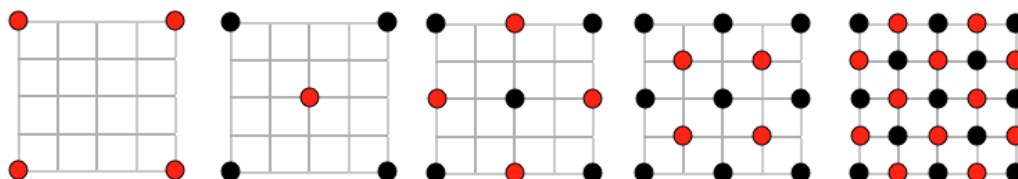
#### Generování výškové mapy

Výškové mapy (uložené jako obrázky) lze velice snadno vytvářet a editovat v běžných grafických editorech. Případně je možné je náhodně generovat. Nejznámějším algoritmem pro tvorbu realisticky vypadajících terénů je algoritmus "**Perlin noise**". V oblasti počítačové grafiky je velmi rozšířený, využívá se například ke generování textur, mraků, ohně... Jeho základem je pseudonáhodný generátor čísel, který vygeneruje s určitým rozestupem posloupnost pseudonáhodných čísel v zadaném intervalu. Získané hodnoty lze pak proložit křivkou. Následně vygenerujeme novou posloupnost s polovičním rozestupem a intervalem než v předchozím kroku. Tento postup několikrát opakujeme. Nakonec zkombinujeme všechny získané výškové hodnoty, a tak získáme výslednou nepravidelnou křivku terénu[7].



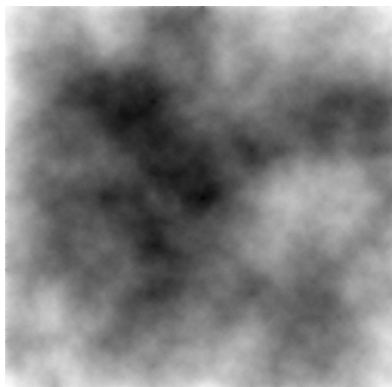
Obrázek 5: Různá podoba výsledků použití Perlinova šumu v závislosti na počtu iterací (zdroj obrázku: [http://www.m3xbox.com/GPU\\_blog/?p=28](http://www.m3xbox.com/GPU_blog/?p=28)).

Dalším ze způsobů, jak generovat poměrně realistické výškové mapy je **Diamond-square** algoritmus. Jedná se o vylepšenou verzi **Midpoint-displacement** algoritmu. Jeho princip je následující: Na začátku máme 4 body spojené 4 úsečkami tak, aby tvořily čtverec. Každému bodu přiřadíme náhodnou výškovou hodnotu. Poté čtverec rozdělíme na 4 menší čtverce a za výškové hodnoty v jejich rozích dosadíme průměr hodnoty hodnot na hranách velkého čtverce. Při výpočtu hodnoty nového bodu ve středu čtverce přidáme náhodnou odchylku. Čím vyšší tato odchylka bude, tím bude výsledný terén zvlněnější. Každý ze 4 nových čtverců následně rozdělíme na 4 nové čtverce a celý postup se iterativně opakuje libovolně dlouho.



*Obrázek 6: kroky Midpoint-displacement algoritmu: červené body jsou nové hodnoty, černé body jsou hodnoty starší. (zdroj obrázku [6])*

Midpoint-displacement algoritmus ve své původní podobě se již téměř nepoužívá. Ve výsledné výškové mapě totiž často vznikaly nepřírozené čtvercové útvary. Jeho vylepšená verze zvaná **Diamond-square algoritmus** pracuje i se šikmo natočenými čtverci, čímž tento problém řeší[6].



*Obrázek 7: Výšková mapa vygenerovaná pomocí diamond-square algoritmu [6]*

## 2.5.2 Zdroje kartografických dat

Tato podkapitola představí několik zdrojů, pomocí nichž je možné získat kartografická data reálného světa, ať už se jedná o výškové mapy nebo snímky povrchu.

### Český úřad zeměměřický a katastrální

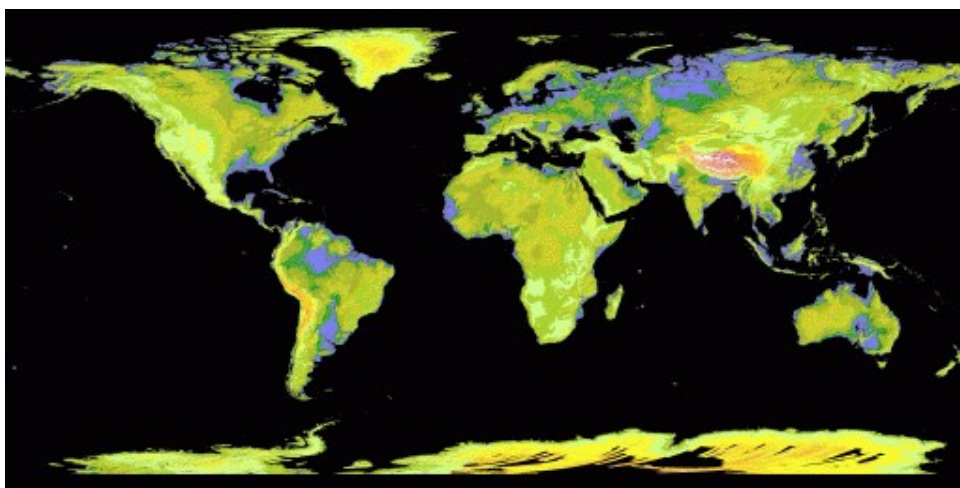
ČÚZK patří mezi ústřední orgány státní správy. Mezi jeho hlavní funkce patří evidence nemovitostí v Katastru nemovitostí České republiky a také zeměměřičské činnosti.[8] Tento úřad spravuje rozsáhlou databázi geografických dat pro Českou republiku s označením ZABAGED, včetně výškové mapy v detailu 10 metrů, nebo neustále aktualizované letecké mapy. ZABAGED běžně slouží jako hlavní zdroj pro tvorbu základních map České republiky.

Databáze není volně dostupná, přístup k ní je zpoplatněn. [9]

### ASTER GDEM

Roku 2009 byl dokončen společný projekt japonského Ministerstva ekonomie a průmyslu a americké NASA nazvaný ASTER GDEM. Jedná se o globální výškovou mapu planety Země s dosud nejvyšším dosaženým detailem a pokrytím povrchu.

ASTER je označení multispektrálního skeneru, pomocí kterého byla výšková data nasnímána. Na oběžnou dráhu byl vypuštěn v prosinci 1999 a do té doby snímá záření ze zemského povrchu ve 14 spektrálních pásmech. Infračervené záření dokáže snímat ve dvou směrech. Díky tomu je možné pořídit pro jednu oblast dvojici snímků z různých úhlů. Ty pak poslouží k výpočtu výškových dat.



Obrázek 8: Celková výšková mapa světa databáze ASTER GDEM.

Výsledná výšková mapa vznikla složením z asi 1.5 milionu družicových snímků, které byly shromažďovány přibližně jeden rok. Od doby svého vydání je stále zdokonalována. Je dostupná zdarma na adrese projektu <http://www.gdem.aster.ersdac.or.jp>.

