

IrrEdit

Tvorba 3D scén

IrrEdit
3D scenes creation

Tomáš Kavecký



Univerzita Tomáše Bati ve Zlíně

Fakulta aplikované informatiky

Ústav aplikované informatiky

akademický rok: 2008/2009

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Tomáš KAVECKÝ**

Studijní program: **B 3902 Inženýrská informatika**

Studijní obor: **Informační technologie**

Téma práce: **IrrEdit – tvorba scén pro Irrlicht engine**

Zásady pro vypracování:

1. Vypracujte literární rešerši na dané téma.
2. Podrobně se seznamte s prostředím programu irrEdit v jeho aktuální verzi.
3. V práci podrobně charakterizujte program irrEdit včetně oblasti jeho využití.
4. Vytvořte elektronickou dokumentaci k programu irrEdit a umístěte ji na CD do přílohy práce.
5. Demonstrujte možnosti IrrEditu na praktickém příkladu komplexní scény. Pro tuto scénu si vytvořte modely v programu Blender a textury v programu Gimp.
6. Vytvořenou scénu včetně modelů a textur umístěte opět do elektronické přílohy práce.

Rozsah práce:

Rozsah příloh:

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

1. **POKORNÝ, Pavel. Blender – naučte se 3D grafiku. 1. vyd. Praha : BEN – technická literatura, 2006. 247 s. ISBN 8-7300-203-5.**
2. **Ambiera:irrEdit – a free realtime 3D world editor and radiosity lightmap generator [online]. 2007 [cit. 2009-01-23]. Dostupný z WWW:<http://www.ambiera.com/irredit>.**
3. **BlenderWiki [online]. 2006 [cit. 2009-01-23]. Dostupný z WWW:<http://wiki.blender.org> .**
4. **Blender Art magazine [online]. 2005 [cit. 2009-01-23]. Dostupný z WWW:<http://www.blenderart.org> .**
5. **GIMP – the GNU Image manipulation program [online]. 2007 [cit. 2009-01-23]. Dostupný z WWW:<http://www.gimp.org> .**

Vedoucí bakalářské práce:

Ing. Pavel Pokorný, Ph.D.

Ústav aplikované informatiky

Datum zadání bakalářské práce:

20. února 2009

Termín odevzdání bakalářské práce:

1. června 2009

Ve Zlíně dne 13. února 2009



prof. Ing. Vladimír Vašek, CSc.

děkan



doc. Ing. Ivan Zelinka, Ph.D.

ředitel ústavu

ABSTRAKT

Cílem této práce je podrobně popsat editor pro tvorbu 3D scén IrrEdit, ve kterém lze použít 3D objekty vytvořené v Blenderu a otexturované v programu GIMP. Praktická část obsahuje příručku k IrrEditu a ukázkou tvorby scény v tomto editoru.

Klíčová slova: IrrEdit, IrrLicht Engine, Blender, Gimp

ABSTRACT

The aim of my bachelor thesis is to describe the editor for creating 3D scenes IrrEdit, where can be used 3D objects of Blender and textured by the programme GIMP. The practical part includes a user guide for IrrEdit and preview of making the scenes in this editor.

Keywords: IrrEdit, IrrLicht Engine, Blender, Gimp

Rád bych touto cestou poděkoval vedoucímu své práce Ing. Pavlu Pokornému PhD. Za veškeré rady, připomínky a dohled, který této práci věnoval. Také svým rodičům za podporu a neutuchající povzbuzování.

Prohlašuji, že

- beru na vědomí, že odevzdáním bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – bakalářskou práci nebo poskytnout licenci k jejímu využití jen s předchozím písemným souhlasem Univerzity Tomáše Bati ve Zlíně, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše);
- beru na vědomí, že pokud bylo k vypracování bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval.

V případě publikace výsledků budu uveden jako spoluautor.

Ve Zlíně

.....
podpis diplomanta

OBSAH

ÚVOD.....	9
I TEORETICKÁ ČÁST	10
1 IRREDIT	11
1.1 POPIS EDITORU	11
1.1.1 Lightmap generátor	12
1.1.2 Podporované souborové formáty.....	13
1.1.3 World,Terrain a Particle system editor.....	14
1.1.4 Pluginy a skriptování.....	15
1.2 HISTORIE IRREDITU	15
1.3 POPIS ROZHRANÍ	18
1.3.1 Zobrazovací pole	18
1.3.2 SceneGraphExplorer	19
1.3.3 Properties.....	19
1.3.4 Lightmapping	20
1.3.5 Message Log.....	20
1.3.6 Textury	20
1.3.7 Window Managment.....	21
1.4 ZÁKLADNÍ OPERACE	21
1.4.1 Tvarová manipulace	21
1.4.2 Orientace a posun objektů	21
1.4.3 Textury	21
1.4.4 Kamera	21
1.4.5 Částicový systém.....	22
1.4.6 Billboard.....	22
1.4.7 Terén.....	22
1.4.8 Model	22
1.4.9 Animovaný model	22
1.4.10 OctTree.....	22
1.4.11 Wireframe.....	22
1.4.12 Lightmapping	23
1.4.13 Prohlížení výsledků	24
1.4.14 Skriptování	24
II PRAKTICKÁ ČÁST	27
2 TVORBA SCÉNY POMOCÍ IRREDITU.....	28
2.1 ZÁKLAD SCÉNY	28
2.1.1 Vytvoření Slunce a mlhovin.....	29
2.2 TVORBA MODELŮ.....	31
2.2.1 Modelování objektů	32
2.2.2 Texturování objektů	36
2.2.3 Animace objektů	38
2.3 VÝSLEDNÁ SCÉNA.....	40
2.3.1 Úprava jednotlivých planet a dalších objektů	40

2.3.2	Úprava SkyBoxu a prohlížení výsledné scény	41
2.3.3	Spuštění v IrrLicht Enginu	44
ZÁVĚR		46
RESULT		47
SEZNAM POUŽITÉ LITERATURY.....		48
SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK.....		49
SEZNAM OBRÁZKŮ		50
SEZNAM TABULEK.....		52
SEZNAM PŘÍLOH.....		53
PŘÍLOHA PI:CD		54

ÚVOD

Technologický pokrok, zasahující různá technická odvětví od automobilů až po spotřební elektroniku, ovlivňuje i vývoj počítačových komponentů a softwaru. Vývoj počítačových technologií je pro normálního člověka asi nejznatelnější. Uživatel má čím dál vyšší nároky na výkon počítače samotného díky stále výkonnějším programům nebo počítačovým hrám, kde se jedná především o výkon grafické karty pro výpočty daných 3D modelů.

Počítačová trojrozměrná grafika (dále jen 3D grafika) využívá 3D reprezentaci geometrických dat, která jsou uložena v počítači a použita pro výpočty a renderování 2D obrázků. 3D grafika je, přes zdánlivou neslučitelnost, s 2D prostorem velmi úzce spjata. 3D grafika se opírá o velmi podobné či stejné algoritmy jako 2D vektorová grafika v drátěném modelu a 2D rastrová grafika ve finálním renderovém vyobrazení. [8]

Základem každé aplikace je engine. Engine je jádro počítačové hry nebo programu. Některé firmy vyvíjejí vlastní engine, jiné používají engine jiných firem a existují i volně šířitelné jako IrrLicht engine. Grafický engine pracuje jak s 3D, tak s 2D grafikou a nejčastěji se používá při vývoji počítačových her. Do enginu je možné doplnit různé prvky jako např. modely či úrovně (levely). [7]

Existuje nespočet programů pro tvorbu modelů a prostorových scén a jejich následná aplikace v konkrétním enginu. Avšak pořízení velké řady z nich není nejlevnější záležitost, a proto nastupují volně šířitelná vývojová prostředí.

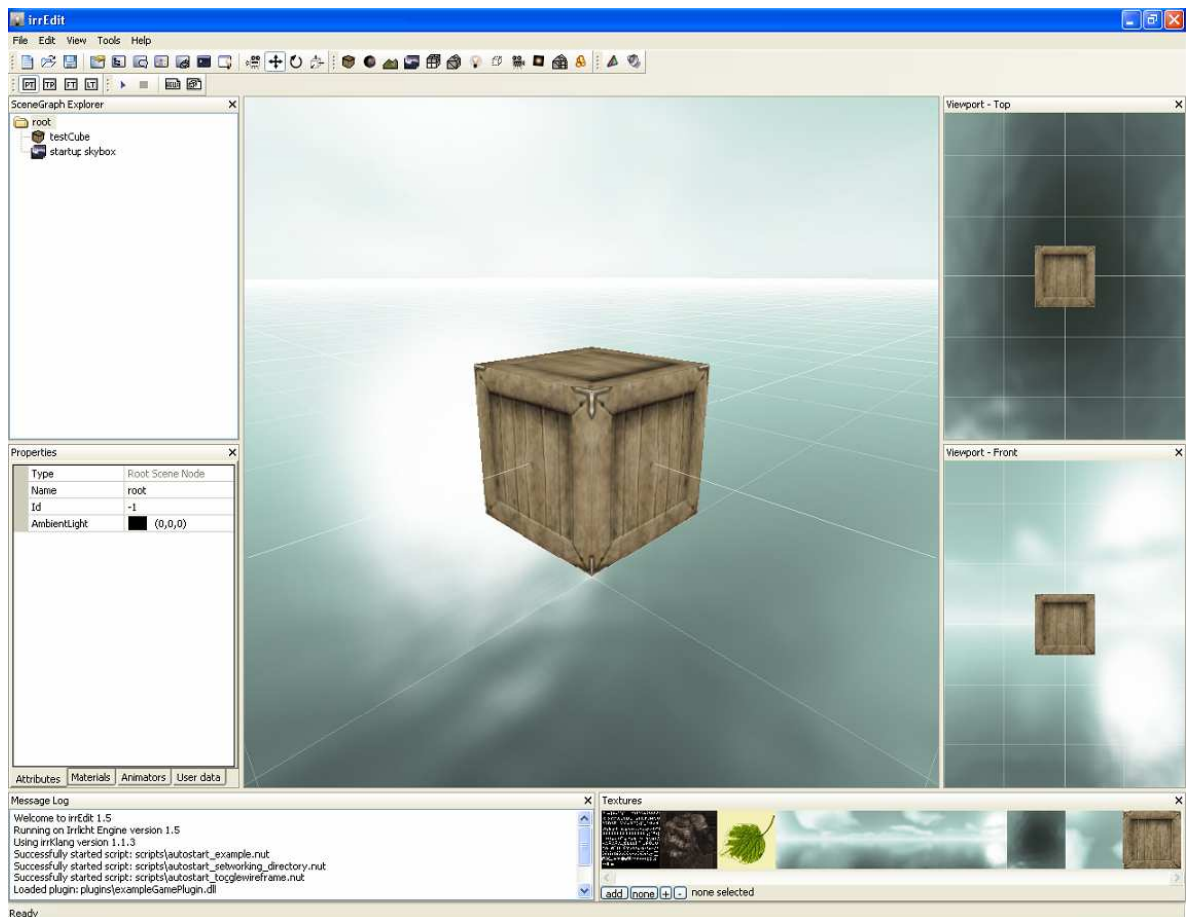
Tato práce se zabývá popisem vlastností editoru a jeho použití pro tvorbu 3D scén IrrEdit.

I. TEORETICKÁ ČÁST

1 IRREDIT

1.1 Popis editoru

IrrEdit je volně šiřitelný realtime 3D world editor s lightmap generátorem. Je používán jako editor pro hry a 3D multimediální aplikace. Byl vytvořen mimo jiné pro open source Irrlicht 3D engine a může být upravován a rozšířen pro použití vlastních aplikací. Narozdíl od Irrlicht engine nemá IrrEdit otevřený zdrojový kód. Od verze 0.7.1 si uživatelé mohou zakoupit redistribuční licenci a source licenci. Editor je psaný v C++ a kompilovaný ve VisualStudio.NET 2003. [2]



Obr. č. 1 – Prostředí IrrEditu po spuštění

Editor také obsahuje:

- Vysoce kvalitní radiosity light map generátor,
- editor scén, terénu, generátor částic,
- useredata editor, editor pro animátory,
- mocný opensource a volný API pro čtení ,irr souborů, podporované velkou a aktivní komunitou,
- zásuvný a skriptovací systém pro rozšiřování editoru,
- řadu importních a exportních souborových formátů.

1.1.1 Light Map generátor

IrrEdit má vestavěný vlastní light map generátor. Ten je užitečný pro hry se statickou geometrií, které potřebují nehybné, dobře vypadající osvětlení. Informace o světle a stínu jsou jednoduše uloženy v textuře, tak jak je to známé například z herního engine Quake3.

Light map generátor IrrEditu má spoustu možností pro generování light map:

- Difúzní osvětlení bez stínů,
- stíny (extrémně rychlé),
- globální osvětlení (hardwarově nebo softwarově založený výpočet radiozity).

Možnost „Shadows“ je schopna vypočítat stíny pro středně velké úrovně během několika vteřin. A ostatními dostupnými možnostmi jako například Shadow Opacity, Subsampling a Ambient Light může být podobný globální iluminaci. Pokud to nestačí, má IrrEdit vestavěný tender skutečného globálního osvětlení. Toto je podobné možnosti shadows, ale také je schopen produkovat hezké efekty, jako je barva krve. [3]

1.1.2 Podporované souborové formáty

IrrEdit může importovat, exportovat a zobrazovat následující 3D mesh formáty

souborů:

Formát	Import	Export
3D Studio meshes (.3ds)	ANO	-
Atras Wavefront Maya (.obj)	ANO	-
B3D files (.b3d)	ANO	-
Cartography shop 4 (.csm)	ANO	-
COLLADA (.xml, .dae)	ANO	ANO
DeleD (.dmf)	ANO	-
FSRad oct (.oct)	ANO	-
rrlicht static meshes (.irrmesh)	ANO	ANO
Irrlicht scenes (.irr)	ANO	ANO
Microsoft DirectX (.x)	ANO	-
Milkshape (.ms3d)	ANO	-
My3DTools 3 (.my3D)	ANO	-
OGRE meshes (.mesh)	ANO	-
Pulsar LMTools (.lmts)	ANO	-
Quake 3 levels (.bsp)	ANO	-
Quake 2 models (.md2)	ANO	-

Exportování mesh je nezbytné, pouze když je modifikujeme v IrrEditu například při generování UV souprav pro lighmapy. Při odkazování mesh ve scéně nejsou obvykle modifikovány a nepotřebují být exportovány do jiného formátu, zůstávají v poskytovaném formátu. Následující obrázky jsou podporovány pro textury:

Formát	Import	Export
Adobe Photoshop (.psd)	ANO	-
JPEG File Interchange Format (.jpg)	ANO	-
Portable Network Graphics (.png)	ANO	ANO
Truevision Targa (.tga)	ANO	-
Windows Bitmap (.bmp)	ANO	ANO
Zsoft Paintbrush (.pcx)	ANO	-

Export pro textury tady znamená, v jakých formátech je editor schopen ukládat generované light mapy. Stejně jako mesh nejsou textury modifikovány a zůstávají v poskytnutém formátu při odkazování ve scéně.

Je možné snadno rozšířit import formátů souborů pomocí pluginů. Pokud je třeba, aby IrrEdit exportoval či importoval souborový formát, který ještě nepodporuje, je výhodné kontaktovat pracovníky na www.ambiera.com a tento formát bude přidán. [3]

1.1.3 World, terrain a particle system editor

Protože byl IrrEdit vytvořen jako plně vybavený editor scén, je možné editovat a umisťovat vše potřebné pro hry a multimediální aplikace:

- Terén,
- částicový systém,
- animované a statické meshe,
- světla,
- billboardy,
- sky boxy,
- uživatelské subjekty, poskytované Plutony a další.

Umístěné subjekty jsou nazývány uzly scén a mohou být také hierarchicky uspořádány, jeden uzel scény může být rodičem druhého. Jestliže se rodič přesune, všechny děti ho budou následovat. Tento mechanismus je užitečný pro strukturování velmi komplexních scén, v případě nutnosti rozdělit úroveň na sektory nebo místnosti a ze všech předmětů v místnosti vytvořit děti této místnosti.

Scény jsou uloženy do .irr souborů, které mohou být čteny a zobrazeny přímo Irrlich engine API, který je vybaven úplným C++ zdrojem v balíčku IrrEditu a je také k dispozici i pro různé programovací jazyky jako Java, Python, Perl, Ruby, Lua, C#, VisualBasic, .NET a mnoho dalších. [3]

1.1.4 Pluginy a skriptování

Pokud IrrEdit neposkytuje funkce, které uživatel potřebuje, je snadné je přidat. IrrEdit nabízí tři mechanismy:

- C++ pluginy rozšiřující funkce IrrEditu jako přidávání uzlů scén a importování souborových formátů,
- skriptování rozhraní API pro přidání nových příkazů a nástrojů do editoru,
- C++ pluginy pro tvorbu hry/aplikace běžící přímo v irrEditu pro testování aktuální scény.[3]

1.2 Historie IrrEditu

V této subkapitole je stručný popis historie vývoje editoru včetně některých verzí až po verzi 1.5, která je v současnosti aktuální. Celý vývoj vzal pod svá ochranná křídla Nikolaus Gebhardt, jenž vyvíjel také IrrLicht engine.

Červen 2006 - IrrEdit 0.1

Počáteční vydání

Červenec 2006 - IrrEdit 0.2

IrrEdit je skriptovatelný. K dispozici je 17 skriptovacích funkcí, se kterými je možné manipulovat téměř se vším, s čím editor umí pracovat. Navíc obsahuje autostartovací skripty, které mohou být přidány do menu a panelu nástrojů.

Srpen 2006 - IrrEdit 0.3

- Přidána podpora .b3d; vylepšení importerů .3ds, .obj a podpora běžných ovládacích prvku pro XP.
- Náhled lightmapperu.
- Vylepšená renderovací rychlost.

Spren 2006 - IrrEdit 0.4

- Bulit-In light mapper nyní plně funkční.
- Ambient barvy světla je editovatelný pro celou scénu v kořenovém uzlu scény.
- Tutoriál lightmappingu pro IrrEdit.
- Možnost exportovat meshe jako COLLADA soubory.
- Ligh mapped scény mohou být ukládány a načítány.

Říjen 2006 - IrrEdit 0.5

- Lighmapper má nyní možnost použití subSamplingu, vylepšující kvalitu obrazu.
- Vylepšen 3ds importer.
- Světla nyní již nemohou pohlcovat sama sebe.
- Opravena položka „toggle debugdata of all lights“.
- Vylepšeno barvení při výběru a vylepšena modifikace nástrojů.

- IrrKlang je začleněn do IrrEditu. V době vzniku této verze programu to mělo přínos především v oblasti přehrávání zvukových a streamovaných souborů přímo v editoru.

Duben 2007 - IrrEdit 0.7

- Možnost editovat uzly terénu scény.
- Podpora binárních pluginů k existujícím skriptům.
- Kompatibilita s IrrLichtem 1.3.

Září 2007 - IrrEdit 1.4 alpha

- Všechny verze IrrEditu byly upraveny na odpovídající Irrlicht verze.
- Kompatibilita s Irrlichtem 1.4 alpha.
- Opravená chyba s light mapperem.
- IrrKlang 1.0 využíván jako zvukový engine.
- Okno lightmapperu má rolovací lišty.

Červen 2008 - IrrEdit 1.4.1

- Kompatibilita s Irrlichtem 1.4.1.
- IrrKlang 1.0.4 využit jako zvukový engine.
- Opraveno několik dalších menších chyb.

Září 2008 - IrrEdit 1.4.2

- Kompatibilita s Irrlichtem 1.4.1.
- IrrKlang 1.1.0 je použit jako zvukový engine.
- Nyní je možno změnit hodnoty ZFar kamery dialogového okna.
- IrrEdit nyní pracuje s managerem Vista Aero window (dříve se nezobrazovaly správně textury oken).

Leden 2009 - IrrEdit 1.5

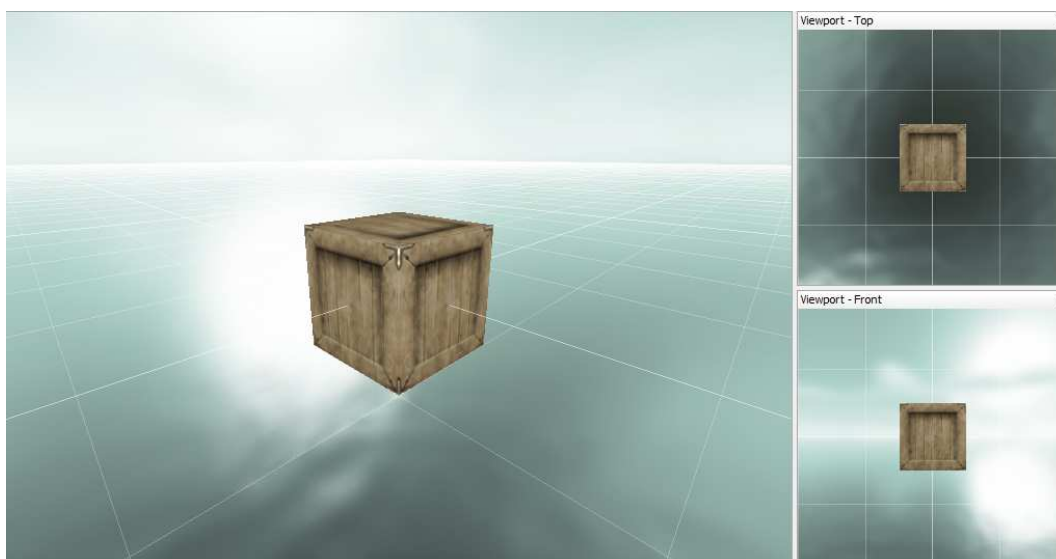
- Kompatibilita s Irrlichtem 1.5.
- Nyní se používá IrrKlang jako zvukový engine.
- Přidána možnost exportovat jako STL a OBJ soubor.
- Přidán loader STL a LWO.

1.3 Popis rozhraní

Následující kapitola se zaměřuje na popis rozhraní, funkcí a ovládání IrrEditu pro tvorbu 3D scén. Po spuštění se objeví hlavní okno obsahující menu bar, několik lišt s nástroji a směs oken plnících různé funkce. [4]

1.3.1 Zobrazovací pole

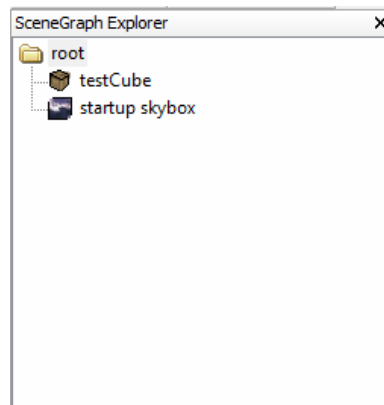
Základem je zobrazovací pole perspektivy nacházející se uprostřed, kde uživatel vidí vytvořený virtuální svět. Ihned po spuštění editoru se v tomto místě nachází testovací krabice a SkyBox. Editor obsahuje kromě základního zobrazovacího pole i pole horního a předního pohledu. Tyto tzv. Viewporty se nacházejí vpravo. Hlavní zobrazení lze také měnit pomocí kláves F1-F4 mezi perspektivou, horním, předním a pohledem zleva. Nebo také pomocí ikon na hlavní liště. [4]



Obr. č. 2 – Základní zobrazovací pole s přidavnými zobrazovacími poli

1.3.2 SceneGraph Explorer

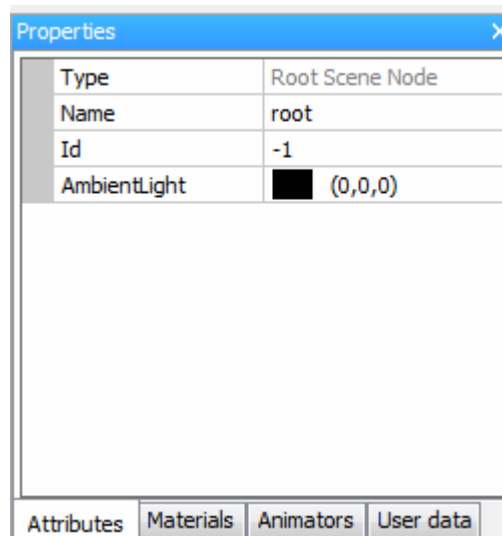
Jedná se o uspořádaný pohled na všechny grafické součásti tvořené scény. Klíčové je, že kliknutím na položku v tomto průzkumníku se vybírá položka ve zobrazovacím poli, která je zvýrazněná zeleným ohraničením. Dvouklikem na objekt proběhne vycentrování tohoto objektu v zobrazovacím poli. Tato vlastnost je velmi užitečná při hledání ve větší scéně. [4]



Obr. č. 3 – Zobrazení SceneGraph Exploreru

1.3.3 Properties

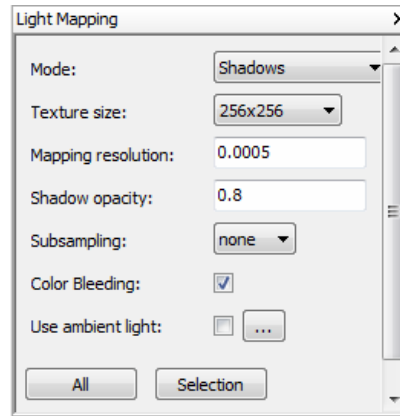
Properties, neboli vlastnosti, se nachází těsně pod SceneGraph Explorerem a umožňuje upravovat detaily jednotlivých uzlů scény. [4]