Mapa pokrývá pevninu v rozmezí od 83 stupňů severní šířky do 83 stupňů jižní šířky. Jednotlivé výškové body jsou rozmístěny přibližně ve vzdálenostech 30 metrů [10].

|                   | <b>ASTER GDEM</b>             | <b>SRTM3</b>                         | <b>GTOPO30</b>                           |
|-------------------|-------------------------------|--------------------------------------|--|
| Zdroj dat         | Skener ASTER na Družici Terra | Upravený radar raketoplánu Endeavour | Různé společnosti vlastníci výšková data |
| organizace        | METI/NASA                     | NASA/USGS                            | USGS                                     |
| Rok vydání        | 2009                          | 2003                                 | 1996                                     |
| Doba sběru dat    | Od roku 2000                  | 11 dní                               |  |
| detail            | 30m                           | 90m                                  | 1000m                                    |
| Výšková odchylka  | 7-14m                         | 10m                                  | 30m                                      |
| Pokrytí povrchu   | 99%                           | 80%                                  |  |
| Nepokryté oblasti | Oblasti s trvalou oblačností  | Příliš strmé oblasti                 |  |

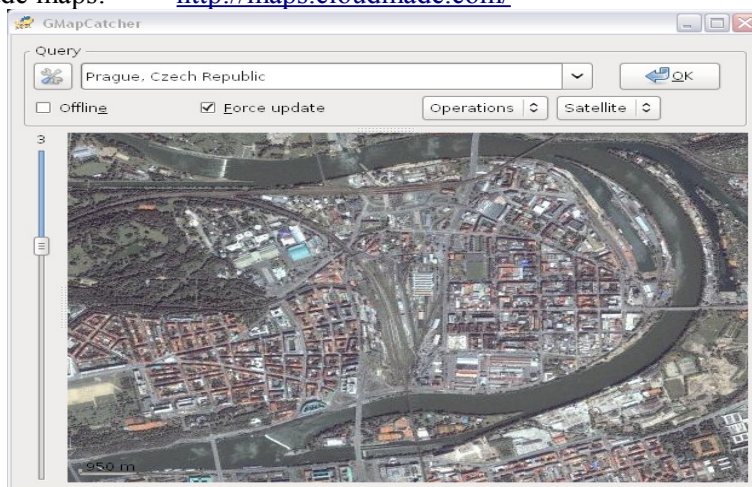
*Tabulka 1: Srovnání ASTER GDEM s jinými globálními výškovými mapami (zdroj [10]).*

### Google Map Catcher

Volně dostupný program, který umožňuje stahování, prohlížení a export mapových podkladů v různém stupni přiblížení. Při prohlížení map online program ukládá na disk, a je proto možné se k nim kdykoli vrátit offline. Další možností je přímo naplánovat stažení mapových dat určené oblasti. Na výběr jsou klasické mapy, satelitní snímky a kombinace obojího.

Poskytovatelé mapových služeb, jejichž použití Google Map Catcher aktuálně podporuje:

- Yahoo maps: <http://maps.yahoo.com/>
- OpenStreetMap: <http://www.openstreetmap.org/>
- Cloudmade maps: <http://maps.cloudmade.com/>



*Obrázek 9: Rozhraní aplikace Google Map Catcher*

## 3 Návrh editoru

Tato kapitola se bude zabývat návrhem později implementovaného editoru. Bude zde představeno uživatelské rozhraní, hlavní funkce a také formát dat, se kterými tato aplikace pracuje.

### 3.1 Grafické uživatelské rozhraní

Zde budou rozebrány hlavní prvky grafické podoby aplikace. Při jejím návrhu je třeba klást důraz na přehlednost a jednoduchost s přihlédnutím k podobě samotných zobrazovaných dat. Také je vhodné už dopředu počítat s možnostmi dalšího rozšiřování aplikace a její grafický design tomu přizpůsobit.



#### 3.1.1 Mapa terénu

V této části bude pomocí OpenGL rozhraní vykreslena načtená část povrchu, se kterou bude moci uživatel pracovat a pomocí zvoleného nástroje ji upravovat. Pro větší zdůraznění tvaru povrchu bude možné přepnout pohled do módu drátěného modelu. Bude zde implementováno ovládání pohybu kamery tažením mapy určitým směrem a také změna úhlu pohledu kamery.

#### 3.1.2 Panely nástrojů

Jedná se o seznam dostupných nástrojů pro editaci terénu. Nástroje jsou umístěny v jednotlivých záložkách a jejich vybráním se daný nástroj aktivuje. Současně se zobrazí panel s podrobnostmi o nastavení daného nástroje. Které zde může uživatel modifikovat podle momentálních potřeb.

### 3.1.3 Lišta záložek

Při práci s rozsáhlejším terénem je velice užitečná možnost uložit si aktuální pozici na mapě a později se k ní zase jednoduše vrátit. A právě k tomu slouží lišta záložek. Je zde 6 volných slotů, které může uživatel k tomuto účelu využít.

### 3.1.4 Seznam vrstev

Jednoduchá nabídka, která umožňuje vytvářet vrstvy a přepínat mezi nimi. Jedná se v podstatě o různé verze upraveného terénu. Více se vrstvám věnuje kapitola 3.5.1.

### 3.1.5 Stavový panel

Poskytuje informace, jako jsou aktuální souřadnice kurzoru na mapě, výška v daném bodě, aktuálně zvolený nástroj a seznam vybraných objektů

## 3.2 Vykreslování scény

Pro vykreslování scény bylo zvoleno rozhraní OpenGL. Modely objektů jsou nahrány a vykresleny jako display listy. Pro tento účel je to naprosto postačující řešení. Modely skládají z mnoha grafických primitiv s různými vlastnostmi a povrchovými texturami. Uložení všech příkazů pro jejich vykreslení do jediného display listu velice usnadňuje následnou manipulaci s těmito objekty.

Pro práci s terénem bylo zvoleno použití vertex buffer objektů. Hlavním důvodem pro toto rozhodnutí byla skutečnost, že VBO je možné snadno modifikovat, což je výhodné při úpravách tvaru terénu a také se tím mírně urychluje proces nahrávání nových mapových dlaždic.

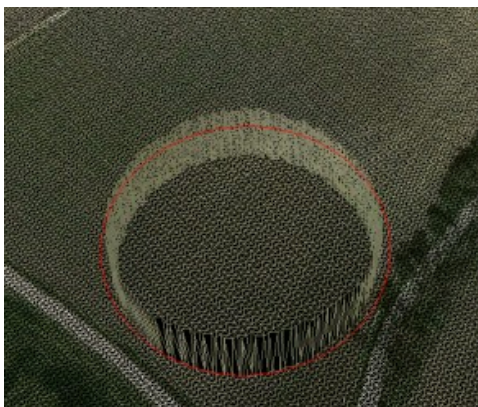
U vyvíjeného editoru není grafická stránka prioritou. Zobrazení scény by mělo být především přehledné a rychlé. Proto nebyl využit programovatelný vykreslovací řetězec (programmable rendering pipeline). Použití shaderů by sice umožnilo implementovat nejrůznější pokročilé grafické efekty, ty ovšem pro dosažení požadované úrovně zobrazení nejsou potřeba.

## 3.3 Hlavní funkce

Tato podkapitola podrobně představí jednotlivé nástroje editoru pro přetváření mapových podkladů.

### 3.3.1 Editace výškových dat

V editoru pracujícím s trojrozměrnou mapou je právě možnost upravovat tvar terénu jednou z nejvíce očekávaných funkcí. K tomuto určený nástroj po kliknutí na mapu upraví výšku zvoleného bodu, stejně jako všech okolních bodů v kruhové oblasti do zadané vzdálenosti. Nejjednodušším způsobem je prosté zvýšení či snížení bodů o určitou hodnotu (obrázek 10). Toto se může hodit při práci na detailech, ovšem při úpravách větší části terénu příliš využití nenajde. Na okrajích editované plochy totiž vznikají nepřírozené ostré hrany. Zde uživatel spíše využije možnost vyhlazení okrajů podle Gaussovy křivky. To zaručí, že bude nově vzniklá vyvýšenina nebo dolík plynule navazovat na okolní terén (obrázek 11).



Obrázek 10: Prosté zvýšení terénu o pevně danou hodnotu.



Obrázek 11: Zvýšení terénu se zarovnáním Gaussovou křivkou.

U obou metod je možné zapnout zarovnávání s nejnižším či nejvyšším bodem v okolí. V takovém případě modifikované body nepřekročí stanovenou hranici a je tak možné vytvořit rovinu. To se hodí například při korekci vodní hladiny.

Poslední mód, Zahlazování, slouží k zarovnávání ostrých hran a přechodů ve výškové mapě. Každému upravovanému bodu přiřadí výškovou hodnotu získanou jako aritmetický průměr hodnot sebe samého a osmi okolních bodů. Tento princip byl blíže popsán v kapitole 2.4. Tímto způsobem lze ostré hrany a „schody“ na mapě převést na plynulé přechody.

### 3.3.2 Úprava povrchových textur

Jemné a detailní úpravy povrchových textur již vyžadují nástroje poměrně pokročilých grafických editorů. Vývoj dalšího editačního nástroje na požadované úrovni by však bylo příliš náročné, a navíc tato oblast nepatří mezi úkoly vyplývající ze zaměření této práce. Z toho důvodu jsem zvolil variantu, kdy uživatel vybere texturu, kterou chce editovat, a ta bude otevřena v nastavené externí aplikaci. Zde se provedou úpravy a tlačítkem Aktualizovat se nový modifikovaný soubor nahraje zpět do editoru terénu. Není to zcela ideální způsob, ale je v tomto směru plně postačující.