Obr. č. 4 – Okno s vlastnostmi

1.3.4 Light Mapping

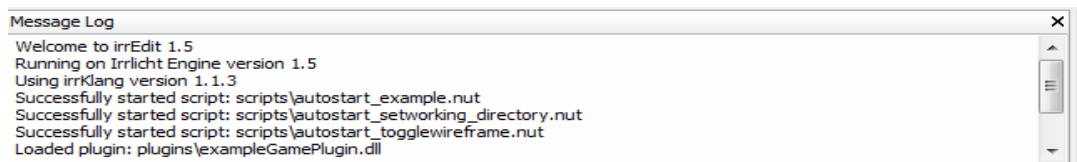
Zde jsou umístěny detaily osvětlení užívané ve scéně. Toto okno není standardně zobrazené, je však velmi užitečné si ho přidat pomocí ikony lišty s názvem „Show Lightmapper Window“ . [4]



Obr. č. 5 – Okno pro úpravy light map

1.3.5 Message Log

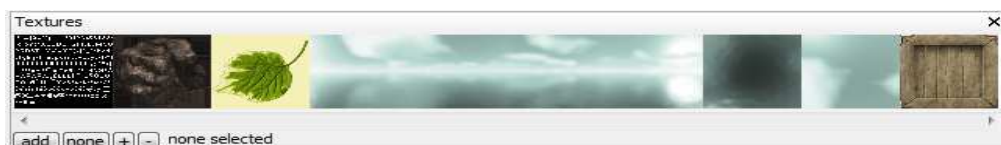
V levém dolním rohu je výpis zpráv. Na prvním řádku lze vidět, jakou verzi IrrEditu uživatel používá. Na dalších se nachází podporovaná verze IrrLichtu Engine a použitá verze irrKlangu. [4]



Obr. č. 6 – Message log

1.3.6 Textury

V tomto prostoru lze vidět textury, které může uživatel používat. Další se mohou přidat pomocí tlačítka „add“. [4]



Obr. č. 7 – Výběr textur

1.3.7 Window managment

Všechna okna zobrazená v IrrEditu mají tlačítka pro uzavření. Všechna lze zavřít až na hlavní zobrazovací pole. Přidání dalších oken umožňuje ikona na hlavní liště a nebo pod záložkou View hlavního menu. [4]

1.4 Základní operace



Obr. č. 8 – Základní lišta s nástroji

1.4.1 Tvarová manipulace

U vložených objektů lze vedle výměny textur měnit i jejich tvar. Po označení příslušného objektu stačí kliknout na ikonu „Scale Object“ (nebo klávesa R) a pomocí barevných „rukojetí“ měnit tvar označeného objektu. [4]

1.4.2 Orientace a posun objektů

Objektem se pohybuje pomocí nástroje „Move object“ (klávesa W), jehož ikona se nachází na hlavní liště. Rotace se provádí pomocí „Rotate Object“ (klávesa E). Podobně jako u tvarové manipulace se změny provádějí pomocí barevných „rukojetí“. [4]

1.4.3 Textury

Výměna textury je velmi jednoduchá. Stačí označit objekt, u kterého se má textura změnit, a v sekci „Textures“ se dvojklikem zvolí textura, jež má nahradit tu stávající. Změna se projeví okamžitě. [4]

1.4.4 Kamera

Do scény lze vložit kameru, jejíž typ se upřesní v Properties v záložce Animators, kde jsou na výběr kamera Maya a kamera typu FPS (First Person Shooter). Také lze použít dalších animačních vlastností, například rotaci atd.

1.4.5 Částicový systém

U tohoto systému lze měnit velikost prostoru, ve kterém budou částice emitované. Je možno také měnit jednotlivé textury, přičemž kombinovat lze až čtyři.

1.4.6 Billboard

Jedná se o 2D objekty, které jsou vloženy v prostoru a vždy směřují normálou do kamery. Je možné je použít jako model pro světlo atd.

1.4.7 Terén

Terén umožňuje zobrazování velkých modelů krajiny. Tvoří se na základě informací získaných z výškové mapy.

1.4.8 Model

IrrEdit implementoval dva základní objekty, a to krychli a kouli. Další objekty lze přidat pomocí ikony „Mesh“. V kapitole 1.1.2 je výpis podporovaných formátů. [4]

1.4.9 Animovaný model

Všechny podporované formáty z kapitoly 1.1.2 mohou být použity pro vložení animovaného objektu, za předpokladu, že daný formát animace podporuje.

1.4.10 OctTree

Jde o datovou strukturu pro hledání rozsahu a lokace tří dimenzionálních bodů. Prostor je dekomponován na 8 subprostorů.

1.4.11 Wireframe

Pomocí tohoto tlačítka lze přepínat mezi normálním pohledem a „drátovou konstrukcí.“

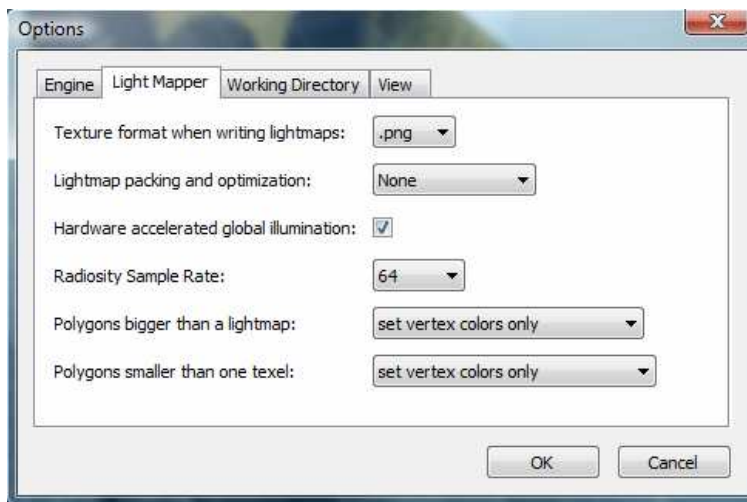
1.4.12 Light mapping

Slouží k propočítání světel jednotlivých uzlů scén, nebo i všech najednou. Do scény se světla přidávají přes ikonu „Light“. V sekci properties lze měnit radius, barvu světla a další vlastnosti. V sekci Light Mapping se pomocí tlačítka „Selection“ propočítají světla pro označený uzel scény. Pomocí „All“ se propočítají pro všechny uzly najednou. [5]

Níže jsou uvedeny měnitelné parametry hodnot light mapped:

- **Mode:** Vybírá stíny, rozptýlené či globální osvětlení. To velmi ovlivní rychlost výpočtů light map, ale také zvýší jejich kvalitu.
- **Texture size:** Vyšší rozlišení textur zvýší rozlišení také pro lightmapy. Výsledkem jsou kvalitnější obrazy. Ale vyšší rozlišení také znamená mnohem složitější výpočet light map.
- **Mapping resolution:** Specifikuje poměr mapování mezi obecnými souřadnicemi a souřadnicemi textur light map. Větší pozornost je třeba věnovat regulování této hodnoty. Vyšší hodnota zvýší rozlišení mapování, ale také zvětší požadavek na počet textur light map.
- **Shadow opacity:** Tato volba je používána pouze tehdy, pokud je označen „Shadow“ mode. Udává míru průhlednosti stínu. Nastavení této hodnoty na 1.0 učiní stín 100% neprůhledný a hodnota 0.0 poskytne opačný efekt.
- **Subsampling:** Zvyšuje preciznost snímku propočtem více než jednoho vzorku na texel. To má za následek například přesnější ohraničení stínů. Čtyřikrát subsampling zabere čtyřikrát více času na propočet výsledku.
- **Color Bleeding:** Je použit pouze v modu „global illumination“. Specifikuje barvy rozptýlených textur, které by měly být brány v úvahu pro osvětlení. Aktivuje se pro realističtější výsledek.
- **Use ambient light:** Tato možnost nastaví osvětlení okolí do uzlu kořenové základní scény.[5]

Je zde také možnost změnit formát uložených light map na .png nebo .bmp:



Obr. č. 9 – Změna formátu light mapy

1.4.13 Prohlížení výsledků

Výsledná scéna se může prohlížet již přímo v editoru, a to pomocí ikony lišty „Start Plugin“. [4]

1.4.14 Skriptování

Jazyk použitý pro skriptování se nazývá Squirrel. Je podobný programovacím jazykům C, C++ nebo Java. Squirrel je objektově orientovaný programovací jazyk, oproti výše zmiňovaným je dynamičtější. Chybové hlášky a texty psané s příkazem „print“ jsou zobrazeny v okně „3D Engine Log“. Pro přidání menu nebo položky panelu se vytvoří samospustitelný skript a registruje se ve funkci jako menu nebo ikona nástroje. [6]

Podprogramy související s editorem:

- **editorAddSceneNode:** Vytváří nový uzel scény. Aplikuje defaultní nastavení editoru.
- **editorFocusPosition:** Zaměřuje kamery na zadanou pozici.
- **editorGetFileNameFromDialog:** Zobrazuje selection dialog vyzívající uživatele k označení souboru. Vrací název souboru a nebo prázdný string pokud není nic označeného

- **editorGetSelectedSceneNode:** Vrací označený uzel scény nebo 0 pokud není nic označeno.
- **editorGetSelectedTexture:** Vrací označenou texturu z okna texture manager.
- **editorUpdateAllWindows:** Udatuje všechna okna editoru.
- **editorRegisterMenuEntry:** Registruje nový vstup menu pod „Tools“.
- **editorRegisterToobarEntry:** Registruje nový vstup panelu nástrojů-
- **editorSetSelectedSceneNode:** Nastavuje a označuje uzel scény.

Podprogramy související s Irrlicht enginem:

- **irrAddSceneNode:** Vytváří nový uzel scény.
- **irrGetChildSceneNode:** Vrací dítě rodičovského uzlu scény.
- **irrGetRootSceneNode:** Vrací kořenový uzel scény.
- **irrGetSceneNodeChildCount:** Vrátí počet dětí uzlu scény.
- **irrGetSceneNodeFromName:** Prohledává scene graph uzlu scény se zadaným jménem.
- **irrGetSceneNodeMaterialCount:** Vrací množství materiálů uzlu scény.
- **irrGetSceneNodeMaterialProperty:** Vrací vlastnost materiálu uzlu scény.
- **irrGetSceneNodeProperty:** Získá hodnotu vlastnosti uzlu scény.
- **irrGetSceneNodeType:** Vrací typ uzlu scény.
- **irrLoadTexture:** Načte texturu do cache textury.
- **irrRemoveSceneNode:** Odstraní uzel scény. Nefunguje pro kořenový uzel.
- **irrSetSceneNodeMaterialProperty:** Vrací vlastnosti materiálu uzlu scény.
- **irrSetSceneNodeProperty:** Nastavuje hodnotu uzlu scény.
- **irrSetWorkingDirectory:** Mění pracovní adresář.
- **irrPlayMusic:** Přehrává streamovanou hudbu přes vnitřní zvukový engine.
- **irrPlaySound:** Přehrává streamované zvuky přes vnitřní zvukový engine.

Ostatní podprogramy:

- **vector3d**: Třída pro držení tří koordinátů.
- **Standard Library**: Standartní knihovna Squirrelu.

Ostatní podprogramy jsou ve standardní knihovně Squirrelu:

Math: abs(x); acos(x); asin(x); atan(x); atan2(x, y); ceil(x); cos(x); exp(x); fabs(x); floor(x); log(x); log10(x); pow(x, y); rand(); sin(x); sqrt(x); srand(seed); tan(x); PI
RAND_MAX[6]

System: clock(); date([time], [format]); getenv(varaname); remove(path);
rename(oldname, newname); system(cmd); time();[6]

Strings: format(formatstr, ...); lstrip(str); regexp(pattern); rstrip(str); split(str, separators);
strip(str); capture(str, [start]); match(str); search(str, [start]);[6]

Blob: dofile(path, [raiseerror]); loadfile(path, [raiseerror]); writeclosuretofile(destpath,
closure); eos(); flush(); len(); readblob(size); readn(type); seek(seek, [origin]); tell();
writeblob(blob); writen(n, type);[6]

I/O: dofile(path, [raiseerror]); loadfile(path, [raiseerror]); writeclosuretofile(destpath,
closure);[6]

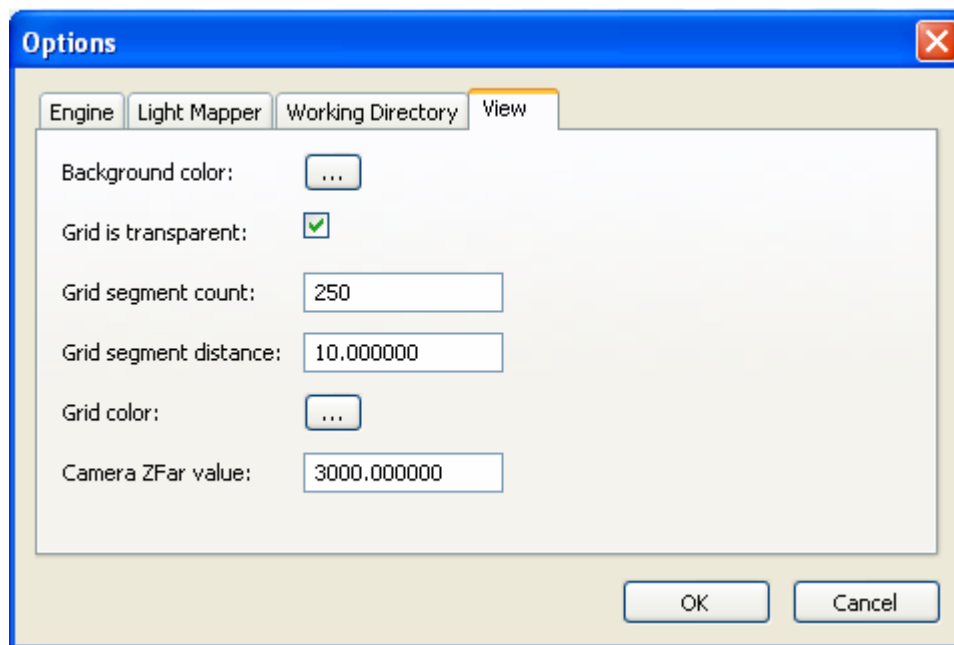
II. PRAKTICKÁ ČÁST

2 TVORBA SCÉNY POMOCÍ IRREDITU

Tato kapitola se zabývá tvorbou scény pomocí IrrEdu. Pro demonstraci možností IrrEdu byla vybrána scéna vesmíru. Vedle použitých postupů je zde také stručně popsána tvorba použitých modelů a textur. Důležitá informace je, že v IrrEdu není možné tvořit modely (kromě krychle a koule), proto byly modely vytvořeny v programu Blender a textury v programu GIMP.

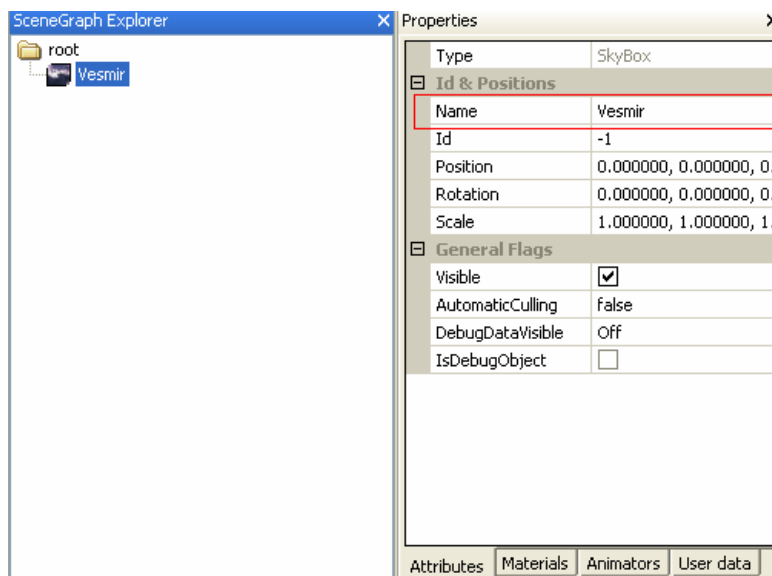
2.1 Základ scény

Základem každé scény IrrEdu je Skybox. Jedná se o prostor, ve kterém lze danou scénu tvořit a upravovat. Zobrazená mřížka reprezentuje pomyslný horizontální střed scény, díky čemuž je usnadněna orientace. Parametry pro úpravu základního rozhraní jsou umístěny v Tools -> Options -> View. Tyto parametry lze měnit i v průběhu tvorby scény. Pokud je potřeba pomyslnou hranici scény rozšířit nebo omezit, stačí změnit hodnotu Grid segment count a distance. Změnou hodnoty Camera ZFar se upravuje viditelnost (dohled) kamery v editoru.



Obr. č. 10 – Základní nastavení rozhraní

Pomocí importování se načítají různé statické a dynamické modely nebo terén. Kvůli přehlednosti je vhodné si pojmenovávat jednotlivé modely, světla, terény a jiné náležitosti.

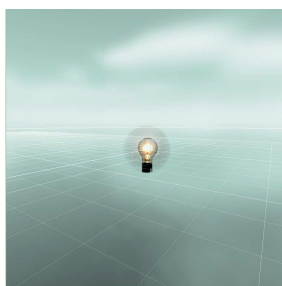


Obr. č. 11 – Změna názvu SkyBoxu

Přidané textury není možné pojmenovávat přímo v editoru, proto je lepší toto učinit při ukládání vytvořených textur.

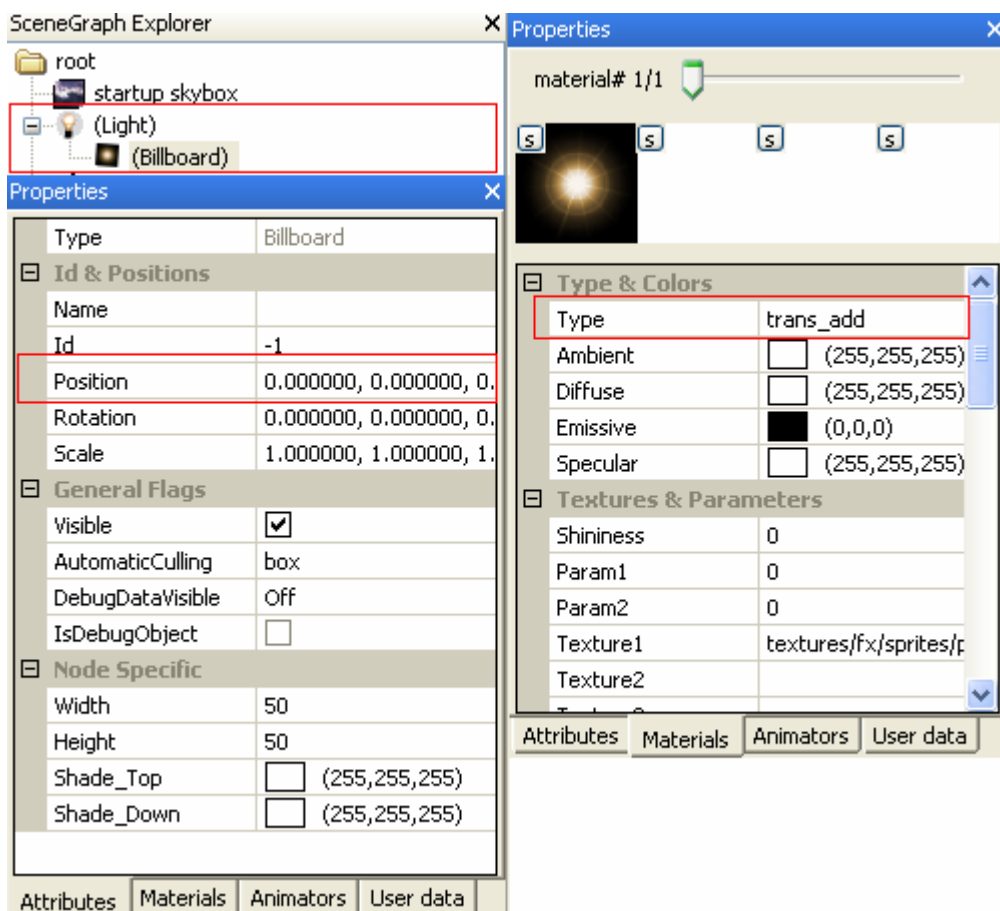
2.1.1 Vytvoření slunce a mlhovin

K vytvoření viditelného Slunce, které vyzařuje světlo sestavené scény, je potřeba vložit světlo samotné pomocí ikony hlavní lišty Light. Toto světlo je reprezentované žárovkou, takže neodpovídá potřebám pro tvorbu dané scény.



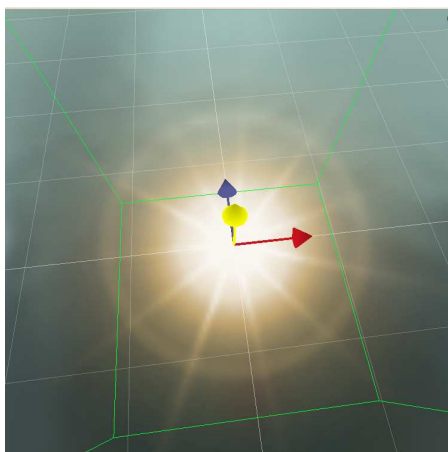
Obr. č. 12 – Vložené světlo

Proto se v sekci Properties->Attributes nastaví DebugDataVisible z hodnoty Full na Off. Následně se přidá do scény Billboard pomocí ikony „Billboard“ hlavní lišty. Označený Billboard se otexturuje pomocí kliknutí na texturu particlewhite.bmp, která byla přidána v okně Textures. Poté je sloučen se světlem. Sloučení se provede označením Billboardu v SceneGraph Exploreru a jeho přesunem na světlo (Light). Hodnoty Billboardu se přenastaví, jak je vidět na následujícím obrázku.



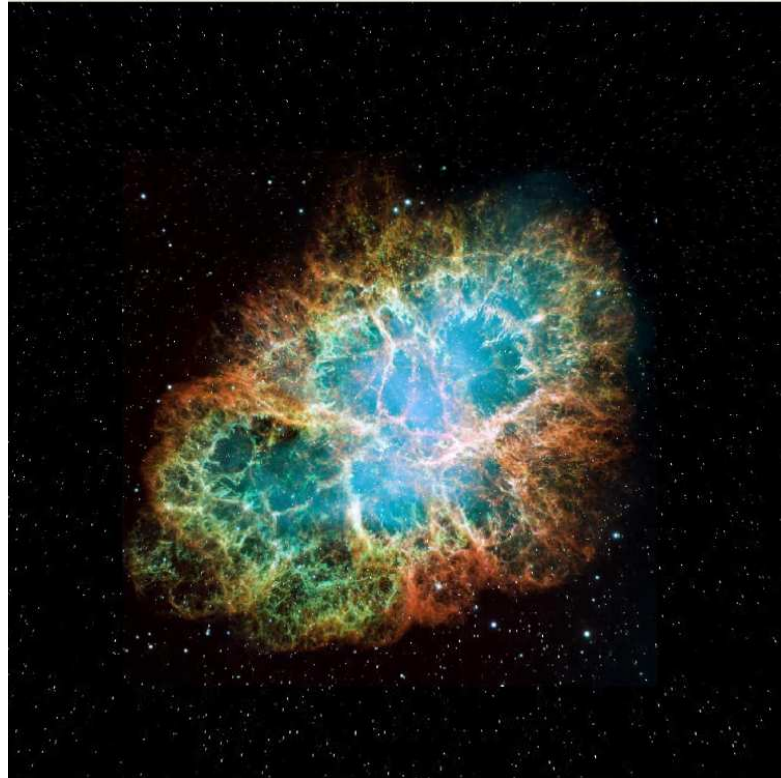
Obr. č. 13 – Nastavení Billboardu

Po těchto úkonech se světlo a Billboard umístí na jedno místo. V sekci Attributes se nastaví pozice (Position) na 0,0,0. Výsledkem je světlo, které se „schová“ v tomto vytvořeném Billboardu na pozici 0,0,0. (Pro lepší osvětlení bylo přidáno ještě jedno světlo.)



Obr. č. 14 – Hotové Slunce

Této vlastnosti Billboardu bylo využito i při vytváření jednotlivých mlhovin. Vhodné obrázky mlhovin byly nalezeny pomocí Google, například na stránce <http://www.dl-digital.com/astrophoto/Nebula-home.htm>. Poté již stačilo aplikovat postup, který byl proveden u vytváření Slunce s tím rozdílem, že nebylo potřeba použití světla.



Obr. č. 15 – Vytvořená mlhovina scény

2.2 Tvorba modelů scény

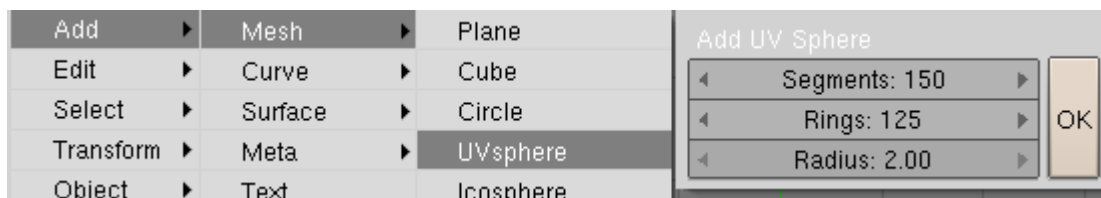
Jak bylo zmíněno dříve, IrrEdit neumožňuje tvorbu 3D objektů. Proto byl pro tento účel použit program Blender - open source software pro 3D modelování, animaci, tvorbu her, rendering, přehrávání a mnoho dalších. [1]

Jelikož se tato práce nezabývá programem Blender, popis tvorby modelů je stručný a ukazuje pouze hlavní kroky a případné obtíže při jejich tvorbě.