### 3.3.3 Rozmíst'ování objektů

Na každé dostatečně podrobné trojrozměrné mapě narazíme na problém, kdy mnoho objektů, jako jsou například stromy nebo budovy, má příliš složitý tvar, než aby je bylo možné vymodelovat pouze pomocí výškové mapy. V takovém případě je výhodnější nahrát trojrozměrný model objektu a umístit jej do scény na příslušnou pozici.

Nástroj Modely obsahuje seznam úspěšně nahraných 3D objektů. Z nich je možné vybírat a v potřebném množství rozmístit na mapu. U každého modelu je možné editovat jeho vlastnosti - název, cestu k souboru a implicitní nastavení velikosti a natočení. U všech objektů je možné i po umístění měnit jejich měřítko, otáčet je, přemístit, nebo smazat.

Údaje o všech nahraných a umístěných objektech se ukládají do XML s daty o modifikacích mapy, společně s úpravami terénu.

## 3.4 Formát vstupních dat

V této podkapitole bude nastíněna konkrétní podoba souborů, se kterými editor pracuje. Možnosti, jak přizpůsobit aplikaci pro práci se vstupními daty v jiném tvaru, bude blíže nastíněna až v kapitole Chyba: zdroj odkazu nenalezen.

### 3.4.1 Kartografická data

Editor pracuje z mapou rozdělenou na jednotlivé bloky (dlaždice). Mají tvar čtverců stejné velikosti. Ke každé dlaždici náleží 2 soubory – výšková mapa a textura. Výškové mapy mohou být uloženy v souborech PNG a RAW. Mezi podporované formáty textur patří obrázky ve formátech BMP, JPG, TGA a PNG. Soubory jsou uloženy v adresářové struktuře, jejíž tvar je následující:

**map\[terrain\_tiles/texture\_tiles]\a\b\c\d.ext**

**map** – kořenová složka

**terrain\_tiles, texture\_tiles** – adresáře obsahující textury povrchu anebo výškové mapy

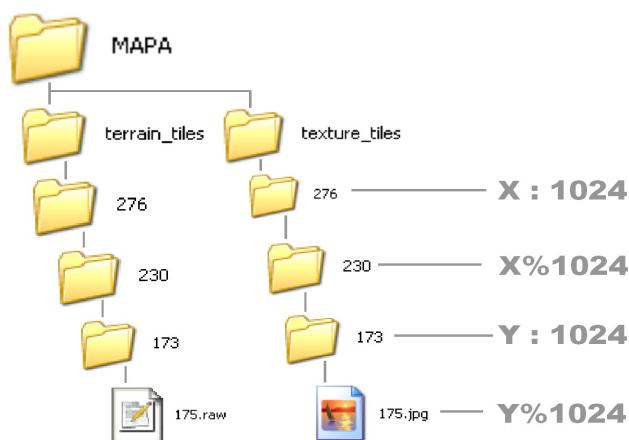
**a** – souřadnice X celočíselně vydělená konstantou 1024

**b** -zbytek po celočíselném dělení souřadnice X konstantou 1024

**c** – souřadnice Y celočíselně vydělená konstantou 1024

**d** -zbytek po celočíselném dělení souřadnice Y konstantou 1024

**ext**- přípona souboru



Obrázek 12: Adresářová struktura pro dlaždici X: 282854; Y: 177327

Kartografická data použitá jako mapové podklady byla získána ze služby ASTER GDEM (více viz kapitola 2.5.2). Výšková mapa byla dále interpolována, aby se dosáhlo dostačujícího detailu pro plynulé zobrazení tvaru terénu.

### 3.4.2 Modely objektů

Aplikace pracuje s modely objektů v podobě KMZ archivů obsahujících model uložený v COLLADA formátu. Hlavními důvody byla jeho rozšířenost, zdarma dostupné editační nástroje, jednoduchá vnitřní struktura a velké množství již hotových modelů budov poskytnutých volně ke stažení na internetu.

#### COLLADA formát

Zkratka Collada znamená COLLaborative Design Activity a představuje formát pro snadný přenos modelů mezi různými formáty a platformami. Data jsou ukládána v textové podobě jako XML dokument, a je proto velice snadné s nimi pracovat. Struktura COLLADA dokumentu není pevně definovaná, jedná se o otevřený standard. To umožňuje snadnou rozšiřitelnost a implementaci nových funkcí. Soubory mají obvykle příponu .dae. [11]

#### KMZ soubory

Jedná se o zip archiv, který kromě samotného COLLADA dae souboru obsahuje i příslušné textury a další soubory. Tento formát používá Google Earth a Google Maps pro celou řadu funkcí, nejen ukládání a načítání modelů. [12]

## 3.5 Ukládání dat

Žádné změny, které uživatel provede, se hned nepromítnou v samotných podkladových datech. Soubory, které mají být upraveny, se zkopírují jinam a teprve pak jsou modifikovány. Následně se údaje o změně zanesou do XML souboru. Zde se pro danou dlaždici vytvoří nový záznam, obsahující informace o umístění upravených souborů a také o veškerých objektech, které jsou zde umístěny.

Při následném nahrávání dlaždic terénu se nejprve vyhledají informace o nových upravených souborech. V případě, že nejsou k dispozici, se místo nich použijí soubory z původní mapy.

Všechny informace o úpravách mapy a umístěných objektech jsou uloženy v jediném XML souboru. Jinou možností by bylo přikládat samostatný XML soubor přímo do adresářové struktury ke každé upravené dlaždici. To by zjednodušilo proces načítání a údaje o změnách by byly více svázány s uloženými daty. Nakonec se ale ukázalo, že pro implementaci hromadných operací, jako například mazání všech objektů daného typu nebo zanášení změn do podkladové mapy, by to bylo naprosto nevhodné. Bylo by totiž nutné procházet celou mapu a u každé dlaždice kontrolovat, zda pro ni existuje soubor s úpravami. Proto je výhodnější mít všechna data pohromadě na jednom místě.

### 3.5.1 Vrstvy

V editoru bude možné mít pro stejnou oblast hned několik alternativních textur a výškových map. Aby se mezi sebou nepletly nesouvisející verze dlaždic, budou organizovány v tzv. vrstvách. Uživatel se mezi těmito kolekcemi bude moci přepínat a změny, které provede v jedné vrstvě, nijak neovlivní ostatní.

Ve chvíli, kdy je uživatel se všemi úpravami spokojený, může zvolit možnost zanést vrstvu do podkladové mapy. V tu chvíli modifikované soubory nahradí ty původní a dotyčná (v tu chvíli již nepotřebná) vrstva bude smazána.

Vrstvy obsahují pouze informace o editaci terénu, tedy textur a výškových map. Údaje o umístění objektů se sem nezapisují. Ty jsou pro všechny vrstvy společné.

## 4 Rozšiřování editoru

Pro další vývoj jakéhokoli software je důležitá jeho snadná modifikovatelnost a rozšiřitelnost o nové funkce. Zde popisovaný editor nepodporuje zásuvné moduly v podobě externích souborů. Modularita nebyla při jeho vývoji prioritou a při jeho úpravách se neobejdeme bez editace původního zdrojového kódu. Přesto by však postup měl být ve většině případů relativně jednoduchý.

Schéma objektového návrhu aplikace ukazující hierarchii tříd spolu s jejich stručným popisem je ukázáno na straně 19.

### 4.1 Nové nástroje

Pro přidání nového nástroje stačí vytvořit potomka třídy TTool v jednotce CustomStatus.pas. Zde nadefinujeme, jak bude nový nástroj reagovat na vstupy od uživatele, jako je pohyb myši či stisk kláves. Aby nebyly možnosti funkcionality nového nástroje omezeny, je potřeba mu umožnit přístup k proměnným a metodám z jiných částí aplikace. Toho je dosaženo pomocí několika globálních proměnných. CCursor (v jednotce KONSTANTY.pas) je záznam, který ovládá grafickou podobu kurzoru na mapě. Potřebujeme-li modifikovat hlavní okno uživatelského rozhraní, dostaneme se k němu přes proměnnou GUI (WinGUI.pas). Naprostá většina ostatních potřebných proměnných a metod je pak přístupná přes globální zastřešující proměnnou Main (MainWin.pas).