2.2.1 Modelování objektů

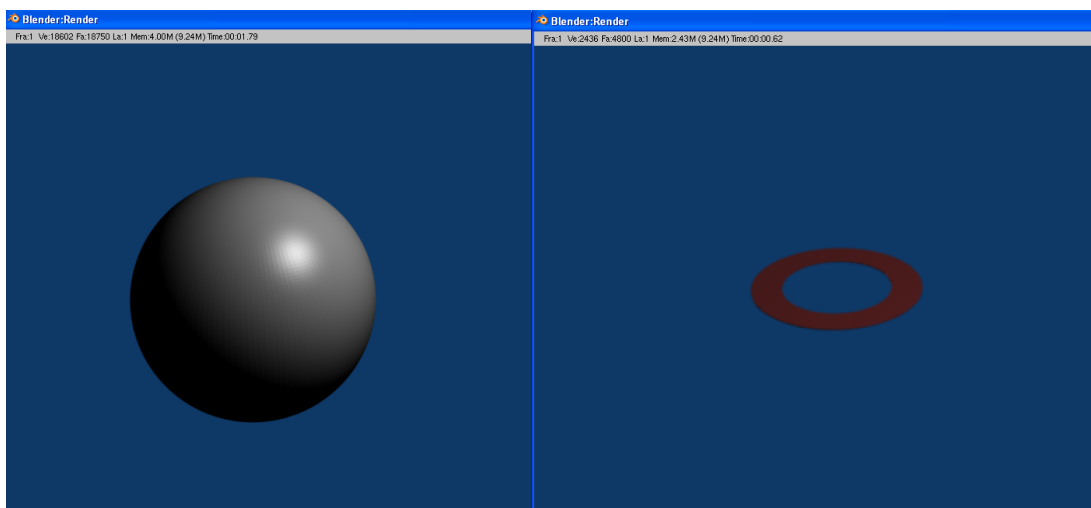
Při modelování je důležité mít na paměti, jak detailní objekty je potřeba vytvořit. Čím detailnější, tím náročnější jsou pro vykreslování. Také jejich fyzické přenášení je vzhledem k jejich velikosti složitější.

Všechny modely jsou nízkopolygonové a byly vytvořeny základními modelovacími technikami. Planety byly vytvořeny přímo ze základní koule, kterou Blender má již implementovanou, pouze s tím rozdílem, že při jejím vytváření byly zvýšeny detaily modelu.



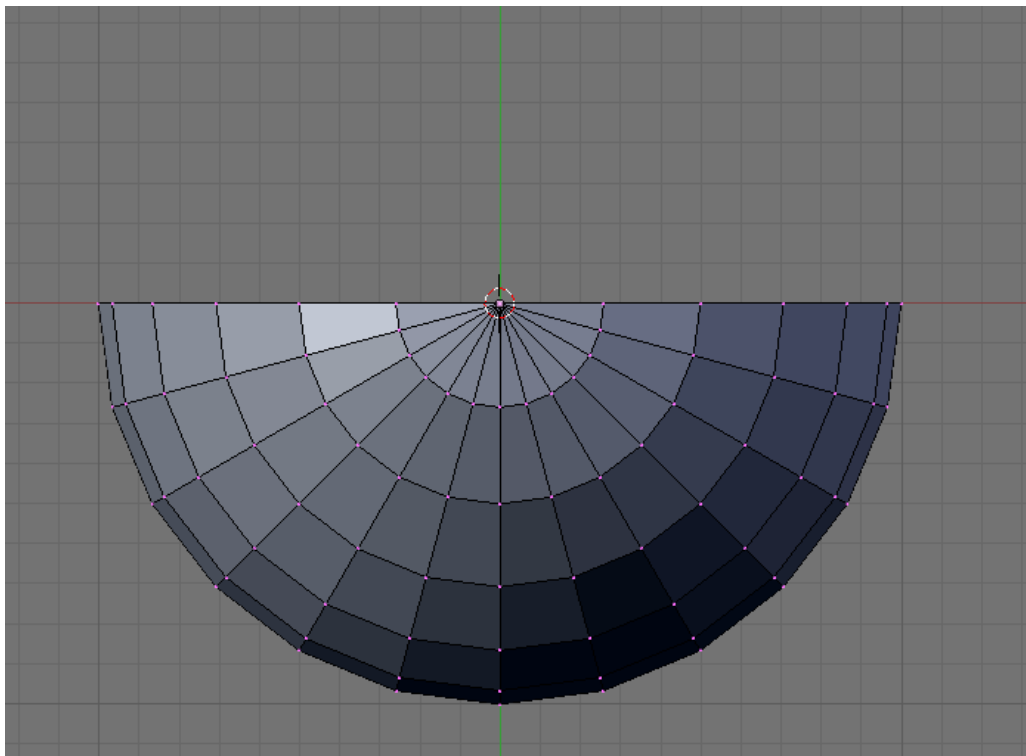
Obr. č. 16 – Vložení koule v Blenderu a nastavení detailů

Prstence Saturnu a Uranu byly následně vytvořeny ze dvou soustředných kružnic a prostor mezi nimi byl vyplněn plochou. Objekt planety a prstence nebyl spojen dohromady, a to z důvodu snadnějšího texturování.



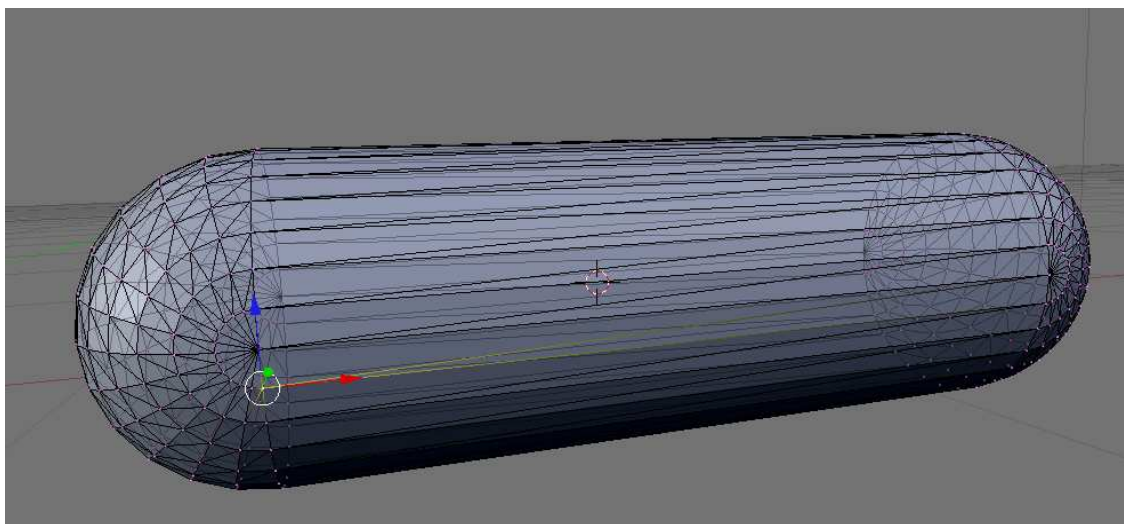
Obr. č. 17 – Model planety a prstence

Vesmírná stanice byla vytvořena ze základních útvarů jako je válec, koule, čtverec a krychle. Válce posloužily jako hlavní trup. Pomocí funkce Scale došlo k natažení objektů podle osy, která byla zvolena (x,y nebo z). Zaoblenější části trupu byly vytvořeny z koule, která byla z jedné poloviny ořezána. To bylo provedeno pomocí funkce BordelTool, díky které se označily vertexy ke smazání a pomocí klávesy x nebo del byly následně smazány.



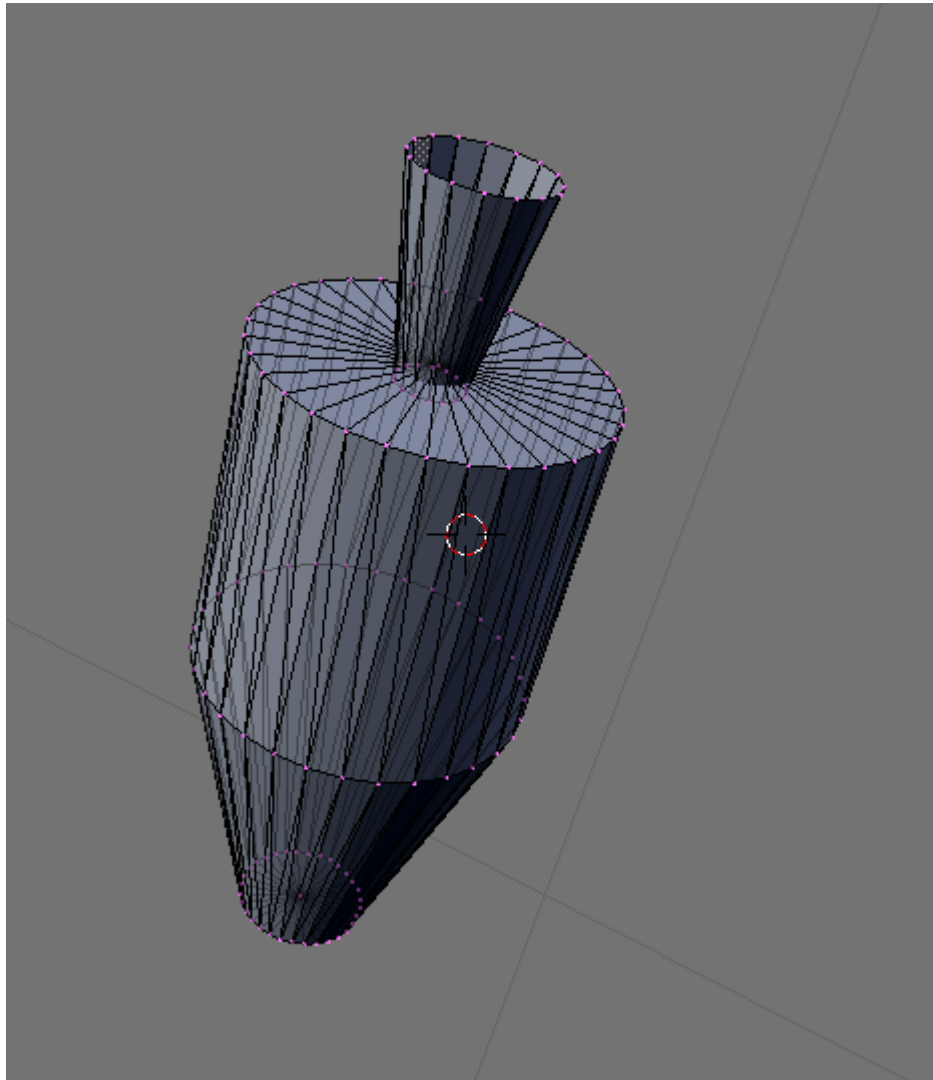
Obr. č. 18 – Ořezaná koule

Poté krajní vertexy těchto koulí byly spojeny a vyplněny plochou pomocí Shift+F.

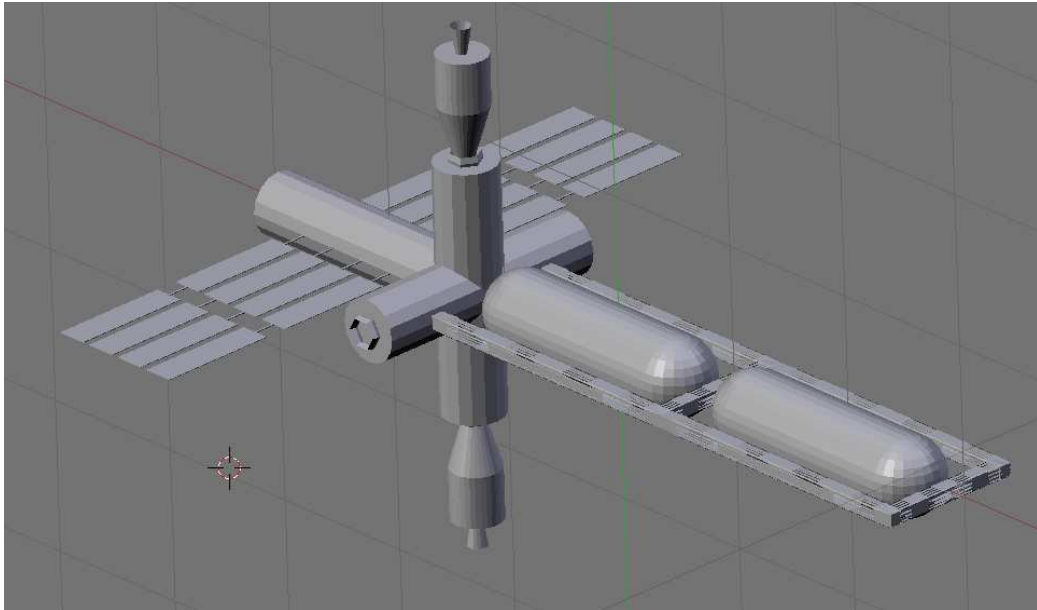


Obr. č. 19 – Výsledná zaoblená část trupu

Další část stanice jako zpevňovací rámy nebo solární panely byly vytvořeny pomocí techniky extruze čtverce nebo krychle a samotným spojováním vertexů jak již bylo popsáno. Přídavné moduly vznikly z válce a jeho následným extrudováním a smazáním některých vertexů.