Pak už jen stačí přidat grafické rozhraní. Do panelu nástrojů ve formuláři GUI se přidá nová záložka, do níž se umístí ovládací prvky, pomocí kterých uživatel nastaví potřebné parametry. Nakonec je případně nutné při výběru záložky daný nástroj aktivovat: Pro novou záložku nadefinovat Event OnShow, který provede CurSwitch.SwitchMode(Tnázev\_třidy\_nového\_nástroje);.

### 4.1.1 Nový formát 3D modelů

V současnosti dokáže editor otevírat modely objektů ze souborů typu Collada Archiv (přípona .kmz). Abychom zprovoznili načítání modelů v jiných formátech, bude stačit drobná úprava v jednotce `Modely.pas`, ve funkci `Tmodely.loadModelFile`. Do těla funkce je potřeba přidat kód ve tvaru:

```
if AnsiEndsText('ext',lowpath)then
begin
  result:=LoadModel(ffpath);
  exit;
end;
```

'Ext' je řetězec s příponou souboru, tedy například '.3ds' nebo '.max'.

`LoadModel` je nová funkce, která bude použita k načtení modelu a vytvoření jeho displaylistu. Jako parametr může brát `ffpath` – řetězec, který obsahuje cestu k načítanému souboru. Musí vrátit výsledek jako `TglModelPtr`, což je ukazatel na záznam `TglModel` (definovaný v jednotce `glModel`). Ten kromě samotného identifikátoru OpenGL displayListu obsahuje také další informace, jako jsou název objektu, jeho rozměry a také seznam identifikátorů použitých textur. Pokud se model nepodaří načíst, funkce vrátí jako svůj výsledek `nil`.

## 4.2 Nový formát mapy

Zde budou nastíněny způsoby, jak přizpůsobit editor pro práci s mapovými podklady odlišného formátu.

### 4.2.1 Nové typy souborů

Editor pracuje s povrchovými texturami v souborech typu PNG nebo JPEG. Výškové mapy načítá ze souborů s příponami RAW či PNG. Samotný typ právě používaných formátů lze nastavit v souboru `Settings.xml`. Je však poměrně snadné přidat podporu pro další formáty (samozřejmě pokud máme již k dispozici algoritmus, který provede samotné načtení potřebných dat ze souboru). V jednotce `Tile` jsou definovány metody `TTile.loadHmap` a `TTile.loadTex`. V nich jsou `case` přepínače, které podle typu souboru vyberou vhodnou funkci. Takže stačí přidat položku s názvem nové přípony, která provede načtení dat ze souboru daného typu.

Samotné načítací funkce berou jako parametr řetězec s cestou k požadovanému souboru. Jako svůj výsledek vracejí logickou hodnotu `TRUE`, pokud vše proběhne úspěšně, anebo `FALSE`, když načítání selže. Při načítání textury dojde v vytvoření OpenGL textury ze zadaného obrázku a uložení jeho identifikátoru do proměnné `TTile.texture`. Funkce sloužící k načtení výškové mapy otevře zadaný soubor a postupně z něj načte hodnoty do dvourozměrného pole `TTile.heightmap`.

### 4.2.2 Odlišná struktura mapových podkladů

Podoba adresářového stromu, ve kterém je uložena mapa, byla popsána v kapitole 3.4. V konfiguračním souboru `Settings.xml` jsou definovány názvy adresářů pro textury, výškové mapy, zvolený detail a také přípony samotných souborů. Pokud však chceme pracovat s mapou, která má odlišnou samotnou strukturu složek, je nutné přepsat tělo funkce, která se stará o překlad souřadnic dlaždice na cestu k jednotlivým souborům. Konkrétně se jedná o funkci `TilePath`, definovanou v jednotce `KONSTANTY.pas`. V současné implementaci bohužel neexistuje žádný přepínač, kterým by se dalo jednoduše vybrat jeden z několika možných způsobů překladu souřadnic.

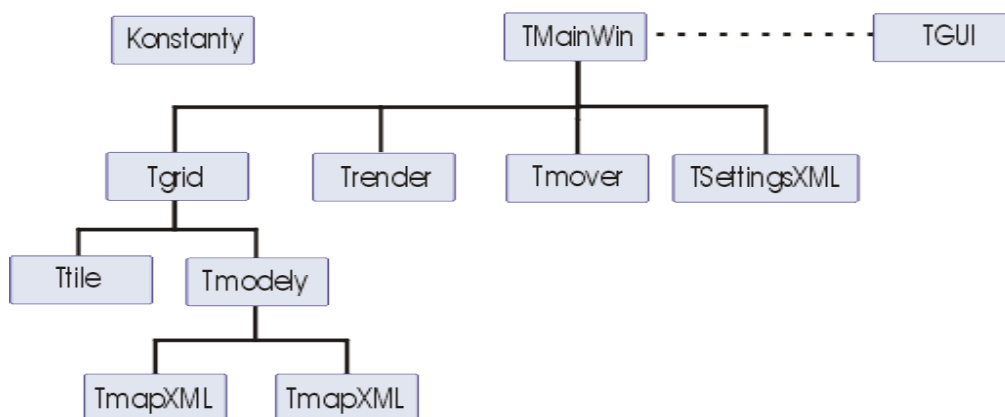
# 5 Implementace

Program TerraEdit.exe byl napsán v jazyce Object Pascal. ve vývojovém prostředí Borland Delphi 7. Jako hlavní přednost tohoto vývojového nástroje vidím rychlé a snadné vytváření uživatelského rozhraní a také relativně rychlé a dobře optimalizované výsledné binární soubory. Na druhou stranu už není v dnešní době tolik rozšířený a dal by se považovat za mírně zastaralý, což se při vývoji projevilo především nedostatečnou nativní podporou funkcí novějších verzí OpenGL rozhraní, kterou jsem tak byl nucen dodatečně implementovat.

Výsledná aplikace byla testována v CVT na operačním systému Windows XP.

## 5.1 Programová struktura

V této podkapitole bude naznačena podoba objektového návrhu implementace s popisem hlavních součástí. Podpůrné a méně důležité součásti byly pro zjednodušení vynechány.



Obrázek 13: Schéma objektového návrhu editoru

- **Konstanty** – jednotka obsahující globální konstanty, proměnné a funkce, ke kterým je potřeba mít přístup na více místech programovém kódu.
- **TMainWin** -hlavní zastřešující třída, která umožňuje přístup k dalším nejpodstatnějším částem aplikace
- **TRender** – vykreslování OpenGL, ovládání kamery
- **TGrid** – matice dlaždic. Zajišťuje většinu funkcí pro manipulaci s mapou. Základem této třídy je dvojrozměrné pole instancí třídy TTile. Tilegrid pak obstarává jejich postupné načítání při pohybu kamery, tvorbu VBO objektů z výškových map, umístování objektů ale také třeba úpravu tvaru terénu více dlaždic najednou.
- **TTile** – jedna dlaždice terénu. Obsahuje načtená data o výškové mapě, použité textuře i umístěných objektech.
- **TMover** – zajišťuje synchronizaci mezi pohybem kamery a načítáním terénu.
- **TcurSwitch** – třída, která se stará o přepínání mezi používanými nástroji.
- **TModely**- seznam použitých modelů a příslušné operace, jako jejich načítání nebo odstraňování.
- **TMapXML** – rozhraní pro čtení a zápis úprav mapy do xml souboru.
- **TGUI** – uživatelské rozhraní, hlavní okno celé aplikace.
- **TLoadModel** – třída pro načítání modelů, provádí parsování COLLADA archivů.
- **TSettingsXML** – třída, která má na starost načítání a ukládání nastavení aplikace do XML souboru.

## 5.2 Načítání dlaždic terénu

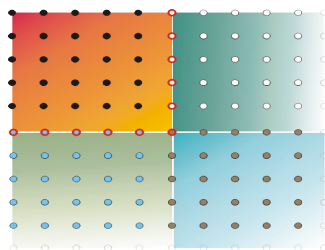
Program sleduje pohyb kamery, a jakmile v jednom směru překročí stanovenou hranici, je nutné načíst nový úsek terénu. Nejprve se vyčistí dlaždice, které již nejsou potřeba. Smaže se jejich povrchová textura i seznam umístěných objektů a označí se jako prázdné.