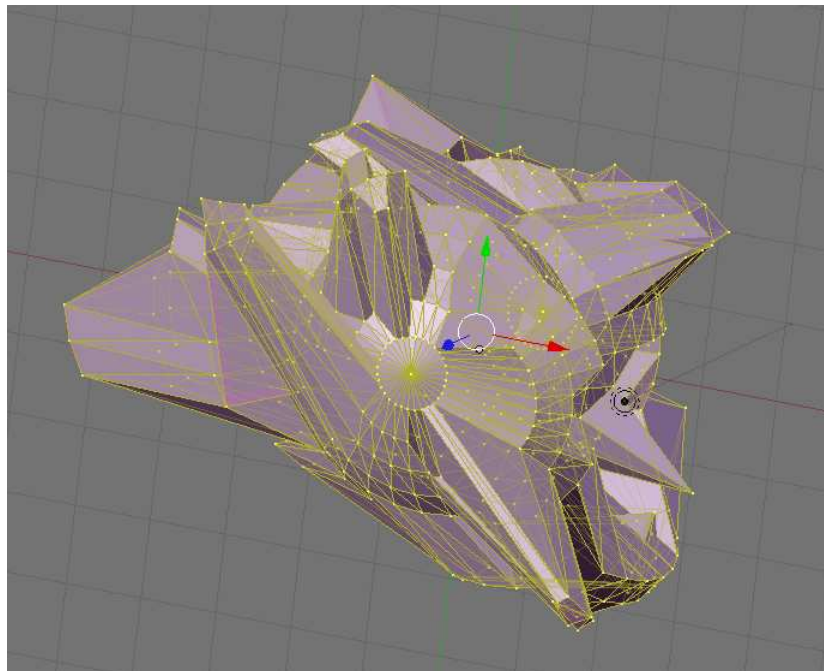


Obr. č. 20 – Přídavný modul



Obr. č. 21 – Výsledný model stanice

Dále byly vymodelovány asteroidy, ze kterých byly vytvořeny pole či asteroidové pásy. Bylo toho docíleno pomocí deformace objektu koule, při níž docházelo k posunutí jednotlivých vertexů a také skupinek vertexů.



Obr. č. 22 – Model asteroidu

Následovalo exportování modelů do formátu, který IrrEdit podporuje. Jedna z možností jak vyexportovat daný neotexturovaný model je pomocí skriptu do formátu .b3d,

u nejnovější verze tohoto skriptu se daný export nezdařil, proto bylo zvoleno přímé vyexportování do .3ds formátu.

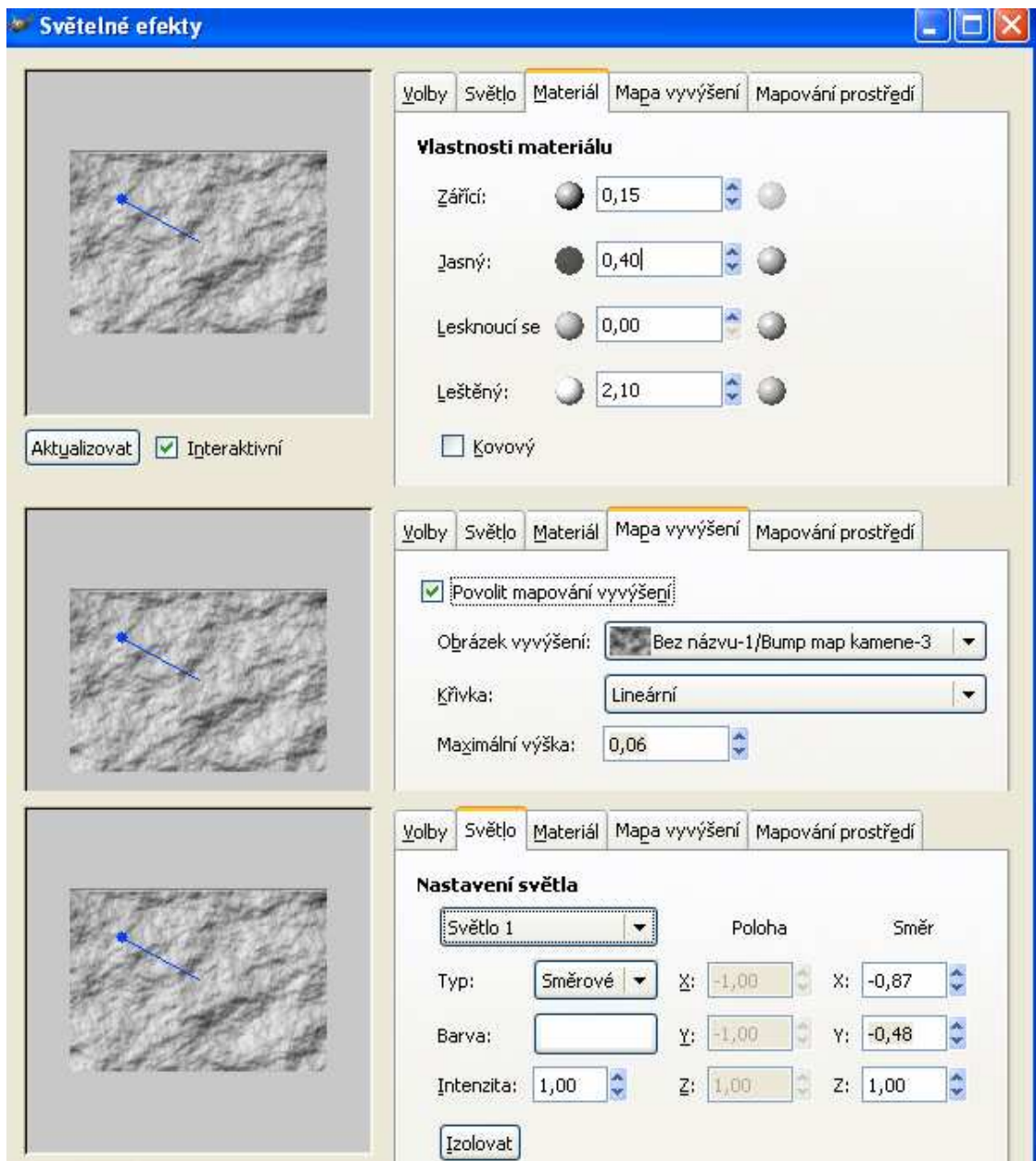
Nastal totiž problém s exportováním modelů z Blenderu do IrrEditu, neboť IrrEdit nepodporuje soubory typu .blend. Při pokusu o vyexportování např. do formátu .obj se daný model zobrazil pouze z jedné čtvrtiny či poloviny. I přes úspěšné exportování do formátu .3ds se nedařilo vyexportovat dané objekty i s jejich texturami, maximálně jenom s barevným odstínem dané textury.

2.2.2 Texturování objektů

Je to technika, která umožňuje dodat realistický vzhled virtuálnímu trojrozměrnému modelu tak, že se určí barva a případně další optické vlastnosti povrchu modelu.[9]. Textury nebo jejich podklady byly získány z různých zdrojů na internetu, zvláště složitější mapy planet, jako je Země.

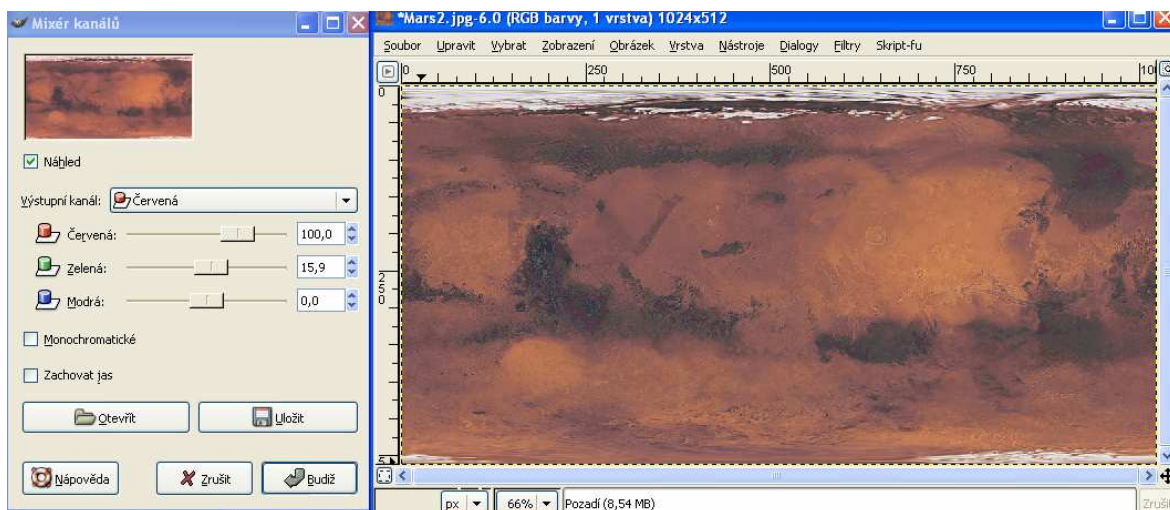
Pro názornou ukázkou byla zvolena tvorba textury v programu GIMP verze 2.2.17 s českou lokalizací pro model asteroidu. Začalo se úplně od nuly, tudíž od prázdné bílé plochy. Nejprve bylo nutno vytvořit nový kanál v sekci Kanály. Poté pomocí utility Filtry -> Vykreslení -> Mraky -> Pevný šum byly nastaveny detaily z 1 na 15. Čím vyšší hodnota detailu, tím byla získána členitější kresba výsledné textury.

Po vypnutí viditelnosti tohoto kanálu byla zvolena vrstva pozadí. Následně se přešlo na nastavení světelných efektů pomocí Filtry -> Efekty se světlem -> Světelné efekty, kde v záložce Mapa vyvýšení bylo povoleno „mapování vyvýšení“ a jako „obrázek vyvýšení“ byl načten již dříve vytvořený kanál. Nakonec bylo potřeba nastavit tyto hodnoty:



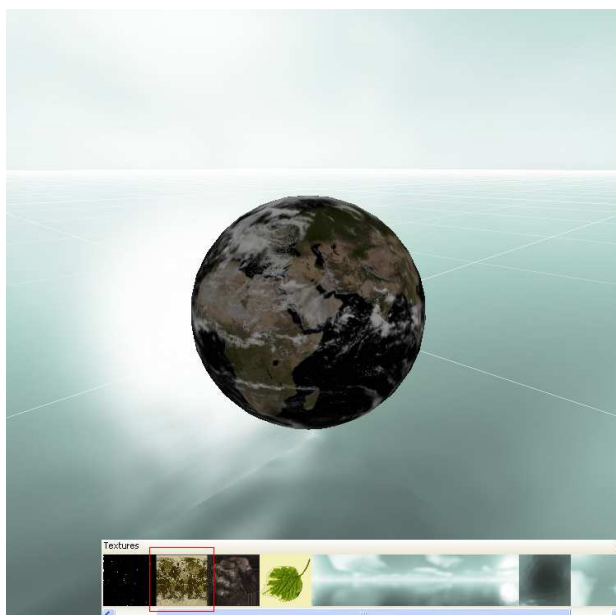
Obr. č. 23 – Nastavení hodnot světelných efektů

Složitější textury planet, byly nalezeny na internetu například na stránce <http://www.oera.net> a upraveny v GIMPU dle potřeby. Jednalo se hlavně o zesvětlení dané textury, pomocí „Mixéru kanálů“ v záložce filtry.



Obr. č. 24 – Ukázka textury Marsu

Samotné texturování probíhalo již přímo v IrrEditu, abychom předešli výše zmíněným problémům s vyexportováním modelů včetně textur.