Do těchto uvolněných pozic se začnou nahrávat data z nových souřadnic. Nejprve se načte seznam všech vrstev pro danou dlaždici. Pokud má dlaždice upravenou texturu nebo výškovou mapu v aktuálně zvolené vrstvě, nahrají se příslušné soubory. Pokud ne, použijí se původní mapová data.

Ze souboru s texturou se vytvoří OpenGL textura a načte se pole s výškovou mapou. Tato operace je časově poměrně náročná, je třeba otevřít a přečíst větší množství souborů. Aby se minimalizoval negativní dopad na plynulost programu, jsou požadavky na načítání nových dlaždic řazeny do fronty a vykonávány postupně v několikamilisekundových intervalech.

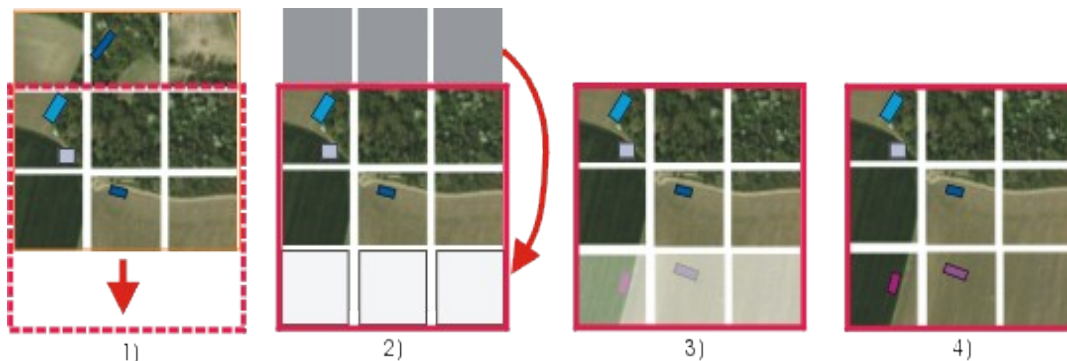
Následuje načtení seznamu umístěných objektů a jejich parametrů. Informace o objektech se nacházejí v XML ve stejné poloze jako data o vrstvách, a proto je jejich vyhledávání zoptimalizováno.

Nakonec je potřeba vytvořit 3D model dlaždice. A zde nastává problém. Sousedící výškové mapy se totiž na okrajích vzájemně nepřekrývají, a my tak nemáme dostatek informací k tomu, abychom z jediné výškové mapy vytvořili model plynule navazující na ostatní. Je totiž potřeba znát i výšku hraničních bodů ze tří sousedících dlaždic. U dostatečně detailních výškových map by se tato komplikace dala ignorovat – pracovat s výškovými daty, jako by se překrývaly. Tím by se zjednodušil a mírně urychlil proces načítání -ovšem za cenu vzniku občasných grafických anomálií u výrazněji zvlněného terénu.



Obrázek 14: Při tvorbě modelu dlaždice (oranžová) jsou potřeba i výškové body okolních dlaždic (červeně označené)

Implementace TeraEdit nakládá v tomto ohledu s výškovými daty korektně a pracuje i s potřebnými hodnotami mimo aktuální dlaždici. Soubory si načte dlaždice sama, ovšem tvorbu modelu má na starosti nadřazená matice dlaždic (třída TGrid), která má přístup ke všem nezbytným informacím. Načítání souborů a vytváření modelu je časově uspořádáno tak, aby byla všechna potřebná data již k dispozici. A pokud tomu tak náhodou není (např. při velmi rychlém chaotickém pohybu a nízké rychlosti načítání), provede se jejich urychlené načtení.



Obrázek 15: Fáze posunu: 1) zahájen posun matice; 2) smazání obsahu nepotřebných dlaždic; 3) nahrání nových textur, výškových map a modelů; 4) vytvoření modelů dlaždic a jejich zobrazení

## 5.3 Načítání 3D modelů

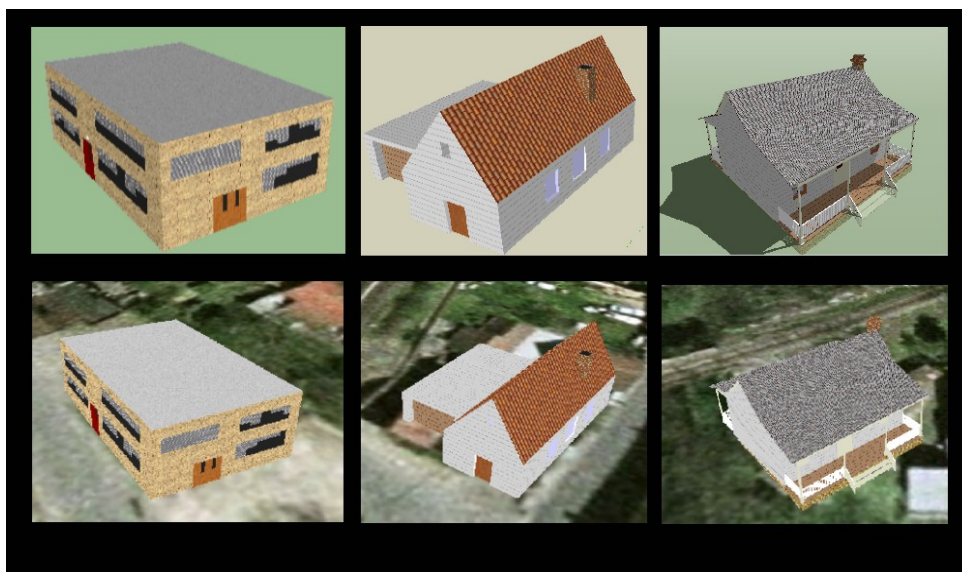
Pro aplikace vyvíjené pomocí Borland Delphi je naprostý nedostatek volně dostupných knihoven pro načítání trojrozměrných modelů do rozhraní OpenGL. Z toho důvodu je součástí implementace i vlastní parser pro načítání COLLADA souborů (více v kapitole 3.4.2).

### 5.3.1 Specifikace knihovny a její omezení

Vyvinutá knihovna zvládá načítání COLLADA modelů v KMZ archivech. Struktura těchto archivů se v jednotlivých verzích liší. Tato knihovna byla primárně vyvíjena pro parsování souborů ve formátu určeném pro Google Earth 4. Při načítání jsou z archivu vyextrahovány soubory s texturami i se samotným modelem. Z nich je pak sestaven OpenGL Display List. Průběhu načítání se podrobněji věnuje následující podkapitola.

Možnosti knihovny jsou zatím značně omezené. Byla vyvinuta pro vytváření Display Listů, což postačuje v rámci této konkrétní aplikace. V dnešní době je však mnohem rozšířenější metoda vykreslování přes Vertex Buffer Objekty, případně pomocí Shaderů. Světelné vlastnosti povrchových ploch byly značně zjednodušeny a zčásti dokonce ignorovány. Mnohé modely také obsahují efekty a vlastnosti ploch, které nebyly implementovány, a nemusí být proto zobrazeny správně. Na obrázku 16 je vidět srovnání podoby modelu v aplikaci Google SketchUp, použité k jeho vytvoření a poté níže stejný model zobrazený v programu TerraEdit.

Knihovna v současné podobě postačuje pro načítání jednoduchých modelů, ovšem pro její širší využití mimo tento projekt je nezbytný další vývoj.



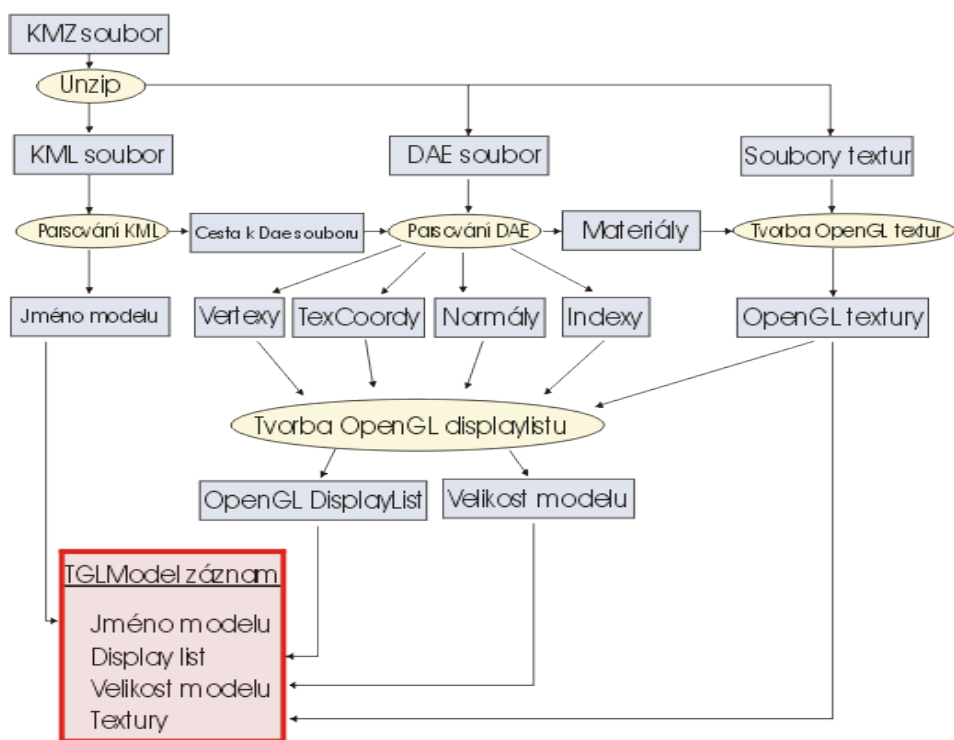
Obrázek 16: Nahoře modely v Google SketchUp, dole zobrazeny v TerraEdit.

### 5.3.2 Průběh načítání modelu

Načítání modelu z KMZ souboru probíhá následovně: Nejprve je třeba archiv rozbalit. K tomu je použita knihovna unzip32.dll . Dalším krokem je nalezení KML souboru ve vyextrahované složce (podle specifikace musí každý KMZ archiv obsahovat pouze jediný KML soubor). Zde jsou obsaženy především informace o geografickém umístění modelu na mapě světa, což nás nezajímá. Jediné užitečné údaje jsou název modelu a cesta k dae souboru.

Nyní již můžeme přistoupit k samotnému parsování COLLADA souboru. V seznamu materiálů se postupně ke každému použitému materiálu dohledá přidělený soubor s texturou (ze které se vygeneruje OpenGL textura), anebo alespoň barva. Následně se načtou všechna pole vertexů, indexů, normál a souřadnic textur.

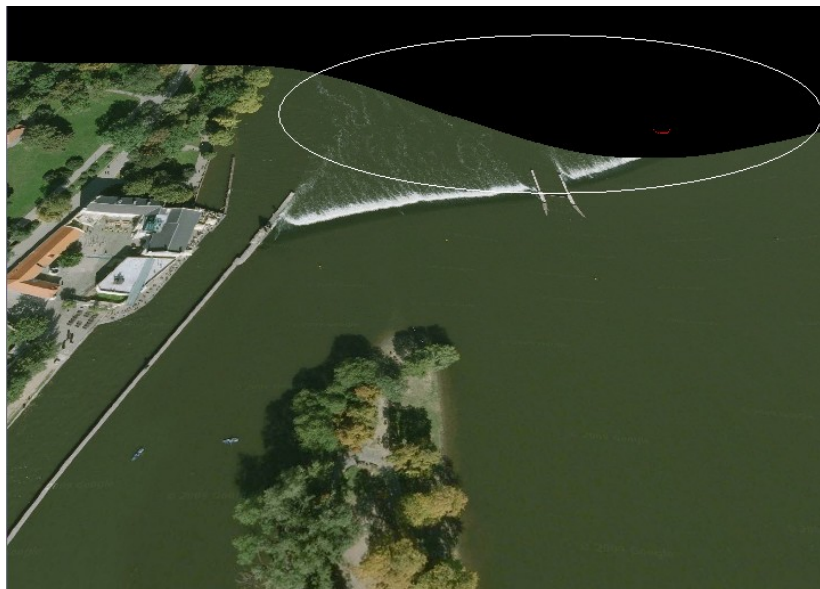
Tím jsou shromážděna všechna potřebná data pro sestavení modelu. Takže už jen stačí postupně všechny příkazy pro vykreslení jednotlivých trojúhelníků uložit do display listu. Ten je vrácen jako výsledek společně s doprovodnými informacemi, jako je název modelu, rozměry a také seznam načtených textur (aby je bylo možno později vymazat z paměti grafické karty).



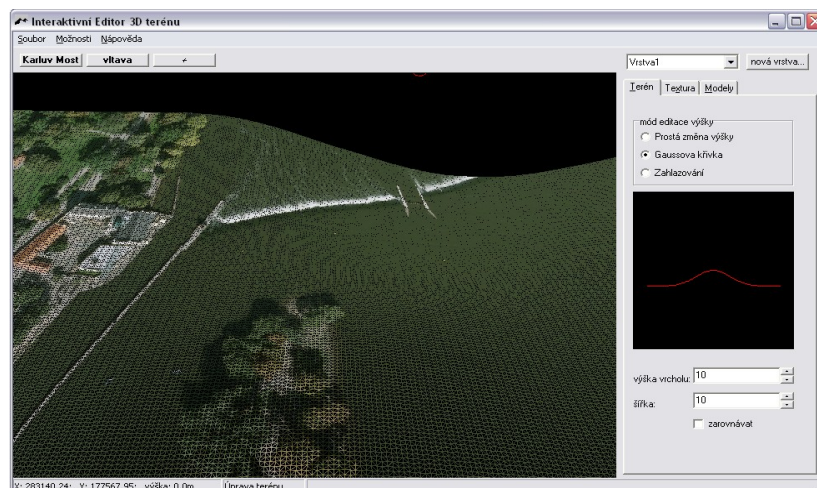
Obrázek 17: Postup načítání modelu

## 6 Vzorová scéna

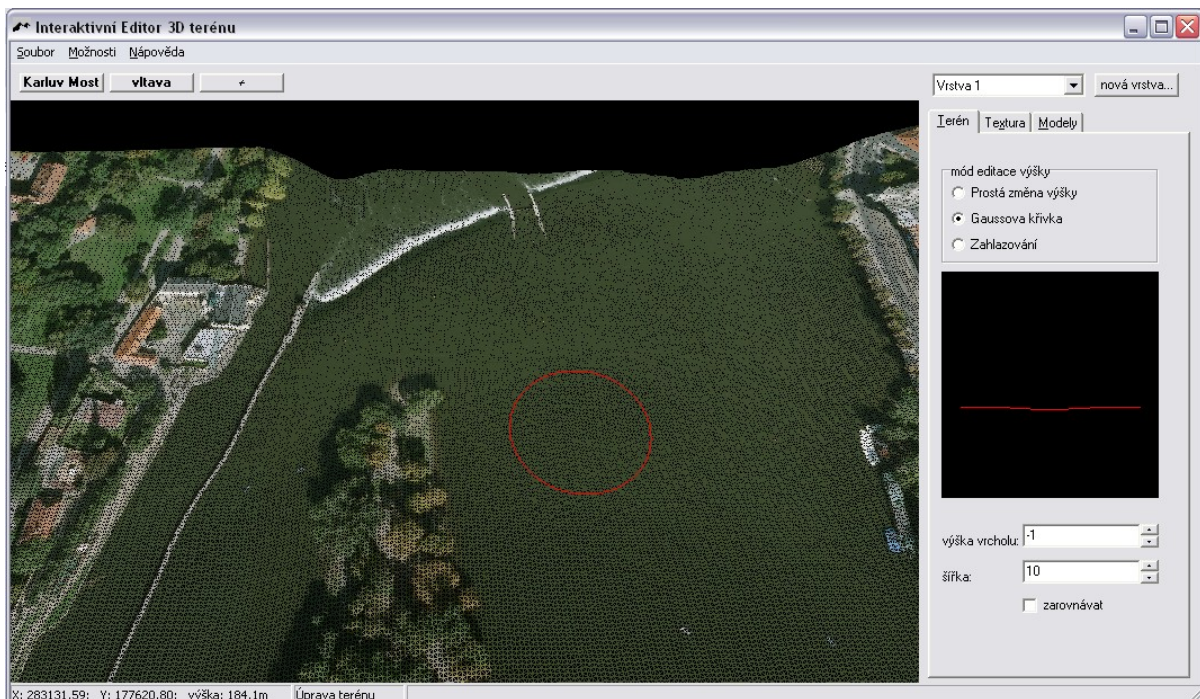
V této kapitole bude předvedeno použití implementovaného editoru. Série obrázků bude ilustrovat postupné odstraňování nepřesností ve zdrojové výškové mapě, rozmístění několika objektů a úpravu textur.