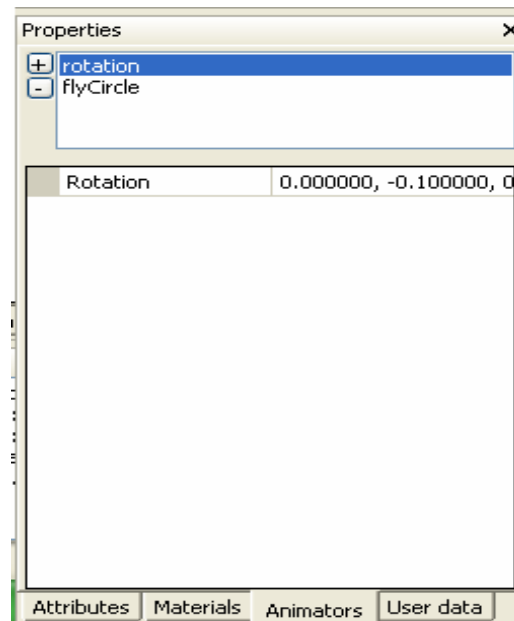


Obr. č. 25 – Otexturovaná planeta Země

2.2.3 Animace objektů

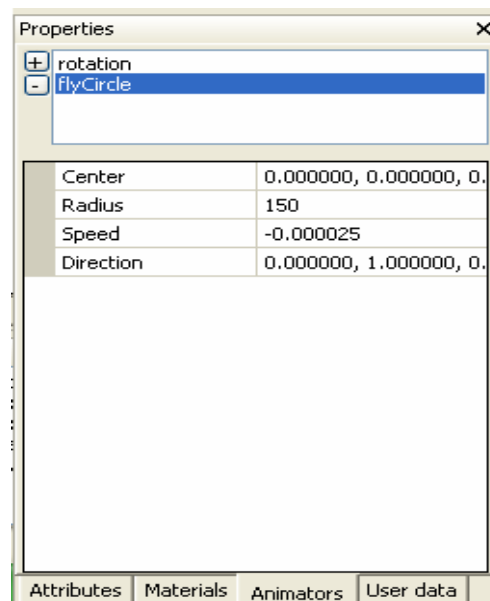
Pro animace planet a asteroidů, jako jejich rotaci kolem vlastní osy a obíhání kolem Slunce, nebyl použit Blender, ale vlastnost Animators v sekci Properties.

Přidáním rotation lze nastavit rotace objektu podle os v pořadí x, y, z, včetně směru rotace reprezentovaného kladnou hodnotou (ve směru hodinových ručiček) a zápornou hodnotou (proti směru hodinových ručiček)



Obr. č. 26 – Nastavení rotation

Pomocí flyCircle, jak již název napovídá, obíhá objekt kolem zvoleného bodu (nastavuje se pomocí Center). Dále lze nastavit Radius (vzdálenost od daného bodu), Speed (rychlost obíhání po směru hodinových ručiček s kladnou hodnotou, v protisměru se zápornou hodnotou) a Direction (určuje, podle které osy bude objekt obíhat). Z výběru Animatoros byla navíc přidána vlastnost collisionResponse.

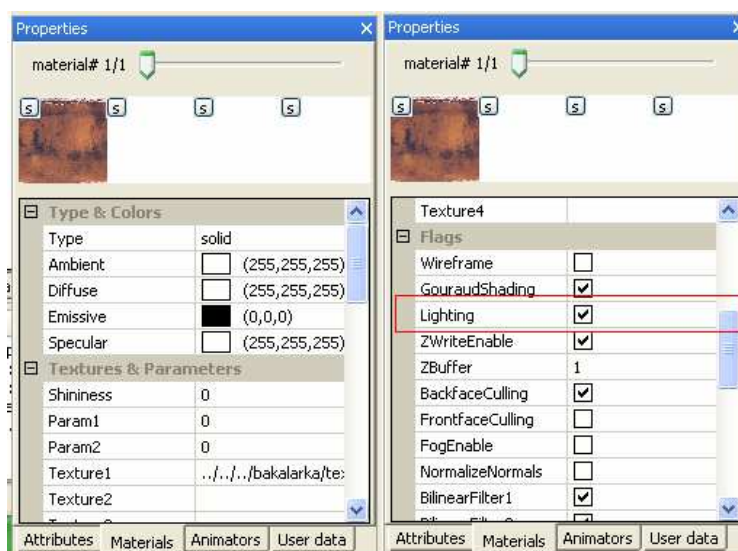


Obr. č. 27 – Nastavení flyCircle

2.3 Výsledná scéna

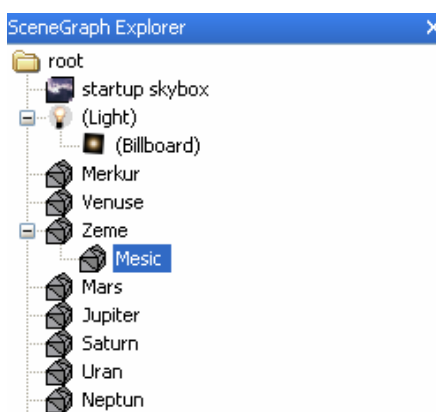
2.3.1 Úprava jednotlivých planet a dalších objektů

Velmi důležité je nastavení objektů pro aktuální scénu. Jsou nutné úpravy parametrů tak, aby objekty správně reagovaly na světlo či světla a stíny. Proto u jednotlivých planet byly provedeny změny typu na solid (pokud tak nebylo nastaveno), nastavení barev a zaškrtnutí políčka Lighting, díky kterému objekt reaguje na změnu světla (nebo světel) v reálném čase.



Obr. č. 28 – Úprava vlastností objektů

U objektu, reprezentujícího Zemi, byl vytvořen i Měsíc. Bylo tedy nutno tyto objekty propojit stejně, jako dříve u Billboardu a světla. Neboť Země obíhá Slunce a zároveň je obíhána Měsícem.



Obr. č. 29 – Přiřazení Měsíce k Zemi

Tak je zabezpečeno, že Měsíc bude obíhat Zemi, a proto není nutno nastavovat u parametru flyCircle střed dráhy oběhu. Stačí pouze nastavit vzdálenost, rychlost a jiné parametry.

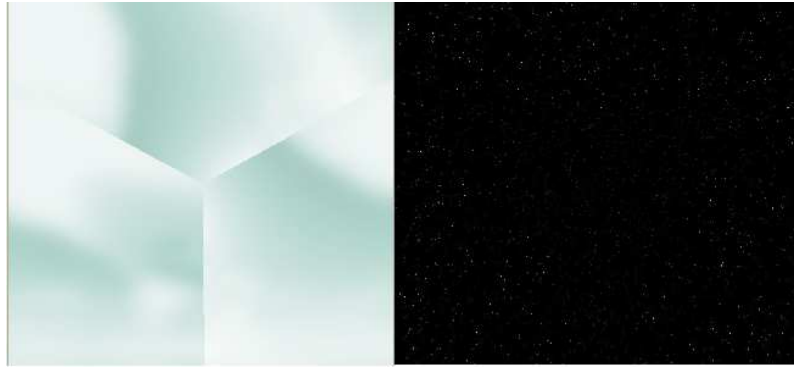
Z asteroidu byl vytvořen postupně pás a dvě pole. Problém IrrEditu spočívá v tom, že neumožňuje hromadné označení objektů a jeho následné kopírování. Proto nezbývalo nic jiného než importovat jeden asteroid, upravit jeho vlastnosti a textury a potom ho následně kopírovat (ctrl+c) po jednom. U každé další kopie byly opět změněny vlastnosti a pozice, aby bylo docíleno interaktivnosti. Je zřejmé, že tento proces byl velmi zdlouhavý. Mnohem jednodušší by bylo vytvořit v Blenderu jednotný statický asteroidový pás. Ale tím by výsledná scéna velmi utrpěla.



Obr. č. 30 – Vytvořený pás asteroidů

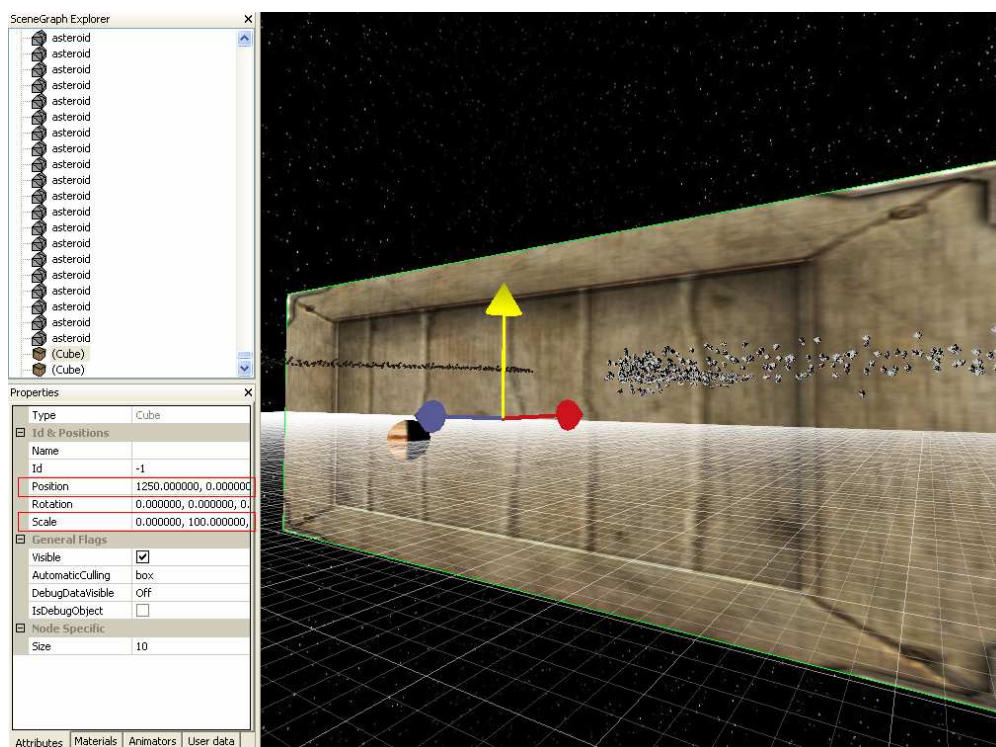
2.3.2 Úprava SkyBoxu a prohlížení výsledné scény

Po celou dobu tvorby nebyla textura SkyBoxu. Změnou textury oblohy na vesmír by daná scéna poměrně ztmavla, což by práci s jednotlivými objekty značně zkomplikovala. Proto se na závěr přidá textura vesmíru, označí se SkyBox a dvojklikem na danou texturu vesmíru dojde k její změně.



Obr. č. 31 – Změna textury SkyBoxu

Jelikož se jedná o otevřenou scénu, bylo nutné vytvořit její hranice. Pokud by byla scénou například uzavřená místnost, hranice by byly samotné zdi. U tohoto příkladu byly vymezení okraje vytvořeny pomocí krychle (cube), jež byla vložena pomocí ikony nástrojové lišty. Krychle byla upravena vlastností Scale v atributech a umístěna na požadované souřadnice. Podle osy x to byly souřadnice (1250,0,0) a (-1250,0,0) atd.



Obr. č. 32 – Vytvořené hranice scény

Takto byly jednotlivé krychle upraveny ve všech směrech, bylo tedy zapotřebí celkem šest objektů. Nakonec se u všech zakázala vlastnost Visible v sekci Attributes a v sekci Animators byla vložena možnost collisionResponse, která slouží k budoucímu propočítávání kolizí objektů a kamery. U pravidelného rozměru scény, by stačilo použít

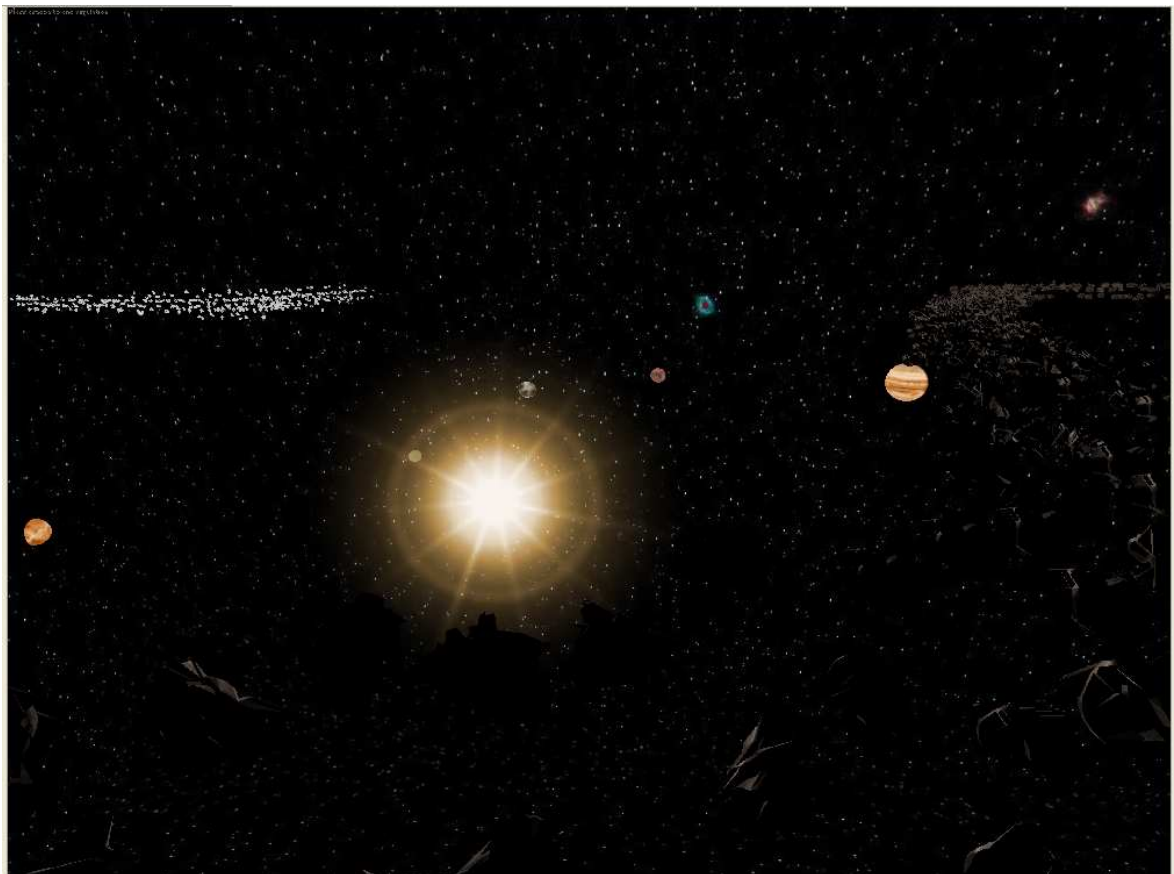
jednu krychli, u které by se změnila velikost Scale tak, aby pokryla celou scénu. Pak by opět byla zakázána vlastnost Visible.