Obrázek 18: Koryto řeky. Jak lze vidět na zvlněném "horizontu", výšková data jsou velice nepřesná.



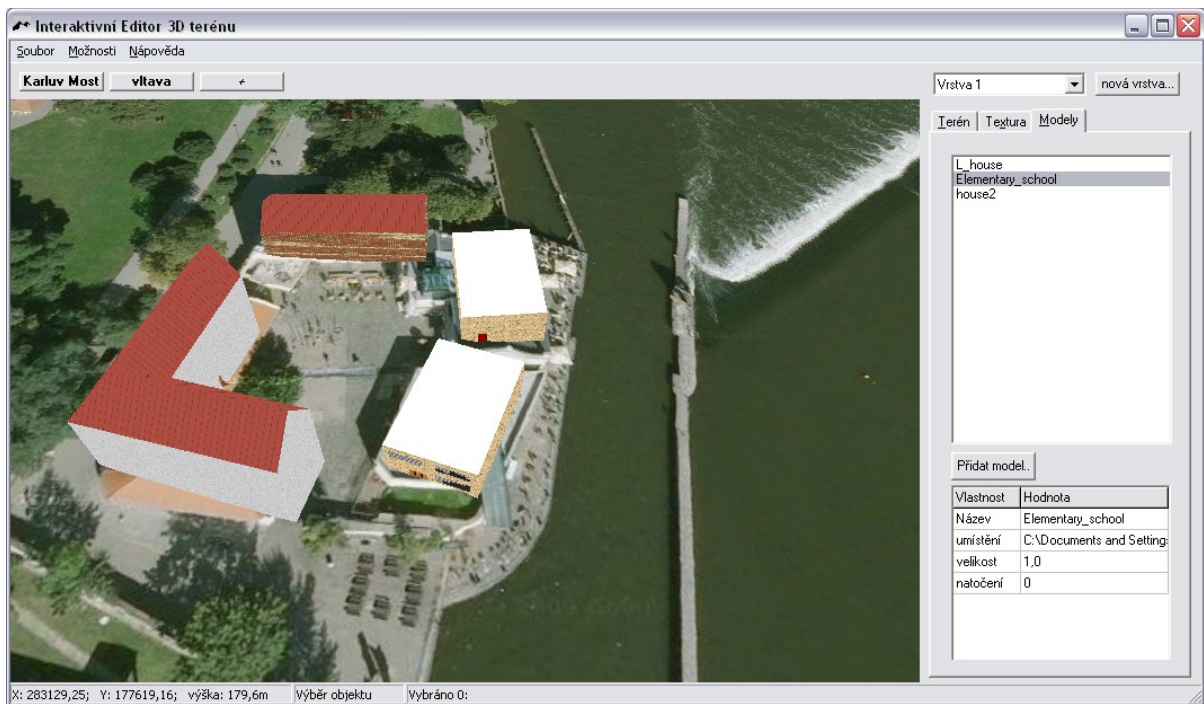
Obrázek 19: Pro zřetelnější vykreslení tvaru terénu je vhodné přepnout režim zobrazení na "drátěný model".



*Obrázek 20: Pomocí nástroje pro úpravu terénu v režimu "Gaussova křivka" byla hladina řeky vytvarována do přibližně správné výšky. Stále je však poměrně nerovná.*



*Obrázek 21: Použitím nástroje Zahlazování bylo docíleno rovné hladiny řeky.*



Obrázek 22: Několik budov bylo nahrazeno 3D modely.



Obrázek 23: V externím grafickém editoru byla otevřena textura dlaždice a vyretušovány všechny objekty na nádvoří.

## 7 Závěr

V rámci této bakalářské práce se podařilo vytvořit Interaktivní 3D Editor terénu. Od počátku byl vyvíjen se zaměřením na úpravy a korekce existujících mapových podkladů. Právě v oblasti kartografie vidím jeho hlavní využití. Poskytuje sadu jednoduchých, ale přesto použitelných a dostačujících funkcí pro základní editaci trojrozměrného terénu. Mohl by se stát jedním z nástrojů použitých při sestavování takovýchto map, pro jednoduché a rychlé odstraňování chyb, odchylek a nepřesností.

Využití tohoto editoru pro jiné účely je poněkud omezené. Například pro vytváření nového terénu zde nejsou implementovány funkce jako je generování náhodných výškových map nebo umístování textur v závislosti na tvaru terénu. Pro využití jako prohlížeč mapy by bylo potřeba zdokonalit program po grafické stránce a také přidat možnost načítat mnohem větší plochu povrchu s dodatečnými informacemi v různém stupni detailu.

### 7.1 Další pokračování projektu

Další vývoj projektu je možný ve více směrech. Při použití k editaci dodaných dat by bylo nezbytné implementovat podporu více různých formátů map. Anebo alespoň těch nejpoužívanějších. Rozumným krokem by v tomto směru nabídnout uživatelům možnost, aby si mohli jednoduše nadefinovat formát vstupních dat sami – například konfiguračním souborem.

S tím také souvisí podpora více formátů 3D modelů, opět především těch nejpoužívanějších, jako je AutoCAD nebo Autodesk 3ds Max.

Dále je pak několik funkcí, které by proces editace mohly velmi usnadnit, ovšem ve stávající implementaci nejsou obsaženy. Především bych zmínil možnost hromadného rozmístování populací objektů na určených oblastech. To by velice urychlilo například modelování lesních porostů.

Dalším směrem je odstranění nedostatků stávající implementace. Mám na mysli především optimalizaci kódu. Například při pohybu mapy je třeba načíst data z velkého množství souborů. To je časově velmi náročná operace a na strojích s pomalejším pevným diskem může mít negativní dopad na plynulost chodu celé aplikace. Tento problém by částečně zmírnila implementace načítání souborů v samostatném vlákně. O omezeních použitého parseru COLLADA souborů už psáno v kapitole 5.3.1. Je třeba jej zdokonalit a rozšířit jeho schopnosti.

Současná aplikace přenechává editaci textur externím grafickým editorům. Bylo by ale vhodné vyvinout (alespoň jako alternativu) vlastní dostatečně robustní grafický editor, aby se eliminovala závislost na aplikacích třetích stran.

Anebo je možné se při vývoji editoru ubírat směrem rozšiřování jeho funkčnosti tak, aby byl použitelný i v oblastech, pro které nebyl primárně vyvíjen. Mohlo by se například jednat o již zmíněné vytváření nového terénu - implementace algoritmů pro náhodné generování terénu, rozmístování řek, textur různých povrchů atd. Také v realističnosti zobrazení má TerraEdit značné rezervy a bylo by možné přidat prvky jako stínování terénu a modelů, mlhu, různé světelné vlastnosti povrchů a podobně.