Výslednou scénu lze prohlížet přímo v IrrEditu pomocí Start Pluginu, nacházejícího se na hlavní liště.



Obr. č. 33 – Spuštění prohlížeče scény

Pro náhled vytvořené scény není potřeba žádných programů, ve kterých by scéna byla zobrazitelná nebo spustitelná. Ovládání při procházení scény obstarávají šipky na klávesnici pro pohyb dopředu, dozadu, doleva, doprava a také myš, jejíž kurzor se znehybní a slouží pro otáčení kamery libovolným směrem.



Obr. č. 34 – Finální výsledek

2.3.3 Spuštění v IrrLicht Enginu

Výše uvedená varianta neumožňuje přenášet dané scény mezi různými počítači, protože uložená scéna .irr si uchovává informace pouze o cestách k jednotlivým modelům a texturám. Proto byla potřeba danou scénu spustit pomocí IrrLicht enginu, díky kterému byl vytvořen spouštěcí exe soubor. Celý engine je psán v jazyce C++.

Samotné spuštění scény je velmi jednoduché. Pro tyto účely byla použita verze 1.5. Engine nemá žádný instalační soubor, stejně jako u IrrEditu se jedná o rozbalovací archiv, jenž stačí rozbalit na zvolené místo. Ke spuštění scény byl použit zdrojový kód sloužící k načítání souborů typu .irr. Kód se nachází v adresáři 15.LoadIrrFile. Celá cesta k tomuto adresáři by měl vypadat takto irrlicht-1.5\examples\15.LoadIrrFile. Zde se nachází verze pro DevC++ a pro Microsoft Visual Studio.

```
#include <irrlicht.h>
#include <iostream>
using namespace irr;

#ifdef _MSC_VER
#pragma comment(lib, "Irrlicht.lib")
#endif

int main()
{
    // ask user for driver

    video::E_DRIVER_TYPE driverType;

    printf("Please select the driver you want for this example:\n"
        " (a) Direct3D 9.0c\n (b) Direct3D 8.1\n (c) OpenGL 1.5\n"
        " (d) Software Renderer\n (e) Burning's Software Renderer\n"
        " (f) NullDevice\n (otherKey) exit\n\n");

    char i;
    std::cin >> i;

    switch(i)
    {
        case 'a': driverType = video::EDT_DIRECT3D9;break;
        case 'b': driverType = video::EDT_DIRECT3D8;break;
        case 'c': driverType = video::EDT_OPENGL; break;
        case 'd': driverType = video::EDT_SOFTWARE; break;
        case 'e': driverType = video::EDT_BURNINGSVIDEO;break;
        case 'f': driverType = video::EDT_NULL; break;
        default: return 1;
    }

    // create device and exit if creation failed

    IrrlichtDevice* device =
        createDevice(driverType, core::dimension2d<s32>(640, 480));

    if (device == 0)
        return 1; // could not create selected driver.

    device->setWindowCaption(L"Load .irr file example");

    video::IVideoDriver* driver = device->getVideoDriver();
    scene::ISceneManager* smgr = device->getSceneManager();
```

Obr. č. 35 – Ukázka zdrojového kódu

V této práci byla použita verze pro Microsoft Visual Studio, neboť po kompilaci v DevC++ se program nedal spustit s podporou DirectX 3D. S největší pravděpodobností, byla tato možnost špatně kompilována v knihovně, kterou používá DevC++. Jeho rekompilace či rekonfigurace by byla velice náročná. Microsoft vedle placeného Visual Studia nabízí také freeware Visual Studio 2008 Express Edition, kterou lze stáhnout ze stránek <http://www.microsoft.com/express/vc>. Bylo zapotřebí provést několik úprav v kódu, jako například zadání cesty umístění vytvořené scény.

```
// load the scene  
smgr->loadScene("../..../media/SlunecniSoustavaFinal.irr");
```

Vytvořený program neobsahuje únikovou cestu, neboli neumožňuje standardní opuštění spuštěného programu. Proto byla přidána tato část kódu `#include <windows.h>` a také byla upravena část:

```
while(device->run()) na while(device->run() && !GetAsyncKeyState(VK_LBUTTON)).
```

Nyní lze z programu vyskočit kliknutím levého tlačítka myši.

ZÁVĚR

V teoretické části této práce byla zpracována literární rešerše o freeware editoru pro tvorbu 3D scén IrrEditu. Teoretická část je rozdělena na několik menších celků.

První celek se zabývá základní charakteristikou IrrEditu, obsahem a popisem jeho základních rysů. Dále obsahuje historii vývoje tohoto editoru a popis některých verzí včetně provedených změn.

Další celek se zaměřuje na popis uživatelského rozhraní pro práci na dané scéně, načítání statických i dynamických modelů, jejich texturování, upravování či samotné animování v omezené míře, tvorbu kamer, emitorek částic, skriptování, tvoření light map atd.

Praktická část práce popisuje možnosti použití jednotlivých funkcí editoru při tvorbě scény. Je zde přiblížena tvorba příslušné scény, včetně načítání modelů, jejich exportování, texturování, upravování a částečné provedení animací. Je zde také stručně popsána tvorba jednotlivých objektů pomocí open source programu Blender. Úprava textur byla provedena ve volně šířitelném editoru GIMP. Nastaly problémy, jež byly částečně vyřešeny výběrem formátu .3ds, který oba programy podporují. Také se věnuje spuštění vytvořené scény pomocí IrrLicht engine, u kterého se vyskytly problémy s podporou DirectX 3D. Tyto problémy byly vyřešeny kompilátorem Visual Studio 2008 Express Edition.

Další celek praktické části je příručka k IrrEditu, která se nachází na přiloženém CD. Příručka popisuje uživatelské rozhraní IrrEditu, její funkce, jednotlivé komponenty a utility, včetně jednoduchých ukázek použití nástrojů.

RESULT

Theoretical part of this thesis deals with basic information about freeware editor for production of 3D scenes, IrrEdit. Theoretical part is divided to several smaller parts.

First part deals with primary characteristics of IrrEdit, that includes and describes his keytone. Includes information about history of development of this editor, describes some versions and their changes..

Next part is focusing on description of user's background on given scene. Counting static and dynamic models, their textures, trimming or animating in dull measure. Production cameras, particle systems, scripting, creation light - maps and so on.

Practical part describes possibilities of using functions to create the scene. Here is approached production of appropriate practical scene, including loading of meshes, their exposition, texturing, trimming plus partial fulfilment animation. Also a short circumscribed production of objects in open source programme Blender. Adjustment of textures in freeware editor GIMP. Problems has occurred at exporting from Blender to the IrrEdit. This problem was partly solved by directly exporting to format .3ds that supports both programs. Also describes running of created scene by IrrLicht engine. Problems appeared with the support of DirectX 3D. This problems were solved with the help of compiler Visual Studio 2008 Express Edition.

Next section of practical part is handbook for IrrEdit that is on enclosed CD. Handbook describes user interface of IrrEdit, its functions, individual components plus utilities inclusive simple demonstrations of using tools.

SEZNAM POUŽITÉ LITERATURY

- [1] POKORNÝ, Pavel. Blender - naučte se 3D grafiku. 1. vyd. Praha : BEN - technická literatura, 2006. 247 s. ISBN 8-7300-203-5.
- [2] Ambiera: irrEdit [online]. Verze 1.0. 2006-2007, [cit. 2009-04-26]. ISO-8859-1. Text v angličtině. Dostupný z WWW: <<http://www.ambiera.com/irredit/index.html>>.
- [3] Ambiera: irrEdit [online]. Verze 1.0. 2006-2007 [cit. 2009-04-26]. ISO-8859-1. Text v angličtině. Dostupný z WWW: <<http://www.ambiera.com/irredit/features.html>>.
- [4] Ambiera - Irredit [online]. Verze 1.0. 2006-2007 [cit. 2009-04-26]. ISO-8859-1. Text v angličtině. Dostupný z WWW:
<<http://www.ambiera.com/irredit/gettingstarted.html>>.
- [5] Ambiera - Irredit [online]. Verze 1.0. 2006-2007 [cit. 2009-04-26]. ISO-8859-1. Text v angličtině. Dostupný z WWW:
<<http://www.ambiera.com/irredit/lightmapper.html>>.
- [6] Ambiera - Irredit [online]. Verze 1.0. 2006-2007 [cit. 2009-04-26]. ISO-8859-1. Text v angličtině. Dostupný z WWW: <<http://www.ambiera.com/irredit/scripting.html>>.
- [7] Engine - Wikipedie [online]. Verze 1.0. [1999-2009] [cit. 2009-04-26]. UTF-8. Text v češtině. Dostupný z WWW:
<http://cs.wikipedia.org/wiki/Hern%C3%AD_engine>.
- [8] Počítačová 3D grafika [online]. Verze 1.0. [1999-2009] [cit. 2009-04-26]. UTF-8. Text v češtině. Dostupný z WWW:
<http://cs.wikipedia.org/wiki/Po%C4%8D%C3%ADta%C4%8Dov%C3%A1_3D_grafika>.
- [9] Texturovani - Wikipedie [online]. Verze 1.0. [1999-2009] [cit. 2009-04-26]. UTF-8. Text v češtině. Dostupný z WWW:
<<http://cs.wikipedia.org/wiki/Texturov%C3%A1n%C3%AD>>.

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

API	Application Programming Interface (rozhraní pro programování aplikaci)
I/O	Input/Output (vstupně výstupní zařízení)
FPS	First Person Shooter, pohled z první osoby
Lightmap	Stromová datová struktura, která využívá na optimalizování vykreslování 3D prostoru rekurzivní rozdělení

SEZNAM OBRÁZKŮ

Obrázek 1	Prostředí IrrEditu po spuštění	11
Obrázek 2	Základní zobrazovací pole s přídatnými zobrazovacími poli	18
Obrázek 3	Zobrazení SceneGraph Exploreru.....	19
Obrázek 4	Okno s vlastnostmi.....	19
Obrázek 5	Okno pro úpravu Light map.....	20
Obrázek 6	Message log.....	20
Obrázek 7	Výběr textur.....	20
Obrázek 8	Lišta s nástroji.....	21
Obrázek 9	Změna formátu Lighmapy.....	24
Obrázek 10	Základní nastavení rozhraní.....	28
Obrázek 11	Změna názvu SkyBoxu	29
Obrázek 12	Vložené světlo	29
Obrázek 13	nastavení Billboardu	30
Obrázek 14	Hotové Slunce.....	30
Obrázek 15	Vytvořená mlhovina scény	31
Obrázek 16	Vložení koule v Blenderu a nastavení detailů	32
Obrázek 17	Model planety a prstence	32
Obrázek 18	Ořezaná koule	33
Obrázek 19	Výsledná zaoblená část trupu	33
Obrázek 20	Přídavný modul	34
Obrázek 21	Výsledný model stanice	35
Obrázek 22	Model asteroidu.....	35
Obrázek 23	Nastavení hodnot světelných efektů	37
Obrázek 24	Ukázka textury Marsu	38

Obrázek 25 Otexturovaná planeta Země	38
Obrázek 26 Nastavení rotation	39
Obrázek 27 Nastavení flyCircle.....	39
Obrázek 28 Úprava vlastností objektů	40
Obrázek 29 Přiřazení Měsíce k Zemi	40
Obrázek 30 Vytvořený pás asteroidů	41
Obrázek 31 Změna textury SkyBoxu.....	42
Obrázek 32 Vytvořené hranice scény.....	42
Obrázek 33 Spuštění prohlížeče scény	43
Obrázek 33 Finální výsledek.....	43
Obrázek 34 Ukázka zdrojového kódu.....	44

SEZNAM TABULEK

Tabulka 1:Souborové formáty.....	13
Tabulka 2:Podporované textury.....	14

SEZNAM PŘÍLOH

CD

PŘÍLOHA P I: CD

Adresářová struktura CD:

/prirucka – zde se nachází příručka k editoru

/prace – zde se nachází vytvořená scéna

/modely – zde se nacházejí modely Blenderu

/textury – zde se nacházejí jednotlivé textury

/zdroj – zde jsou umístěny zdrojové kódy