## 8 Literatura

- [1] PlanetSide Software: Terragen, [ONLINE] , dostupné na: <http://www.planetside.co.uk>
- [2] ŽÁRA, Jiří. Moderní počítačová grafika. 2., přeprac. a rozš. vyd. Praha : Computer Press, 2004. 609 s., 16 s. barev. obr. příl ISBN 8025104540
- [3] OpenGL reference pages, [ONLINE], dostupné na: <http://www.opengl.org/sdk/docs/man/>
- [4] Graham Wihlidal : Box Filtering Height Maps for Smooth Rolling Hills, [ONLINE], dostupné na: [http://www.gamedev.net/page/resources/\\_/reference/programming/sweet-snippets/box-filtering-height-maps-for-smooth-rolling-hills-r2164](http://www.gamedev.net/page/resources/_/reference/programming/sweet-snippets/box-filtering-height-maps-for-smooth-rolling-hills-r2164)
- [5] Radek Luner: Optimalizace renderování rozsáhlého terénu, diplomová práce, Brno, FIT VUT v Brně, 2010
- [6] Daniel Beard: Diamond square algorithm, [ONLINE], dostupné na: <http://danielbeard.wordpress.com/2010/08/07/terrain-generation-and-smoothing/>
- [7] Wikipedia: Perlin Noise [ONLINE] , dostupné na: [en.wikipedia.org/wiki/Perlin\\_noise](en.wikipedia.org/wiki/Perlin_noise)
- [8] Wikipedia: ČÚZK, [ONLINE], dostupné na: [http://cs.wikipedia.org/wiki/český\\_úřad\\_zeměměřický\\_a\\_katastrální](http://cs.wikipedia.org/wiki/český_úřad_zeměměřický_a_katastrální)
- [9] Český úřad zeměměřičský a katastrální: Geoportál ČÚZK, [ONLINE], dostupné na: <http://geoportal.cuzk.cz/>
- [10] Earth Remote Sensing Analysis Center: ASTER GDEM, [ONLINE], dostupné na: <http://www.ersdac.or.jp/GDEM/E/>
- [11] Kronos Group, COLLADA, [ONLINE], dostupné na: [http://www.khronos.org/developers/library/overview/collada\\_overview.pdf](http://www.khronos.org/developers/library/overview/collada_overview.pdf)
- [12] Google: Documentation - KMZ archive, [ONLINE], dostupné na: <http://code.google.com/apis/kml/documentation/kmzarchives.html>

# Seznam příloh

Příloha 1. Popis Ovládání editoru

Příloha 2. DVD s implementovaným editorem terénu, zdrojovými kódy, vzorovými vstupními daty, kopií zprávy a dalšími soubory.

# 9 Příloha 1

## 9.1 Ovládání aplikace

Zde bude podrobně vysvětleno ovládání jednotlivých prvků implementovaného editoru.

### 9.1.1 Ovládání kamery

Pohyb po mapě je řešen tak, aby byl pokud možno co nejintuitivnější a zároveň k němu stačila pouze myš. Pro posouvání mapy stačí stisknout pravé tlačítko myši a mapu potáhnout pořadovaným směrem. Případně lze použít klávesy 'W','A','S','D'. Výšku kamery lze upravit pohybem kolečka myši, anebo kombinací kláves 'R' a 'F'. Chce-li uživatel změnit úhel pohledu kamery, je třeba stisknout a držet pravé tlačítko myši a poté stisknout kolečko. Poté již může pohybem kurzoru volně ovládat úhel natočení kamery, dokud nepustí pravé tlačítko.

### 9.1.2 Panel nástrojů

Jedná se nabídku nástrojů pro úpravu terénu. Jednotlivé nástroje se vyberou a aktivují kliknutím na příslušnou záložku v záhlaví panelu. Tím se zobrazí rozhraní, kde lze nastavit parametry nástroje.

#### **Nástroj Terén:**

Tento nástroj umožňuje měnit výšku terénu, vytvářet kopce, údolí, nebo je zarovnávat. V Panelu nástroje si může Uživatel nastavit poloměr plochy, která bude upravena, a také výšku v metrech, o kolik bude povrch zvednut či snížena v bodě kliknutí. Zatržením položky "zarovnávat" bude výška terénu zarovnána s nejvyšším (resp. nejnižším) bodem v upravované oblasti.

Samotná editace výšky terénu se provede kliknutím levým tlačítkem myši na vybranou oblast. Pokud je při kliknutí stisknut 'Ctrl', provede se inverzní změna výšky, tzn. terén se místo zvýšení o stejnou hodnotu sníží a naopak.

#### **Nástroj Textura:**

Nástroj sloužící k editaci povrchových textur jednotlivých dlaždic povrchu. Uživatel si levým tlačítkem myši vybere požadovanou dlaždici. Tlačítkem "Upravit texturu" v panelu nástroje ji pak otevře ve vybraném externím grafickém editoru (pokud není žádný editor vybrán, zobrazí se dialogové okno pro jeho výběr). Po provedení a uložení změn pak stačí kliknout na tlačítko "Aktualizovat", čímž se provede znovunačtení příslušné dlaždice.

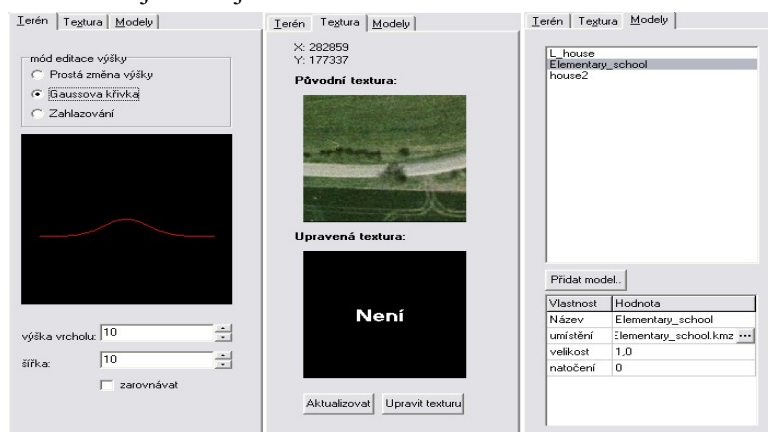
#### **Nástroj Modely:**

Nástroj pro umístování a úpravu 3D objektů na mapě. V horní části Panelu je seznam všech nahraných modelů, z nichž je možné vybírat. Nový model nahrajeme tlačítkem "Přidat model". Naopak jejich smazání z mapy se provede označením dotyčných modelů, pravým tlačítkem myši

vyvoláme kontextovou nabídku a vybereme položku "Smazat označené". Označením určitého modelu se v dolní části zobrazí jeho details, jako je název, umístění souboru nebo také jeho implicitní velikost a natočení. Všechny tyto hodnoty je možné upravovat. Pro umístění nového objektu na mapu stačí kliknutím vybrat konkrétní model ze seznamu a dalším kliknutím jej položit na vybranou pozici. Chceme-li umístit model ve více exemplářích, je nutné při pokládání podržet klávesu 'Shift'.

Samozřejmě je možné upravovat už položené objekty. Nejprve je potřeba kliknutím myši daný objekt vybrat (výběr více objektů najednou se provádí se stisknutým 'Shift'em).

Pak lze klávesami 'J' a 'L' měnit jeho natočení a kombinace kláves 'I' a 'K' slouží k úpravě jeho velikosti. Klávesou 'Delete' smažeme vybrané objekty. Pro přesun na jiné místo je potřeba na již vybraný objekt kliknout ještě jednou a následně kliknout na nově vybranou pozici (celou operaci lze přerušit klávesou 'Escape'). Opět je možné se stiskem klávesy 'Shift' objekt umístit vícekrát. Bohužel není možné přemísťovat více objektů najednou.



Obrázek 24: Detaily jednotlivých nástrojů

### 9.1.3 Záložky

Umožňují uložit si určitou pozici na mapě a později se k ní zase vrátit. Je možné si takto uložit až 6 pozic. Vytvoření nové záložky se provede kliknutím na tlačítko '+'. Návrat na uložené souřadnice se provede po kliknutí levým tlačítkem na příslušnou záložku. Pro odstranění konkrétní záložky na ni stačí kliknout pravým tlačítkem a z kontextového menu vybrat položku „Smazat záložku“.

### 9.1.4 Seznam vrstev

Slouží k přepínání kontextu, ve kterém jsou úpravy terénu prováděny. Přidání nové vrstvy se provede tlačítkem „Nová vrstva“. Poté se pro ni v dialogovém okně zvolí název a adresář, kam se budou ukládat modifikované soubory.

Změny provedené v příslušné vrstvě lze zapsat do podkladové mapy. Kliknutí pravým tlačítkem myši na nabídku záložek vyvolá kontextové menu, ve kterém zvolíme položku „Uložit do zdrojové mapy“.

Ve stejné kontextové nabídce je také možnost smazat vrstvu. Přitom se odstaní všechny údaje o vrstvě z XML a smažou se všechny modifikované soubory i vygenerovaný adresářový strom.

### 9.1.5 Klávesové zkratky:

- Ctrl+B - Nová záložka
- Ctrl+J - Skočit na zadané souřadnice

- Ctrl+T - Vybere nástroj pro úpravu terénu.
- Ctrl+X - Vybere nástroj pro úpravu povrchové textury.
- Ctrl+M - Vybere nástroj pro vkládání a editaci modelů.
- F1 -Zobrazí okno „O aplikaci“.
- F2 - Přepíná mezi plným zobrazením a drátěným modelem.